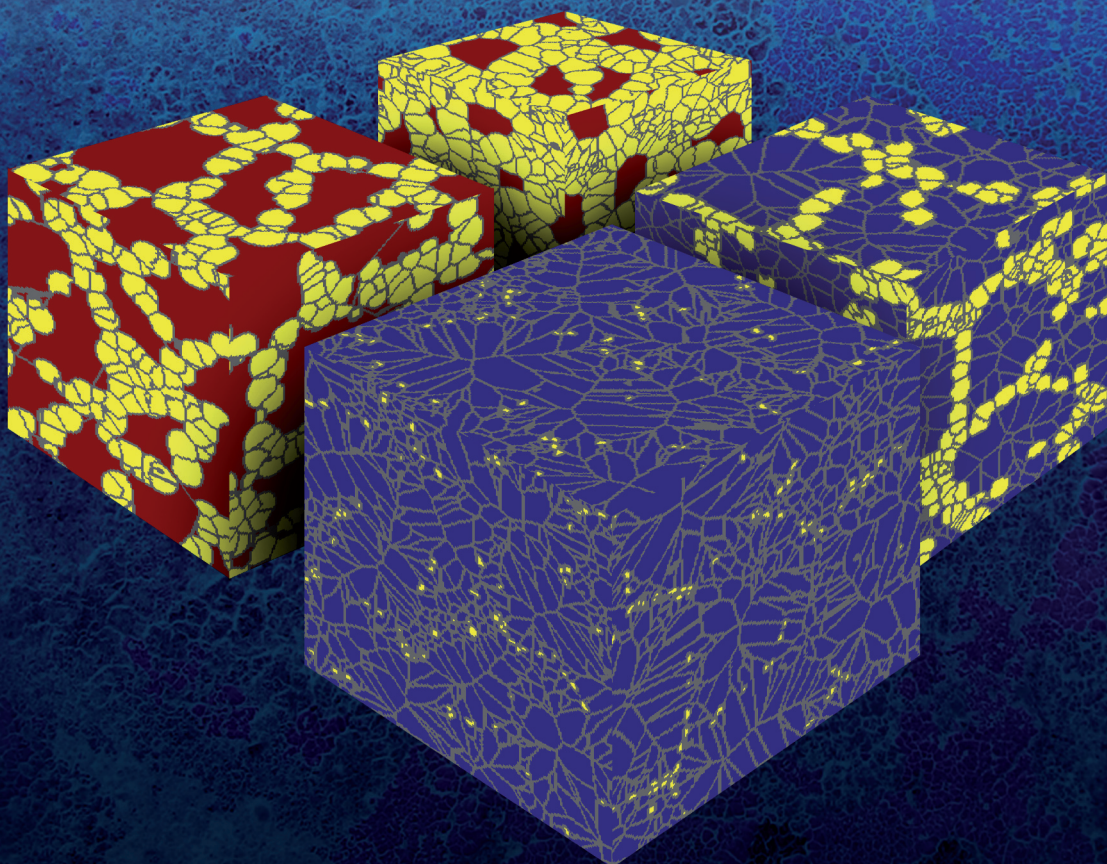


Dmytro Svyetlichnyy

FRONTALNE AUTOMATY KOMÓRKOWE DO MODELOWANIA MIKROSTRUKTURY



WYDAWNICTWA AGH

KRAKÓW 2013

Dmytro Svyetlichnyy

**FRONTALNE
AUTOMATY KOMÓRKOWE
DO MODELOWANIA
MIKROSTRUKTURY**



WYDAWNICTWA AGH

KRAKÓW 2013

0494 pozycja wydawnictw naukowych
Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie

© Wydawnictwa AGH, Kraków 2013
ISBN 978-83-7464-569-0

Redaktor Naczelny Wydawnictw AGH: *Jan Sas*

Komitet Naukowy Wydawnictw AGH:
Zbigniew Kąkol (przewodniczący),
Marek Cała,
Borys Mikułowski,
Tadeusz Sawik,
Mariusz Ziółko

Recenzenci: *prof. dr hab. inż. Zbigniew Malinowski*
prof. dr hab. inż. Norbert Szczygiół

Afiliacje autora:
AGH Akademia Górniczo-Hutnicza

Korekta: *Krzysztof Zadros, Joanna Ciągala*

Projekt okładki i strony tytułowej: *Mariusz Saduś*

Skład komputerowy: „Andre”, tel. 12 422 83 23

Redakcja Wydawnictw AGH
al. Mickiewicza 30, 30-059 Kraków
tel. 12 617 32 28, tel./faks 12 636 40 38
e-mail: redakcja@wydawnictwoagh.pl
<http://www.wydawnictwa.agh.edu.pl>

Spis treści

Przedmowa	9
Część pierwsza. Podstawy frontalnych automatów komórkowych	11
1. Wstęp	13
1.1. Rys historyczny	14
1.2. Klasyfikacja automatów	18
1.3. Automaty komórkowe a modelowanie zjawisk mikrostrukturalnych	21
1.4. Automaty komórkowe a problemy związane z ich stosowaniem	23
2. Algorytmy rozrostu ziarna	29
2.1. Klasyczny algorytm rozrostu ziarna	29
2.2. Algorytm rozrostu ziarna z zastosowaniem losowych reguł	34
2.3. Algorytm rozrostu ziarna ze sterowaniem prędkością rozrostu	38
2.4. Izotropia przestrzeni	41
2.4.1. Algorytm z niezmiennymi warunkami rozrostu ziarna	42
2.4.2. Algorytm uwzględniający zmienną prędkość rozrostu	44
2.4.3. Algorytm uwzględniający odkształcenie lub dowolny kształt komórek	48
2.4.4. Zmodyfikowany algorytm do wyznaczenia czasu przejścia	52
2.5. Sterowanie kształtem rosnącego ziarna	54
2.5.1. Globalne i lokalne współrzędne ziarna	54
2.5.2. Algorytm rozrostu ziarna w kształcie kuli	57
2.5.3. Algorytm rozrostu ziarna w kształcie elipsoidy	57
2.5.4. Algorytm rozrostu ziarna w kształcie prostopadłościanu	59
2.5.5. Algorytm rozrostu ziarna w kształcie ośmiościanu	61
2.5.6. Algorytm rozrostu ziarna w kształcie walca	62
3. Warunki brzegowe i reorganizacja przestrzeni komórkowej	64
3.1. Wymagania stawiane warunkom brzegowym	64
3.2. Warunki brzegowe przy stałej topologii automatów komórkowych	66

3.3. Warunki brzegowe	
przy zmiennej topologii automatów komórkowych	75
3.3.1. Odrzucenie połowy modelowej przestrzeni (<i>halving</i>)	76
3.3.2. Przycinanie i składanie przestrzeni (<i>cutting and bonding</i>)	80
3.3.3. Podwojenie przestrzeni	83
3.3.4. Wyprostowanie przestrzeni automatów komórkowych	84
3.4. Wskazówki dotyczące wyboru warunków brzegowych	85
4. Opracowanie i zastosowanie frontalnych automatów komórkowych	87
4.1. Konwencjonalne automaty komórkowe	87
4.2. Frontalne automaty komórkowe	89
4.3. Przykłady oszacowań nakładów obliczeniowych	94
4.4. Uniwersalny frontalny automat komórkowy	95
5. Przygotowanie danych do tworzenia siatki	
metody elementów skończonych	101
5.1. Zasady przygotowania danych	101
5.1.1. Wczytywanie informacji o mikrostrukturze	103
5.1.2. Sprawdzenie ciągłości ziaren	103
5.1.3. Eliminacja zbyt drobnych ziaren	106
5.1.4. Stworzenie listy komórek leżących na granicach ziaren	106
5.1.5. Lista płaszczyzn styku ziaren	106
5.1.6. Uporządkowanie ścian komórek	
na powierzchni styku dwóch ziaren, sprawdzenie ich ciągłości	
i wyznaczenie konturu każdej płaszczyzny	107
5.1.7. Stworzenie listy krawędzi (linii) każdej płaszczyzny	108
5.1.8. Stworzenie listy wszystkich linii	109
5.1.9. Algorytm sprawdzania linii	109
5.1.10. Stworzenie i wypełnienie listy linii dla każdego ziarna	111
5.1.11. Stworzenie i wypełnienie listy wierzchołków	111
5.1.12. Stworzenie i wypełnienie listy wierzchołków	
dla każdej płaszczyzny i ziarna	
oraz list płaszczyzn i ziaren dla każdego wierzchołka	112
5.1.13. Zapisywanie wyników do plików	112
5.2. Przykłady siatek MES	114
Część druga. Modelowanie zjawisk mikrostrukturalnych	115
6. Modelowanie początkowej mikrostruktury o zadanych parametrach	117
6.1. Zasady i struktura modelu kształtowania	
początkowej mikrostruktury	117
6.2. Wybór warunków symulacji i symulacja	118

6.3. Uzyskanie mikrostruktury o zadanym rozkładzie wielkości ziaren	126
6.3.1. Przedstawienie rozkładu teoretycznego w postaci szeregu rozdzielczego	126
6.3.2. Wyznaczenie liczby ziaren i reprezentatywnej objętości modelowej oraz sprawdzenie rozmiarów przestrzeni komórkowej	127
6.3.3. Zadanie warunków zarodkowania	128
6.3.4. Modelowanie mikrostruktury i wyznaczenie empirycznego rozkładu wielkości ziarna	128
6.3.5. Porównanie rozkładu empirycznego z teoretycznym i wyznaczenie błędu dopasowania	129
6.3.6. Sprawdzenie kryterium dopasowania i korekta warunków zarodkowania	130
6.3.7. Obliczenia końcowe prędkości zarodkowania oraz modelowanie sprawdzające	133
6.3.8. Określenie warunków zarodkowania	134
6.4. Przykład symulacji mikrostruktury i jej wyniki	137
6.5. Orientacja krystalograficzna ziaren	142
6.5.1. Dopasowanie rozkładu orientacji ziaren	142
6.5.2. Dopasowanie rozkładu kąta dezorientacji granic ziaren	143
6.5.3. Dopasowanie rozkładu oraz orientacja ziaren i kąta dezorientacji granic	145
7. Krzepnięcie	146
7.1. Model FCA do modelowania makrostruktury podczas krzepnięcia	146
7.2. Założenia wstępne	149
7.3. Modelowanie zjawiska zmiany stanu skupienia materiału (krystalizacja)	151
7.4. Uwzględnienie pola temperatury podczas modelowania krzepnięcia	157
7.5. Opis modułu FCA	159
7.6. Wyniki modelowania	161
7.6.1. Modelowanie wlewków za pomocą FCA o dużej liczbie komórek	161
7.6.2. Wyznaczenie parametrów zarodkowania	166
7.7. Modyfikacje programu FCA	168
7.8. Weryfikacja modelu i jego optymalizacja ze względu na nakłady obliczeniowe	173
8. Rekrystalizacja	183
8.1. Rozwój metod badania i modelowania rekrystalizacji	183
8.2. Główne założenia do modelowania rekrystalizacji	187

8.3. Podstawowe modele	188
8.3.1. Rozwój dyslokacji	188
8.3.2. Naprężenie uplastyczniające	190
8.3.3. Zarodkowanie	191
8.3.4. Rozrost ziaren	194
8.4. Kinetyka rekrytalizacji	197
8.4.1. Warunki brzegowe	197
8.4.2. Wymiar przestrzeni	198
8.4.3. Kształt ziaren	202
8.5. Rekrytalizacja statyczna	204
8.5.1. Zarodkowanie podczas rekrytalizacji statycznej	204
8.5.2. Kinetyka rekrytalizacji statycznej	208
8.6. Rekrytalizacja dynamiczna	214
8.6.1. Zarodkowanie i rozrost ziaren	215
8.6.2. Uwzględnienie geometrii odkształcenia	217
8.6.3. Naprężenie uplastyczniające	219
8.6.4. Rekrytalizacja metadynamiczna	223
8.7. Wielkość ziarna	224
8.8. Kierunki dalszych badań rekrytalizacji	228
9. Przemiany fazowe w stanie stałym	230
9.1. Podstawowe wiadomości o przemianach fazowych w stali	230
9.2. Modelowanie przemian fazowych	233
9.3. Model przemian fazowych	236
9.4. Wyniki modelowania wstępnego	239
10. Rozdrobnienie mikrostruktury	243
10.1. Materiały drobnoziarniste	243
10.2. Model rozwoju mikrostruktury i zmian własności mechanicznych materiałów silnie rozdrobnionych	246
10.2.1. Modele oparte na metodzie elementów skończonych: MES I i MES II	247
10.2.2. Automaty komórkowe do modelowania powstania nowych granic ziaren	247
10.2.3. Teoria plastyczności kryształów	248
10.2.4. Naprężenie uplastyczniające	249
10.2.5. Model oparty na metodzie elementów dyskretnych	249
10.3. Modele automatów komórkowych do rozwoju struktur dyslokacyjnych	249
10.3.1. Model I	250
10.3.2. Model II	255
10.3.3. Model III	255

10.4. Weryfikacja parametrów automatów komórkowych	256
10.5. Zastosowanie opracowanego modelu do przeprowadzenia testowych obliczeń	257
10.5.1. Walcowanie pakietowe	257
10.5.2. Symulacja procesu MAXStrain®	263
11. Przykłady zastosowania frontalnych automatów komórkowych do modelowania mikrostruktury w procesie walcowania	266
11.1. Procesy wieloetapowego odkształcenia (walcowanie w wykrojach)	267
11.1.1. Warunki symulacji	268
11.1.2. Charakterystyka materiału	269
11.1.3. Wyniki modelowania MES	269
11.1.4. Modelowanie rozwoju mikrostruktury	271
11.2. Walcowanie wyrobów płaskich	278
11.2.1. Dane wejściowe	279
11.2.2. Wyniki symulacji	280
Podsumowanie i analiza wyników badań teoretycznych	285
Część trzecia. Praktyczny kurs automatów komórkowych	289
Ćwiczenia laboratoryjne 1. Automaty komórkowe. Podstawowe pojęcia. Jednowymiarowe CA. Lokalne reguły	293
Ćwiczenia laboratoryjne 2. Jednowymiarowe CA. Synchroniczne i asynchroniczne CA	297
Ćwiczenia laboratoryjne 3. Dwuwymiarowe CA	302
Ćwiczenia laboratoryjne 4. Otoczenie (sąsiedztwo)	308
Ćwiczenia laboratoryjne 5. Sterowanie prędkością rozrostu. Izotropia przestrzeni	312
Ćwiczenia laboratoryjne 6. Frontalne automaty komórkowe	316
Ćwiczenia laboratoryjne 7. Kształt ziaren. Prostokąt	321
Ćwiczenia laboratoryjne 8. Początkowa mikrostruktura	324
Ćwiczenia laboratoryjne 9. Granice ziaren. Warunki brzegowe	326
Ćwiczenia laboratoryjne 10. Otwarte warunki brzegowe. Krok czasowy	329
Ćwiczenia laboratoryjne 11. Rozwój mikrostruktury. Przemiana fazowa	333
Literatura	335

Przedmowa

Moja przygoda z automatami komórkowymi zaczęła się około czterdziestu lat temu, na początku lat siedemdziesiątych ubiegłego wieku. Kiedy byłem w szóstej, siódmej klasie, rodzice, wiedząc o moim zamiłowaniu do różnych matematycznych zadań, zagadek i ciekawostek z tej dziedziny, kupili mi drugą już książkę Martina Gardnera pod tytułem *Математические досуги* (1972). Książka ta była tłumaczeniem na język rosyjski kilku rozdziałów z dwóch angielskojęzycznych wydań. Tłumacz i wydawca zaznaczyli, że opublikowana w 1970 roku gra „Życie” jest bardzo niezwykła i niepodobna do żadnej innej. Zachwyceni fantazyjnymi zmianami sytuacji w grze z trudno przewidywalnymi zdarzeniami i skutkami porównali ją ze scenariuszem fascynującego, niekończącego się filmu. Nazwali ją prawdziwą grą XX wieku, w którą najlepiej grać z komputerem.

Nie wiedziałem, bo nie było tam nawet małej wzmianki o tym, że książka opisuje nową metodę, która nosi nazwę automatów komórkowych (*cellular automata*). Nowy świat „sztucznego życia” pochłoniął mnie na długie miesiące. Teraz, kiedy przyjaciele pytają, czym się zajmuję, i otrzymują odpowiedź, że automatami komórkowymi, nie rozumieją, o co chodzi, ale kiedy uzupełniam moją odpowiedź, mówiąc: gra „Życie” – natychmiast się ożywiają.

Różne wieści o tej metodzie lub bliskich jej zagadnieniach krążyły wśród ludzi. Kojarzono z nią i automaty skończone, i maszynę Turinga z dwuwymiarowym wariantem w postaci mrówki Langtona, i same automaty komórkowe, jednowymiarowe, malujące zadziwiające wzorce „dywaników” oraz dwuwymiarowe, samoreplikujące się, ruchome i demoniczne. Dopiero pod koniec lat osiemdziesiątych ubiegłego wieku miałem możliwość głębszego zajęcia się niektórymi zagadnieniami związanymi z automatami komórkowymi. Po raz pierwszy miałem dostęp do komputera osobistego. Pisałem programy w Basicu i Pascalu. Jako komórki wykorzystywałem bezpośrednio pamięć karty graficznej. Właśnie w owym czasie wprowadziłem do pamięci komputera fraktale i automaty komórkowe. Mrówka Langtona rysowała coś podobnego do rozrastającego się jabłka, cykliczna przestrzeń tworzyła „demony”, rozszerzał się i genetycznie ewoluował tworzony przeze mnie dziwny świat, którego załączkiem były „pokarm” i „ślepe bakterie” zastąpione później przez „organizmy obdarzone zmysłami”, a potem „drapieżniki”. Ale gierki pozostawały tylko gierkami bez żadnego zastosowania praktycznego.

Przełom nastąpił na początku XXI wieku, kiedy pracując nad modelem rozwoju dyslokacji i dostrzegając niezgodności w istniejących modelach, nie mogłem połączyć różnych jego części. Zwróciłem wówczas uwagę na automaty komórkowe jako narzędzie, które mogło mi pomóc zrozumieć i rozwiązać problem. Skorzystałem z jednego już nieco zapomnianego swojego programu, który przetłumaczyłem na Fortran, trochę zmodyfikowałem jego reguły i w dwa dni miałem już gotowy swój oparty na automatach komórkowych pierwszy model rekrytalizacji dynamicznej. Uzyskałem wyniki jakościowo zbieżne z teorią i po niedługich rozważaniach rozwiązałem problem pierwotny, dotyczący kształtu opracowywanego modelu gęstości dyslokacji i naprężenia uplastyczniającego.

Od tej pory automaty komórkowe stały się elementem mojej pracy naukowej i jednym z głównych jej kierunków. Automaty rozwijały się i rozszerzał się obszar ich zastosowań. Za jedno z przełomowych osiągnięć w tej dziedzinie uważam opracowanie frontalnych automatów komórkowych, które wielokrotnie przyspieszają obliczenia i tym samym pozwalają na zwiększenie modelowej przestrzeni i precyzyjności owych obliczeń. Ważne są także liczne zastosowania owego modelu, w tym symulacje podstawowych zjawisk, takich jak krystalizacja, rekrytalizacja, przemiany fazowe czy rozdrobnienie ziaren. Uwzględnienie rzeczywistego odkształcenia, wprowadzenie modeli materiałowych i jednocześnie modelowanie kilku zjawisk pozwoliło wprowadzić modele symulujące całe procesy, czy to walcowania, czy to krzepnięcia w maszynie ciągłego odlewania.

Książka niniejsza z jednej strony zawiera zestaw podstawowych wiadomości na temat automatów komórkowych, które przydadzą się każdemu, kto rozpoczyna swoją przygodę z nimi, a z drugiej – jest prezentacją osiągnięć autora. Nie jest jednak kompleksowym opracowaniem, zawierającym omówienie wszystkich możliwych zastosowań automatów komórkowych, lecz opisuje, jak stosować je do modelowania zjawisk mikrostrukturalnych zachodzących w materiałach. Może być przydatna naukowcom, specjalistom i studentom kierunków związanych z inżynierią materiałową i procesami technologicznymi.

Praca ta składa się z trzech części: pierwszej – teoretycznej, drugiej – prezentującej zasady i przykłady zastosowania automatów komórkowych oraz trzeciej – zawierającej krótki praktyczny kurs mający pomóc czytelnikom przyswoić sobie zawartą w książce wiedzę. W tej ostatniej części zostały zaprezentowane programy do symulacji metodą automatów komórkowych.

Chciałbym w tym miejscu złożyć podziękowania wszystkim, bez których pomocy nie powstałaby ta praca. Najpierw chcę podziękować recenzentom książki, profesorom Zbigniewowi Malinowskiemu i Norbertowi Sczygiolowi, za życzliwość i bardzo pomocne uwagi. Pragnę też wspomnieć o mojej bardzo owocnej współpracy z profesorami Januszem Majtą, Andriyem Mileninem, Henrykiem Adrianem oraz doktorantami Jarosławem Matachowskim i Łukaszem Łachem. Za pomoc w testowaniu i sprawdzaniu kursu praktycznego jestem wdzięczny doktorowi Jarosławowi Nowakowi i mojemu synowi Andriyowi Svyetlichnemu.

Część pierwsza

Podstawy frontalnych automatów komórkowych

W pierwszej części niniejszej pracy zostały omówione teoretyczne podstawy frontalnych automatów komórkowych. W rozdziale pierwszym przedstawiono podstawowe pojęcia i rys historyczny, dalej opisano główne cechy automatów komórkowych, ich klasyfikację i problemy związane z ich stosowaniem do modelowania mikrostruktury. W kolejnych rozdziałach przedstawiono rozwiązanie tych problemów. W rozdziale drugim omówiono algorytmy rozrostu ziaren, poczynając od najprostszych deterministycznych, poprzez reguły losowe, do skomplikowanych, dokładnych, wprowadzających sterowanie prędkością rozrostu i kształtem ziaren. Rozdział trzeci jest poświęcony warunkom brzegowym i przekształceniu przestrzeni komórkowej. Zasady działania frontalnych automatów, ich zalety oraz przejście od automatów klasycznych do frontalnych stanowią treść rozdziału czwartego. Ostatni rozdział części pierwszej dotyczy przygotowania danych do tworzenia siatek elementów skończonych, co pozwala na rozbudowę modeli wieloskalowych.

1. Wstęp

Automaty komórkowe (*cellular automata* – CA) – to model dyskretny stosowany w matematyce, teorii obliczeń, fizyce, biologii teoretycznej, mikromechanice i innych dziedzinach. CA jest strukturą obejmującą siatkę komórek, z których każda może się znajdować w jednym ze skończonego zbioru stanów. Siatka może mieć dowolny wymiar. Dla każdej komórki definiowany jest zestaw wielu komórek, nazywany otoczeniem. Działanie CA wymaga wyznaczenia początkowego stanu wszystkich komórek oraz reguł (zasad) przejścia tych komórek z jednego stanu do innego. Na każdym kroku, stosując reguły przejścia i uwzględniając stany sąsiadującym komórek, wyznacza się stan każdej komórki. Zazwyczaj reguły przejścia są takie same dla wszystkich komórek i są stosowane bezpośrednio do całej siatki, zatem CA ma cztery podstawowe cechy: geometrię użytą do przestrzeni, która zawiera komórki, lokalną regułę przejścia, stany komórki oraz otoczenie (sąsiedztwo) komórek.

Geometria albo topologia automatów określa konfigurację przestrzenną automatów komórkowych i globalny porządek rozmieszczenia oraz współdziałania ich elementów. Może ona przyjąć formę regularnej lub nieregularnej siatki, skończonej lub nieskończonej, posiadającej jeden wymiar lub więcej, z nieskończoną lub niewyznaczoną wymiarowością. Akronim CA jest zazwyczaj używany w odniesieniu do szerokiego zakresu różnych struktur. W przypadku siatek skończonych konieczne jest zdefiniowanie warunków brzegowych.

Stan komórki. Jest to najbardziej istotna cecha automatu komórkowego. Każda komórka CA może w dowolnej chwili przyjmować jeden ze skończonej liczby możliwych stanów. Od stanu komórek w bieżącej chwili zależy nowy stan komórek w kolejnej chwili. Stan komórki wpływa na jej zachowanie w kolejnych krokach oraz na zachowanie jej otoczenia. Często (ale nie zawsze) wprowadza się jeden szczególny stan, nazywany pierwotnym (niezakłóconym, początkowym), taki, że lokalna reguła wprowadzi komórkę do tego stanu, jeśli wszyscy jej sąsiedzi się w tym stanie znajdują.

Otoczenie. Sąsiedzi to elementy, od których zależy dany element CA i na które on sam wpływa. Pojęcie otoczenia (sąsiedztwa) jest dla CA kluczowe. W niektórych przypadkach już sama topologia (geometria) określa otoczenie komórki. Otoczenie należy rozumieć nie tyle w sensie geometrycznym, co informacyjnym (istotne jest otrzymanie

i przekazanie informacji, a nie bliskość geometryczna), chociaż strumienie informacyjne często związane są z geometrią. Jeśli rozważa się regularne siatki, możliwe jest określenie różnych rodzajów otoczenia. Dla każdego automatu siatki przeważnie określa się otoczenie na stałe, za pomocą specjalnego wektora – indeksu sąsiedztwa. Z reguły rozpatrywano siatki d -wymiarowe w punktach o współrzędnych będących liczbami całkowitymi, w których rozmieszczano kopie danego automatu. Stan elementu w kolejnym kroku obliczano na podstawie stanu komórki i jej sąsiadów. Sąsiedztwo komórki w znacznej mierze zależy od geometrii CA. Możliwe jest rozróżnienie otoczenia wejściowego i wyjściowego.

Lokalna reguła. Zgodnie z lokalną regułą zmienia się stan komórki w trakcie obliczeń. Głównym elementem reguły są stany komórki i jej sąsiadów, ale nie ogranicza się ona do nich, w regułach mogą być również wykorzystane inne zmienne (w tym procesowe i materiałowe: temperatura, krystalograficzna orientacja, gęstość dyslokacji i inne). Lokalne reguły mogą być wspólne (takie same) dla wszystkich elementów, różnić się lub też być zadawane indywidualnie dla każdego z nich, ponadto mogą być zupełnie deterministyczne lub mieć składową losową.

Ważne w działaniu automatów komórkowych są warunki początkowe, czyli stany poszczególnych komórek przed rozpoczęciem obliczeń. Od stanu początkowego często zależą ewolucja automatu, jego zachowanie i stan końcowy. Czasami stan początkowy zalicza się do elementów konstrukcyjnych i niektóre automaty komórkowe z założenia muszą mieć warunki początkowe w odpowiedni sposób ustalone.

1.1. Rys historyczny

Automat po raz pierwszy opisany przez Johna von Neumanna był dwuwymiarową nieskończoną tablicą jednolitych komórek, przy czym każda z nich była połączona z czterema ortogonalnymi sąsiadami. Została ona przedstawiona na Uniwersytecie Illinois w grudniu 1949 roku i szczegółowo zaprezentowana w późniejszej publikacji (Neumann 1966) wydanej przez Burksa. Pierwotna koncepcja von Neumanna opierała się na idei robota, który konstruuje innego robota. Model ten znany jest jako kinematyczny. Po opracowaniu owego modelu von Neumann dostrzegł trudność tworzenia samoreplikującego się robota, a w szczególności zapewnienia niezbędnych „zapasów części”, z których można robota zbudować. W tym samym czasie, kiedy John von Neumann rozpoczynał prace nad problemem samoreplikujących się układów, jego kolega z Los Alamos National Laboratory, Stanisław Ulam (1952; 1962), studiował symulację wzrostu kryształów za pomocą prostego modelu siatkowego, w którym zastosował dyskretną przestrzeń i dyskretny czas. W ten sposób powstał pierwszy układ automat-komórka. Początkowo był on nazywany komórkową przestrzenią, ale termin „automaty komórkowe” zyskał z czasem większą popularność (został on wprowadzony przez von Neumanna). Tak więc każdy z wymienionych naukowców wniósł po jednym słowie do określenia automatów komórkowych: von Neumann – „automaty” (*automata*), a Ulam – „komórkowe” (*cellular*).

Podobnie jak siatka Ulama, automat komórkowy von Neumanna jest dwuwymiarowy, a samoreplikujący się robot jest opisany algorytmicznie. Wynikiem pracy von Neumanna był uniwersalny konstruktor. Później von Neumann udowodnił, że w takim modelu istnieje wzorec, który może w nieskończoność kopiować sam siebie.

Również w latach czterdziestych XX wieku Norbert Wiener i Arturo Rosenblueth (1946) opracowali model automatu komórkowego tkanek pobudliwych. Celem ich pracy był matematyczny opis propagacji impulsu w węzłach nerwowych serca. Udało im się zasymulować zjawiska fibrylacji i regularnych drgań mięśni sercowych. Ich praca jest cytowana we współczesnych opracowaniach na temat zaburzeń rytmu serca i tkanek pobudliwych.

W związku z powstaniem automatów komórkowych bywa przywoływane nazwisko jeszcze jednego człowieka – twórcy pierwszych niemieckich komputerów, których prototypy powstały przed drugą wojną światową – Konrada Zusego, który złożył komputer jako klasyczną maszynę Turinga.

Wracając do pracy von Neumanna (1966), jej kluczowa koncepcja była opracowana z myślą o modelowaniu układów biologicznych. Głównym celem tego uczonego było wprowadzenie do analizy skomplikowanych naturalnych układów rygorystycznych aksjomatów i dedukcyjnych związków. Przewodnią ideą jego prac było zbudowanie uniwersalnej maszyny Turinga, która zostanie przedstawiana poniżej.

Po pierwsze, nie trudno wyobrazić sobie następujące dwa rodzaje automatów: pierwszy to automat A , który po otrzymaniu instrukcji I może stworzyć automat zakodowany przez I . W rzeczywistości instrukcja I może zostać uznana za program złożony z prostych instrukcji, z których każda jest wykorzystywana do konstruowania podstawowych elementów, wraz z instrukcjami, które określają, jak złożyć te podstawowe części ze sobą. Drugi automat, na przykład B , jest jeszcze prostszy: kopiuje on instrukcję I do części sterującej innego automatu. Teraz można rozważyć A i B wraz z automatem sterującym C , który działa w następujący sposób: otrzymawszy instrukcję I , automat C uruchamia A w celu stworzenia automatu A_1 odpowiadającego I , a następnie uruchamia automat B , kopiując instrukcję I do części sterującej A_1 . Niech automat D składa się z A , B i C . Zatem D jest automatem, który do działania wymaga instrukcji I . Niech I_D będzie instrukcją, która koduje D . Niech E będzie automatem utworzonym z D przez kopiowanie I_D do części sterującej D . Teraz łatwo zauważyć, że E stwarza sam siebie, a więc jest zdolny do samodzielnej replikacji. Ten prosty opis pomija kodowanie i inne szczegóły, które zostały później uwzględnione przez samego von Neumanna. Dotyczy on komórkowej przestrzeni, w której każda komórka może się znajdować w jednym z 29 możliwych stanów. Struktura jest zdolna do samodzielnej replikacji. Proces samoreplikacji można przedstawić w następujący sposób: początkowo maszyna jest umieszczana w środowisku, gdzie wszędzie dostępna jest dowolna ilość sprzętu („zupa sprzętu” – ang. *hardware soup*). Zgodnie z lokalnymi regułami wstępna konfiguracja automatów komórkowych, którą można nazwać pierwszą maszyną, przechodzi przez sekwencję kroków, w których wyciąga „rękę” do „zupy sprzętu”, tworzy swoją własną kopię i odłącza od siebie nowo utworzoną maszynę.

Opisana przez von Neumanna samoreplikacja jest bezpłciowa w tym sensie, że „potomstwo” pochodzi od jednego z „rodziców”. W tej formie replikacji (reprodukcji) „potomstwo” powstaje na podstawie pojedynczego „kodu genetycznego”. „Seksualna” reprodukcja również była rozważana i opisana – maszyna konstruuje automat w oparciu o dwa „kody genetyczne”, a powstałe w ten sposób „potomstwo” nie jest dokładną kopią któregośkolwiek z „rodziców” (Vitanyi 1973).

Ważne jest, by zauważyć, że samoreplikująca się maszyna ma być nietrywialna w sensie zdolności do uniwersalnych obliczeń. Inaczej jednowymiarowy szereg z jedną pierwotną komórką i lokalną regułą, kopiującą siebie na lewego i prawego sąsiada, może być uważany za samoreplikację. Prowadzi to do kwestii, czy CA jest zdolny do uniwersalnych obliczeń i uniwersalnego konstruowania. Jeśli maszyna może skonstruować zestaw automatów, to nazywamy ją uniwersalnym konstruktorem. Jeśli ten zestaw zawiera również i tę maszynę, to jest to samoreplikacja.

W latach sześćdziesiątych ubiegłego wieku automaty komórkowe rozpatrywane były jako szczególny typ układów dynamicznych oraz ustalono ich powiązanie z dziedziną dynamiki symbolicznej. Najbardziej znaczącym wynikiem tych prac było opisanie zestawu reguł automatu komórkowego jako zbioru ciągłych endomorfizmów w przestrzeni przesunięć.

W roku 1970 (Gardner 1970) znano dwuwymiarowy model automatu komórkowego o dwóch stanach, określane jako „Życie” lub „Gra w życie” (*Life, The game of life*). Główna idea wymyślonej przez Johna Conwaya i spopularyzowanej przez Martina Gardniera gry polegała na tym, żeby rozpoczynając od jakiegoś dość prostego rozmieszczenia organizmów, prześledzić ich ewolucję z początkowej pozycji pod wpływem „praw genetycznych”, które kontrolują narodziny, życie i śmierć.

Conway dość ostrożnie dobierał swoje reguły i długo sprawdzał je w praktyce, próbując spełnić trzy warunki:

- 1) Nie powinno być żadnej konfiguracji, w przypadku której można by łatwo dobieść możliwości nieograniczonego rozrostu.
- 2) Powinny istnieć początkowe konfiguracje, które posiadają możliwości nieograniczonego rozwoju.
- 3) Powinny istnieć proste początkowe konfiguracje, które w ciągu długiego czasu rosną, zmieniają się i kończą swoją ewolucję na jeden z trzech sposobów: zupełnie znikają, przechodzą w konfigurację stabilną lub w stan nieskończonych oscylacji.

Uzyskane reguły okazały się zaskakująco proste, ale przed ich sformułowaniem należy zwrócić uwagę na wyznaczenie sąsiadów. Wybrano ich ośmiu, a takie otoczenie nazywano sąsiedztwem Moore’a (1962).

Zatem „genetyczne” reguły gry są następujące:

- Przetrawianie. Każda „żywa” komórka, która ma dwóch lub trzech sąsiadów, przetrwa i przejdzie do nowej generacji.

- Śmierć. Każda komórka, która ma więcej niż trzech sąsiadów, umiera z powodu zbyt gęstego zagęszczenia, a jeżeli ma mniej niż dwóch sąsiadów – z braku wsparcia.
- Narodziny. Jeżeli pusta komórka ma równo trzech „żywych” sąsiadów, następuje narodziny nowego „organizmu”.

Ważne jest, że śmierć i narodziny następują jednocześnie i wszystkie komórki stanowią jedną generację lub są na jednym etapie ewolucji.

W 1969 roku wspomniany już pionier komputeryzacji i programistyki Konrad Zuse wydał książkę *Calculating Space* (w oryginale *Rechnender Raum*) (Zuse 1969). Zasugerował w niej, iż Wszechświat sam jest wielką maszyną obliczeniową, podważając w ten sposób istnienie stałych praw fizyki. Skupił się na automatach komórkowych jako możliwym narzędziu dokonywania takich obliczeń. A zatem jeżeli układ może być obliczony, jest on deterministyczny, a w takim przypadku (między innymi) klasyczne rozumienie pojęcia entropii i jej wzrostu w takich układach (w tym we Wszechświecie) nie ma sensu. Nadal nie ma niezaprzeczalnych fizycznych argumentów przeciwko temu, że „wszystko jest tylko obliczeniem”. Była to pierwsza książka z dziedziny nazywanej teraz fizyką cyfrową.

Stephen Wolfram (1983) opublikował pierwszy z serii artykułów na temat badań bardzo prostej, ale wówczas jeszcze niestudiowanej klasy automatów komórkowych nazywanych elementarnymi. Nieoczekiwana złożoność ich zachowania skłoniła Wolframa do wywiedzenia założenia, że złożoność układów naturalnych oparta jest na podobnym mechanizmie. Ponadto w tym okresie Wolfram sformułował koncepcję prawdziwej losowości i nieredukowalności obliczeniowej oraz zasugerował, że reguła 110 może być uniwersalna, co zostało udowodnione w 1990 roku przez jego asystenta Matthew Cooka. Tezy Wolframa zawarte w jego publikacji zatytułowanej *A New Kind of Science* (Wolfram 2002) podsumowują jego osiągnięcia w nurcie fizyki cyfrowej.

W roku 1987 Brian Silverman (zob. Dewdney 1990) zaproponował automat komórkowy znany teraz jako „Wireworld” (świat przewodów), stanowiący część jego programu o nazwie „Phantom Fish Tank”. Stał się on popularny po opublikowaniu poświęconego mu artykułu w dziale „Computer Recreations” czasopisma „Scientific American”. Automat ów składa się z dwóch rodzajów komórek: niezmiennych lub pustych, które nigdy nie zmieniają swojego stanu, i zmiennych, cyklicznie przechodzących przez trzy stany, które pozwalają odtworzyć ukierunkowany ruch „elektronów”. „Wireworld” jest często używany do symulacji elementów elektronicznych operujących na wartościach bitowych. Mimo prostoty reguł, jakie nim rządzą, za pomocą automatu komórkowego „Wireworld” można stworzyć działający komputer.

W roku 2002 Paul Chapman zbudował model „życia” – RMM (rejestracja maszyna Minsky’ego). W rzeczywistości RMM jest równoważna maszynie Turinga. Pierwsza wersja była duża (268 096 żywych komórek w przestrzeni 4558×21469 komórek) i powolna (20 pokoleń/sekundę przy użyciu Life 32 Johana Bontesa na 400 MHz AMD K6-II). Tak więc w grze „Życie” można wykonać dowolny algorytm, który może być realizowany na nowoczesnym komputerze.

1.2. Klasyfikacja automatów

Praktycznie każda cecha automatów komórkowych może pełnić funkcję czynnika klasyfikującego. Na początku niniejszego rozdziału wspomnieliśmy o pewnym podziale automatów komórkowych. Tutaj różnice zostaną opisane bardziej szczegółowo.

Można wyróżnić podział automatów **ze względu na stan komórek**. Najbardziej rozpowszechnione są automaty, w których zestaw stanów jest taki sam dla wszystkich komórek. Takie automaty nazywano monogenicznymi. Jednak istnieją CA, w których komórki mogą mieć różny zestaw stanów. Są to automaty poligeniczne. Takie CA są rzadziej spotykane. Czasami trudno odróżnić automat monogeniczny od poligenicznego. Już wspomniany tu automat „Wireworld” można zaliczyć zarówno do monogenicznych, gdy się rozpatrzy wszystkie komórki o tym samym zestawie czterech stanów, jak i do poligenicznych, jeżeli założyć dwa rodzaje automatów: z jednym i trzema stanami, zadanymi na wstępie przestrzennie. Można też mówić o automatach binarnych (o dwóch stanach) i wielostanowych. Liczba stanów może być zmienna.

W połączeniu z regułami przejść można rozpatrywać **strukturę automatów komórkowych**: wówczas zestaw stanów może mieć strukturę algebraiczną (liniową) lub bardziej złożoną. Liniowość bywa zatem rozpatrywana jako moduł sumy stanów komórki i jej sąsiadów lub jako zaokrąglenie ciągłej zmiennej. Automaty komórkowe, w których lokalna reguła zależy od sumy stanów sąsiednich komórek, nazywa się totalistycznymi, a pojęcie to zostało wprowadzone przez Stephena Wolframa.

Można je również podzielić **ze względu na kolejność zmiany stanów** na automaty ze strukturą jednokierunkową, dwukierunkową lub cykliczną, które mogą mieć jednotorową, wielotorową lub rozgałęzioną strukturę przejść.

Ze względu na sposób zmiany stanu komórek automaty można podzielić na synchroniczne i asynchroniczne. W tych pierwszych wszystkie komórki przechodzą w nowy stan jednocześnie, na sygnał globalnego zegarka. Wówczas można mówić o kolejnych generacjach i naturalnym wprowadzeniu czasu globalnego. W CA asynchronicznych natomiast komórki przechodzą w nowy stan w kolejności losowej, przy czym ów nowy stan może być od razu po zmianie wykorzystany przez sąsiadów jako wejściowy. Działanie asynchronicznych automatów jest z powodu ich właściwości bardzo zbliżone do metody Monte Carlo i głównym problemem w porównaniu ich z CA synchronicznymi są trudności z wprowadzeniem do obliczeń skali czasowej. Istnieje również możliwość zastosowania w automatach synchronicznych takich samych jak w automatach asynchronicznych metod obliczeń.

Według kryterium geometrycznego najpierw należałoby rozróżnić automaty w postaci siatek i w postaci grupy grafów. Rozbijanie przestrzeni D -wymiarowej na elementy jest powszechnym sposobem wprowadzenia geometrii, natomiast bardziej abstrakcyjnym i rzadziej spotykanym sposobem jej definiowania, jednakowym dla przestrzeni o dowolnej wymiarowości i od liczby wymiarów niezależnym, jest wykorzystanie grupy grafów. Grupa grafów to zestaw powiązanych ze sobą elementów $N = (G, h)$,

gdzie $G = (g_1, g_2, \dots, g_k)$ jest grupą, która definiuje komórki, a h określa mapę połączeń. Faktycznie h określa sąsiedztwo (otoczenie) komórek. Pojęcie grupy grafów jest dogodnym sposobem opisanja jednolitej geometrii – wzorca połączeń, który wygląda tak samo we wszystkich komórkach. Niejednolite połączenia są stosowane również wtedy, kiedy różne komórki mogą mieć różną liczbę sąsiadów i ich konfigurację.

Jak już wspomniano, **siatka** może być regularna lub nieregularna, skończona lub nieskończona, jedno- lub więcejwymiarowa. Może się składać z kwadratów, trójkątów, czworokątów, sześciokątów (w przestrzeni dwuwymiarowej) lub innych elementów. Mogą być one niezmiennie lub zmieniać swój kształt i rozmiar.

W przypadku siatek skończonych pojawia się dodatkowy stopień swobody i możliwe jest zdefiniowanie różnych **warunków brzegowych**. Siatka może mieć periodyczny (okresowy) warunek brzegowy w jakimś wymiarze, jeżeli brzegi są połączone w tym wymiarze. Wymiar ma stały warunek brzegowy, jeśli komórki skrajne przylegają do innych komórek w jakimś uprzednio określonym stanie, który nie zmienia się w trakcie obliczeń. W przypadku gdy ten stan jest stanem pierwotnym, warunek brzegowy nazywa się zerowym. Możliwe jest również, że na jednej granicy będzie periodyczny warunek brzegowy, a na drugiej stały warunek brzegowy. W literaturze polskiej stosowany jest również podział warunków brzegowych o stałej wartości na pochłaniające i odbijające, a periodyczny nazywano czasami przenikającym. Warunkom brzegowym poświęcony jest rozdział 3 niniejszej książki, w którym omówiono kilka innych wariantów takich warunków. Warunki brzegowe również mogą się zmieniać z upływem czasu. Mogą być takie same dla całego brzegu lub uzależnione od rozmaitych czynników.

Geometria może się zmieniać i jeśli jest ona różna w różnych strefach przestrzeni, takie CA nazywa się niejednorodnymi. Są to CA statyczne, w przypadku których zestaw komórek i wzorec wzajemnych połączeń nie ulegają istotnym zmianom z upływem czasu. Możliwe jest rozważenie CA, w których zestaw komórek nie zmienia się w czasie, ale ich wzajemne połączenia mogą się zmieniać. Taka struktura nadal uważana jest za statyczną. Natomiast w CA dynamicznych zarówno zbiór komórek, jak i ich połączenia mogą się zmieniać i takie CA są często wykorzystane do modelowania systemów biologicznych.

Istnieje duża klasa ruchomych CA, które cechują się możliwością zmiany lokalizacji komórki w siatce (lub nawet w układzie bez siatki) podczas ewolucji układu. W automatach nieruchomych lokalizacja komórek pozostaje niezmienna. Właśnie na przykładzie ruchomych i nieruchomych CA można zrozumieć, iż sąsiedztwo w automatach komórkowych ma wymiar bardziej informacyjny niż geometryczny. Ruch „elektronów” w „Wireworld” lub „szybowców” w „Życiu” polega na zmianie stanów, a w ruchomych automatach na przemieszczeniu się komórki w siatce, jednak w obydwu wariantach następuje przekazanie informacji. W jednym przypadku znaczące jest przekazanie informacji o stanie, w drugim o zmianie współrzędnych, ale efektem obu jest takie samo przemieszczenie informacji z jednego punktu siatki do innego. Różnica jest zauważalna raczej w algorytmach i kodach niż widocznym rezultacie.

Istnieją również podziały **według typu otoczenia**. Najpierw należy odróżnić takie opisywane przez grupy grafów automaty, w których już sama topologia określa otoczenie komórki. W przypadku siatek regularnych otoczenie zależy od kilku czynników: wymiarowości przestrzeni i wybranych elementów siatki oraz promienia sąsiedztwa, który wyznacza odległość, na jaką jest przekazywana informacja w jednym cyklu. W najprostszych regularnych siatkach o najmniejszym promieniu (równym 1) najczęściej wykorzystywane jest sąsiedztwo von Neumanna (ortogonalne) i sąsiedztwo Moore'a (sześcian jednostkowy).

Możliwe jest określenie wejściowego i wyjściowego otoczenia komórki – wówczas komórka pobiera informacje z otoczenia wejściowego, a jej stan jest dostępny dla otoczenia wyjściowego. Jeżeli liczba jej wejściowych i wyjściowych sąsiadów jest równa, wtedy CA jest zrównoważony.

Sąsiedztwo może być niezmiennie albo uzależnione od czasu lub czynników losowych.

Lokalna reguła zwykle jest deterministyczna. W deterministycznych CA stan komórki w kolejnym kroku obliczeń zależy wprost od stanu samej komórki i jej najbliższych sąsiadów w poprzednim kroku czasowym. W tym przypadku jej stan zależy tylko od tych dwóch czynników. Taki automat nie posiada pamięci. CA z pamięcią mogą przepuszczać, zatrzymywać lub hamować przekazywaną informację, czyli przyjmować informację bardziej odległą w czasie. CA, w których stan komórki jest wyznaczany na podstawie niektórych przypadkowości, nazywa się automatami z niedeterministycznymi, stochastycznymi lub losowymi regułami.

W klasycznych automatach reguły są abstrakcyjne i niezwiązane z rzeczywistymi procesami, które zachodzą w modelowanych układach. W takich automatach podczas modelowania procesu dla każdej komórki generowane są losowo liczby, które wyznaczają prawdopodobieństwo realizacji procesu.

CA, w których każda komórka ma swoją własną lokalną regułę, są nazywane hybrydowymi. Tego typu struktury zostały opracowane w związku z zastosowaniem w VLSI (*very-large-scale integration*) – zminiaturyzowanym układzie elektronicznym zawierającym w swym wnętrzu od kilku do setek milionów podstawowych elementów elektronicznych, takich jak tranzystory, diody, rezystory, kondensatory. Możliwe jest, by komórka zmieniała swoje lokalne reguły w każdym kroku czasowym. W kontekście VLSI nazywane są one programowalnymi CA lub *tessellation automata*.

Czasami używano reguł zapisanych w postaci równań różniczkowych. W takim przypadku stany komórek są określane przez zestaw zmiennych, których wartości mogą być dowolnymi liczbami rzeczywistymi. W przypadku takich automatów równania różniczkowe są rozwiązywane dla każdej komórki indywidualnie, każda komórka może mieć różne warunki początkowe. Ta klasa CA jest ściśle związana z równaniami różniczkowymi cząstkowymi. Modele takie zajmują pozycję pośrednią między CA a metodą różnic skończonych.

Przyjęło się stosować podział automatów komórkowych **ze względu na ich zachowanie**. Wolfram badając zachowanie jednowymiarowych automatów komórkowych, wyróżnił cztery ich typy:

- 1) automaty sprowadzające do jednorodnych konfiguracji sieci komórek, w których niezależnie od stanu początkowego ustala się jednakowy stan;
- 2) automaty prowadzące do niezmienniających się lub periodycznie powtarzających się konfiguracji;
- 3) automaty charakteryzujące się brakiem porządku zarówno lokalnego, jak i globalnego (chaotyczne);
- 4) automaty prowadzące do złożonych konfiguracji lokalnych; takie automaty charakteryzują się bardzo złożonym, długotrwałym zachowaniem.

Najprostszym modelem CA jest jednowymiarowy zbiór komórek (prawdopodobnie dwukierunkowy, nieskończony). Czas jest dyskretny, a każda komórka w każdym punkcie czasowym znajduje się w jednym ze stanów, wybranym ze skończonego zbioru stanów możliwych. Komórki zmieniają stan na każdym kroku czasowym, a nowy stan jest całkowicie określony przez obecny stan komórki i jej sąsiadów. Funkcja (tzw. reguła lokalna), która określa tę zmianę stanu, jest taka sama dla wszystkich komórek. Automat nie ma żadnych wejść, a więc jest autonomiczny. Zestawienie stanów wszystkich komórek w dowolnej chwili czasu jest nazywane konfiguracją lub globalnym stanem danego CA i odzwierciedla etap ewolucji tego CA. W czasie $t = 0$ CA znajduje się w jednej z konfiguracji początkowych i od tej pory, pod wpływem lokalnych reguł, które są stosowane do każdej komórki na każdym kroku czasowym, zachowuje się deterministycznie. Zastosowanie lokalnej reguły do każdej komórki CA powoduje transformację od jednej konfiguracji do innej. Transformacja ta jest nazywana globalną mapą lub globalną regułą CA.

Automaty komórkowe są bardzo skutecznym narzędziem. Ich główne atuty to: prostota implementacji, wysoka efektywność i dobre własności obliczeniowe modelu, osiągnięte dzięki zastosowaniu dyskretnego czasu i przestrzeni, brak błędów numerycznych oraz wysoka przydatność do obliczeń równoległych i rozproszonych.

Do wad automatów komórkowych można zaliczyć uproszczenia i błędy związane z dyskretyzacją, ograniczenia rozmiarów przestrzeni, wysokie wymagania w stosunku do pamięci podręcznej komputera i długi czas obliczeń.

1.3. Automaty komórkowe a modelowanie zjawisk mikrostrukturalnych

Automaty komórkowe nie są jedyną metodą modelowania mikrostruktury materiałów, zachodzących w nich procesów i zmian wpływających na ich własności. Oprócz najprostszej metody Woronoja (Voronoi 1908), pozwalającej uzyskać mikrostrukturę, ale niemożliwiającej modelowania żadnych zjawisk w niej zachodzących, metody

Monte Carlo Potts, którą można zaliczyć do jednego z rodzaju automatów komórkowych z asynchronicznym losowym uaktualnieniem stanu komórek i wprowadzeniem lokalnych reguł uwzględniających minimalizację funkcji mającej często sens energetyczny, stosuje się również metodę „vertex” i bardzo zbliżoną do niej metodę śledzenia frontu (*front tracking*), a także metody pól fazowych (*phase-field*) i multifazowych (*multi-phase-field*). Istnieją też modele wykorzystujące metody elementów skończonych lub różnic skończonych. Nie ma jednak jednej metody odpowiedniej do stosowania we wszystkich przypadkach oraz nie zawsze zalety i wady poszczególnych sposobów postępowania są wyłącznie obiektywne, przeciwnie – często są postrzegane subiektywnie, a nawet tendencyjnie. Podobnie jak nie można jednoznacznie odpowiedzieć na pytanie, co jest lepsze: marynarka czy lotnictwo. Zawsze można znaleźć obszary zastosowań, w których dana metoda będzie lepsza i bardziej skuteczna. Dlatego chciałbym się skupić wyłącznie na automatach komórkowych, które uważam za bardzo przydatne i uniwersalne narzędzie do modelowania zjawisk mikrostrukturalnych.

Rozpoczęcie pracy z dowolną metodą, w tym z automatami komórkowymi, powoduje pojawienie się wielu problemów, które często pozostają ukryte, dopóki nie zostaną przez tego, kto podjął się danego zadania, sobie uzmysłowione i sformułowane. Następnie powinny one zostać sformalizowane i rozwiązane. W ten sposób zanikają wady i ograniczenia metody, zwiększa się jej adekwatność, skuteczność, a tym samym zakres możliwych zastosowań.

Czynnikami motywującymi do doskonalenia modelu zawsze są cel, któremu ów model ma służyć, i wyniki, które chcemy uzyskać. Ten cel często wyznacza strukturę i środki, które powinny zostać w danym przypadku zastosowane. Cel może być globalny, formułowany jako stworzenie uniwersalnego narzędzia do modelowania zmian mikrostruktury w różnych procesach: odlewania, przeróbki plastycznej, obróbki cieplnej itd. Może to być również cel lokalny, szczegółowy, na przykład odzwierciedlenie rzeczywistej kinetyki rekrytalizacji lub wyznaczenie właściwości materiału. W takim przypadku, określając wybrane zagadnienia, zjawiska mikrostrukturalne i procesy technologiczne, które chce się zasymulować, można przedstawić ogólny schemat blokowy modelu (rys. 1.1). Podstawą tego modelu są automaty komórkowe, natomiast jego wierzchołkiem – zestaw procesów, w których będzie zasymulowany rozwój mikrostruktury. Między nimi znajdują się modele zjawisk mikrostrukturalnych, które je ze sobą łączą. Model dopełnia baza materiałowa.

Opracowanie modelu również składa się z kilku etapów. Rozpoczyna się ono od podstawy, czyli od automatów komórkowych. Dalej opracowuje się modele poszczególnych zjawisk i na końcu odpowiedni proces technologiczny. Opracowanie modelu przypomina budowę domu. Na fundament (automaty komórkowe) nakładane są kolejne bloki (zjawiska), które tworzą ściany, następnie wszystko zostaje przykryte dachem (proces technologiczny), a na koniec wykańcza się budowlę odpowiednim materiałem (baza materiałowa). Modelując kolejne procesy technologiczne, pozostaje dobierać odpowiednie, istniejące już „klocki” i „materiały”.

Przedstawione na rysunku procesy zachodzące w materiałach można podzielić na dwie grupy. Pierwsza z nich kształtuje pierwotną strukturę i do niej zaliczamy rzeczywiste

zjawisko krystalizacji (krzepnięcia), czyli zmiany stanu skupienia materiału i „nierze-
czywisty” proces wytworzenia początkowej mikrostruktury. Zarówno pierwsze, jak
i drugie zjawisko może służyć do dalszego modelowania kolejnych, które są zaliczane
do drugiej grupy – czyli modelowanie zjawisk tej grupy, w odróżnieniu od zjawisk gru-
py pierwszej, wymaga istnienia początkowej mikrostruktury. Do drugiej grupy zalicza-
my więc rekrytalizację, przemiany fazowe i rozdrobnienie mikrostruktury.



Rys. 1.1. Struktura modelu rozwoju mikrostruktury w procesach technologicznych

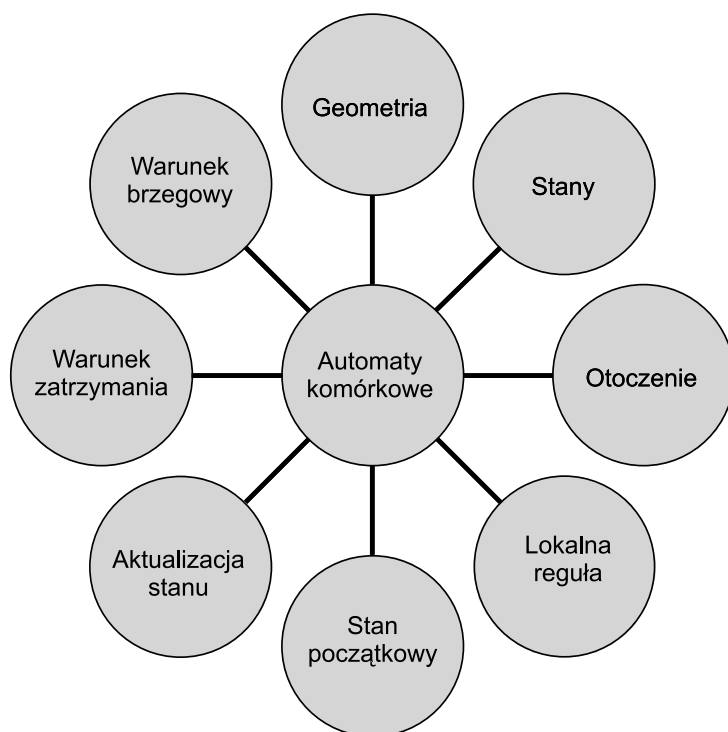
W czterech z pięciu przedstawionych podstawowych zjawisk można wyróżnić dwa elementarne procesy: zarodkowanie i rozrost ziarna (kryształu, krystalitów, nowej fazy), czyli mogą one być modelowane za pomocy tego samego automatu komórkowego (z uwzględnieniem odpowiednich szczegółowych modyfikacji). Symulacja zarodkowania w automatach komórkowych polega głównie na wyznaczeniu komórki, od której rozpoczyna się rozrost. Zarodkowanie jest w zasadzie operacją zewnętrzną w stosunku do automatu komórkowego. Wyznaczenie lokalizacji zarodka jest ważnym elementem modelu, istotnie wpływa na przebieg i końcowy wynik modelowania, jednak zarodkowanie nie powoduje żadnych zasadniczych problemów związanych bezpośrednio z automatami komórkowymi. Natomiast rozrost ziarna doprowadza do powstania kilku zasadniczych problemów, które istotnie ograniczają zakres zastosowań automatów komórkowych.

1.4. Automaty komórkowe a problemy związane z ich stosowaniem

Analizując przedstawione na początku niniejszego rozdziału podstawowe cechy automatów komórkowych w aspekcie stanowiącym istotę niniejszej pracy, a mianowicie symulacji rozrostu ziarna, można opracować schemat podstawowych cech automatów komórkowych przedstawiony na rysunku 1.2.

Przy tworzeniu automatów komórkowych należy zatem w mniejszym lub większym stopniu uwzględnić następujące elementy:

- geometrię,
- warunki brzegowe,
- otoczenie,
- stany komórek,
- stan początkowy,
- lokalną regułę,
- aktualizację stanów,
- warunek zatrzymania.



Rys. 1.2. Podstawowe cechy i parametry automatów komórkowych

Geometria

Niektóre parametry geometrii są narzucone przez modelowane zagadnienie. Przestrzeń powinna być trójwymiarowa (czasami, jako wyjątek, dopuszczalna jest przestrzeń dwuwymiarowa). Rozmiary i kształt przestrzeni podczas symulacji mogą się zmieniać (w szczególności jeżeli modelowano odkształcenie materiału). Siatka powin-

na być skończona i najlepiej regularna. Te parametry oraz właściwości automatów od razu powodują powstanie problemów, które należy rozwiązać.

Problem pierwszy: dyskretność przestrzeni. Nałożenie siatki, czyli dyskretyzacja przestrzeni, wprowadza niedokładności, które jednak można zredukować, zmniejszając rozmiary komórki. Zmniejszając rozmiary komórki, można z kolei doprowadzić do sytuacji, kiedy rozdzielczość staje się wystarczająca, żeby można było pominąć nieciągłość przestrzeni. Dodatkowym utrudniającym cały proces czynnikiem jest zniekształcenie przestrzeni przy uwzględnieniu rzeczywistego odkształcenia w procesach przeróbki plastycznej. Rozwiązanie tego problemu opiera się na dwu współzależnych działaniach. Pierwsze to zwiększenie liczby komórek, drugie – reorganizacja przestrzeni komórkowej, która z kolei polega na zmianie rozmiarów i kształtu przestrzeni modelowej oraz zastąpieniu siatki zniekształconej przez inną, mniej zniekształconą lub niezdeformowaną siatkę. Możliwość zwiększenia liczby komórek jest często ograniczona albo przez objętość pamięci, albo też przez wymagany lub sensowny czas obliczeń. Wynika z tego kolejny problem: wymiary przestrzeni – objętość pamięci – czas obliczeń. Reorganizacja przestrzeni nie powoduje dodatkowych problemów, jednak komplikuje się jeszcze z powodu innych czynników, które są rozpatrywane poniżej.

Problem drugi: ograniczoność przestrzeni. Ograniczoność przestrzeni i wykorzystanie siatki skończonej powoduje wprowadzenie odpowiednich warunków brzegowych. Podejścia do rozwiązania tego problemu zostały krótko opisane poniżej, przy opisie parametru „warunki brzegowe”, i bardziej szczegółowo w rozdziale 3.

Problem trzeci: wymiary przestrzeni – objętość pamięci – czas obliczeń. Parametry geometrii zasadniczo wpływają na obliczenia. Rozmiary przestrzeni modelowej (wyrażone w jednostkach długości) decydują o tym, jak duży obszar może być zamodelowany. Na ten obszar jest nałożona regularna siatka. Liczba elementów jest parametrem podstawowym, który wpływa na trzy zasadnicze wskaźniki modeli opartych na automatach komórkowych: rozdzielczość (dokładność), objętość pamięci i czas obliczeń. Przy niewielkiej objętości pamięci liczba elementów może być istotnie ograniczona, co rzutuje na rozdzielczość lub rozmiary przestrzeni. Natomiast wykorzystanie w pełni dużej pamięci podręcznej (4–8 GB), czyli wielkiej liczby komórek, często prowadzi do znacznego wydłużenia czasu obliczeń, ponieważ jest on przynajmniej liniowo zależny od liczby elementów. Pokonanie tak zwanego przekleństwa wymiarowości jest niemożliwe i jest to naturalne ograniczenie. Czas obliczeń zależy od dwóch czynników: liczby kroków czasowych i czasu obliczeń w jednym kroku. Liczba kroków często (ale nie zawsze) wprost zależy od liniowego wymiaru przestrzeni komórkowej (liczonej w komórkach). Czas obliczeń w jednym kroku zazwyczaj liniowo zależy od liczby komórek, czyli w przestrzeni trójwymiarowej jest sześcianiem wymiaru przestrzeni komórkowej (a w dwuwymiarowej – kwadratem). Wówczas przejście od dwuwymiarowej przestrzeni 1000×1000 komórek do przestrzeni trójwymiarowej powoduje albo zmniejszenie dokładności i rozdzielczości przy zachowaniu tej samej liczby komórek ($100 \times 100 \times 100$), albo istotne wydłużenie czasu obliczeń (ok. 3000 razy) przy zachowaniu tego samego wymiaru liniowego przestrzeni ($1000 \times 1000 \times 1000$). Zmniejszyć

wpływ „przekleństwa wymiarowości” próbowano na kilka sposobów. Problem ten został omówiony dokładnie w rozdziale 4, tam też podano jego rozwiązanie według autora niniejszej pracy.

Warunki brzegowe

Wprowadzenie warunków brzegowych jest narzucone ograniczonością przestrzeni modelowej i nałożeniem skończonej siatki. Adekwatne warunki brzegowe powinny spełniać jedno podstawowe wymaganie: powinny wyeliminować efekt brzegu, co oznacza, że wszystkie elementy (komórki) powinny być w tych samych warunkach równoważne, niezależnie od tego, czy znajdują się w środku przestrzeni, czy na jej granicy. Są dwa rozwiązania tego problemu. Pierwszy to wyeliminowanie wpływu zewnętrznych obszarów na automaty komórkowe, drugi – zrównoważenie wpływu automatu komórkowego na zewnątrz obszary i wpływu zewnętrznego obszaru na automaty komórkowe. W pierwszym wypadku automaty zamykają się na siebie (są izolowane od obszarów zewnętrznych), w drugim pozostają otwarte, a narzucone warunki brzegowe są bardziej skomplikowane. Zatem warunki brzegowe można rozpatrywać jako środki rozwiązywania problemu drugiego: ograniczoności przestrzeni. Szczegółowo warunki brzegowe omówiono w rozdziale 3.

Otoczenie

Otoczenie jest naturalną, podstawową cechą automatów komórkowych. Jest to element niezbędny, którego nie można pominąć. Otoczenie, jak już wspomniano, nosi cechy geometryczne i informacyjne. Geometryczna natura otoczenia jest związana z nałożeniem siatki i przypisaniem elementom siatki statusu komórek. Wyłonienie otoczenia z całej przestrzeni ma dwa przeciwne skutki. Z jednej strony ograniczenie strefy wpływów prowadzi do skrócenia czasu obróbki informacji i zmniejszenia całkowitego czasu obliczeń. Z drugiej strony, zmniejszenie promienia otoczenia prowadzi do częściowej utraty informacji i, co bardziej istotne, do zwiększenia anizotropii przestrzeni, co z kolei wywołuje kolejny problem.

Problem czwarty: izotropia przestrzeni. Wprowadzenie prostego sąsiedztwa krótkiego zasięgu powoduje powstanie małej liczby uprzywilejowanych kierunków rozpowszechniania informacji i odpowiednio ograniczonej liczby kierunków rozrostu ziaren, czyli prowadzi do anizotropii przestrzeni. Na przykład w otoczeniu von Neumanna w przestrzeni dwuwymiarowej każda komórka ma cztery sąsiednie komórki, z którymi ma wspólne boki, zatem występują w nim tylko cztery główne kierunki wzrostu i naturalnym tego wynikiem jest rozrost ziaren w postaci kwadratów zorientowanych swoimi przekątnymi zgodnie z ułożeniem siatki. Ten problem wymaga szczególnej uwagi i jego rozwiązaniu jest poświęcony rozdział 2 niniejszej pracy. Rozwiązać ten problem można, stosując różne sąsiedztwa i (lub) wprowadzając elementy sąsiedztwa długiego zasięgu.

Jeżeli geometryczne cechy sąsiedztwa ukierunkowują przepływ informacji, ograniczając jej równomierne rozpowszechnienie, to niezależnie od geometrii cechy można zaliczyć do sąsiedztwa dalekiego zasięgu. Pozwalają one poprawić izotropowość

przestrzeni, zmniejszyć wykorzystanie pamięci i istotnie przyspieszyć obliczenia. Do sąsiedztwa długiego zasięgu można zaliczyć takie nadstruktury, jak zgrupowanie komórek w różny sposób (często niegeometryczny). Jako przykład można tu podać ziarno, które zawiera informację dotyczącą wszystkich należących do niego komórek: m.in. o orientacji krystalograficznej, miejscu zarodkowania, gęstości dyslokacji, albo też grupę komórek przypisanych do różnych list, na przykład na granicy ziarna, co skraca czas poszukiwania miejsc zarodkowania i przyspiesza obliczenia. Należy również pamiętać o tym, że istnieje otoczenie wejściowe i wyjściowe, a wykorzystanie odpowiedniego otoczenia pozwala zwiększyć skuteczność algorytmów.

Tak więc sąsiedztwo wprowadza ograniczenia i powoduje pojawienie się problemu izotropii, ale jednocześnie umożliwia jego rozwiązanie, a także wspomaga rozwiązywanie innych problemów.

Stany komórek

Stany komórek są podstawowym parametrem, na którym opierają się lokalne reguły i który pozwala adekwatnie modelować podstawowe zjawiska mikrostrukturalne. Najprostsze automaty posiadają tylko dwa stany, jednak można stosować i automaty wielostanowe. Mała liczba stanów upraszcza lokalne reguły, ale albo istotnie ogranicza możliwości automatów, albo przenosi ciężar na inne, niegłówne, dodatkowe zmienne, które stają się rozszerzeniem pojęcia stanu komórki. Zwiększenie liczby stanów komplikuje reguły przejścia, ale i wzbogaca je, poszerzając ich możliwości. Komórki znajdujące się w tym samym stanie można badać oddzielnie lub pomijać badanie komórek, skracając tym samym czas obliczeń.

Stan początkowy

Jak można wywnioskować z poprzednich rozdziałów, istnieją dwa rodzaje stanu początkowego. Niektóre zjawiska potrzebują już gotowej mikrostruktury, w innych punktem wyjścia jest stan niezakłócony, kiedy wszystkie komórki znajdują się w tym samym stanie początkowym. Modelowanie krystalizacji lub krzepnięcia rozpoczyna się bez utworzenia wstępnej mikrostruktury, natomiast pozostałe procesy wymagają początkowej mikrostruktury o wyznaczonych parametrach. Należy zatem opracować „pseudoprocess” kształtowania początkowej mikrostruktury, który może być prowadzony w oparciu o te same zasady.

Lokalna reguła

Lokalna reguła opiera się głównie na stanach komórek, ale nie ogranicza się tylko do nich. Reguły powinny uwzględniać parametry procesowe. Również od reguł zależy skuteczność całego modelu oraz czas obliczeń. To przede wszystkim reguła wykorzystuje otoczenie wejściowe i właśnie tak działają klasyczne automaty komórkowe – wykonują dużą liczbę zbędnych operacji i dlatego bardzo długo liczą. Wykorzystanie w algorytmach również otoczenia wyjściowego pozwoliło opracować frontalne automaty komórkowe, dzięki którym możliwe było przyspieszenie obliczeń przynajmniej kilkaset razy. Frontalne automaty komórkowe opisano w rozdziale 4.

Aktualizacja stanów

Powszechnie przyjęta synchroniczna aktualizacja stanów ma jeden bardzo istotny atut: modelowanie łatwo związać z czasem rzeczywistym, natomiast czas obliczeń nigdy nie jest krótszy niż przy aktualizacji asynchronicznej. Za wadę asynchronicznej aktualizacji uważano jej niekontrolowany wpływ na sąsiednie komórki, który może być „przedwczesny”. Jednak wynika z tego, że wprowadzenie aktualizacji asynchronicznej, którą można by było sztywno związać z czasem rzeczywistym i która eliminuje „przedwczesny” wpływ na swoje otoczenie, pozwala przyspieszyć obliczenia. Taka aktualizacja została opracowana i wykorzystana, jako jeden z elementów, we frontalnych automatach komórkowych. Aktualizacja ta jest ściśle związana z krokiem obliczeniowym, z krokiem czasowym, a zatem dobór długości kroków pozwala zmniejszyć ich liczbę i skrócić całkowity czas obliczeń. We frontalnych automatach komórkowych zastosowano zmienny krok czasowy adaptacyjnie dopasowywany do bieżących warunków modelowych.

Warunek zatrzymania

Głównym kryterium końca obliczeń jest czas modelowy, jednak można wprowadzić warunek bardziej skomplikowany – osiągnięcie stanu ustalonego, czyli takiego, kiedy w całej przestrzeni nie będą już zachodzić żadne zmiany lub wszystkie komórki osiągną stan końcowy.

Podsumowując ten krótki przegląd charakterystyk automatów komórkowych, można jeszcze raz wymienić główne problemy, na które należy zwrócić szczególną uwagę:

- dyskretność przestrzeni,
- ograniczoność przestrzeni,
- wymiary przestrzeni – objętość pamięci – czas obliczeń,
- izotropia przestrzeni.

Rozwiązaniu tych problemów podczas opracowywania frontalnych automatów komórkowych będą poświęcone kolejne trzy rozdziały monografii.

2. Algorytmy rozrostu ziarna

Większość zjawisk, które planowano modelować, składa się z dwóch etapów: zarodkowania i rozrostu ziaren. Zarodkowanie jest ważnym etapem, jednak w stosunku do automatów komórkowych jest to zazwyczaj operacja zewnętrzna, a rola automatów sprowadza się jedynie do znalezienia odpowiedniego miejsca zarodkowania. Szczególnie istotna jest kwestia zarodkowania, gdy mamy do czynienia z modelowaniem rzeczywistych zjawisk, natomiast tworząc pierwszy układ do modelowania początkowej struktury, miejsce zarodkowania można narzucić lub zastosować losowy sposób jego wyboru. Wówczas można skupić się na etapie drugim – rozroście ziarna – i rozpocząć modelowanie od rozrostu pojedynczego ziarna. Jest kilka różnych podejść do modelowania rozrostu ziarna, niektóre będą przeanalizowane w tym rozdziale. Rozpatrywanie algorytmów rozpocznie od najprostszego wariantu, który będzie uzupełniany o warianty coraz bardziej skomplikowane, uwzględniające więcej warunków i pozwalające uzyskać wymagany wynik. W niniejszym rozdziale zostaną rozpatrzone różne warianty sąsiedztwa, zastosowanie losowych reguł, wprowadzenie sterowania rozrostem, zostaną w nim też opisane wykorzystane rozwiązania problemu izotropii oraz przedstawione algorytmy uzyskania niektórych wybranych kształtów rosnących ziaren i przykłady mikrostruktur. Większość przedstawionych rozwiązań dotyczy przestrzeni dwuwymiarowej, ale zawierają wskazówki umożliwiające przejście do przestrzeni trójwymiarowej. Rozdział rozpoczyna się od omówienia klasycznego zagadnienia: rozrostu pojedynczego ziarna w dwuwymiarowej przestrzeni automatów komórkowych z najprostszą regułą przejścia.

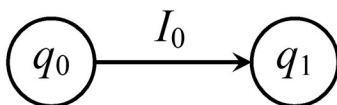
2.1. Klasyczny algorytm rozrostu ziarna

Klasyczny algorytm rozrostu ziaren jest bardzo prosty i szybki, opiera się na deterministycznej regule, która w wyznaczonej przestrzeni automatów komórkowych pozwala uzyskać tylko jeden kształt rosnącego ziarna. Główną cechą takiego algorytmu jest prostota reguły przejścia. Kształt zależy wówczas tylko od wybranego otoczenia. Na rysunku 2.1 widoczna jest jedna komórka główna (oznaczona szarym kolorem) i osiem komórek w jej sąsiedztwie, które nazwano otoczeniem lub sąsiedztwem Moore'a.

Przy realizacji klasycznego algorytmu można ograniczyć się do najprostszego automatu, który jest określany tylko jednym parametrem, a mianowicie stanem komórki. Wówczas komórka przyjmuje tylko jeden z dwóch stanów: początkowy i końcowy. Różne stany można opisać liczbami (0 albo 1) lub zmienną logiczną (*true* – *false*). Wtedy automat komórki można schematycznie przedstawić w postaci grafu jak na rysunku 2.2, gdzie q_0 – stan początkowy, q_1 – stan końcowy, a I_0 – warunek, którego spełnienie powoduje przejście w kierunku wskazanym przez strzałkę.

6	7	8
4		5
1	2	3

Rys. 2.1. Komórka i jej otoczenie



Rys. 2.2. Automat komórki oparty na zastosowaniu algorytmu rozrostu ziaren

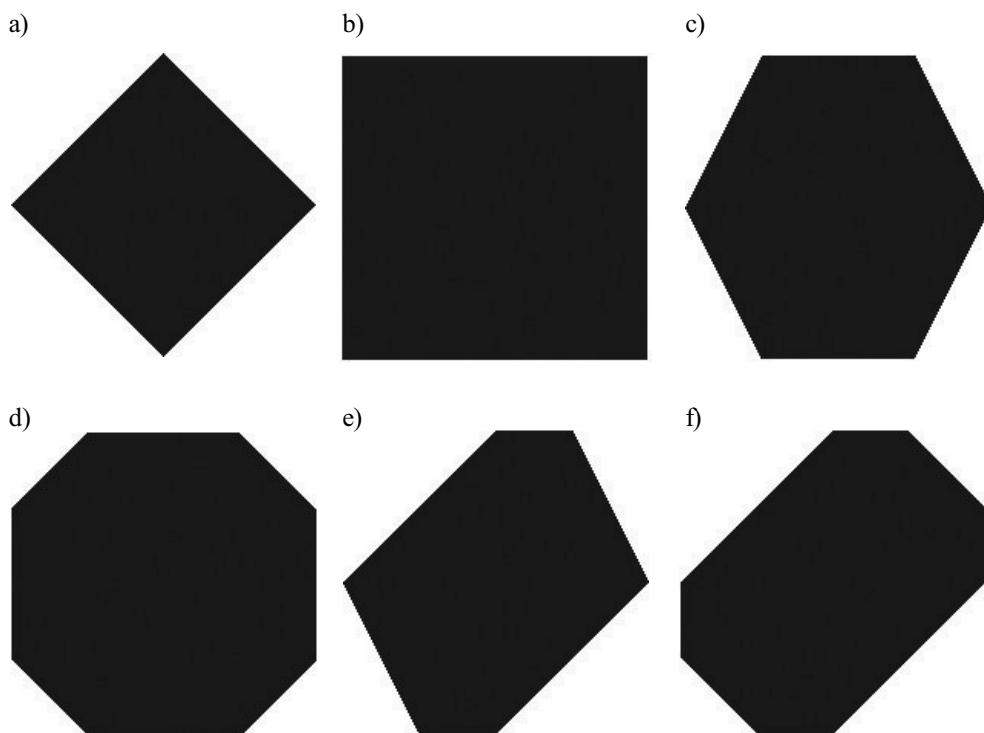
Położenie automatu (komórki) w przestrzeni dwuwymiarowej jest jego podstawową charakterystyką geometryczną i może być określone za pomocą dwóch współrzędnych (kartezjańskich). Współrzędne te mogą być przypisane do modelowej przestrzeni lub do automatów komórkowych. Pierwsze wskazują na położenie w przestrzeni wyrażone w jednostkach długości, drugie określają miejsce w regularnej siatce lub tabeli i wskazują odpowiednią kolumnę i wiersz. Wówczas współrzędne mogą być podawane w komórkach (jako numer wiersza i kolumny) i są liczbami całkowitymi, a położenie można oznaczyć za pomocą dwóch indeksów. Na przykład położenie centralnej, szarej komórki na rysunku 2.1 może być wskazane przez indeksy i (w kierunku poziomym) i j (w kierunku pionowym), czyli c_{ij} . Pozostałe komórki wówczas będą oznaczone następująco: $c_1 = c_{i-1,j-1}$, $c_2 = c_{i,j-1}$, ..., $c_8 = c_{i+1,j+1}$.

Zachowanie automatów komórkowych wyznacza lokalna reguła. W celu uzyskania rozrostu we wszystkich możliwych kierunkach należy ustalić odpowiednią regułę. Można ją sformułować za pomocą jednego zdania: komórka zmienia swój stan (przechodzi ze stanu początkowego q_0 w stan końcowy q_1) wtedy i tylko wtedy, gdy w jej otoczeniu znajduje się przynajmniej jedna komórka w stanie końcowym q_1 .

W przypadku otoczenia Moore’a tę regułę można zapisać następująco:

$$I_0 = c_1 \vee c_2 \vee c_3 \vee c_4 \vee c_5 \vee c_6 \vee c_7 \vee c_8 \quad (2.1)$$

Na rysunku 2.3 pokazano wyniki symulacji po stu iteracjach przy zastosowaniu różnego sąsiedztwa. Sąsiedztwo von Neumanna (komórki c_2 , c_4 , c_5 i c_7) i Moore’a (komórki $c_1 \div c_8$) tworzą kwadrat: w pierwszym przypadku przekątne kwadratu są zorientowane wzdłuż osi przestrzeni (rys. 2.3a), w przypadku drugim wzdłuż osi skierowane są boki kwadratu (rys. 2.3b). Przy zastosowaniu sąsiedztwa heksagonalnego uzyskano kształt sześciokątny (rys. 2.3c). Ośmiokątny kształt (rys. 2.3d) można uzyskać, stosując sąsiedztwo pseudoheksagonalne.



Rys. 2.3. Przykłady symulacji po stu iteracjach przy zastosowaniu różnego sąsiedztwa:

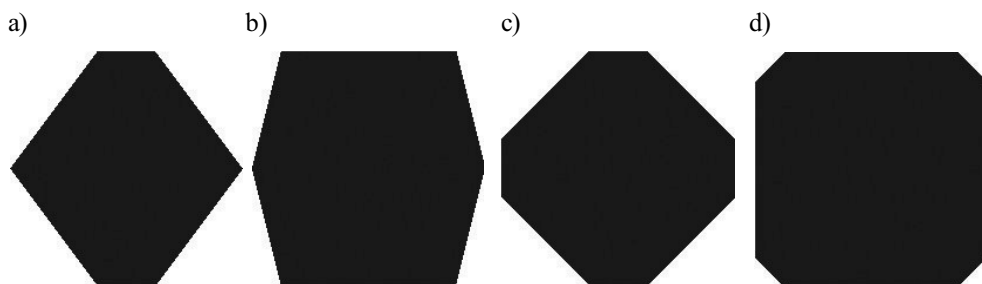
- a) von Neumanna; b) Moore’a; c) heksagonalnego; d) pseudoheksagonalnego;
- e) niesymetrycznego pentagonalnego; f) niesymetrycznego pseudopentagonalnego

Sąsiedztwo heksagonalne na siatce kwadratowej może być zrealizowane w różny sposób, również w różny sposób można zadać sąsiedztwo dla parzystych i nieparzystych numerów wierszy (lub kolumn). Na przykład usuwano na przemian po dwa dowolne narożniki (c_1 , c_3 , c_5 lub c_8) w sąsiedztwie Moore’a lub wycinano jeden i trzy narożniki bądź też cztery albo żadnego. Ten ostatni wariant odpowiada stosowaniu sąsiedztwa

von Neumanna i Moore'a na przemian w przypadku parzystych i nieparzystych numerów wierszy. Graficznie każdy z tych wariantów sąsiedztwa heksagonalnego skutkuje takim samym kształtem rosnącego ziarna. Na przykład, zadanie różnego sąsiedztwa dla parzystych i nieparzystych numerów wierszy daje kształt pokazany na rysunku 2.3c. Zadawanie różnego sąsiedztwa dla kolumn (a nie dla wierszy) spowoduje obrót sześciokąta o 90° . Sąsiedztwo pseudoheksagonalne różni się od heksagonalnego tym, że zmieniano je nie w zależności od położenia w przestrzeni automatów, a zależnie od numeru kroku czasowego, czyli wybierano odpowiednie sąsiedztwo dla wszystkich komórek w iteracjach o parzystych i nieparzystych numerach na przemian. Wówczas zachowana pozostaje symetria względem obydwu osi przestrzeni i powstaje ośmiokąt (rys. 2.3d).

Pozostałe dwa przykłady (rys. 2.3e i f) odnoszą się do kombinacji różnych sąsiedztw lub sąsiedztwa pentagonalnego, czyli takiego, w którym każda komórka ma średnio pięciu sąsiadów. Różnica między ziarnem przedstawionym na rysunku 2.3e a tym widocznym na rysunku 2.3f również polega na zależności od położenia w przestrzeni i od numeru kroku czasowego. Na tych rysunkach pokazano sąsiedztwa niesymetryczne.

Na rysunku 2.4 uwidoczniono natomiast przykłady symetrycznego sąsiedztwa pentagonalnego (rys. 2.4a), heptagonalnego (rys. 2.4b) i ich pseudoodpowiedników (rys. 2.4c i d).



Rys. 2.4. Przykłady symulacji po stu iteracjach przy zastosowaniu różnego sąsiedztwa:

- a) symetrycznego pentagonalnego; b) symetrycznego heptagonalnego;
- c) symetrycznego pseudopentagonalnego; d) symetrycznego pseudoheptagonalnego

Analizując rysunki 2.3 i 2.4, można zauważyć, że kształt uzyskanych figur na ogół nie odpowiada nazwie sąsiedztwa. Uzyskanie przestrzeni pokrytej wyłącznie pięcio- lub siedmiokątami jest fizycznie niemożliwe. Natomiast sąsiedztwo heksagonalne można uzyskać w kilku wariantach przestrzeni: siatki złożonej z sześciokątów lub trójkątów, albo, jak pokazano na rysunku 2.3c, z kwadratów. Przy zastosowaniu kwadratowych komórek sąsiednie wiersze (lub kolumny) mogą być przesunięte o pół długości boku komórki, jednak i bez takiego przesunięcia zostanie uzyskany taki sam kształt, lecz z maksymalną niedokładnością, nie większą niż połowa długości boku komórki. Należy przy tym podkreślić, że wszystkie te kształty mogą być uzyskane w ramach jednego

rodzaju sąsiedztwa – sąsiedztwa Moore’a, i tylko tego. Sąsiedztwo Moore’a pozwala zasymulować wszystkie inne sąsiedztwa, za to inne sąsiedztwa nie pozwalają w prosty sposób uzyskać efektu sąsiedztwa Moore’a, a więc sąsiedztwo Moore’a jest najbardziej ogólnym sąsiedztwem w przestrzeni dwuwymiarowej, z promieniem równym 1. Natomiast to, co wcześniej zostało nazwane wariantami sąsiedztwa, możemy teraz określić mianem zastosowania różnych reguł przejścia – w takim przypadku nie będziemy już mówić o najprostszej regule przejścia, która została podana na początku, a mianowicie: komórka zmienia swój stan (przechodzi ze stanu q_0 w stan q_1) wtedy i tylko wtedy, gdy w jej otoczeniu znajduje się przynajmniej jedna komórka w stanie 1. Teraz przejście może być uzależnione od współrzędnych przestrzennych i czasu. Dalej we wszystkich rozważaniach zastosowano sąsiedztwo Moore’a, natomiast wprowadzenie sąsiedztwa von Neumanna bądź heksagonalnego zrealizowano za pomocą odpowiednich reguł przejścia. Dla przykładu odpowiednie reguły można zapisać w następujący sposób:

$$I_0 = c_2 \vee c_4 \vee c_5 \vee c_7 \quad (2.2)$$

$$I_0 = (c_2 \vee c_4 \vee c_5 \vee c_7) \vee jj \wedge (c_1 \vee c_3 \vee c_6 \vee c_8) \quad \forall jj = 2n \quad (2.3)$$

$$I_0 = (c_2 \vee c_4 \vee c_5 \vee c_7) \vee kk \wedge (c_1 \vee c_3 \vee c_6 \vee c_8) \quad \forall kk = 2n \quad (2.4)$$

Wzór (2.2) odpowiada otoczeniu von Neumanna, wzór (2.3) – sąsiedztwu heksagonalnemu, wzór (2.4) – pseudoheksagonalnemu, $jj = 2n$ i $kk = 2n$ oznaczają odpowiednio parzysty numer kolumny i kroku czasowego.

Podsumowując przedstawione wyniki, należy podkreślić, że liczba możliwych kształtów rosnących ziaren nie jest duża, boki ziaren są zawsze prostoliniowe, a naroża ostre. Figury przedstawione na rysunkach 2.3 i 2.4 nie wyczerpują wszystkich możliwych wariantów, jednak ich liczba jest ograniczona. Wśród możliwych kształtów ziaren brak okrągłego, ponieważ sąsiedztwo wprowadza do przestrzeni bardzo istotną anizotropię. Uzyskanie kształtu okrągłego jest jednym z niezastąpionych testów potwierdzających izotropię i do jego uzyskania dąży większość badaczy, często zapominając o tym, że głównym celem jest nie kształt, a izotropia przestrzeni. Uzyskanie okrągłego kształtu za pomocą opisanego wyżej algorytmu jest niemożliwe. Drugim testem potwierdzającym anizotropię rozrostu w dowolnym kierunku w izotropowej przestrzeni jest uzyskanie rosnącego ziarna w kształcie kwadratu obróconego o dowolny kąt w stosunku do osi przestrzeni. Kwadrat jest najprostszą figurą, którą można otrzymać za pomocą automatów komórkowych, ale nie ma możliwości, by go obrócić przy zastosowaniu przedstawionego algorytmu.

Główną przyczyną występowania ograniczeń jest niemożność sterowania prędkością rozrostu przy użyciu algorytmu klasycznego. Pociąga to za sobą konieczność opracowania i zastosowania innych, bardziej złożonych algorytmów. Można przy tym posłużyć się sposobem stosowanym w metodzie Monte Carlo. Polega on na sprawdzaniu losowo wybranych komórek przestrzeni na obecność w ich sąsiedztwie rosnącego ziarna.

W razie stwierdzenia, że taka sytuacja zachodzi, dana komórka jest do tego ziarna przyłączana. Tak więc różnica polega na tym, że w automatach komórkowych jeden krok obejmuje sprawdzenie wszystkich komórek przestrzeni (lub wszystkich komórek w pobliżu rosnącego ziarna) oraz zmianę stanu po przeglądzie i wyznaczenie reguł przejścia dla wszystkich komórek, natomiast w metodzie Monte Carlo jeden krok odpowiada sprawdzeniu jednej przypadkowej komórki (lub wyznaczonej liczby komórek). Ten sam wynik można uzyskać również za pomocą automatów komórkowych, stosując asynchroniczna, losową aktualizację stanu komórek. Algorytmy stosowane w metodzie Monte Carlo można znaleźć w odpowiedniej literaturze i nie będą one tu rozpatrywane szczegółowo. Jednak wspomniany algorytm przy uwzględnieniu kilku czynników pozwala otrzymać kształt zbliżony do okrągłego. Metoda Monte Carlo wpłynęła na rozwój automatów komórkowych, wzbogacając je o losowe algorytmy rozrostu, z których niektóre przedstawiono w kolejnym podrozdziale.

2.2. Algorytm rozrostu ziarna z zastosowaniem losowych reguł

Klasyczny algorytm rozrostu ziaren, będąc najprostszym i najszybszym deterministycznym algorytmem, nie pozwala uzyskać dowolnego kształtu ani sterować prędkością rozrostu. Jak już wspomniano, jednym z testów jest próba uzyskania okrągłego kształtu. Sukcesy w otrzymywaniu kształtu zbliżonego do okrągłego za pomocą metody Monte Carlo sprawiły, że w automatach komórkowych zaczęto stosować reguły losowe. Polega to na tym, że uwzględnia się czynnik losowy, który wprowadza przypadkowość przejścia przy spełnieniu reguły klasycznej. Wówczas reguła może być dopełniona ważnym czynnikiem, a mianowicie przejście zależy nie tylko od obecności w sąsiedztwie przynajmniej jednej komórki w stanie końcowym, ale również od liczby tych komórek. Jeżeli założymy, że prawdopodobieństwo przejścia przy obecności jednej komórki w stanie końcowym wynosi 0,5, to przy dwóch sąsiadach w stanie końcowym prawdopodobieństwo w zależności od przyjętej reguły wyniesie 0,75 lub 1,0. Teoretycznie powinno to zatem doprowadzić do wygładzania ostrych kątów.

Rozpatrzmy kilka wariantów zastosowania czynnika losowego.

Wariant pierwszy jest praktycznie identyczny ze swym odpowiednikiem w metodzie Monte Carlo. Sprawdzano wszystkie komórki, lecz reguły stosowano tylko względem komórek wybranych losowo. Regułę tę na przykład w przypadku otoczenia von Neumanna można zapisać w następujący sposób:

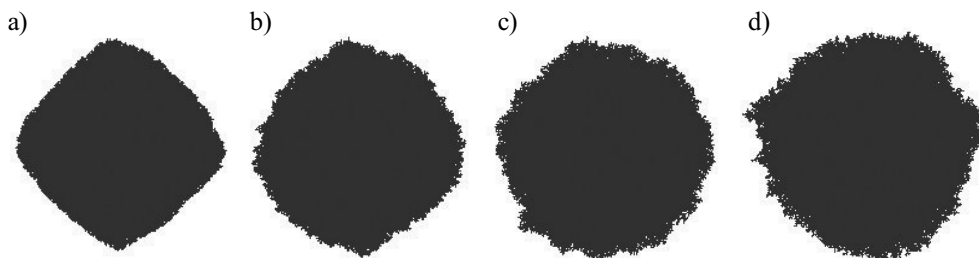
$$I_0 = p \wedge (c_2 \vee c_4 \vee c_5 \vee c_7) \quad (2.5)$$

gdzie p oznacza prawdopodobieństwo zastosowania reguły.

W tym przypadku prawdopodobieństwo nie zależy od liczby sąsiednich komórek już należących do rosnącego ziarna. Ten czynnik losowy może być zastosowany do dowolnego przedstawionego wyżej sąsiedztwa lub reguły.

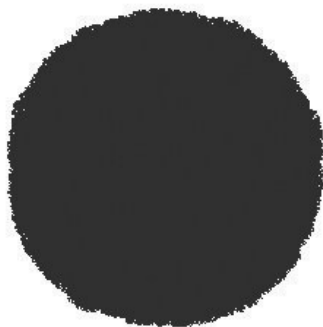
Na rysunku 2.5 pokazano wyniki symulacji w sąsiedztwie von Neumanna. Zmniejszono prawdopodobieństwo zastosowania reguły przejścia, zadając po kolei wartości p równym 0,7; 0,4; 0,1 i 0,05. W celu uzyskania porównywalnych wymiarów odpowiednio zwiększano liczbę kroków w symulacji. Przy dość wysokim prawdopodobieństwie pierwotny kształt pozostaje widoczny w kształcie końcowym, jednak można również zauważyć zaokrąglające działanie czynnika losowego. W miarę zmniejszania się prawdopodobieństwa kształt ziarna coraz bardziej zbliża się do okrągłego, jednak granica jest coraz mniej gładka, coraz mniej regularna.

Wyniki przedstawione na rysunku 2.5 pozwalają wysnuć wniosek, iż nawet w najuboższym otoczeniu von Neumanna można uzyskać kształt zbliżony do okrągłego. Jednak powstaje problem gładkości granic. Z jednej strony prawdopodobieństwo powinno być jak najmniejsze, żeby uzyskać okrągły kształt, z drugiej zaś strony – jak największe, żeby granica była gładka (oraz obliczenia trwały krócej). Więc należy dobrać taki kształt bazowy, którego zaokrąglenie wymaga jak największego prawdopodobieństwa. Wówczas lepiej byłoby wybrać regułę odpowiadającą sąsiedztwu pseudoheksagonalnemu lub heksagonalnemu. Rysunek 2.6 pokazuje wynik symulacji z sąsiedztwem pseudoheksagonalnym i prawdopodobieństwem $p = 0,3$.



Rys. 2.5. Przykłady symulacji rozrostu ziarna z wykorzystaniem sąsiedztwa von Neumanna z różnym prawdopodobieństwem zastosowania reguły (1.5):

a) $p_0 = 0,7$; b) $p_0 = 0,4$; c) $p_0 = 0,1$; d) $p_0 = 0,05$



Rys. 2.6. Wynik symulacji wzrostu ziarna w sąsiedztwie pseudoheksagonalnym z prawdopodobieństwem $p_0 = 0,3$

Wariant drugi różni się od pierwszego innym sposobem sprawdzania prawdopodobieństwa. Reguła przejścia w przypadku otoczenia von Neumanna jest zapisywana za pomocą następującego wzoru:

$$I_0 = (p_1 \wedge c_2) \vee (p_2 \wedge c_4) \vee (p_3 \wedge c_5) \vee (p_4 \wedge c_7) \quad (2.6)$$

W odróżnieniu od poprzedniego wariantu podczas wyznaczenia przejścia stosowano cztery liczby losowe zamiast jednej (Svyetlichnyy i Milenin 2005). Wówczas, jeżeli prawdopodobieństwa $p_1 \div p_4$ są równe (i równe p_0), sumaryczne prawdopodobieństwo przejścia będzie takie samo jak prawdopodobieństwo $p = p_0$ w wariancie pierwszym, kiedy w sąsiedztwie będzie się znajdować tylko jedna komórka. Jednak w obecności dwóch lub więcej komórek prawdopodobieństwo przejścia będzie większe i może być wyznaczone według zależności:

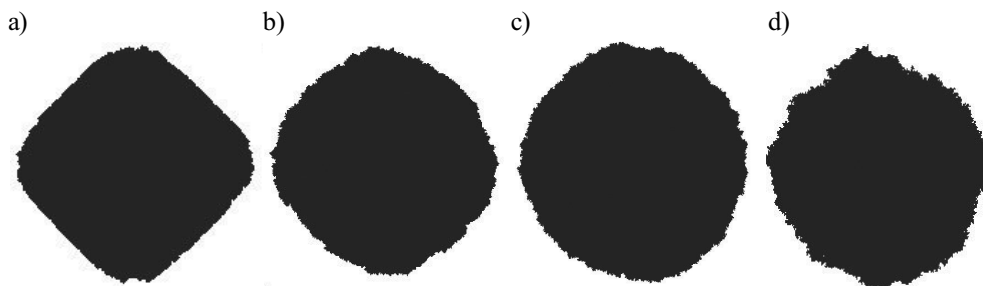
$$p_n = 1 - (1 - p_0)^n,$$

gdzie n – liczba komórek należących do rosnącego ziarna w otoczeniu sprawdzanej komórki.

W inny sposób regułę tę można zapisać następująco:

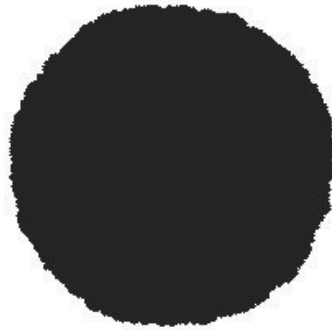
$$I_0 = p_n \quad (2.7)$$

Zwiększenie prawdopodobieństwa przejścia w obecności większej liczby komórek w procesie rozrostu ziarna jest energetycznie uzasadnione i działa stabilizująco na gładkość granicy. Wyniki symulacji w sąsiedztwie von Neumanna z prawdopodobieństwem p_0 równym 0,7; 0,4; 0,1 i 0,05 przedstawiono na rysunku 2.7 i w porównaniu z wynikami przedstawionymi na rysunku 2.5 uzyskano wyraźnie gładkie granice ziarna. Na rysunku 2.8 pokazany jest wynik symulacji z sąsiedztwem pseudoheksagonalnym z prawdopodobieństwem $p_0 = 0,3$.



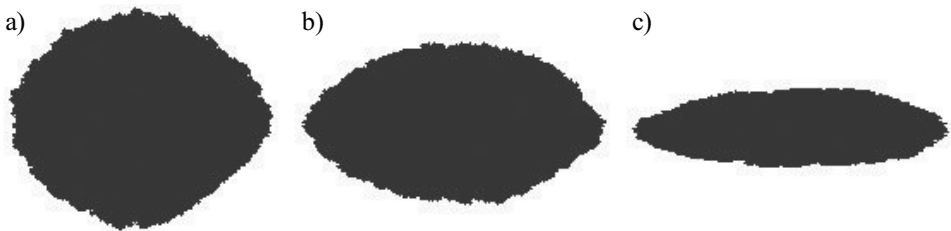
Rys. 2.7. Przykłady symulacji rozrostu ziarna z wykorzystaniem sąsiedztwa von Neumanna z różnym prawdopodobieństwem zastosowania reguły (1.6)–(1.7):

a) $p_0 = 0,7$; b) $p_0 = 0,4$; c) $p_0 = 0,1$; d) $p_0 = 0,05$



Rys. 2.8. Wynik symulacji rozrostu ziarna w sąsiedztwie pseudoheksagonalnym z prawdopodobieństwem $p_0 = 0,7$

Wprowadzenie prawdopodobieństwa dla każdego sąsiada, poza zwiększeniem prawdopodobieństwa przejścia komórki w inny stan w obecności większej liczby komórek w sąsiedztwie, pozwala w niewielkim stopniu sterować prędkością rozrostu w różnych kierunkach, a co za tym idzie – uzyskać większy zestaw kształtów.



Rys. 2.9. Przykłady symulacji rozrostu ziarna w otoczeniu von Neumanna z różnym prawdopodobieństwem w kierunku poziomym p_h i pionowym p_v :

a) $p_h = 0,6$, $p_v = 0,4$; b) $p_h = 0,8$, $p_v = 0,2$; c) $p_h = 0,9$, $p_v = 0,1$



Rys. 2.10. Przykłady symulacji rozrostu ziarna w otoczeniu Moore'a z różnym prawdopodobieństwem w kierunku pionowym, poziomym i po przekątnej

Na przykład na rysunku 2.9 przedstawiono „elipsy” o różnym stosunku długości osi, natomiast na rysunku 2.10 – „elipsy” obrócone o różny kąt.

Obrócenie elipsy osiągnięto, zadając różne prawdopodobieństwo dla różnych komórek w sąsiedztwie. Dzięki temu różni sąsiedzi mają różny wpływ na komórkę rozpatrywaną. Aby uzyskać bardziej wydłużony kształt elipsy, różnica prawdopodobieństw powinna być jak największa. Na rysunku 2.9 pokazano rozrost ziarna w otoczeniu von Neumanna z prawdopodobieństwem w kierunku poziomym p_h i pionowym p_v odpowiednio $p_h = 0,6$, $p_v = 0,4$ (a); $p_h = 0,8$, $p_v = 0,2$ (b) oraz $p_h = 0,9$, $p_v = 0,1$ (c). Obrót typu „elipsa” w otoczeniu von Neumanna nie jest możliwy. W otoczeniu Moore’a powinny być zadane różne prawdopodobieństwa dla czterech kierunków – poziomego, pionowego i wzdłuż dwóch przekątnych. Wówczas można uzyskać wyniki przedstawione na rysunku 2.10.

Można też uwzględnić wspólne działanie sąsiednich komórek w inny sposób. W poprzednim opisie przyjęto multiplikatywne prawo prawdopodobieństw, kiedy wpływ kilku komórek przedstawiono za pomocą wzoru: $p_n = 1 - (1 - p_1)(1 - p_2) \dots (1 - p_n)$. Inną zasadą, którą można tu zastosować, jest arytmetyczne sumowanie prawdopodobieństw, czyli: $p_n = p_1 + p_2 + \dots + p_n$. Oczywiście uzyskane wypadkowe prawdopodobieństwo p_n powinno być ograniczone wartością 1,0. Jest to warunek ostrzejszy, a jego wpływ na gładkość granic jest wyraźniejszy, chociaż uzasadnienie teoretyczne takiego warunku jest trudniejsze.

Jeszcze jednym wariantem losowych reguł przejścia jest wyznaczenie dla rosnącego ziarna kilku otoczeń i przypisanie prawdopodobieństwa wyboru każdemu z tych sąsiedztw. Jednak zastosowanie takich algorytmów w automatach komórkowych wydaje się raczej sztucznym zabiegiem niż praktycznym, przydatnym rozwiązaniem.

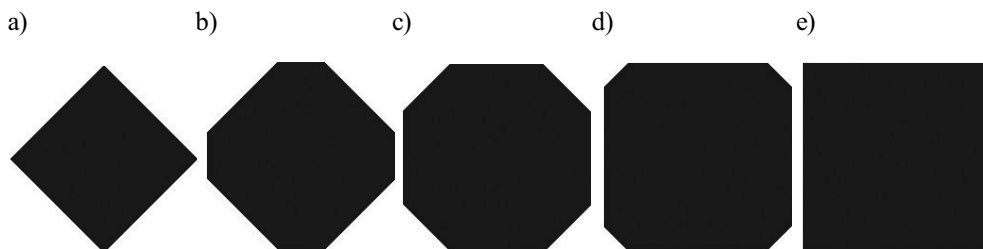
Podsumowując temat algorytmów z zastosowaniem reguł losowych, trzeba przyznać, że posiadają one pewne zalety, na przykład pozwalają uzyskać okrągły kształt rosnącego ziarna, kształt elipsy bardziej lub mniej wydłużony lub obróconej o praktycznie dowolny kąt. Jednak algorytmy te mają niewątpliwe wady, do których trzeba zaliczyć niemożność uzyskania gładkiej granicy dowolnego kształtu i ograniczoność sterowania prędkością rozrostu. Głównym mankamentem algorytmów tego rodzaju jest jednak to, że nie można za ich pomocą rozwiązać problemu anizotropii przestrzeni.

2.3. Algorytm rozrostu ziarna ze sterowaniem prędkością rozrostu

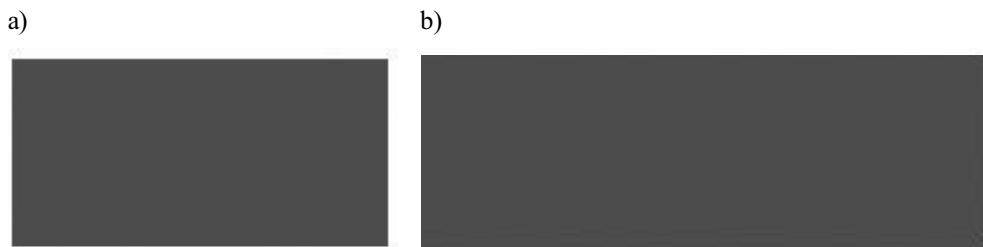
Na początku należy zauważyć, że rozpatrywane w tym rozdziale sterowanie prędkością rozrostu nie dotyczy uzyskania rzeczywistej prędkości rozrostu uzależnionej od parametrów procesowych. Chodzi tu przede wszystkim o możliwość uwzględnienia zależności względnej prędkości ruchu granicy od kierunku rozrostu. Wówczas zadanie odpowiednich zależności pozwoli zredukować anizotropię przestrzeni oraz dokładnie sterować kształtem i orientacją (obrotem) ziaren.

Wracając do klasycznych algorytmów, możemy traktować różne zastosowane w nich otoczenia w różny sposób. Otoczenie Moore’a składa się z czterech komórek

znajdujących się w sąsiedztwie von Neumanna i mających wspólne boki oraz czterech komórek położonych po przekątnej i stykających się z rozpatrywaną komórką wierzchołkami. Symetryczne pseudosąsiedztwa dopełniają sąsiedztwo von Neumanna komórkami po przekątnej w taki sposób, że w sąsiedztwie pseudopentagonalnym każda z tych komórek pojawia się jeden raz w czterech iteracjach, w pseudoheksagonalnym – dwa razy, w pseudoheptagonalnym – trzy razy, a w sąsiedztwie Moore’a w każdym kroku czasowym. Można zatem powiedzieć, iż zmieniając częstość pojawiania się komórek po przekątnej n_p , można sterować prędkością rozrostu wzdłuż głównych osi i po przekątnej. Wyniki takiego sterowania zilustrowano na rysunku 2.11. Wzdłuż osi w każdym przypadku uzyskujemy maksymalną prędkość rozrostu, natomiast po przekątnej prędkość wzrasta od zera do wartości maksymalnej. Czy można uzyskać inne wartości stosunku między prędkością rozrostu wzdłuż osi i po przekątnej? Na pewno nie stoi temu na przeszkodzie. Możemy wykonać przejście na przykład w dwóch z pięciu ($n_p = 0,4$) lub trzech z siedmiu ($n_p = 0,428$) kolejnych kroków czasowych. Czy możemy zastosować różne przejścia dla osi poziomej i osi pionowej? Okazuje się, że i to jest możliwe. Na rysunku 2.12 pokazano wyniki symulacji prowadzących do uzyskania prostokątów, w których stosunek długości boków wynosi 2:1 (częstotliwość pojawiania się komórek w osi pionowej $n_p = 0,5$) i 3:1 ($n_p = 0,333$).

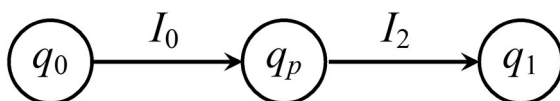


Rys. 2.11. Przykłady symulacji rozrostu ziarna w otoczeniu Moore’a z różną częstotliwością pojawiania się komórek po przekątnej n_p :
a) $n_p = 0$; b) $n_p = 0,25$; c) $n_p = 0,5$; d) $n_p = 0,75$; e) $n_p = 1$



Rys.2.12. Przykłady symulacji rozrostu ziarna w otoczeniu Moore’a z różną częstotliwością pojawiania się komórek w pionie n_p :
a) $n_p = 0,5$; b) $n_p = 0,333$

Wyniki te uzyskano wyłącznie za pomocą klasycznych algorytmów, z zastosowaniem dwóch stanów: początkowego i końcowego. Sterowanie nadal pozostaje trudne, ponieważ ogranicza je mała liczba parametrów wpływających na reguły przejścia oraz wymóg, by komórka znajdowała się tylko w jednym z dwóch stanów. Większe zróżnicowanie prędkości można uzyskać, rozdrabniając krok czasowy lub stosując różne jego długości w odniesieniu do różnych części modelu opartego na automatach komórkowych. Jednak istnieje prostsze rozwiązanie: wprowadzenie trzeciego, przejściowego stanu. Dzięki spełnieniu tego warunku każda komórka może przebywać w jednym z trzech stanów. Pierwszy stan to stan początkowy q_0 , drugi – stan przejściowy q_p , a trzeci – stan końcowy q_1 . Przejście ze stanu początkowego do stanu końcowego może się odbywać bezpośrednio lub poprzez stan przejściowy – zależy to od reguły przejścia oraz od prędkości ruchu granicy w tym czy innym kierunku. Stan przejściowy może mieć różny czas trwania lub, innymi słowy, różną liczbę kroków czasowych. Czas przejściowy może być wyznaczony jednokrotnie podczas inicjacji stanu przejściowego przez wyznaczenie liczby kroków lub w każdym kroku osobno przez obliczenie bieżącego przyrostu zmiennej charakteryzującej stan przejściowy. Schematycznie przedstawiono to na rysunku 2.13.



Rys. 2.13. Automat komórkowy w algorytmie rozrostu ziaren z dwoma podstawowymi stanami i stanem przejściowym q_p

W obliczeniach pierwszym lub drugim sposobem w celu uzyskania większego zróżnicowania prędkości lub większej dokładności zadawania jej wartości wymagane jest odpowiednie rozdrobnienie kroku czasowego. Zastosowanie dwudziestu kroków pozwoli zmieniać prędkości dwudziestokrotnie lub zadawać prędkości z dokładnością do 5%. Jednak wykorzystanie dwudziestu iteracji zamiast jednej, bez zmiany algorytmów przeglądania przestrzeni automatów komórkowych, wydłuży czas symulacji dwudziestokrotnie.

Wydłużenie czasu symulacji jest istotną wadą tej metody, natomiast jej zalet na razie nie widać. Jeżeli by pozostawić bez zmian otoczenie, nie zostanie osiągnięta istotna dodatkowa jakość. Każda komórka w otoczeniu Moore’a będzie miała nadal osiem komórek sąsiednich, co pozwoli wyznaczyć tylko osiem kierunków rozrostu. Faktycznie te osiem kierunków w ten czy inny sposób było już wykorzystane zarówno w algorytmach klasycznych, jak i z zastosowaniem reguł losowych. Nową jakość dałoby się uzyskać, gdyby było możliwe wyznaczanie w ten czy inny sposób dowolnego kierunku rozrostu w przestrzeni oraz uzależnienie prędkości od wyznaczonego kierunku.

Oczywiście, w otoczeniu Moore’a można określić tylko osiem wektorów wyznaczających kierunki rozrostu analizowanej komórki. Zwiększyć liczbę możliwych kierunków można dzięki zwiększeniu promienia otoczenia komórki. Dla promienia równego

dwóm komórkom takich kierunków będzie już 16, a w przypadku zwiększenia promienia do długości równej trzem komórkom uzyskamy 32 kierunki rozrostu.

W algorytmach, które zostaną przedstawione poniżej, zastosowano nieco inne podejście.

2.4. Izotropia przestrzeni

W poprzednich rozdziałach rozpatrywano przestrzeń automatów komórkowych, która od początku jest anizotropowa i w której istnieje kilka przeważających kierunków rozrostu. Anizotropię wprowadza lub wzmacnia narzucone otoczenie. W przestrzeni dwuwymiarowej z komórkami kwadratowymi otoczenie von Neumana uprzywilejowuje cztery kierunki, otoczenie Moore'a – osiem (z komórkami trójkątnymi odpowiednio 3 i 6, z sześciokątnymi również 6). W przestrzeni trójwymiarowej z komórkami sześciennymi w sąsiedztwie o promieniu jednej komórki takich podstawowych kierunków może być do 26. Nie jest to skutek dyskretyzacji, ale wprowadzenia sąsiedztwa – to właściwość przestrzeni narzucona przez geometrię otoczenia. Właściwość ta w istotnym stopniu przeszkadza w sterowaniu kształtem rosnących ziaren, ponieważ sterowanie wymaga rozrostu uzależnionego od kształtu ziarna i jego przestrzennej orientacji, a nie od właściwości samej przestrzeni. Najważniejszym zadaniem przy opracowaniu algorytmów sterowania kształtem ziaren jest więc zbliżenie właściwości przestrzeni automatów komórkowych do izotropowych, czyli stworzenie przestrzeni, w której każdy kierunek rozrostu jest równoważny i niezależny od narzuconych przez sąsiedztwo uprzywilejowanych osi przestrzeni.

Jeżeli chodzi o izotropię, najprostszym testem potwierdzającym ją będzie uzyskanie okrągłego kształtu ziarna, czyli taki przebieg rozrostu, w którym wszystkie kierunki są równoważne. Oczywiście dokładność okrągłego kształtu będzie ograniczona dyskretnością przestrzeni oraz jej rozdzielczością (co jednak nie jest związane z anizotropią). W takim przypadku dla uzyskania okrągłego kształtu wystarczy, aby prędkość rozrostu w każdym kierunku była taka sama.

Zrozumienie zależności pomiędzy sąsiedztwem a anizotropią przestrzeni, pozwoli znaleźć rozwiązanie problemu izotropii. Na rysunku 2.1, na którym pokazano otoczenie Moore'a, łatwo zauważyć, że odległość do komórek sąsiadujących w pionie i poziomie jest mniejsza niż odległość do sąsiadów po przekątnej (dokładnie: odległość ta jest $\sqrt{2}$ razy mniejsza). Ponieważ czas przejścia równa się czasowi trwania kroku czasowego i jest taki sam dla wszystkich komórek przestrzeni, to prędkość przekazywania informacji i w konsekwencji rozrost ziarna są większe po przekątnej. W otoczeniu von Neumanna do przekazania informacji do komórki sąsiadującej po przekątnej konieczne są dwa kroki czasowe, ponieważ nie wchodzi ona do otoczenia tej komórki, lecz do otoczenia sąsiednich komórek (w pionie i w poziomie). Czyli większa odległość ($\sqrt{2}$) jest pokonana w dwukrotnie dłuższym czasie, toteż z prędkością wolniejszą niż w kierunkach poziomym i pionowym. W wyniku powyższego w otoczeniu Moore'a

uprzywilejowane są kierunki po przekątnej, a w otoczeniu von Neumanna – kierunki poziome i pionowe. Wpływ otoczenia na rozrost ziarna i kinetykę rekrytalizacji przedstawił Davies (1995).

Ponieważ warunkiem izotropii jest niezależność prędkości od kierunku, a odległości komórek narzucone są przez wybraną siatkę, otoczenie i kierunek przekazywania informacji, to należy uzależnić czas od kierunku rozrostu. Jednak w otoczeniu Moore’a jest tylko osiem kierunków, a w otoczeniu von Neumanna jeszcze mniej – cztery, inne kierunki w ramach tych sąsiedztw nie istnieją. Zwiększyć liczbę kierunków można przez wydłużenie promienia otoczenia, a żeby uzyskać dowolny kierunek, nie wystarczy nawet promień o długości równej wymiarowi przestrzeni. Z jednej strony automaty nie potrzebują dowolnego kierunku, lecz ich skończonego zestawu, właściwego dla wybranej siatki i rozmiarów przestrzeni, a z drugiej – można wyróżnić sąsiedztwa bliskiego i dalekiego zasięgu. Pierwsze wyznacza, która komórka będzie zmieniała swój stan, a drugie – kiedy to nastąpi. W zasadzie ze wszystkich informacji, których może dostarczać otoczenie dalekiego zasięgu, wykorzystana zostaje tylko ta o kierunku rozrostu. Tak więc w użytkowaniu izotropii przestrzeni można wyróżnić dwa etapy: wyznaczenie kierunku rozrostu i zachowanie wyznaczonej prędkości rozrostu w każdym kierunku, co da się osiągnąć przez dobór czasu, w którym następuje przejście w stan końcowy q_1 , lub w układzie ze stałym krokiem czasowym – wybór kroku, w którym następuje ta zmiana.

Uzyskanie okrągłego kształtu jest pierwszym etapem sprawdzenia izotropii przestrzeni. Najważniejsze w tym teście jest wyznaczenie kierunku rozrostu. W drugim teście prędkość będzie uzależniona od kierunku w celu uzyskania innych kształtów rosnących ziaren i ich dowolnej przestrzennej orientacji. Algorytmy do wyznaczenia kierunku rozrostu w odniesieniu do pierwszego testu zostaną przedstawione poniżej, natomiast o uzyskaniu ziaren o różnych kształtach będzie mowa w podrozdziale 2.5.

2.4.1. Algorytm z niezmiennymi warunkami rozrostu ziarna

Kierunek rozrostu okrągłego ziarna zawsze będzie prostopadły do jego powierzchni i będzie przechodzić przez centrum kuli (okręgu). Czas osiągnięcia przez granicę ziarna dowolnego punktu przestrzeni będzie zależał tylko od odległości tego punktu od miejsca zarodkowania ziarna, nie będzie natomiast uzależniony od kierunku rozrostu. Na tej właściwości może się opierać najprostszy algorytm (Svyetlichnyy i Matachowski 2005a, 2005b, Svyetlichnyy *et al.* 2005).

Pierwszy algorytm może wykorzystać informację o lokalizacji granicy ziarna, miejscu powstania zarodka i na tej podstawie obliczyć odległość komórki na granicy ziarna od miejsca zarodkowania i czas rozrostu ziarna do wyznaczonej komórki. W tym algorytmie w zasadzie nieistotny jest rodzaj otoczenia, nie wpływa ono na aktualny kształt, więc wystarczy zastosować najprostsze otoczenie von Neumanna. Reguła przejścia dla I_0 pozostaje taka jak opisana na początku rozdziału (rys. 2.1 i 2.2). Komórka przechodzi w stan przejściowy q_p (rys. 2.13), a przejście w stan końcowy q_1 następuje po spełnieniu warunku I_2 (osiągnięcie czasu przejścia).

W celu sprawdzenia tego warunku najpierw należy wyznaczyć współrzędne rozpatrywanej komórki (x_b, y_b) i komórki, w której powstał zarodek (x_z, y_z) , oraz obliczyć odległość pomiędzy nimi L :

$$L = \sqrt{(x_b - x_z)^2 + (y_b - y_z)^2} \quad (2.8)$$

Dalej na podstawie prędkości rozrostu v_r można obliczyć czas od chwili zarodkowania ($\Delta t_p = L/v_r$) i czas ($t_p = t_z + \Delta t_p$) do momentu, w którym powinno nastąpić przejście ze stanu przejściowego q_p w stan końcowy q_1 . Z uwzględnieniem stałej długości kroku czasowego τ , można obliczyć, w jakim czasie od momentu rozpoczęcia zarodkowania, licząc w krokach czasowych n_τ , nastąpi przejście do stanu końcowego:

$$n_\tau = \text{int} \left(\frac{L}{v_r \tau} \right) + 1 \quad (2.9)$$

gdzie $\text{int}(a)$ – część całkowita liczby a .

Ponieważ liczba kroków może być tylko liczbą całkowitą, zastosowano odpowiednią funkcję $\text{int}(a)$. Zgodnie ze wzorem (2.9) przejście następuje w kroku kolejnym po osiągnięciu obliczonej chwili czasu $t > t_p$ (z uwzględnieniem dyskretyzacji po czasie: $n > n_\tau + n_z$). Służy temu dodanie jednostki. Zatem w tym algorytmie przejście zawsze następuje z pewnym opóźnieniem. Oczywiście krok czasowy należałoby w tym algorytmie dobrać w taki sposób, żeby czas przejścia granicy ziarna przez dowolną komórkę nie był krótszy od wybranego kroku czasowego.

Po wyznaczeniu długości czasu liczonego od momentu rozpoczęcia zarodkowania do przejścia wybranej komórki w stan końcowy, znając krok czasowy, w którym powstał zarodek, można wyznaczyć krok czasowy, w którym nastąpi przejście w stan q_1 wybranej komórki, czyli spełni się warunek I_2 . W przypadku zastosowania komórek kwadratowych realizacja algorytmu jest najłatwiejsza.

Głównym warunkiem stosowalności algorytmu jest niezmiennosc przestrzeni i prędkości rozrostu. Algorytm ten jest bardzo przydatny wtedy, gdy przestrzeń komórek pozostaje bez zmian, czyli do stworzenia struktury początkowej i w niektórych zagadnieniach związanych z krzepnięciem lub krystalizacją.

Algorytm ten może być łatwo rozbudowany tak, by mógł być stosowany w odniesieniu do przestrzeni trójwymiarowej.

Warunki stosowalności algorytmu nie zawsze są spełniane. Może się zmienić zarówno geometria przestrzeni komórek, jak i prędkość rozrostu, również długość kroku czasowego może być zmienna. Zmienny krok czasowy nie będzie stwarzał istotnego problemu, jeżeli przejścia będą liczone w jednostkach czasu, a nie numerach kroku czasowego. Geometria przestrzeni komórek, czyli ich rozmiary i kształt, może być zmienna, na przykład przy rozpatrywaniu procesów związanych z odkształceniem. Zmiana prędkości rozrostu ziaren jest bardzo często stosowanym zabiegiem podczas symulacji

praktycznie wszystkich zjawisk. Na ogół prędkość rozrostu zależy od temperatury, stężenia pierwiastków, gęstości dyslokacji, krystalograficznej orientacji ziaren i innych parametrów rozpatrywanych procesów. Należałoby zatem zmodyfikować przedstawiony w tym rozdziale algorytm, eliminując jego zależność od wymienionych warunków.

Najpierw zostanie przedstawiony wariant algorytmu, w którym geometria przestrzeni komórek pozostaje bez zmian, natomiast prędkość rozrostu może być dowolną funkcją przestrzeni, czasu i parametrów procesu (podrozdz. 2.4.2). Dalej (podrozdz. 2.4.3) zaprezentuję algorytm, który uwzględni różne warianty odkształcenia komórek lub komórki o dowolnym kształcie.

2.4.2. Algorytm uwzględniający zmienną prędkość rozrostu

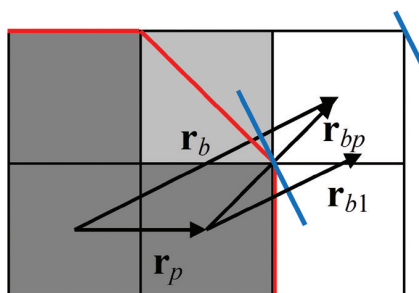
Zmienna prędkość nie pozwala wprost obliczyć czasu przejścia na podstawie tylko położenia granicy i aktualnej lub początkowej prędkości rozrostu. Należałoby całkować prędkość rozrostu ziarna względem czasu aż do osiągnięcia przez granicę wybranej komórki. Nie jest możliwe liczenie czasu przez całkowanie prędkości wzdłuż całej linii rozrostu w sposób jawny, należałoby zatem zapamiętać całą historię rozrostu, co nie jest możliwe ze względu na ogrom niezbędnej do wykonania takiej operacji pamięci. Jednak taka historia jest już zapisana w poprzednich krokach i wystarczy tylko odpowiednio ją wykorzystać. Należy wybrać taki algorytm, który będzie potrzebował jak najmniej dodatkowych informacji przypisywanych do każdej komórki, ale pozwoli dokładnie uwzględnić zmienną prędkość rozrostu.

Pierwszym koniecznym krokiem ku modyfikacji poprzedniego algorytmu jest przejście od bezpośredniego obliczenia odległości granicy od miejsca zarodkowania, jak to było w algorytmie pierwszym, do obliczenia odległości na podstawie przyrostów (Matachowski i Svyetlichnyy 2005, 2006; Svyetlichnyy 2012). Całka będzie zatem zastąpiona przez sumę i zostanie uwzględniona geometria otoczenia oraz dyskretyzacja przestrzeni i czasu.

Na rysunku 2.14 przedstawiono fragment przestrzeni z rosnącym ziarnem. Ciemnoszarym kolorem zaznaczono komórki znajdujące się w stanie końcowym q_1 , jasnoszarym – komórkę w stanie przejściowym q_p , a białym – komórki w stanie początkowym q_0 . Linia łamana (czerwona) odzwierciedla granicę ziarna. W przedstawionej na rysunku chwili granica ziarna właśnie w pełni przeszła przez środkową dolną komórkę i komórka ta znalazła się w stanie końcowym q_1 . Przez środkową górną komórkę granica ziarna jeszcze przechodzi i dlatego znajduje się ona w stanie przejściowym. Przejście granicy ziarna przez komórkę albo komórki w stan końcowy oznacza, że wszystkie sąsiednie komórki znajdujące się w stanie początkowym q_0 (obie komórki po prawej stronie) powinny przejść w stan przejściowy q_p . Dla nich zatem należy obliczyć czas przejścia w stan końcowy.

Zakładamy, że wewnątrz jednej komórki granica ziarna jest prostoliniowa i prostopadła do kierunku rozrostu oraz nie zmienia swego pochylenia podczas rozrostu. Na rysunku 2.14 kierunek rozrostu w środkowej dolnej komórce zaznaczono wektorem \mathbf{r}_p , natomiast w prawej górnej komórce – \mathbf{r}_b . W przypadku prawej górnej komórki, dla

której należy obliczyć czas przejścia w stan końcowy q_1 , odcinki granicy rosnącego ziarna na początku i na końcu jej przejścia przez komórkę zaznaczono na niebiesko. Wtedy czas przechodzenia granicy ziarna przez komórkę (czas, w którym komórka znajduje się w stanie przejściowym q_p , czyli czas opóźnienia Δt_{bp}) będzie wyznaczony na podstawie odległości między liniami niebieskimi oraz aktualnej prędkości rozrostu ziarna. W celu wyznaczenia czasu przejścia t_{bp} w stan końcowy prawej górnej komórki wystarczy czas opóźnienia Δt_{bp} dodać do czasu przejścia w stan końcowy środkowej dolnej komórki t_p ($t_{bp} = t_p + \Delta t_{bp}$). Zatem poczynając od zarodka, liczone czas przejścia każdej komórki w stan końcowy q_1 (lub numer odpowiedniego kroku czasowego), przy czym moment przejścia granicy ziarna poza komórkę jest chwilą odniesienia, od której liczone czas przejścia kolejnej komórki itd.



Rys. 2.14. Wyznaczenie kierunku rozrostu metodą przyrostową (Svyetlichnyy 2012)

W wyniku zastosowania takiego algorytmu zamiast bezpośredniego obliczenia czasu przejścia zostanie on obliczony metodą przyrostową. Do realizacji algorytmu w tej postaci, jak powyżej napisano, konieczne jest zastosowanie otoczenia Moore'a, natomiast przy zastosowaniu otoczenia von Neumanna należy algorytm zmodyfikować, co zostanie opisane nieco dalej.

Omawiany algorytm składa się z następujących elementów:

- wyznaczenie kierunku rozrostu;
- wyznaczenie odległości, którą powinna pokonać granica ziarna, przechodząc przez komórkę;
- obliczenie czasu przejścia granicy ziarna przez komórkę;
- wyznaczenie czasu przejścia komórki w stan końcowy.

Kierunek rozrostu można wyznaczyć bezpośrednio, jak w poprzednim algorytmie, na podstawie współrzędnych rozpatrywanej komórki i komórki, w której powstał zarodek. Jednak takie obliczenie można zastosować tylko w przypadku, kiedy kształt komórki podczas modelowania nie ulega zmianom. Natomiast jeżeli podczas symulacji uwzględniano odkształcenie, takie podejście jest niedokładne i do obliczenia wektora rozrostu należy zastosować metodę przyrostową, której istotę przedstawiono na rysunku 2.14.

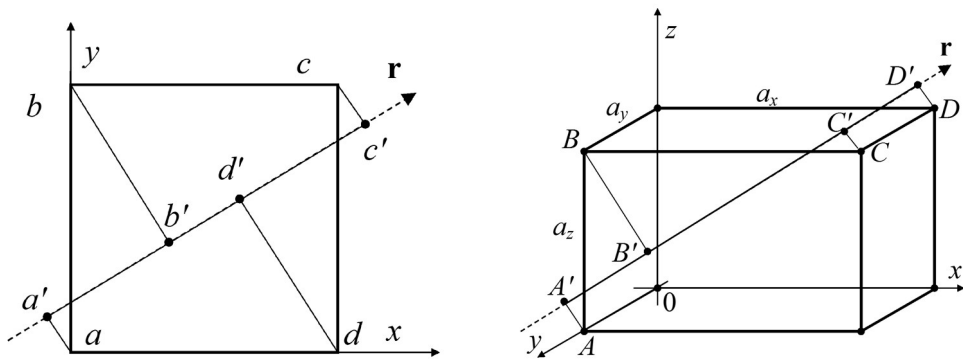
Na przykładzie prawej górnej komórki pokazano, w jaki sposób przyrostowo liczone wektor kierunku rozrostu. Wektor kierunku rozrostu, jak już wspomniano, ziarna rosnącego w postaci kuli (lub okręgu) jest linią łączącą miejsce powstania zarodka z rozpatrywaną komórką. Przyrostowo ten wektor może być obliczony jako suma wektorów kierunku rozrostu poprzedniej komórki i wektora od poprzedniej komórki do rozpatrywanej:

$$\mathbf{r}_b = \mathbf{r}_p + \mathbf{r}_{bp} \quad (2.10)$$

gdzie:

- \mathbf{r}_b – wektor od zarodka do rozpatrywanej komórki,
- \mathbf{r}_p – wektor od zarodka do poprzedniej komórki,
- \mathbf{r}_{bp} – wektor od poprzedniej do rozpatrywanej komórki.

Drugim elementem algorytmu jest wyznaczenie długości odcinka, który musi pokonać granica ziarna, przechodząc przez komórkę (Svyetlichnyy 2006, 2012). Schemat obliczeń pokazano na rysunku 2.15.



Rys. 2.15. Schemat obliczenia czasu przejścia granicy ziarna przez komórkę w kształcie kwadratu i prostopadłościanu (Svyetlichnyy 2012)

Granica rozrostu ziarna jest prostopadła do kierunku rozrostu. Punkty a' i c' są rzutami wierzchołków komórki, w których następuje wejście granicy ziarna w komórkę i wyjście poza nią, natomiast długość odcinka $a'c'$ jest równa sumie rzutów boków komórki na kierunek rozrostu. Oznaczając kierunek rozrostu \mathbf{r} przez wektor jednostkowy $\{r_x, r_y\}$, długość odcinka $a'c'$ można obliczyć jako sumę dwóch iloczynów skalarnych wektora kierunku rozrostu i każdego wektora boków komórki, które w przypadku kwadratów o bokach równych jedności odpowiadają wersorom przestrzeni automatów komórkowych. Długość odcinka $a'c'$ można obliczyć w następujący sposób:

$$a'c' = |r_x| + |r_y| \quad (2.11)$$

Dla przypadku trójwymiarowego $\mathbf{r} = \{r_x, r_y, r_z\}$ można łatwo wykazać, że suma rzutów trzech krawędzi (wektorów) na kierunek rozrostu przyjmuje postać:

$$L = A'D' = |r_x| + |r_y| + |r_z| \quad (2.12)$$

Jeżeli przestrzeń składa się z komórek o kształcie prostopadłościanów o krawędziach a_x , a_y i a_z , łatwo wykazać, że odpowiedni odcinek, który musi pokonać granica ziarna, przyjmuje postać:

$$L = A'D' = a_x|r_x| + a_y|r_y| + a_z|r_z| \quad (2.13)$$

Trzeci element algorytmu to obliczenie czasu przejścia granicy ziarna przez komórkę. Przy stałej prędkości rozrostu v_r lub chwilowej jej wartości czas ten można określić jako czas konieczny do pokonania odcinka $a'c'$: $\Delta t_p = a'c'/v_r$. Ostatecznie wyznaczamy czas i liczbę iteracji koniecznych do zmiany stanu komórki z uwzględnieniem długości kroku czasowego τ :

$$\Delta t_p = \frac{L}{v_r} = \frac{a_x|r_x| + a_y|r_y| + a_z|r_z|}{v_r}; \quad n_\tau = \frac{\Delta t_p}{\tau} \quad (2.14)$$

Jednak prędkość rozrostu może w czasie przejścia granicy przez komórkę ulegać zmianom. W literaturze naukowej można spotkać takie podejścia do tego rodzaju przypadków, w których wyznacza się ułamek wypełnienia komórki. W opisywanym algorytmie wypełnieniem komórki będzie względna odległość granicy w danym momencie od najbliższego zarodka do naroża komórki fr . Można zatem zastosować metodę przyrostową wewnątrz komórki. Liczba iteracji n_τ , wówczas, w odróżnieniu od algorytmu pierwszego (2.9), w równaniu (2.14) nie będzie liczbą całkowitą i nie będzie wykorzystywana bezpośrednio do obliczenia czasu przejścia. Wykorzystywano natomiast wartość odwrotną do liczby iteracji n_τ , czyli przyrost ułamka wypełnienia komórki $dfr = 1/n_\tau$.

Należy zwrócić uwagę na fakt, że rysunek 2.14 ukazuje chwilę przejścia środkowej dolnej komórki ze stanu przejściowego q_p w stan końcowy q_1 bez powiązania z dyskretyzacją w czasie, nie uwzględnia tego również algorytm. W czasie, który minie od chwili przejścia granicy ziarna przez komórkę do końca kroku czasowego, granica pokona pewną odległość, przechodząc przez sąsiednie komórki, więc owa nadwyżka czasu powinna być uwzględniona przy obliczeniach czasu przejścia kolejnych komórek. W wyniku tego dyskretny jest tylko czas wyznaczenia stanu komórki, a nie czas przejścia komórki z jednego stanu w inny. Ten ostatni pozostaje czasem niedyskretnym.

Przejście komórki ze stanu przejściowego q_p w stan końcowy q_1 następuje, kiedy ułamek wypełnienia komórki fr osiągnie albo przewyższy wartość 1,0.

Wówczas w odniesieniu do niektórych sąsiednich komórek należy wykonać następujące czynności:

1. Znaleźć w otoczeniu komórki w stanie początkowym q_0 i zmienić ich stan na stan przejściowy q_p .
2. Na podstawie zawartej w sąsiedniej, znajdującej się w stanie końcowym komórce informacji o kierunku rozrostu ziarna oraz rozmieszczenia tych dwóch komórek wyznaczyć kierunek rozrostu rozpatrywanej komórki zgodnie z (2.11).
3. Wyznaczyć też przyrost ułamka wypełnienia komórki dfr z uwzględnieniem kierunku rozrostu r , prędkości rozrostu v_r , wymiarów komórki a_x , a_y i a_z oraz długości kroku czasowego τ na podstawie równania (2.14).
4. Dla każdej komórki w stanie przejściowym wyznaczyć ułamek wypełnienia fr . W i -tym kroku czasowym ułamek wypełnienia fr wyznacza się zgodnie z następującym wzorem:

$$fr_i = fr_{i-1} + dfr \quad (2.15)$$

gdzie fr_{i-1} – ułamek wypełnienia w poprzednim kroku lub nadwyżka wypełnienia $fr = fr - 1$ przeniesiona z poprzedniej komórki.

5. Po wykonaniu obliczeń ułamka wypełnienia sprawdzić, czy spełniono warunek pełnego wypełnienia ($fr \geq 1$). Jeśli warunek ten jest spełniony, komórka zmienia swój stan na końcowy q_1 , w razie jego niespełnienia pozostaje ona w stanie przejściowym q_p i w kolejnym kroku czasowym algorytm rozpoczyna obliczenia od kroku 4.

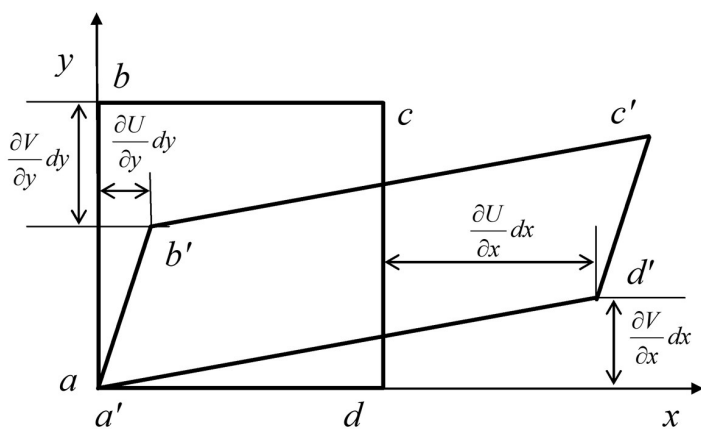
Należy przy tym zauważyć, że w większości obliczeń można przyjąć niezmienną prędkość rozrostu podczas przejścia granicy przez komórkę lub obliczyć wartość średnią. Wówczas czas Δt_p można obliczyć bezpośrednio ze wzoru (2.14). Algorytm z zastosowaniem ułamka wypełnienia może działać bez powiązania z czasem globalnym, ponieważ operuje wyłącznie przyrostami, jednak może być on związany z tym czasem poprzez odpowiedni krok czasowy. Bezpośrednie powiązanie przyrostu czasu Δt_p (2.14) z czasem globalnym wymaga uwzględnienia czasu przejścia poprzedniej komórki t_p : $t_{bp} = t_p + \Delta t_p$.

2.4.3. Algorytm uwzględniający odkształcenie lub dowolny kształt komórek

Ograniczeniem omówionego w poprzednim podrozdziale algorytmu jest prostokątny kształt komórek. Algorytm ten można stosować również do symulacji zjawisk z odkształceniem przestrzeni komórek, jednak główne osie przestrzeni powinny się pokrywać z głównymi kierunkami odkształceń, aby kształt komórek pozostawał prostokątny (lub prostopadłościenny w przestrzeni trójwymiarowej). Można również pozostawić komórki nieodkształcone, uzależnić prędkość rozrostu ziaren od kierunku i tensora

odkształceń, ale wówczas kształt uzyskany po symulacji należy poddać modyfikacji (odkształceniu), która będzie odzwierciedlała odkształcenie rzeczywiste. Takie podejście jest możliwe, ale wedle mojej opinii – nie najlepsze. Prostszy rozwiązaniem jest zmiana kształtu komórek bez uzależnienia prędkości rozrostu w różnych kierunkach od odkształcenia (co nie przeszkadza w uzależnieniu prędkości od innych czynników).

Rozpatrzmy najpierw uwzględnienie odkształcenia. Na rysunku 2.16 przedstawiono przykład odkształcenia komórki w kształcie kwadratu $abcd$. Nowy kształt komórki przedstawiony został za pomocą czworokąta $a'b'c'd'$. Równomierne izotropowe płaskie odkształcenie kwadratowego elementu, kiedy tensor odkształcenia zawiera trzy niezależne elementy: ϵ_x , ϵ_y i ϵ_{xy} , prowadzi do powstania równoległoboku o tej samej powierzchni.



Rys. 2.16. Odkształcenie kwadratowej komórki (Svyetlichnyy 2006)

Nadal podstawowym poszukiwanym parametrem jest czas przejścia komórki z jednego stanu w inny, czyli czas, w którym granica rozrostu ziarna przejdzie przez całą komórkę. Wówczas koniecznymi do obliczeń danymi są kierunek rozrostu ziarna oraz długości i kierunek boków (krawędzi) komórki. Na rysunku 2.16 widać, że ten czworokąt może być opisany dwoma wektorami, które pokrywają się z bokami $a'd'$ i $a'b'$. Z rysunku wynika, że można je opisać w następujący sposób:

$$\begin{aligned} a'd' &= \left\{ \left(1 + \frac{\partial U}{\partial x} \right) dx, \frac{\partial V}{\partial x} dy \right\} \\ a'b' &= \left\{ \frac{\partial U}{\partial y} dx, \left(1 + \frac{\partial V}{\partial y} \right) dy \right\} \end{aligned} \quad (2.16)$$

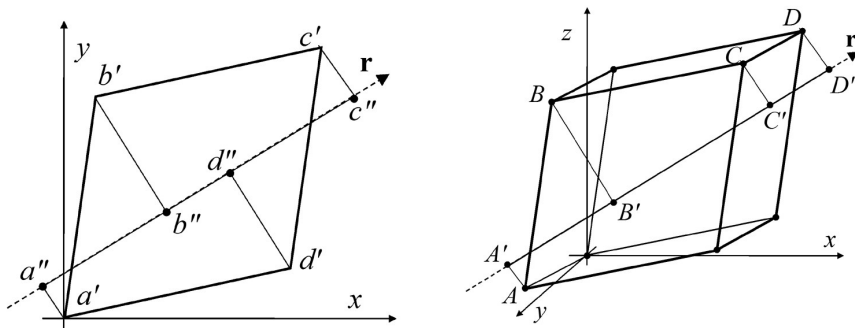
gdzie dx , dy – długości boków nieodkształconej komórki.

Przyjmując jednostkową długość boków wejściowego kwadratu, można wyznaczyć boki przez odkształcenia:

$$\begin{aligned} a'd' &= \left\{ (1 + \epsilon_x), \frac{\gamma_{xy}}{2} \right\} \\ a'b' &= \left\{ \frac{\gamma_{xy}}{2}, (1 + \epsilon_y) \right\} \end{aligned} \quad (2.17)$$

Czas zmiany stanu komórki t_p jest równy czasowi przejścia granicy rozrostu ziarna przez całą komórkę. Na rysunku 2.17 przedstawiono schemat wyjaśniający przeprowadzone obliczenia. Przy stałej prędkości rozrostu v_r lub jej chwilowej wartości ten czas można określić jako czas konieczny do przejścia granicy ziarna przez komórkę $a''c''$: $t_p = a''c''/v_r$. Punkty a'' i c'' są rzutami wierzchołków, w których następuje wejście granicy ziarna w komórkę i wyjście poza nią, a długość odcinka $a''c''$ jest równa sumie rzutów boków komórki na kierunek rozrostu. Oznaczając kierunek rozrostu \mathbf{r} przez wektor jednostkowy $\{r_x, r_y\}$, długość odcinka $a''c''$ można obliczyć jako sumę dwóch iloczynów skalarnych wektora kierunku rozrostu i każdego wektora boków komórki:

$$L = a''c'' = \left| (1 + \epsilon_x)r_x + \frac{\gamma_{yx}}{2}r_y \right| + \left| \frac{\gamma_{xy}}{2}r_x + (1 + \epsilon_y)r_y \right| \quad (2.18)$$



Rys. 2.17. Schemat do obliczenia czasu zmiany stanu odkształconej komórki (Svyetlichnyy 2012)

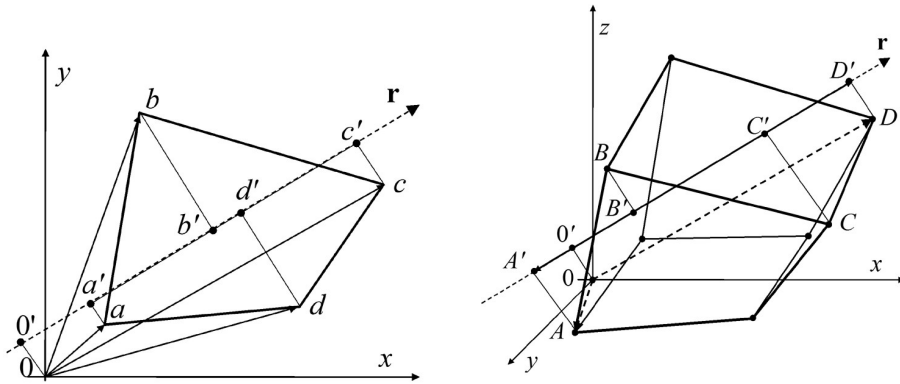
Dla przypadku trójwymiarowego można łatwo wykazać, że suma rzutów trzech krawędzi na kierunek rozrostu przyjmuje postać:

$$\begin{aligned} L = A'D' &= \left| (1 + \epsilon_x)r_x + \frac{\gamma_{yx}}{2}r_y + \frac{\gamma_{zx}}{2}r_z \right| + \\ &+ \left| \frac{\gamma_{xy}}{2}r_x + (1 + \epsilon_y)r_y + \frac{\gamma_{zy}}{2}r_z \right| + \left| \frac{\gamma_{xz}}{2}r_x + \frac{\gamma_{yz}}{2}r_y + (1 + \epsilon_z)r_z \right| \end{aligned} \quad (2.19)$$

Ostatecznie wyznaczamy czas opóźnienia Δt_p i liczbę iteracji koniecznych do zmiany stanu komórki n :

$$\Delta t_p = \frac{L}{v}; \quad n = \frac{\Delta t_p}{\tau} \quad (2.20)$$

Ostatni wariant to komórki o dowolnym kształcie. Na rysunku 2.18 pokazana została komórka o kształcie czworoboku i sześcianu o dowolnym kształcie. Nic nie stoi na przeszkodzie, by zastosować opisane poniżej obliczenia do dowolnego wieloboku, poczynając od trójkąta. Można je również łatwo rozszerzyć na przestrzeń trójwymiarową.



Rys. 2.18. Schemat obliczania czasu zmiany stanu komórki o dowolnym kształcie (Svyetlichnyy 2012)

Zadanie jest takie samo: znaleźć długość rzutu wieloboku lub wielościanu na kierunek rozrostu \mathbf{r} . Można tego dokonać w następujący sposób: wykorzystujemy wektory zaczepione w początku układu dla każdego wierzchołka, wówczas wektor $0a = \{x_a, y_a\}$, $0b = \{x_b, y_b\}$ itd. Odpowiednio rozpatrzmy rzut wektora zaczepionego na kierunek rozrostu \mathbf{r} : dla wektora $0a$ będzie to $0'a' = x_a r_x + y_a r_y$, dla wektora $0b$ będzie to $0'b' = x_b r_x + y_b r_y$ itd. Po obliczeniu rzutów wszystkich wektorów zaczepionych należy wybrać największy i najmniejszy (z uwzględnieniem znaku), a wówczas różnica między nimi będzie równa długości rzutu wieloboku (wielościanu) na kierunek rozrostu L :

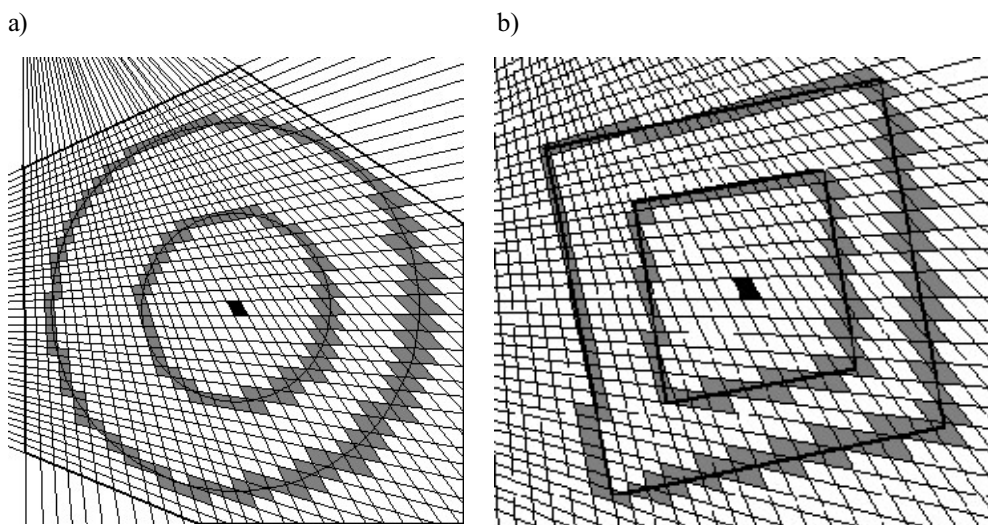
$$L = \max(0'a', 0'b', \dots) - \min(0'a', 0'b', \dots) \quad (2.21)$$

Dalej obliczenia liczby iteracji prowadzono jak poprzednio (2.20).

W przypadku trójwymiarowego rzutu wektora zaczepionego, na przykład wierzchołka A , wzór będzie następujący: $0'A' = x_a r_x + y_a r_y + z_a r_z$. Pozostałe elementy algorytmu się nie zmieniają.

Na rysunku 2.19 pokazano wyniki symulacji rozrostu okrągłego i kwadratowego ziarna na siatce, w której kształt każdej komórki jest unikatowy. „Pionowe” i „poziome”

linie siatki schodzą się w jednym punkcie. Czarna komórka w środku oznacza miejsce powstania zarodka, szare komórki to komórki w stanie przejściowym w dwóch chwilach symulacji. Okręgi i kwadraty zaznaczone ciągłymi liniami wskazują teoretyczne położenie granicy rozrostu ziarna w wybranych chwilach symulacji. Przy oczywistym wpływie dyskretyzacji przestrzeni uzyskane kształty wciąż są dość zbliżone do okrągłych i kwadratowych. Mniejsze zróżnicowanie rozmiarów i kształtu komórek pozwala uzyskać znacznie lepsze wyniki. Algorytm rozrostu okrągłego ziarna został opisany w poprzednich rozdziałach, algorytm uzyskania kwadratowego i innych kształtów ziaren przedstawiono w dalszych częściach niniejszej pracy. Ponieważ moim głównym celem w tym rozdziale jest demonstracja izotropowego rozrostu na dowolnej siatce, jako dodatkowy przykład został pokazany kwadrat pochyłony o 10° . Jak pamiętamy, klasyczny algorytm rozrostu pozwalał uzyskać tylko dwie orientacje kwadratu w przestrzeni automatów komórkowych na regularnej kwadratowej siatce: kiedy jego boki lub przekątne są zorientowane wzdłuż głównych osi przestrzeni. Natomiast uzyskanie okrągłego lub kwadratowego kształtu na nieregularnej siatce metodami klasycznymi jest bardzo problematyczne.



Rys. 2.19. Okrągły (a) i kwadratowy (b) kształt rosnącego ziarna uzyskany na nieregularnej siatce, w której każda komórka ma unikatowe rozmiary i kształt (Malinowski *et al.* 2009)

2.4.4. Zmodyfikowany algorytm do wyznaczenia czasu przejścia

Przedstawiona w tym rozdziale metoda przyrostowa jest jednak niedokładna. Założenie, że granica ziarna wewnątrz komórki jest płaska (prostoliniowa) wprowadza niewielki błąd.

Druga niedokładność polega na tym, że jeżeli prześledzić, jak gromadzi się historia rozrostu odległej od miejsca zarodkowania komórki, można zauważyć, że linia, wzdłuż

której porusza się granica rosnącego ziarna, nie jest linią prostą, lecz łamaną, która naśladuje wszystkie zmiany kierunku rozrostu ziarna w poprzedzających ją komórkach. Wracając do rysunku 2.14, w przypadku prawej górnej komórki linia łamana, która odzwierciedla historię rozrostu, faktycznie będzie zobrazowana przez dwa wektory \mathbf{r}_p i \mathbf{r}_{b1} . Linia ta prawie nigdy nie kończy się w środku rozpatrywanej komórki, do której natomiast prowadzi linia prosta odpowiadająca wektorowi \mathbf{r}_b . Jeżeli linia łamana jest tej samej długości co linia prosta, kończy się ona w innym punkcie. Natomiast, jeżeli kończy się ona w tym samym punkcie, to jest dłuższa od linii prostej i ziarno potrzebuje więcej czasu, żeby dorosnąć do tego punktu.

W pierwszym algorytmie, za którego pomocą była obliczana odległość komórki od zarodka, taki problem nie powstaje, ale algorytm ten nie może być zastosowany, gdy zmianie ulega prędkość i następuje odkształcenie komórek. Błąd jest wynikiem zastosowania metody przyrostowej, kiedy czas przebywania komórki w stanie przejściowym nie uwzględnia historii zmiany kierunku rozrostu. Wówczas należałoby skorygować jeden z następujących parametrów: czas wejścia granicy ziarna w komórkę lub długość odcinka, który powinna pokonać granica, przechodząc przez komórkę. Wybrałem drugi parametr.

W takim przypadku obliczając wektor kierunku rozrostu \mathbf{r}_b według (2.10), długość L należy obliczyć nie według wzorów (2.12), (2.13), (2.18), (2.19) i (2.21), ale według wzoru następującego:

$$L = \mathbf{r}_b \cdot \mathbf{r}_2 - \mathbf{r}_p \cdot \mathbf{r}_1 \quad (2.22)$$

gdzie \mathbf{r}_1 , \mathbf{r}_2 – jednostkowe wektory kierunku rozrostu, odpowiednio, dla komórek poprzedzającej i rozpatrywanej.

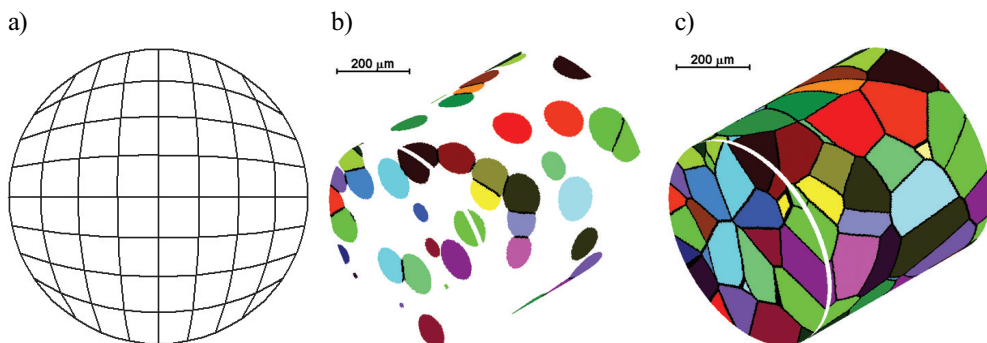
W przypadku rozrostu ziarna w kształcie kuli kierunek wektorów \mathbf{r}_1 i \mathbf{r}_p jest taki sam jak wektorów \mathbf{r}_2 i \mathbf{r}_b . Równanie (2.22) można zapisać następująco:

$$L = \|\mathbf{r}_b\| - \|\mathbf{r}_p\| \quad (2.23)$$

Oznacza to, że długość L można obliczyć jako różnicę długości wektorów \mathbf{r}_b i \mathbf{r}_p . Aby sprecyzować warunek zastosowania tego wzoru, należy zauważyć, że powinna być zmieniona koncepcja określenia czasu przejścia komórki w stan końcowy. Jeżeli poprzednio było to przejście granicy ziarna przez całą komórkę, to teraz będzie to dotarcie granicy ziarna do środka (centrum) komórki. Zatem czas liczony jest nie od granicy do granicy, a od centrum do centrum. Zasadniczo niczego to nie zmienia.

Dodatkowy skutek takiej zmiany jest taki, iż w odróżnieniu od poprzednich przyrostowych algorytmów, w których należało zastosowywać otoczenie Moore'a, w tym algorytmie rodzaj otoczenia nie odgrywa żadnej roli w określaniu czasu przejścia, więc można zastosować prostsze otoczenie von Neumanna, wymagające mniej nakładów na jego badanie.

Na rysunku 2.20a pokazano przykład grubej siatki okrągłej przestrzeni komórkowej, która może być zastosowana do modelowania mikrostruktury w ciekłym drucie. Z kolei na rysunku 2.20b i c przedstawiono etap tworzenia mikrostruktury w cylindrycznej przestrzeni z ziarnami w kształcie kul i jego końcowy wynik. Przedstawione przykłady demonstrowają skuteczność opracowanych algorytmów.



Rys. 2.20. Kształt komórek w przestrzeni okrągłej (lub przekrój przestrzeni cylindrycznej) (a) oraz jeden z etapów (b) i końcowy wynik (c) tworzenia początkowej mikrostruktury w cylindrycznej przestrzeni automatów komórkowych z rosnącymi ziarnami w kształcie kul

2.5. Sterowanie kształtem rosnącego ziarna

2.5.1. Globalne i lokalne współrzędne ziarna

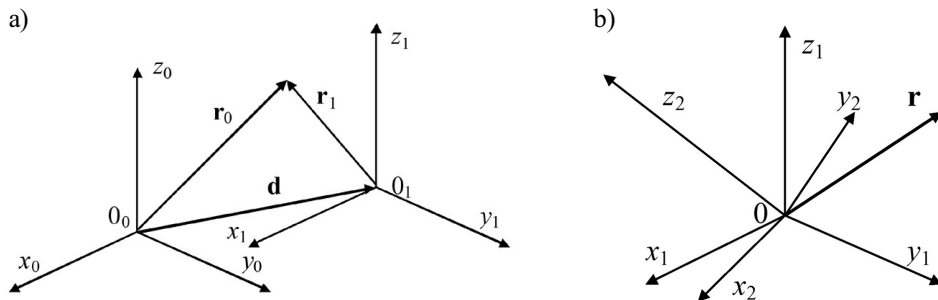
Pierwszym warunkiem sterowania kształtem rosnącego ziarna jest izotropia rozrostu na dowolnej siatce. Ten warunek może być spełniony za pomocą jednego z algorytmów przedstawionych w poprzednim podrozdziale 2.4. Jednak nie w każdym procesie mamy do czynienia z rozrostem ziaren w kształcie kuli. Często kształt rosnących ziaren jest zbliżony do sześciangu, ośmiościanu lub wydłużonych płytek, powinno być zatem możliwe uzyskanie również innych kształtów, a najlepiej, by można było uzyskać kształt dowolny.

Każdy kształt różny od kuli może być zorientowany w przestrzeni w dowolny sposób, natomiast najprościej jest opisać kształt ziarna, rozmieszczając je w sposób szczególny w stosunku do układu współrzędnych. Tak więc taki czy inny kształt łatwiej uzyskać w lokalnym układzie współrzędnych, rozmieszczając ziarno w wybranym punkcie przestrzeni z wyznaczoną orientacją i przekształcając współrzędne lokalne we współrzędne globalne.

Za układem współrzędnych globalnych prostokątnych (kartezjańskim) w przestrzeni dwuwymiarowej czy też trójwymiarowej przemawia wiele argumentów. Choć nie jest to obowiązujące, możliwe są zastosowania w przypadku których można korzystać z układów współrzędnych walcowych (co opisano w poprzednim rozdziale)

lub kulistych. Natomiast w przypadku rosnącego ziarna wybranie lokalnego układu współrzędnych może uprościć opis i ułatwić uzyskanie wybranego kształtu ziarna.

W związku z tym istotne jest przejście od współrzędnych globalnych do lokalnych. Przekształcenie danego układu lub przejście od jednego układu do innego składa się z dwóch operacji: przesunięcia równoległego i obrotu. Rysunek 2.21 ilustruje zastosowane przekształcenia układów.



Rys. 2.21. Przesunięcie (a) i obrót (b) układu współrzędnych

Pierwszy układ lokalny ($0_1x_1y_1z_1$) różni się od układu globalnego ($0_0x_0y_0z_0$) tym, że jego początek jest przesunięty w miejsce zarodkowania ziarna (rys. 2.21a). Operacja ta może być zapisana wektorowo w następującej postaci:

$$\mathbf{r}_1 = \mathbf{r}_0 - \mathbf{d} \quad (2.24)$$

gdzie:

- $\mathbf{r}_0, \mathbf{r}_1$ – wektor określający położenie dowolnego punktu w układzie, odpowiednio, globalnym i lokalnym,
- \mathbf{d} – wektor przemieszczenia układu lokalnego lub położenia układu lokalnego w układzie globalnym.

Drugi lokalny układ jest obrócony względem początku pierwszego lokalnego układu współrzędnych (rys. 2.21b). Wówczas wspólny wektor \mathbf{r} ma dwie reprezentacje – \mathbf{r}_1 i \mathbf{r}_2 , odpowiednio w pierwszym i drugim układzie:

$$\mathbf{r} = \mathbf{r}_1 = x_1\mathbf{i}_1 + y_1\mathbf{j}_1 + z_1\mathbf{k}_1 = x_2\mathbf{i}_2 + y_2\mathbf{j}_2 + z_2\mathbf{k}_2 = \mathbf{r}_2 \quad (2.25)$$

gdzie:

- $\mathbf{i}_1, \mathbf{j}_1, \mathbf{k}_1$ – wektory jednostkowe (wersory) odpowiednio wzdłuż osi x_1, y_1, z_1 ,
- x_1, y_1, z_1 – współrzędne w układzie $0x_1y_1z_1$; $\mathbf{r}_1 = \{x_1, y_1, z_1\}$,
- $\mathbf{i}_2, \mathbf{j}_2, \mathbf{k}_2$ – wektory jednostkowe (wersory) odpowiednio wzdłuż osi x_2, y_2, z_2 ,
- x_2, y_2, z_2 – współrzędne w układzie $0x_2y_2z_2$; $\mathbf{r}_2 = \{x_2, y_2, z_2\}$.

Wówczas obrót może być zapisany w następujący sposób:

$$\mathbf{r}_2 = \mathbf{R}\mathbf{r}_1 \quad (2.26)$$

W wzorze (2.26) \mathbf{R} to macierz obrotu:

$$\mathbf{R} = \begin{bmatrix} \mathbf{i}_1\mathbf{i}_2 & \mathbf{j}_1\mathbf{i}_2 & \mathbf{k}_1\mathbf{i}_2 \\ \mathbf{i}_1\mathbf{j}_2 & \mathbf{j}_1\mathbf{j}_2 & \mathbf{k}_1\mathbf{j}_2 \\ \mathbf{i}_1\mathbf{k}_2 & \mathbf{j}_1\mathbf{k}_2 & \mathbf{k}_1\mathbf{k}_2 \end{bmatrix} \quad (2.27)$$

Przestrzenna orientacja ziarna w przedstawianych automatach komórkowych jest zadawana przez trzy kolejne obroty wokół osi x , y i z . W dalszych obliczeniach orientacja jest wyznaczana przez wektory układu lokalnego, które obliczone są za pomocą równań (2.26) i (2.27). Zatem kiedy granica ziarna dochodzi do komórki, oblicza się współrzędne lokalne według wzorów (2.24) i (2.26). Dalej do obliczenia czasu przejścia komórki w stan końcowy w przypadku ziaren o różnych kształtach wykorzystuje się algorytmy przedstawione poniżej.

Ogólnie rzecz biorąc, algorytmy te nie różnią się od algorytmu omówionego w rozdziale poprzednim, składają się one z tych samych elementów (por. podrozdz. 2.4.2), ale istotne jest w ich przypadku wyznaczenie kierunku rozrostu (innego niż dla kuli) i wyznaczenie prędkości wzdłuż wybranego kierunku rozrostu (anizotropia rozrostu ziarna).

Ze względu na kierunek rozrostu algorytmy można podzielić na algorytmy z globalnym i lokalnym wyznaczeniem kierunku rozrostu. Do pierwszej grupy można zaliczyć algorytm umożliwiający uzyskanie okrągłego kształtu w takiej postaci, w jakiej został on przedstawiony w poprzednich rozdziałach. Kierunek rozrostu wyznaczano globalnie od komórki zarodkowania, natomiast prędkość rozrostu we wszystkich kierunkach jest taka sama. Jeżeli wprowadzić zależność prędkości od kierunku rozrostu, można uzyskać praktycznie dowolny kształt.

Zacznijmy od ogólnego opisu algorytmu rozrostu ziaren o dowolnym kształcie. Algorytm składa się z następujących działań:

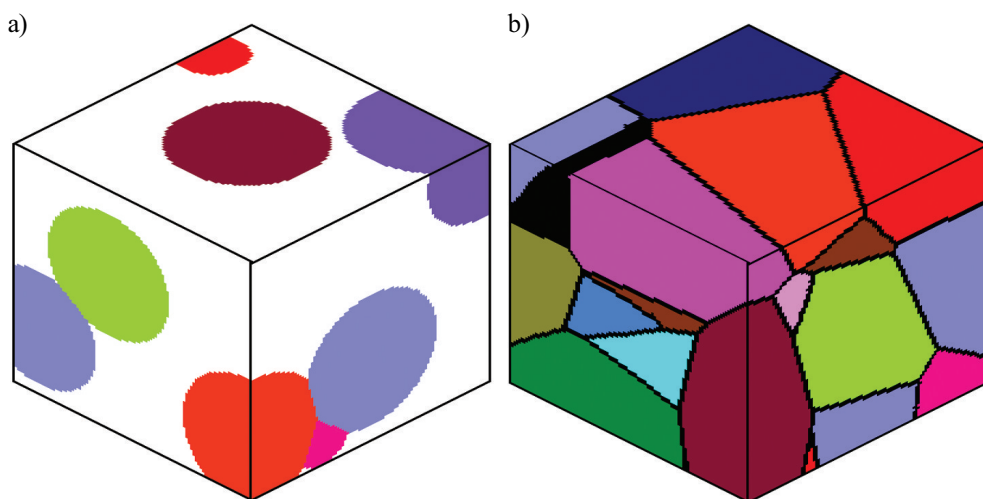
1. Wyznaczenie kierunku rozrostu. Wyznaczyć go można albo globalnie, na podstawie współrzędnych rozpatrywanej komórki i komórki zarodkowania, przy niezmienniej geometrii przestrzeni komórek, przyrostowo, przy uwzględnieniu odkształcenia komórek, albo lokalnie.
2. Wyznaczenie prędkości v_r jako stałej wartości lub funkcji kierunku rozrostu.
3. Obliczenie czasu trwania stanu przejściowego, który na ogół zależy od kierunku rozrostu, kształtu i wymiarów komórki.
4. Obliczenie chwili, w której następuje przejście komórki w stan końcowy.

Faktycznie kształt rosnącego ziarna w tym algorytmie będzie wyznaczany przez zadanie w kroku 2. odpowiedniej funkcji. Jednak ten sam kształt może być czasami uzyskany zarówno za pomocą zadania globalnego, jak i lokalnego kierunku rozrostu. W dalszej części tekstu będą przedstawione podstawowe sposoby uzyskiwania różnych kształtów ziarn.

2.5.2. Algorytm rozrostu ziarna w kształcie kuli

Kierunek rozrostu można uzyskać, stosując metodę globalną, która nie wymaga dużych nakładów obliczeniowych (por. podrozdz. 2.4.4). Lokalnie kierunek rozrostu można wyznaczyć na podstawie normalnej do powierzchni. Metoda ta jest bardziej skomplikowaną i wymaga znacznie większych nakładów, dlatego uważam, że powinna być stosowana tylko w szczególnie uzasadnionych przypadkach.

Prędkość rozrostu ziaren kulistych (lub okrągłych w przestrzeni dwuwymiarowej) v_r jest stałą wartością lub funkcją czasu, niezależną od kierunku rozrostu. Przykład symulacji rozrostu ziaren i uzyskanej mikrostruktury pokazano na rysunku 2.22.



Rys. 2.22. Rozrost ziaren w kształcie kuli: a) ziarna w trakcie rozrostu;
b) uzyskana struktura

2.5.3. Algorytm rozrostu ziarna w kształcie elipsoidy

Przedstawione tu podejście opiera się na wyznaczeniu globalnego kierunku rozrostu. Elipsoida wymaga wprowadzenia do obliczeń kilku dodatkowych parametrów, takich jak stosunek długości trzech głównych osi oraz w razie dowolnej orientacji w przestrzeni – odpowiednich kątów wyznaczających taką orientację. Stosunek długości trzech osi decyduje o kształcie ziarna i może być zadany przez dwie liczby: stosunek dwóch krótszych osi na przykład do najdłuższej. Jednak uważam, że zadając trzy długości półosi w jednej symulacji, można tworzyć ziarna o tym samym kształcie, ale rosnące z różną prędkością. Wówczas trzy parametry a , b , c są długościami trzech półosi elipsoidy, a v_x , v_y , v_z oznaczają prędkości rozrostu wzdłuż odpowiednich lokalnych osi.

Sposobów opisu powierzchni elipsoidy jest kilka, przy zastosowaniu różnych układów współrzędnych. Tu zastosuję metodę parametryczną, wówczas równania opisujące elipsoidę przyjmą następującą postać:

$$\begin{aligned}x &= a \sin u \cos v \\y &= b \sin u \sin v \\z &= c \cos u\end{aligned}\tag{2.28}$$

gdzie:

x, y, z – współrzędne punktu na powierzchni elipsoidy,
 u, v – parametry.

Należy jednak pamiętać, że parametry u i v nie są kątami w układzie współrzędnych kulistych. Związek między parametrami a współrzędnymi opisują następujące wzory:

$$\begin{aligned}\operatorname{tg} v &= \frac{a}{b} \operatorname{tg} \varphi \\ \operatorname{tg} u &= \frac{c}{\sqrt{a^2 \cos^2 v + b^2 \sin^2 v}} \operatorname{tg} \theta\end{aligned}\tag{2.29}$$

gdzie:

θ – kąt między osią z a kierunkiem do punktu na powierzchni elipsoidy,
 φ – kąt pomiędzy osią x a rzutem kierunku do punktu na powierzchni elipsoidy na płaszczyznę Oxy .

Prędkość rozrostu v_r obliczamy w następujący sposób:

$$v_r = \sqrt{(v_x \sin u \cos v)^2 + (v_y \sin u \sin v)^2 + (v_z \cos u)^2}\tag{2.30}$$

Kierunek rozrostu jest wyznaczany globalnie i wektorem tym jest \mathbf{r}_b . Z orientacji ziarna w przestrzeni mamy trzy wersory lokalnego układu współrzędnych ziarna \mathbf{i}, \mathbf{j} i \mathbf{k} . Wówczas do wyznaczenia kątów φ i θ należy zastosować następujące wzory:

$$\cos \theta = \frac{\mathbf{r}_b \cdot \mathbf{k}}{|\mathbf{r}_b|}\tag{2.31}$$

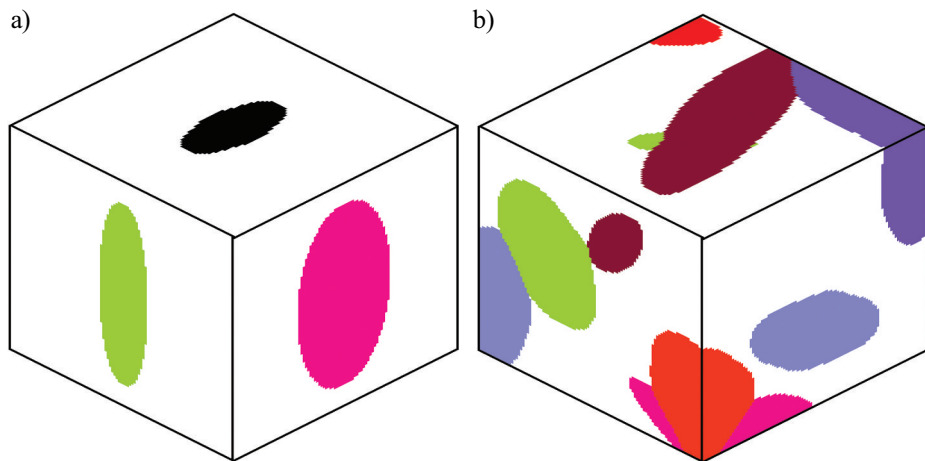
$$\cos \varphi = \frac{\mathbf{r}_b \cdot \mathbf{i}}{|\mathbf{r}_b| \sin \theta}\tag{2.32}$$

Na rysunku 2.23 pokazano rozrost ziaren w kształcie elipsoidy ze stosunkiem półosi $a:b:c = 2:1:4$. W pierwszym przypadku ziarna są zorientowane wzdłuż głównych osi przestrzeni, w drugim ich orientację wybierano losowo.

W przypadku dwuwymiarowym wystarczy założyć $\theta = 90^\circ$.

Algorytm może być zastosowany również do uzyskania kulistego kształtu ziaren, wówczas $a = b = c$, jednak jest on bardziej skomplikowany, wymaga dodatkowych

obliczeń i z tego względu jego użycie nie zawsze jest uzasadnione. Może być to celowe w przypadku, gdy podczas rozrostu ziarna zmieniają się warunki, co powoduje, że na początku rosnące ziarna przybierają kształt kuli, a później elipsoidy lub na odwrót.



Rys. 2.23. Rozrost ziaren w kształcie elipsoidy: a) ziarna zorientowane wzdłuż głównych osi przestrzeni; b) losowa orientacja ziaren

2.5.4. Algorytm rozrostu ziarna w kształcie prostopadłościanu

Prostopadłościan składa się z sześciu powierzchni:

$$x = \pm a, \quad y = \pm b, \quad z = \pm c \quad (2.33)$$

Dodatkowym elementem algorytmu rozrostu ziarna w przypadku wielościanów lub innych brył ograniczonych ścianami i mających jawnie wyrażone krawędzie jest ustalenie, na której ze ścian bryły leży analizowana komórka. Tu można zastosować dwa podejścia: globalne lub lokalne. Zarodek zajmuje jedną komórkę i faktycznie jeszcze nie ma ścian, więc musimy w jakiś sposób określić wszystkie powierzchnie, czyli zastosować podejście globalne.

W celu wyznaczenia, na której ze ścian leży badana komórka, należy wyznaczyć trzy kosinusy kierunkowe w lokalnym układzie współrzędnych ziarna:

$$\begin{aligned} \cos \alpha &= \frac{\mathbf{r}_b \cdot \mathbf{i}}{|\mathbf{r}_b|} \\ \cos \beta &= \frac{\mathbf{r}_b \cdot \mathbf{j}}{|\mathbf{r}_b|} \\ \cos \gamma &= \frac{\mathbf{r}_b \cdot \mathbf{k}}{|\mathbf{r}_b|} \end{aligned} \quad (2.34)$$

Wówczas w zależności od tego, czy największa wartość przyjmuje $\frac{|\cos \alpha|}{a}$, $\frac{|\cos \beta|}{b}$ czy $\frac{|\cos \gamma|}{c}$, wybieramy odpowiednią płaszczyznę $x = \pm a$, $y = \pm b$ lub $z = \pm c$. Znak \pm odpowiada znakowi kosinusa.

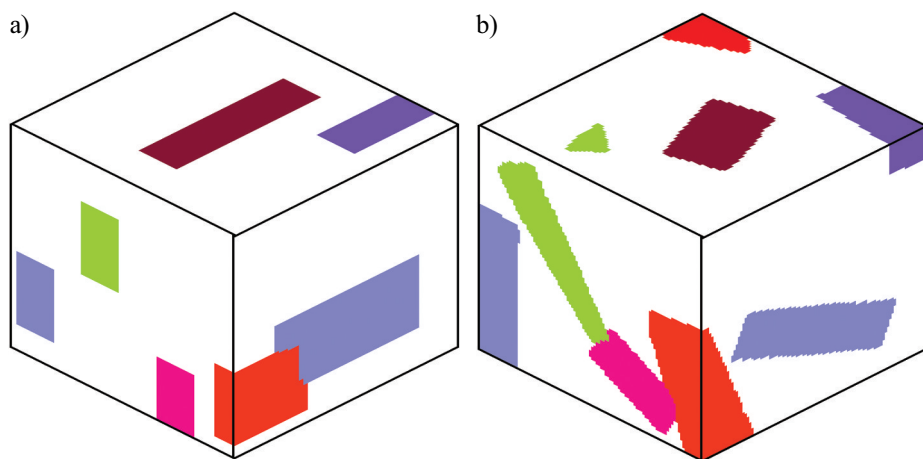
W przypadku lokalnego wyznaczania ściany zakłada się, że przynależność do danej płaszczyzny jest przez komórkę automatycznie „przejmowana w spadku” po komórce rodzicielskiej, od której „diedziczy” ona także rozrost ziarna. Jednak zastosowanie tego algorytmu wiąże się z trudnościami w dokładnym wyznaczeniu położenia krawędzi i wierzchołków, w szczególności gdy różnica między a , b i c jest duża.

Przy lokalnym wyznaczaniu kierunku rozrostu kierunek jest równy odpowiedniemu wersorowi lokalnego układu współrzędnych ziarna. Czyli dla płaszczyzn $x = \pm a$ będzie to $\pm \mathbf{i}$ itd. Natomiast prędkość rozrostu w zależności od płaszczyzny będzie równa v_x , v_y , lub v_z . W przypadku globalnego wyznaczania kierunku rozrostu jest on równy \mathbf{r}_b , a prędkość rozrostu będzie równa odpowiednio $\frac{v_x}{\cos \alpha}$, $\frac{v_y}{\cos \beta}$ lub $\frac{v_z}{\cos \gamma}$.

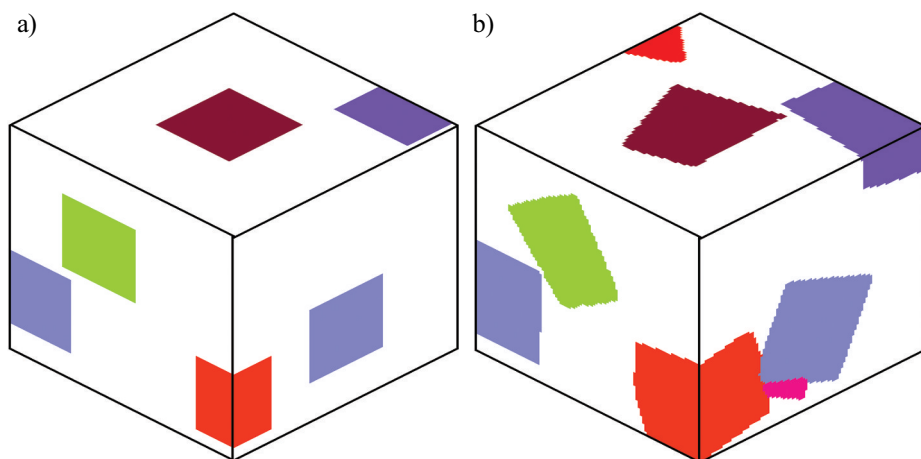
Przykłady wzrostu ziaren w kształcie prostopadłościanu przedstawiono na rysunku 2.24. Wyniki różnią się w zależności od tego, czy orientacja ziarna jest narzucona, czy losowa. Stosunek krawędzi wszystkich prostopadłościanów $a:b:c = 4:1:2$. Oczywiście ten stosunek może być również zadany dla każdego prostopadłościanu niezależnie.

W przypadku sześcianu $v_x = v_y = v_z$. Przykłady rozrostu ziaren w kształcie sześcianu pokazano na rysunku 2.25. Jak poprzednio, w pierwszym przypadku orientacja jest zadana z góry, w drugim wybierano ją przypadkowo.

Gdy analizujemy rozrost ziaren w przestrzeni dwuwymiarowej, bierzemy pod uwagę tylko dwie współrzędne: x i y .



Rys. 2.24. Rozrost ziaren w kształcie prostopadłościanu: a) ziarna zorientowane wzdłuż głównych osi przestrzeni; b) losowa orientacja ziaren



Rys. 2.25. Rozrost ziaren w kształcie sześciianu: a) ziarna zorientowane wzdłuż głównych osi przestrzeni; b) losowa orientacja ziaren

2.5.5. Algorytm rozrostu ziarna w kształcie ośmiościanu

Algorytm rozrostu ziarna w kształcie ośmiościanu jest bardzo podobny do wyżej opisanego algorytmu rozrostu ziarna w kształcie prostopadłościanu. Różnica polega tylko na opisie ścian rosnącego ziarna (por. wzory (2.33) i (2.35)) i wyznaczeniu kierunku rozrostu. Płaszczyzny ośmiościanu foremego można opisać następującym wzorem:

$$\pm x \pm y \pm z = 1 \quad (2.35)$$

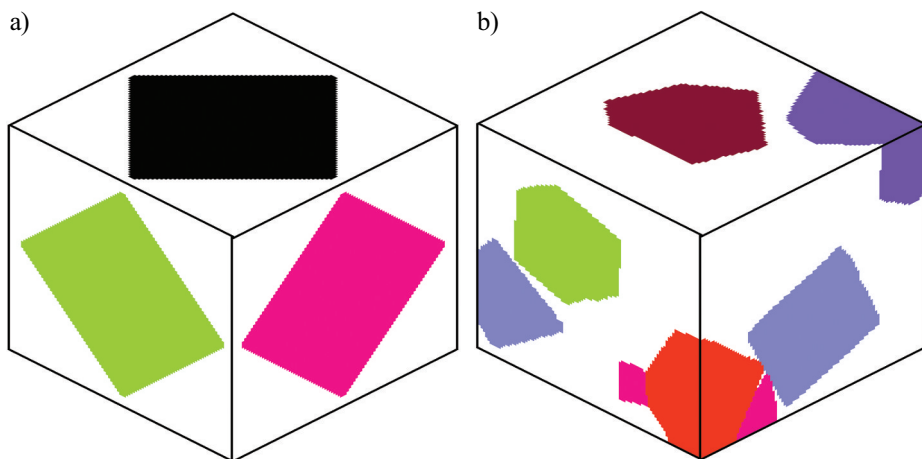
W przypadku ośmiościanu o różnych długościach przekątnych wzór jest następujący:

$$\pm \frac{x}{a} \pm \frac{y}{b} \pm \frac{z}{c} = 1 \quad (2.36)$$

Dalej, aby określić, na której ścianie leży komórka, należy wyznaczyć trzy kosinusy kierunkowe w lokalnym układzie współrzędnych według wzoru (2.34). Znak kosinusów wskaże, do którego z oktantów lokalnych współrzędnych, czyli na której ze ścian, leży komórka. Kierunek rozrostu odpowiada normalnej do powierzchni. W przypadku ośmiościanu foremego będzie go wyrażał następujący wzór:

$$\mathbf{r} = \frac{(\mathbf{i} + \mathbf{j} + \mathbf{k})\sqrt{3}}{3} \quad (2.37)$$

Przykłady rozrostu ziaren w kształcie ośmiościanów foremnych przedstawiono na rysunku 2.26.



Rys. 2.26. Rozrost ziaren w kształcie ośmiościanu foremnego: a) ziarna zorientowane wzdłuż głównych osi przestrzeni; b) losowa orientacja ziaren

2.5.6. Algorytm rozrostu ziarna w kształcie walca

Ten algorytm jest podobny do omówionych wyżej. Powierzchnia walca składa się z trzech płaszczyzn, dwóch płaskich i jednej cylindrycznej:

$$\begin{aligned} x &= \pm H \\ y^2 + z^2 &= R^2 \end{aligned} \quad (2.38)$$

gdzie:

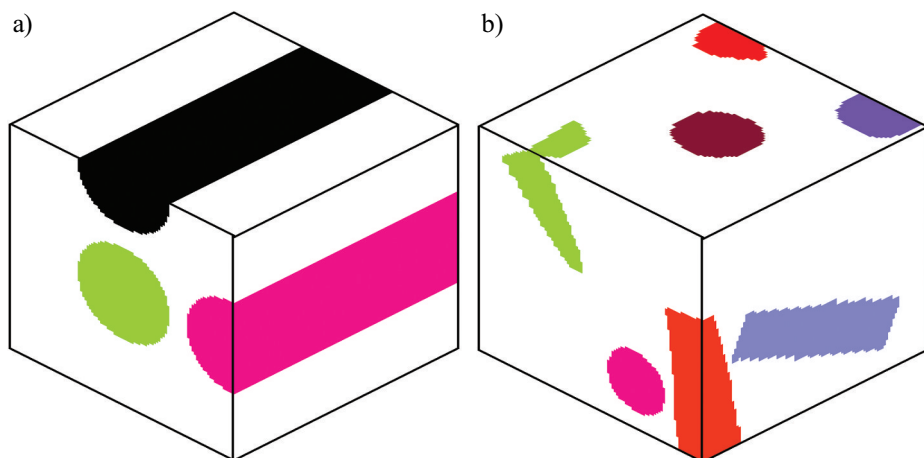
H – wysokość walca,
 R – średnica walca.

Kształt walca wyznaczano na podstawie stosunku jego wysokości H do średnicy R lub zadając odpowiednie prędkości rozrostu ziarna v_H i v_R . W celu wyznaczenia przynależności komórki do jednej z powierzchni wystarczy obliczyć tylko jeden kosinus kierunkowy z trzech podanych wzorem (2.34), a mianowicie $\cos \alpha$, czyli kąt z osią x . Przy czym jeżeli:

$$|\cos \alpha| < \frac{H}{\sqrt{H^2 + R^2}},$$

to komórka należy do powierzchni cylindrycznej, w przeciwnym razie – do powierzchni płaskich. Kierunek rozrostu w przypadku powierzchni płaskich odpowiada kierunkowi wektora \mathbf{i} , a w przypadku powierzchni walcowej – prostopadłej do powierzchni, czyli kierunkowi promienia przekroju walca prostopadłego do jego osi.

Dwa przykłady zadaną i losową orientacją walca zostały pokazane na rysunku 2.27. Stosunek wysokości do średnicy wynosi w tych przykładach $H:R = 3:1$.



Rys. 2.27. Rozrost ziaren w kształcie walca: a) ziarna zorientowane wzdłuż głównych osi przestrzeni; b) losowa orientacja ziaren

Można też uzyskać walec o innym kształcie, na przykład eliptycznym, parabolicznym lub hiperbolicznym. Przeciwległe powierzchnie walca mogą być nie tylko płaskie, ale również wypukłe, wklęsłe, o kształcie kulistym lub innym. Można też zastąpić powierzchnię cylindryczną powierzchnią z krawędziami, na przykład o ścianach sześć- lub ośmiobocznych. Stosując opisane w tym rozdziale zasady, można uzyskać ziarna o dowolnym kształcie.

3. Warunki brzegowe i reorganizacja przestrzeni komórkowej

Warunków brzegowych nie zalicza się do podstawowych parametrów automatów komórkowych, ale są one pochodną wybranej geometrii przestrzeni. Przestrzeń nieskończona nie ma granic, więc nie potrzebuje warunków brzegowych, nie ma gdzie ich zastosować. Jednak występujące w materiałach zjawiska mikrostrukturalne zachodzą albo w niewielkiej objętości, i wówczas powstają ograniczenia naturalne związane z wymiarami próbki, albo konieczne jest zasymulowanie tylko pewnej przestrzeni ze względu na ograniczenia modelowe (objętość pamięci, czas obliczeń itd.).

W podrozdziale 3.2 rozpatrzono większość możliwych warunków brzegowych w przestrzeni, w której nie zachodzą zmiany lub zmiany zachodzące nie wpływają istotnie na jej topologię.

Wprowadzenie odkształcenia do modelowej przestrzeni automatów komórkowych może doprowadzić do istotnego zniekształcenia zarówno całej przestrzeni, jak i komórek. Wówczas powstaje konieczność przebudowy przestrzeni, modyfikacji kształtu komórek i reorganizacji automatów komórkowych. Podczas reorganizacji następują zmiany granic przestrzeni, może być zmieniona jej objętość oraz globalne wymiary. Takie działanie rzutuje na warunki brzegowe, które nie pozostają bez zmian – zmienia się zatem rodzaj warunków brzegowych i cała topologia przestrzeni, która również podlega przekształceniu podczas modelowania. Wybór odpowiednich warunków oraz sterowanie nimi również zostaną omówione w dalszej części tej pracy. W podrozdziale 3.3 zostały przedstawione podstawowe zasady zastosowania warunków brzegowych w sytuacji, gdy w związku ze zmianą wymiarów przestrzeni i komórek powstaje konieczność przebudowy przestrzeni.

3.1. Wymagania stawiane warunkom brzegowym

W przypadku modelowania małego obiektu, o wymiarach porównywalnych z wymiarami przestrzeni automatu, należy uwzględnić jedynie warunki fizyczne. W klasyfikacji podanej w podrozdziale 1.2 taka sytuacja została opisana warunkami stałymi lub zerowymi. Nic nie stoi jednak na przeszkodzie, by te warunki zmieniać – wówczas można je ogólnie nazwać warunkami narzuconymi. Ze względu na modelowanie przedstawionych

w tej książce zjawisk mikrostrukturalnych, kiedy automaty nie oddziałują na strefy poza granicami materiału i na nie nie wpływają obszary zewnętrzne, będą rozpatrywane jako otwarte (a ściśle rzecz biorąc: półotwarte), czyli takie, w odniesieniu do których zakłada się, że wszystko, co dociera do granicy i ją przekracza, znika, i cała informacja o strefach zewnętrznych jest niedostępna. Innymi słowy, na granicy nie dzieje się nic szczególnego.

W drugim przypadku przestrzeń automatów komórkowych reprezentuje małą część materiału i graniczy z tym samym materiałem, w którym zachodzą te same zjawiska. Na modelową przestrzeń wpływają zewnętrzne strefy, a z kolei automaty komórkowe wpływają na strefy zewnętrzne. Właśnie wówczas i powstaje problem (drugi) ograniczoności przestrzeni i określenia warunków brzegowych. Można tu powtórzyć tezę z pierwszego rozdziału, że warunki brzegowe powinny spełnić jedno podstawowe wymaganie: powinny wyeliminować efekt brzegu, czyli wszystkie elementy (komórki) powinny być w tych samych warunkach z takim samym otoczeniem, niezależnie od tego, czy znajdują się w środku przestrzeni, czy na jej granicy. Komórka znajdująca się na brzegu przestrzeni ma mieć takie warunki i takich sąsiadów jak każda inna (umiejscowiona w środku przestrzeni), ponieważ ma ona wokół siebie taki sam materiał jak inne komórki, a w punkcie tym zachodzą takie same zjawiska i procesy, jak w każdym innym miejscu przestrzeni modelowej. Wszystko to powinny uwzględnić warunki brzegowe – wówczas będą one adekwatne.

Stosuje się dwa podejścia do tego zagadnienia. Pierwsze polega na wyeliminowaniu wpływu zewnętrznych obszarów na automaty komórkowe. Jeżeli automaty nie będą wpływały na strefy zewnętrzne (jak w przypadku narzucenia warunków półotwartych), to i one nie będą oddziaływały na automaty. Efektem tego podejścia będą warunki brzegowe zamknięte, czyli automaty, które zamykają się na siebie. Drugim, bardziej skomplikowanym sposobem jest zrównoważenie wpływu automatów komórkowych na zewnętrzne obszary i wpływu zewnętrznych stref na automaty. Umożliwiają to warunki brzegowe otwarte (w pełni otwarte, w odróżnieniu od półotwartych).

Ograniczoność przestrzeni modelowej zazwyczaj implikuje zastosowanie skończonej siatki. Rodzaj, kształt i rozmiary siatki mogą ułatwiać wprowadzenie odpowiednich warunków brzegowych lub to utrudniać. W przypadku warunków otwartych i półotwartych siatka zwykle nie odgrywa istotnej roli, natomiast gdy warunki brzegowe są zamknięte, łatwiej jest uzyskać pożądane rozwiązanie przy zastosowaniu siatki regularnej.

Zatem warunki brzegowe często sprowadzane są do periodycznych lub okresowych, w przypadku których określa się współzależność przeciwległych granic przestrzeni.

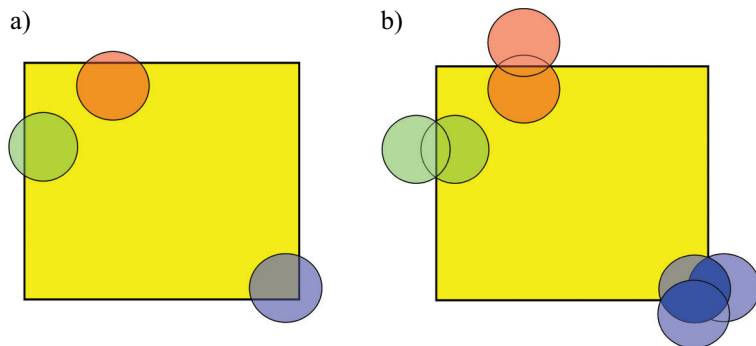
W Polsce często jest stosowana klasyfikacja warunków brzegowych odnosząca się do automatów komórkowych do modelowania poruszających się obiektów. Warunki brzegowe po raz pierwszy sformułowano w odniesieniu do poruszających się cząstek, więc pierwsza klasyfikacja odzwierciedlała oddziaływanie cząstek (promieniowanie) z powierzchnią. Istnieją trzy rodzaje tego oddziaływania: odbicie, przepuszczanie i pochłanianie. Warunki brzegowe z odbiciem wprowadzały idealne sprężyste zderzenie; przepuszczanie oznaczało, że cząstka przechodziła przez granicę i pojawiała się po przeciwnej stronie. Wówczas gdy wprowadzano pochłanianie, cząstka opuszczała

przestrzeń i przestawała istnieć. Przestrzeń z odbiciem lub przepuszczaniem jest zamknięta, natomiast z pochłanianiem – otwarta lub półotwarta, ponieważ cząstki często poruszają się tylko na zewnątrz. Jednak w CA do modelowania mikrostruktury przyjęto nieco inną klasyfikację. Warunki brzegowe określają, jak automaty zachowują się w komórkach znajdujących się na granicy przestrzeni. Warunki brzegowe z przepuszczalnymi granicami nazwano periodycznymi (okresowymi), z granicami pochłaniającymi – otwartymi. Faktycznie, innych warunków brzegowych nie da się znaleźć w literaturze dotyczącej modelowania mikrostruktury za pomocą automatów komórkowych. Jednak odmienne warunki brzegowe lub modyfikacje już powstałych istnieją. Mogą być one skuteczniejsze i bardziej przydatne w niektórych zastosowaniach, co zostanie omówione w dalszej części tego rozdziału.

3.2. Warunki brzegowe przy stałej topologii automatów komórkowych

W literaturze naukowej dotyczącej modelowania mikrostruktury za pomocą automatów komórkowych rozróżnia się dwa podstawowe rodzaje warunków brzegowych: otwarte (w zależności od właściwości będziemy je nazywać półotwartymi lub otwartymi) i zamknięte.

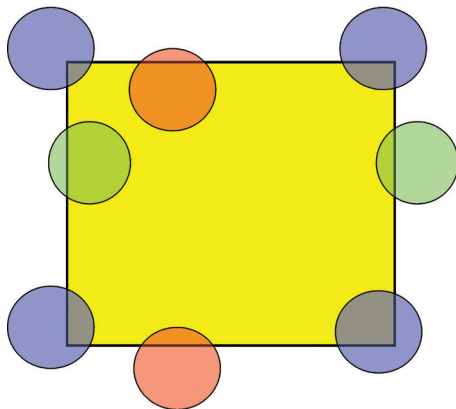
Półotwarte warunki brzegowe charakteryzują się pełną niezależnością od siebie przeciwległych boków (lub płaszczyzn w przestrzeni trójwymiarowej), tak jakby przestrzeń poza wyznaczonymi granicami nie istniała. Jeżeli na przykład granica rozrastającego się ziarna dociera do granicy przestrzeni, ziarno nadal rośnie, jednak o tym już nic nie wiadomo (rys. 3.1), odbywa się to już bowiem poza zasięgiem naszej obserwacji. Nieznane są rozmiary tego ziarna ani jego kształt, nie ma zatem pełnej o nim informacji, nie jest ona kompletna. Można tę sytuację rozpatrywać w nieco inny sposób, mniej naturalny, jak to pokazano na rysunku 3.1b. W takim przypadku ziarno pozostaje w całości wewnątrz modelowej przestrzeni.



Rys. 3.1. Półotwarte warunki brzegowe. Wyjście rosnących ziaren poza granice przestrzeni

Drugi rodzaj warunków brzegowych zwykle nazywano periodycznymi, chociaż termin „warunki zamknięte” jest ogólniejszy. Brzegi przestrzeni automatów można w pewien sposób ze sobą połączyć, „sklejając” przeciwległe granice. Wówczas dwuwymiarowa przestrzeń automatów będzie rozmieszczona na powierzchni torusa, a przestrzeń trójwymiarowa na hipertorusie. W przestrzeni z takimi periodycznymi warunkami komórka położona na granicy przestrzeni sąsiaduje z komórką położoną na przeciwległej granicy przestrzeni. Na przykład rozpatrzmy automaty komórkowe o wymiarach n_x , n_y i n_z . Komórka ze współrzędnymi $\{i = n_x, j, k\}$ leży na granicy przestrzeni i ma sąsiadów w każdym kierunku, również w kierunku x , czyli w kierunku wykraczającym poza obszar automatów komórkowych. Tym sąsiadem przy zadaniu zwykłych periodycznych warunkach brzegowych jest komórka z przeciwległej granicy, o współrzędnych $\{1, j, k\}$. Natomiast w półotwartej przestrzeni komórka $\{n_x, j, k\}$ nie ma tego sąsiada $\{1, j, k\}$. Liczba sąsiadów takiej komórki jest mniejsza i działa ona w innych warunkach.

Wszystkie komórki periodycznej przestrzeni są równoważne i nie oddziałuje na nie żaden obszar zewnętrzny, ponieważ nie istnieją granice przestrzeni. Kiedy ziarno dorasta do nominalnej granicy przestrzeni, rośnie dalej, pojawiając się z przeciwległej strony (rys. 3.2). Nie jest to inne ziarno, nie mamy również dwu- lub wielokrotnej reprezentacji tego samego ziarna. Jest to to samo ziarno, tyle że znalazło się ono po drugiej stronie, ponieważ torus został przecięty właśnie w tym miejscu. Tak samo na zwykłej mapie świata możemy zobaczyć podwójną reprezentację Alaski, chociaż wszyscy wiemy, że jest ona jedna i dotrzeć do niej można zarówno poruszając się na wschód, jak i na zachód. Jeżeli torus zostanie przecięty w innym miejscu, ziarno może być całe, w jednej reprezentacji. Ten fakt często pozostaje nieuświadamiany i takie ziarno może być traktowane albo jako kilka ziaren, albo jako kilka reprezentacji jednego ziarna. Tego typu rozbieżności nie powstają, kiedy zadane są otwarte warunki brzegowe. W nich ziarno zawsze jest w jednej reprezentacji, niezależnie od tego, gdzie się znajduje: w środku (całe) czy na granicy (obcięte).



Rys. 3.2 Periodyczne warunki brzegowe. Dojście rosnących ziaren do granicy przestrzeni, przekroczenie jej i pojawienie się ich z przeciwległej strony

Jeżeli istnieją przynajmniej dwa rodzaje warunków brzegowych, to – po pierwsze – mogą istnieć różne ich kombinacje (na przykład niektóre półotwarte, niektóre periodyczne: przestrzeń na powierzchni cylindra), a po drugie – powstaje problem wyboru. Zadanie nie jest łatwe. Literatura naukowa dotycząca zastosowań automatów komórkowych do modelowania zjawisk mikrostrukturalnych daje miażdżącą przewagę periodycznym warunkom brzegowym. Brakuje jednak uzasadnienia tej preferencji. Spróbujemy takie uzasadnienie znaleźć i odpowiedzieć na pytanie, czy taki wybór zawsze jest słuszny. Udzielone na kilku międzynarodowych konferencjach, przez przedstawicieli kilku ośrodków z różnych krajów odpowiedzi na pytania dotyczące warunków brzegowych nie wydają się do końca przekonujące. Najczęściej spotykane jest rozumowanie typu: „Wszyscy wykorzystują periodyczne warunki, ponieważ one są lepsze do symulacji rozwoju mikrostruktury”. Trudno uznać taką argumentację za przekonującą, gdyż w istocie nie jest to żadne uzasadnienie.

Niektórzy jednak wysnuwają z tego następujący wniosek: połączenie brzegów (granic) stwarza iluzję przestrzeni nieskończonej z powtarzającym się wzorcem, zatem w tym przypadku możemy wykorzystać przestrzeń mniejszą niż w przypadku zastosowania otwartych warunków brzegowych. Jest to uzasadnienie nieco bardziej przekonujące, ale również niewystarczające. Nawet jeżeli pierwsza jego część jest słuszna, druga tylko mnoży pytania. Jeżeli możemy wykorzystać przestrzeń mniejszą, to o ile mniejszą, o ile można ją zmniejszać, żeby powtarzający się wzorec nie wpływał na wyniki symulacji? Po co dążymy do zwiększenia przestrzeni automatów komórkowych, do zastosowania coraz większej liczby komórek? Przyjrzyjmy się dokładniej pierwszej części przytoczonego uzasadnienia. Dla porównania na rysunku 3.3 przedstawiono wyniki generatora początkowej struktury z periodycznymi (rys. 3.3a) i półotwartymi (rys. 3.3b) warunkami brzegowymi. Zarodki zostały rozmieszczone w tych samych miejscach, kształt rosnących ziaren był okrągły. Można zauważyć, że środkowa część rysunków jest taka sama, różnice zaznaczają się tylko bliżej brzegów. Wydaje się, że struktura uzyskana przy zadaniu półotwartych warunków brzegowych jest grubsza. Jednak jeżeli przypomnimy sobie, że nie są to różne ziarna po dwóch stronach przestrzeni, lecz te same, iluzja drobniejszej struktury znika. W jaki sposób średnia wielkość ziaren może być różna, skoro mamy takie same wymiary przestrzeni i taką samą liczbę ziaren? Zatem jeżeli jedno ziarno jest większe, to inne ziarno musi być mniejsze. Wydaje się więc, że periodyczne warunki powodują, iż drobne ziarna przy granicach stają się większymi, a grube mniejszymi. Jest to pierwsza różnica, którą można zaobserwować w strukturze końcowej.

Przejdziemy teraz do „nieskończoności” przestrzeni z warunkami periodycznymi. Przykład takiej przestrzeni przedstawiono na rysunku 3.4a. Widoczne są na nim rozmieszczone obok siebie cztery struktury pokazane uprzednio na rysunku 3.3a. Podobny efekt można otrzymać za pomocą struktur uzyskanych z zastosowaniem półotwartych warunków brzegowych – należy jedynie wprowadzić odpowiednie odbicie wzorca (jak na rysunku 3.1b). Wynik takiego rozprzestrzeniania się początkowej struktury (rys. 3.3b) przedstawiono na rysunku 3.4b. W tym przypadku ziarna na granicach są różne i należałoby je oznaczyć innym kolorem. Z tego przykładu wynika, że „nieskończoność”

przestrzeni nie jest jednak właściwością wyłącznie warunków periodycznych i nie może stanowić uzasadnienia wyboru. Na pewno można stwierdzić, że powstała symetryczna struktura (rys. 3.4b) jest nierealna, ale również nierzeczywista jest struktura z periodycznie powtarzającymi się fragmentami (rys. 3.4a). Jedyna przewaga warunków periodycznych polega na tym, że nie dają one widocznych granic połączenia przeciwnych obszarów, natomiast przy warunkach półotwartych łatwo wyznaczyć granice przestrzeni automatów i można rozdzielić złożone fragmenty przestrzeni.

a)

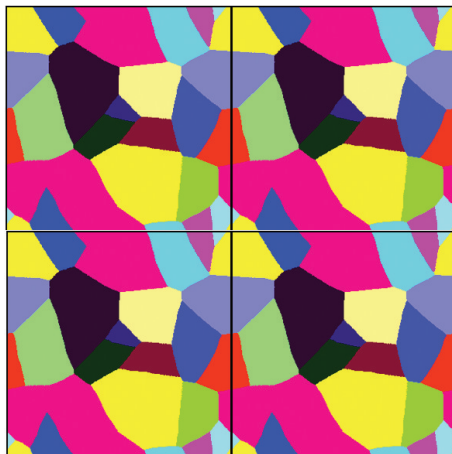


b)

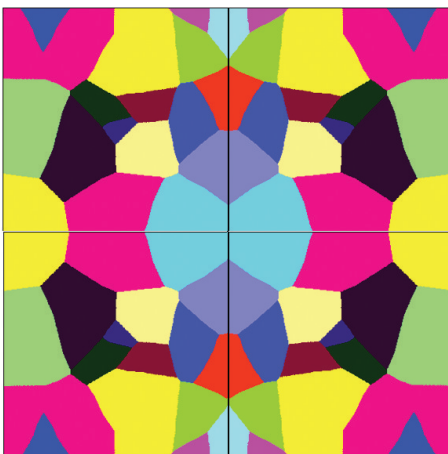


Rys. 3.3. Początkowa struktura uzyskana z różnymi warunkami brzegowymi:
a) periodycznymi; b) półotwartymi

a)



b)

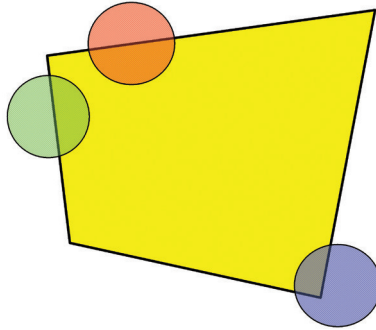


Rys. 3.4. Stworzenie nieskończonej struktury w wyniku łączenia struktur uzyskanych z różnymi warunkami brzegowymi: a) periodycznymi; b) półotwartymi

Jednak bardzo istotna jest inna różnica między periodycznymi a półotwartymi warunkami brzegowymi przy symulacji rozrostu ziaren (na przykład wpływ na kinetykę rekrytalizacji). Brak „efektu brzegu” ze względu na równoważność wszystkich komórek (brak granicy przestrzeni rozmieszczonej na torusie) umożliwia bardziej realistyczną symulację procesu. Dlatego w modelowaniu stosowano głównie periodyczne warunki brzegowe. Najistotniejszym mankamentem półotwartych warunków brzegowych przy rozroście ziaren jest ich półotwartość, półprzezroczystość, sprawiają, że granica jest przezroczysta tylko w jednym kierunku – na zewnątrz i nieprzezroczysta w drugim – do wewnątrz. Wewnętrzne ziarna mogą wyrastać poza obszar automatów, ale ziarna zewnętrzne nie mogą wrastać w przestrzeń komórkową. Takie warunki nazwano w niniejszej książce półotwartymi, w odróżnieniu od warunków określanych jako otwarte. Idea w pełni otwartych lub skrótkowo: otwartych warunków brzegowych będzie przedstawiona nieco dalej w tym rozdziale.

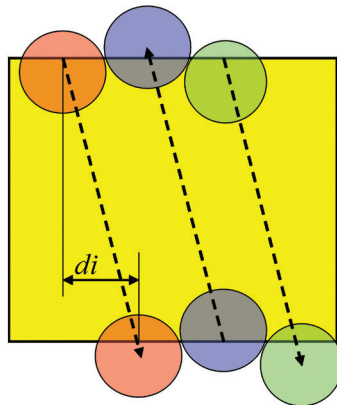
A zatem przestrzeń automatów komórkowych zwykle jest zamknięta, czyli stosowano periodyczne warunki brzegowe. Związane to jest głównie z modelowaniem rozrostu ziarna, czy to podczas krystalizacji, czy rekrytalizacji, czy też związanego z innymi zjawiskami fizycznymi. Wówczas zamykanie przeciwległych brzegów (granic) przestrzeni eliminuje wpływ obszaru zewnętrznego na przebieg rozrostu. Rozpatrując przestrzeń automatów komórkowych jako małą część całej przestrzeni materiału, powinniśmy uwzględnić fakt, że prawdopodobieństwo wyjścia rosnącego ziarna poza przestrzeń automatów komórkowych jest takie samo jak prawdopodobieństwo wejścia ziaren powstałych na zewnątrz przestrzeni automatów komórkowych do tej przestrzeni. Najprostszym rozwiązaniem jest zamknięcie przestrzeni w torus lub hipertorus.

Dopóki modelowanie odbywa się wyłącznie za pomocą automatów komórkowych, fakt ten nie odgrywa istotnej roli i na ogół jest pomijany. Natomiast łączenie kilku metod w jednym modelu (na przykład automatów komórkowych z elementami skończonymi) zmienia tę sytuację zasadniczo. W metodzie elementów skończonych warunki brzegowe na przeciwległych granicach będą różne i spowodują pojawienie się nieciągłości w zamkniętej przestrzeni automatów komórkowych przy modelowaniu odkształcenia. Wówczas rodzi się pytanie, jakie warunki zastosować w odniesieniu do przestrzeni automatów komórkowych, która została odkształcona, jak to pokazano na rysunku 3.5. Jak to przedstawić na powierzchni torusa, kiedy przeciwległe granice mają różne długości? Dlatego w modelu łączącym elementy skończone z automatami komórkowymi przestrzeń nie zawsze będzie periodyczna. Gdy istnieje nieciągłość pomiędzy jej częściami, nie można już rozpatrywać ziarna leżącego po różnych stronach przestrzeni jako jednego. Takie nieciągłości mogą powstać również podczas modelowania procesu rekrytalizacji, kiedy stare ziarno może zostać podzielone przez nowe rosnące ziarna. A zatem model oparty na automatach komórkowych koniecznie powinien zawierać algorytm sprawdzania nieciągłości ziarna i odpowiedniego przypisania części tego ziarna innym ziarnom. Taki algorytm został opisany w rozdziale 5.



Rys. 3.5. Warunki brzegowe dla odkształconej przestrzeni

Wróćmy jednak do poprzedniego zagadnienia. Można wprowadzić inne zamknięte warunki brzegowe. Zamykanie przestrzeni może być przeprowadzone z przesunięciem po płaszczyźnie (lub linii) sklejaną. Idea periodycznych warunków brzegowych z przesunięciem polega na tym, że sąsiednia komórka będzie leżała nie dokładnie wprost na przeciwległej granicy, a z pewnym przesunięciem (Svyetlichnyy 2008, 2012). Przesunięcie to może być przeprowadzone na jednej lub na kilku granicach w kierunku pionowym, poziomym, wzdłużnym lub poprzecznym, lub z prawie dowolną kombinacją takich kierunków. Fakt ten jest mało znany, a więc w praktyce przesunięcie takie prawie nie jest stosowane. Nadal mamy zatem do czynienia z warunkami periodycznymi i nazywamy je periodycznymi z przesunięciem, o ile przesunięcie to jest niezmiennie na całej granicy, jak to pokazano na rysunku 3.6. Strzałki pokazują miejsca powstania urojonych reprezentacji tego samego ziarna, gdy przekracza ono granicę przestrzeni.



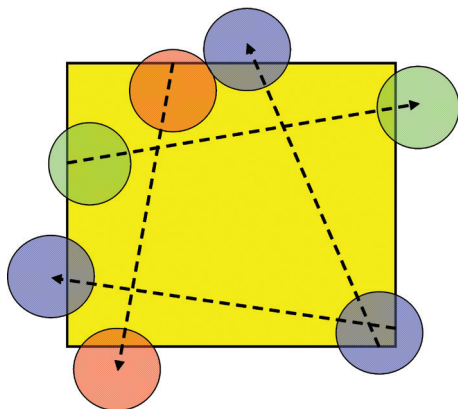
Rys. 3.6. Periodyczne warunki brzegowe z przesunięciem

Na rysunku 3.6 komórka o współrzędnych $\{i, n_y\}$ będzie miała sąsiada nie o współrzędnych $\{i, 1\}$, jak w przypadku zadania zwykłych periodycznych warunków brzegowych, ale o współrzędnych $\{i + di, 1\}$, gdzie di oznacza przesunięcie w kierunku

poziomym. Liczba wszystkich możliwych kierunków przesunięć, które można zadać w modelu w przestrzeni trójwymiarowej, wynosi sześć – po dwa kierunki przesunięcia na trzech płaszczyznach połączenia (jednocześnie można zadać maksymalnie trzy z sześciu). W wyniku zastosowania takiej przestrzeni również powstają ziarna, które leżą na przeciwległych jej granicach.

Jest to stała właściwość przestrzeni, niezależna od zachodzących w niej zmian. Jednak ten sam efekt uzyskamy, jeżeli przypiszemy równe przesunięcia dla każdego ziarna, natomiast przestrzeń pozostawimy bez przesunięcia (Svyetlichnyy 2009). Ten przypadek również ilustruje rysunek 3.6. Jednak różnica polega na tym, że przesunięcie nie jest już właściwością przestrzeni, lecz ziarna i odpowiednio może ono być różne dla różnych ziaren, zadane indywidualnie dla każdego z nich. Przestrzeń nadal pozostaje zamknięta, ale czy takie warunki brzegowe nadal możemy nazywać periodycznymi?

Inny obraz uzyskamy, zadając różne przesunięcie dla różnych ziaren. Sytuację tę zilustrowano na rysunku 3.7. Przedstawione na tym rysunku zarodkowanie ziaren odbywało się w tych samych miejscach co na rysunkach 3.1 i 3.2, jednak po przejściu przez granice przestrzeni ziarna pojawiają się już w innych miejscach.



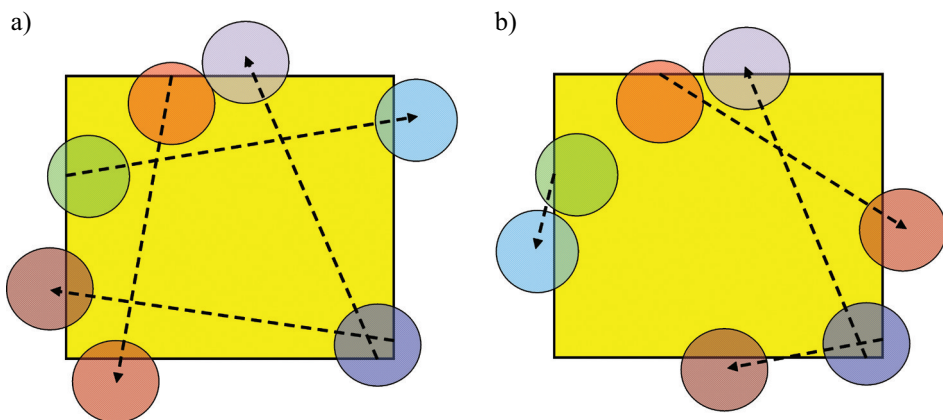
Rys. 3.7. Kombinowane warunki brzegowe

Zdefiniujmy więc periodyczne warunki brzegowe (niestety, w literaturze przedmiotu takiej definicji nie ma) jako takie, przy których można wielokrotnie złożyć przestrzeń w spójny obraz bez odbicia (rys. 3.4a). Wówczas przestrzeń zilustrowana na rysunku 3.6 (pod warunkiem przesunięcia tylko na jednej granicy) nadaje się na takie złożenie, natomiast ta na rysunku 3.7 już nie. Nie są to więc już warunki periodyczne, a warunki określane jako „kombinowane”. Czy wyniki modelowania kinetyki rozwoju mikrostruktury i jej końcowy wygląd w warunkach kombinowanych i periodycznych będą się od siebie różnić? Zasadniczo – nie, będą jedynie występować różnice mało istotne.

Możemy wykonać jeszcze jeden krok, wychodząc od periodycznych warunków brzegowych. Zmienimy kolor ziarna po przejściu przez granicę przestrzeni, jak na rysunku 3.8a. Nie będą to już te same ziarna, ale inne. Wciąż możemy rozpatrywać te warunki jako zamknięte, chociaż niewiele już nas dzieli od warunków otwartych. W szcze-

gólności jeżeli dodamy jeszcze jeden element – nie tylko zadamy przesunięcie wzdłuż granicy, ale również przybliżymy lub oddalimy ziarno od tej granicy. Wymaga to nieco bardziej skomplikowanego algorytmu, ale jest możliwe. Jeżeli jest to już inne ziarno, to dlaczego powinno się ono pojawić na przeciwległej granicy, a nie na dowolnej, na przykład na tej samej? Tak więc będzie to już przestrzeń z otwartymi warunkami brzegowymi, w pełni otwartymi, a nie półotwartymi, jak na rysunku 3.1. Realizacja tych warunków wymaga jednak bardziej skomplikowanego algorytmu.

Przy otwartych warunkach brzegowych wejście dowolnego ziarna na granicę przestrzeni spowoduje pojawienie się dodatkowego ziarna poza granicą przestrzeni w dowolnym miejscu, mniej więcej w tej samej odległości od granicy co rozpatrywane ziarno. Miejsce wrastania nowego ziarna jest więc wybierane losowo (na dowolnej ścianie sześciangu i z ograniczonym losowym przesunięciem w stosunku do powierzchni przestrzeni). Nowe ziarno będzie wrastać w przestrzeń z niewielkim wyprzedzeniem lub opóźnieniem dotarcia ziarna do granicy. Wówczas ani wynikowa struktura, ani kinetyka rozwoju nie będą się różniły od tych uzyskanych z zastosowaniem warunków periodycznych lub kombinowanych. Jeden z algorytmów takich otwartych warunków brzegowych rozpatrzono w instrukcjach do ćwiczeń w części trzeciej niniejszej książki.



Rys. 3.8. Kombinowane (a) i otwarte (b) warunki brzegowe z wprowadzeniem do przestrzeni nowych ziaren

Uzmysłowienie sobie wielkości nakładów na stworzenie automatów komórkowych z odpowiednimi warunkami i algorytmami daje odpowiedź na pytanie, dlaczego w większości przypadków stosowane są warunki periodyczne.

Te same wyniki można uzyskać, stosując półotwarte warunki brzegowe, bez skomplikowanych algorytmów. Jedynie w końcowej mikrostrukturze powinny być odcięte obszary w pobliżu granic i nie powinno się uwzględniać tych obszarów w obliczeniach kinetyki. W tym właśnie tkwi odpowiedź na drugie pytanie: o ile przestrzeń będzie mniejsza. Nie ma możliwości dokładnego określenia, jakie obszary należy obciąć, ponieważ zależy to od szczególnego przypadku symulacji, jednak na ogół można założyć, że odległość miejsca cięcia od granic będzie się równała połowie średnicy ziarna.

Ostatnie stwierdzenia powinny przysporzyć zwolenników periodycznym warunkom brzegowym, jednak w przypadku przedstawionym na rysunku 3.5 oraz w odniesieniu do zagadnień omówionych w podrozdziale 3.3 nie można zastosować prostych periodycznych warunków brzegowych.

Podsumowując warunki brzegowe przy nieziennej topologii i wracając do przedstawionych w rozdziale 1 głównych zasad i wymagań stawianych warunkom brzegowym, należy stwierdzić, że wyeliminowanie efektu brzegu lub wpływu stref zewnętrznych jest osiągane przez zamknięcie przestrzeni. Można to uzyskać dzięki zastosowaniu periodycznych warunków brzegowych bez przesunięcia i z nim lub zadając przejściowe warunki kombinowane, kiedy przestrzeń pozostaje zamknięta, jednak zamknięcie to nie jest regularne, jest lokalne i zależy od topologii mikrostruktury (ziaren), a nie stanowi stałej własności przestrzeni. Brak efektu brzegu w zamkniętej przestrzeni jest wynikiem braku granicy przestrzeni. W przypadku zastosowania otwartych warunków brzegowych wpływ stref zewnętrznych jest zrównoważony wpływem na nie automatów komórkowych poprzez wprowadzenie symetrii w oddziaływaniu. Wpływ automatów na strefy zewnętrzne jest znany z modelowania, więc pozostaje wprowadzić równoważną odpowiedź stref zewnętrznych, co umożliwia zadanie otwartych warunków brzegowych. Otwarte warunki brzegowe, w odróżnieniu od periodycznych, pozwalają rozwiązać problem pokazany na rysunku 3.5.

W tym rozdziale rozpatrzono cztery główne rodzaje warunków brzegowych:

- 1) Półotwarte – najprostsze, nieuwzględniające ograniczoności przestrzeni i wpływu nietożsamości otoczenia komórek, a więc wprowadzające do modelu wpływ brzegów. Poza kilkoma wyjątkami nie zaleca się ich stosowania.
- 2) Periodyczne (zamknięte) – pozwalające dzięki połączeniu przeciwległych granic utworzyć zamkniętą przestrzeń bez brzegów i wyeliminować w ten sposób wpływ tych brzegów. Możliwe jest połączenie bezpośrednie, bez przesunięć albo z przesunięciem w jednym lub kilku kierunkach.
- 3) Kombinowane – posiadające cechy zarówno otwartych (półotwartych), jak i zamkniętych warunków brzegowych. Pozwalają tworzyć struktury identyczne ze strukturami uzyskanymi przy zastosowaniu periodycznych warunków brzegowych (zarówno z przesunięciem, jak i bez) z tą tylko różnicą, że przestrzeń jest otwarta, a pojawiające się po przeciwległej stronie ziarna nie są kolejną reprezentacją tego samego ziarna, a zupełnie innymi, zewnętrznymi ziarnami. Ponadto przesunięcie przestaje być właściwością przestrzeni, a staje się właściwością każdego ziarna, więc niekoniecznie musi być w przypadku wszystkich ziaren jednakowe, ale można je zadawać indywidualnie dla każdego z nich.
- 4) Otwarte – najbardziej skomplikowane warunki, w których zanika wszelka zależność pomiędzy przeciwległymi granicami, ale również wyeliminowany jest wpływ brzegu dzięki fenomenologicznemu uwzględnieniu warunków na brzegu przestrzeni. Osiągnięcie przez ziarno brzegu przestrzeni powoduje powstanie w podobnej odległości od brzegu innego ziarna, w dowolnym miejscu, na zewnątrz.

Oczywiście można tworzyć przestrzenie o różnej kombinacji warunków brzegowych.

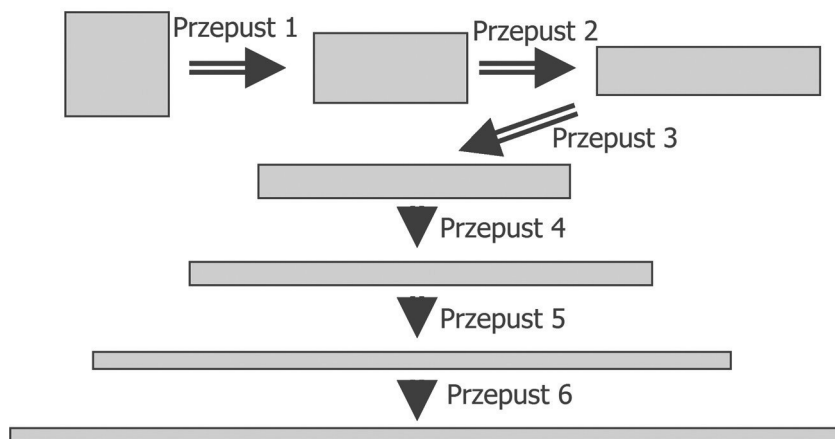
3.3. Warunki brzegowe przy zmiennej topologii automatów komórkowych

Z lektury poprzedniego rozdziału można wysnuć wniosek, że przy wyborze warunku brzegowego jako pierwsze należałoby rozpatrywać warunki periodyczne. W niezmienniej przestrzeni automatów komórkowych są one faktycznie jedynymi, które stosowano w praktyce do modelowania. Przykładem symulacji z niezmienną przestrzenią jest modelowanie krzepnięcia i krystalizacji, często też rozwój pęknięć, mikropasm i pasm ścinania, rekrytalizacji statycznej i dynamicznej (przy niedużych odkształceniach), czyli takie procesy, w których nie zachodzą zmiany wymiarów przestrzeni modelowej automatów komórkowych lub zmiany te nie są duże. Natomiast odkształcenie powoduje zmianę geometrii przestrzeni, która to zmiana może być uwzględniona lub pominięta. W większości omówionych w literaturze przedmiotu przypadków, kiedy rozpatrywano rekrytalizację dynamiczną, była ona pominięta lub uwzględniana przez modelowanie odkształconej struktury stworzonej na nieodkształconych komórkach, czyli praktycznie nie rozpatrywano odkształcenia komórki. Niektóre zasady i algorytmy uwzględnienia odkształcenia komórki i zmniejszenia anizotropii przestrzeni zostały przedstawione w rozdziale 2 niniejszej książki, przy okazji omawiania modelowania rozrostu ziarna. Natomiast w tym rozdziale rozpatrzono skutki odkształcenia i wybór warunków brzegowych.

Modelowanie jednoetapowego odkształcenia na ogół nie pociąga za sobą konieczności przebudowy przestrzeni. Spłaszczenie w jednym kierunku i wydłużenie w innym nie doprowadzi do istotnego zniekształcenia ani przestrzeni, ani komórki. Poza tym zawsze można dobrać wymiary przestrzeni i komórki w taki sposób, żeby podczas modelowania odkształcenia pozostawały one odpowiednie. Problem powstaje wówczas, kiedy zachodzą bardzo duże odkształcenia w procesie z pojedynczym odkształceniem lub w procesach z odkształceniem wielokrotnym, takich jak na przykład walcowanie w kilku przepustach (rys. 3.9). Zwiększenie odkształcenia od przepustu do przepustu doprowadzi do zmiany zarówno całej przestrzeni automatów komórkowych, jak i poszczególnych komórek. Wydłużenie, spłaszczenie przestrzeni może spowodować, że w końcu przestanie się ona nadawać do modelowania ze względu na swoją niewystarczającą grubość, a wielkość ziarna może być nawet większa od grubości przestrzeni. Wydłużona płaska komórka jest mniej przydatna do modelowania, ponieważ w wypadku tego typu komórek trudniej zachować izotropię przestrzeni.

Punktem wyjścia w obliczeniach za pomocą automatów komórkowych mogą być komórki w kształcie sześciangu (ale niekoniecznie). Podczas symulacji z odkształceniem komórki nie pozostają niezmiennie. Odkształcenie działa nie tylko na rozmiary i kształt komórek, ale również na stopień anizotropii modelowej przestrzeni i reprezentatywnej objętości. Dysproporcje między liniowymi wymiarami komórek mogą stać się zbyt duże (rys. 3.9), obliczenia będą wolniejsze i jakość uzyskanej struktury niezadowalająca.

Podczas modelowania z zastosowaniem modeli opartych na innych metodach, na przykład na metodzie elementów skończonych (MES), często stosowano procedurę nazywaną „remeshing”. Zbliżoną procedurę należałoby zastosować i w przypadku modelowania automatami komórkowymi. Tak więc duże odkształcenie lub zniekształcenie przestrzeni i komórek wymaga odpowiedniej przebudowy przestrzeni.



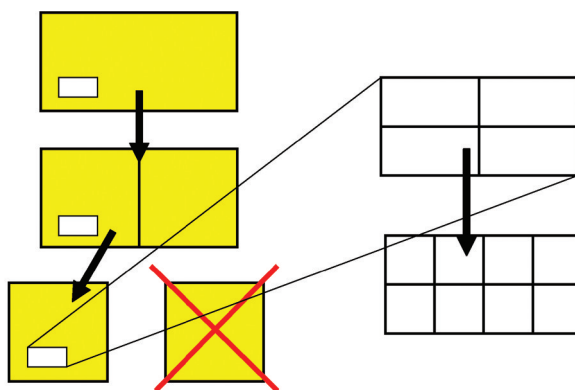
Rys. 3.9. Zmiana kształtu przestrzeni podczas modelowania walcowania

Przebudowę przestrzeni z zachowaniem niezmiennej liczby komórek można wykonać na trzy różne sposoby, w zależności od wymagań i założeń co do wymiarów przestrzeni i reprezentatywnej objętości modelowej. Wybór jednego z owych trzech sposobów zależy od tego, czy przestrzeń może być dwukrotnie zmniejszona (podrozdz. 3.3.1), czy powinna pozostać niezmienną (podrozdz. 3.3.2), czy też objętość należy dwukrotnie zwiększyć (podrozdz. 3.3.3). Istnieją również warianty odkształcenia, w przypadku których żaden z tych sposobów nie jest przydatny. Wówczas można zastosować mniej dokładny, ale bardziej uniwersalny sposób opisany w podrozdziale 3.3.4.

3.3.1. Odrzucenie połowy modelowej przestrzeni (*halving*)

Pierwsza metoda polega na przecięciu modelowej przestrzeni na pół, pozostawieniu jednej połowy i odrzuceniu drugiej, co pokazano na rysunku 3.10. Kolorami zaznaczono na nim całą przestrzeń modelową, natomiast białe prostokąty i kwadraty reprezentują komórki. Po odjęciu jednej połowy w drugiej pozostaje dwukrotnie mniej komórek, których kształt nie ulega zmianie, więc należy podzielić każdą z nich na dwie mniejsze, kwadratowe. W ten sposób w przestrzeni pozostaje ta sama liczba komórek.

Podczas odkształcenia cała przestrzeń i pojedyncze komórki zmieniają swój kształt w ten sam sposób. W przypadku z równą liczbą komórek wzdłuż każdej osi przestrzeni komórka jest zmniejszoną kopią przestrzeni. Tak więc zastosowanie pierwszej z omówionych metod, czyli odrzucenie jednej połowy modelowej przestrzeni, powoduje skrócenie jednego wymiaru przestrzeni i zmniejszenie jej objętości. Stosować tę metodę można na przykład przy modelowaniu procesu walcowania, kiedy następuje spadek temperatury i po rekrytalizacji wielkość ziaren maleje w porównaniu z ich rozmiarami przed odkształceniem, natomiast liczba ziaren wzrasta. Wówczas po odkształceniu (rekrytalizacji) ziaren następuje dzielenie przestrzeni i liczba ziaren od podziału do podziału nie maleje w istotnym stopniu. Celowe jest wówczas skrócenie najdłuższego wymiaru komórki i przestrzeni, co spowoduje zbliżenie się kształtu komórki do początkowego, czyli do sześcianu (kwadratu). W celu zachowania po odrzuceniu jednej połowy przestrzeni tej samej liczby komórek powinna być ona podwojona przez podzielenie każdej komórki na dwie. Wówczas kształt komórki będzie podobny do kształtu przestrzeni, czyli trójkąt zbliży się do sześcianu (kwadratu).



Rys. 3.10. Schemat odrzucenia połowy przestrzeni modelowej
(Svyetlichnyy 2012)

W tym algorytmie przebudowy (podobnie jak w pozostałych) należy zwrócić szczególną uwagę na warunki brzegowe. Jest to dość prosty algorytm i na ogół na początku modelowania można założyć dowolne warunki brzegowe. Jeżeli zastosować półotwarte, kombinowane lub otwarte warunki brzegowe, to – ponieważ wszystkie one należą w zasadzie do warunków otwartych, niezamkniętych – odcinanie i odrzucanie połowy przestrzeni nie wymaga dodatkowych działań.

Jednak ze względów opisanych w poprzednim rozdziale często są zadawane warunki okresowe. Wówczas odcięcie i odrzucenie jednej połowy modelowej przestrzeni prowadzi do naruszenia okresowych warunków brzegowych. Przestrzeń można przedstawić w formie torusa, więc jej dzielenie odpowiada dwóm cięciom na torusie, ale nie da się połączyć przeciwległych granic, ponieważ mikrostruktura na

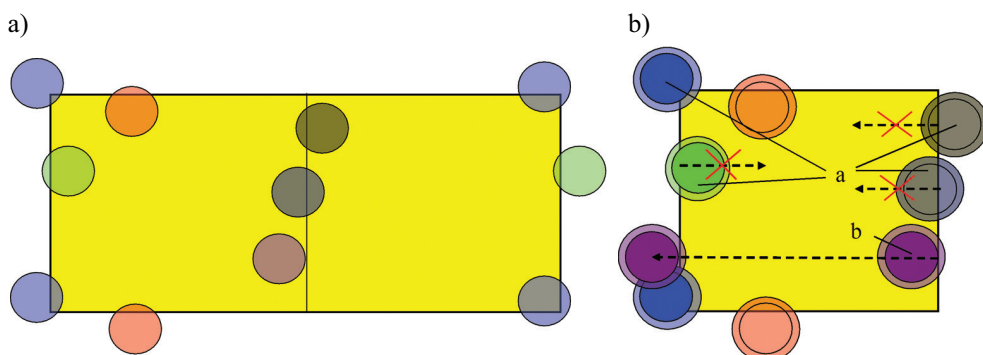
tych dwóch przekrojach będzie różna. Po przeprowadzeniu podziału warunki brzegowe powinny być zatem zmienione z periodycznych na kombinowane (lub półotwarte). Półotwarte warunki brzegowe nie są zalecane, a otwarte granice wymagają bardziej skomplikowanych algorytmów. Stosowanie skomplikowanych algorytmów nie ma sensu, ponieważ w takim przypadku można by je było zastosować od samego początku i cięcie nic by nie zmieniło. Pozostaje więc tylko jedno rozwiązanie: zastosowanie kombinowanych warunków brzegowych. Spośród kilku rodzajów warunków kombinowanych najciekawszy i najbardziej użyteczny może być ten, który różni się od periodycznych tylko tym, że warunki brzegowe związane są w nim z ziarnem, a nie z przestrzenią. Wówczas można powiedzieć, że stan otwartości granic nie będzie permanentny, lecz czasowy.

Różnica między właściwościami warunków brzegowych otwartych (w tym kombinowanych) a zamkniętych ujawnia się dopiero wtedy, gdy struktura zmienia się na skutek ruchu granic ziaren. Jeżeli mamy stabilną, niezmienną strukturę, nie ma sensu brać pod uwagę jakichkolwiek warunków brzegowych. Innymi słowy, warunki brzegowe zależą również od ziaren, o czym wspomniano już w poprzednim rozdziale. Wówczas przeniesienie pojęcia warunków brzegowych z kategorii bezwzględnych własności przestrzeni do kategorii własności przestrzeni uzależnionych od ziarna pozwala rozwiązać problem otwartości granic po zmniejszeniu przestrzeni. Oznacza to, że każde ziarno wyznacza, czy warunki brzegowe dla niego będą otwarte, czy zamknięte.

Rozpatrzmy ziarna znajdujące się w pobliżu miejsca przecięcia przestrzeni (rys. 3.11). Ziarnom, które zostały przecięte (litera „a” na rysunku 3.11b), niezależnie od tego, czy rosły, czy nie, przypisujemy półotwarte warunki brzegowe. Faktyczne, nadal rosną one na zewnątrz pozostawionej części przestrzeni, ale rosną dalej również i w tej części, która została odrzucona i dlatego nie pojawiają się z przeciwległej strony części pozostawionej. Natomiast te ziarna (litera „b”), które dorosły do granicy, po obcięciu połowy przestrzeni pojawiają się z przeciwnej strony. Można ten układ rozpatrywać jak dwa torusy – jeden, większy, w przypadku ziaren przeciętych, oraz mniejszy w przypadku ziaren nieprzeciętych. Ponieważ odległość ziaren od granic w większym torusie jest duża, przecięte ziarna nie mają szansy dorosnąć do drugiego brzegu mniejszego torusa, więc można w odniesieniu do nich rozpatrywać warunki brzegowe jako półotwarte. Natomiast rosnące ziarna, które osiągają brzegi po przecięciu, nie muszą pokonywać odległości dzielącej je od granicy w większym torusie, ponieważ poruszają się po mniejszym torusie i od razu pojawiają się na przeciwległym brzegu pozostawionej części przestrzeni. Wobec tego ziarna, które zostały przecięte, zyskują półotwarte warunki brzegowe, zaś w wypadku pozostałych ziaren warunki brzegowe pozostają zamknięte (periodyczne). Granica powstała w miejscu przecięcia od początku jest półotwarta, dalej ewoluje ona w częściowo półotwartą, częściowo periodyczną i w czasie symulacji może stać się w pełni periodyczna. Mówiąc precyzyjnie, są to kombinowane warunki brzegowe spełniające rolę warunków periodycznych, ponieważ przypisane są do ziaren, a nie do przestrzeni.

Analiza pierwszej metody przebudowy przestrzeni wskazuje, że jeżeli w przypadku wszystkich ziaren zostałyby zachowane zamknięte warunki brzegowe, to, po pierwsze, obydwa brzegi stałyby się granicami ziaren, ponieważ po drugiej stronie na pewno

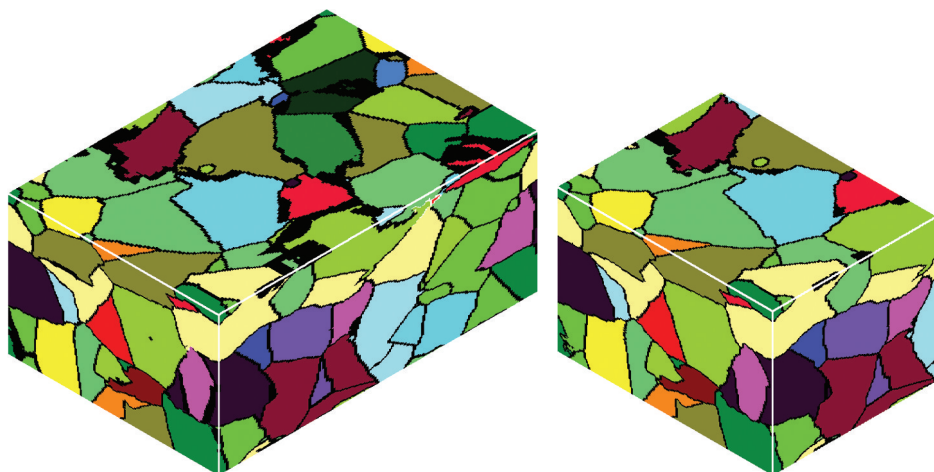
będzie inne ziarno, a po drugie, jeżeli rosnące, przecięte ziarna nadal rosłyby, pojawiając się z innej strony, wyglądałyby one z tej strony, w zależności od algorytmu rozrostu, jak tarcza lub pierścień.



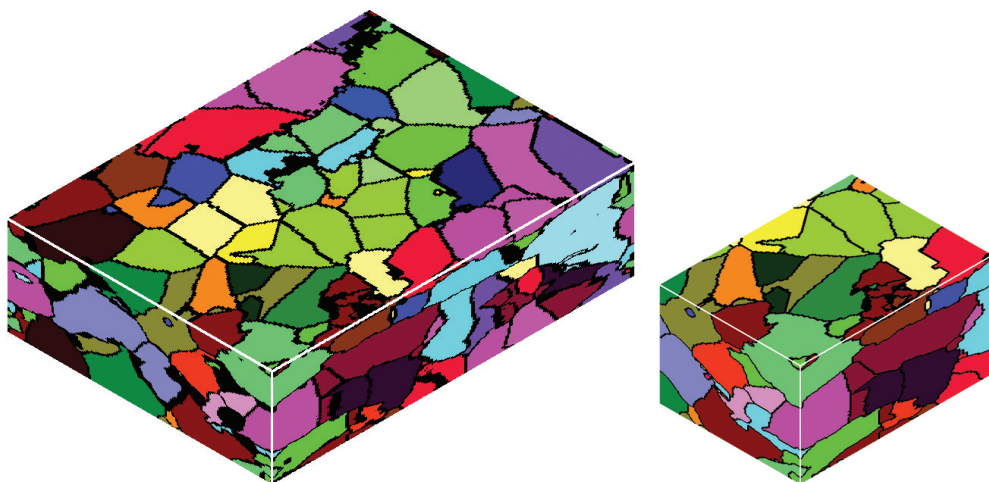
Rys. 3.11. Warunki brzegowe przed podziałem (a) i po podziału przestrzeni (b)

Jako wariant na początku można zastosować warunki periodyczne z przesunięciem, wówczas w ten sam sposób należy zamiast jednej granicy rozpatrywać dwie granice. Ten wariant jest stosowany przy automatycznym wyborze rodzaju przebudowy przestrzeni automatów komórkowych.

Dwa przykłady zastosowania tej metody przedstawiono na rysunkach 3.12 i 3.13. Pierwszy rysunek ilustruje podział przestrzeni w poprzek długości, drugi w poprzek długości i szerokości. Na rysunkach tych można zauważyć wyostrenie granic ziaren, związane z tym, że kształt komórki jest podobny do kształtu przestrzeni, więc szerokość granicy wzdłuż jednej lub dwóch osi przestrzeni się zmniejsza.



Rys. 3.12 Przebudowa przestrzeni przez jej obcięcie w poprzek długości (Svyetlichnyy 2009)



Rys. 3.13 Przebudowa przestrzeni przez jej obcięcie w poprzek długości i szerokości (Svyetlichnyy 2009)

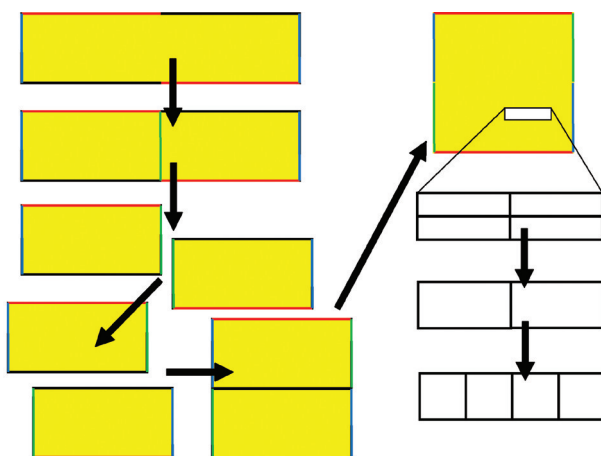
3.3.2. Przycinanie i składanie przestrzeni (*cutting and bonding*)

W drugiej ze wspomnianych metod przestrzeń również jest dzielona na pół, ale potem obydwie jej połowy są znów ze sobą składane i sklejane. Tak więc w metodzie tej następuje skrócenie o połowę jednego wymiaru przestrzeni i podwojenie innego, bez zmiany objętości. Stosować tę metodę można wówczas, kiedy w cyklach symulacji (odkształcenie – rozwój mikrostruktury) liczba ziaren się nie zwiększa w istotnym stopniu.

Celowe jest wówczas pomniejszenie największego wymiaru i zwiększenie najmniejszego, co poprawi kształt przestrzeni (rys. 3.14). Podobnie należy postąpić z komórkami, tylko kolejność powinna być inna: najpierw sklekanie sąsiednich komórek, a potem ich dzielenie. Związane jest to z faktem, że możliwość podwojenia liczby komórek może być ograniczona pamięcią. Aby nie wprowadzić dużego błędu, podczas łączenia komórek należących do różnych ziaren należy uwzględnić ich otoczenie. Na przykład komórkę można dołączyć do tego ziarna, do którego należy większość komórek w sąsiedztwie komórek łączonych. Jeżeli liczba komórek w sąsiedztwie jest równa, można je dołączyć do większego ziarna lub wybrać ziarno losowo. Po sklejeniu komórek dzielone są one na dwie i wówczas ich kształt będzie podobny do kształtu przestrzeni, czyli również zbliżony do kwadratu (sześciangu).

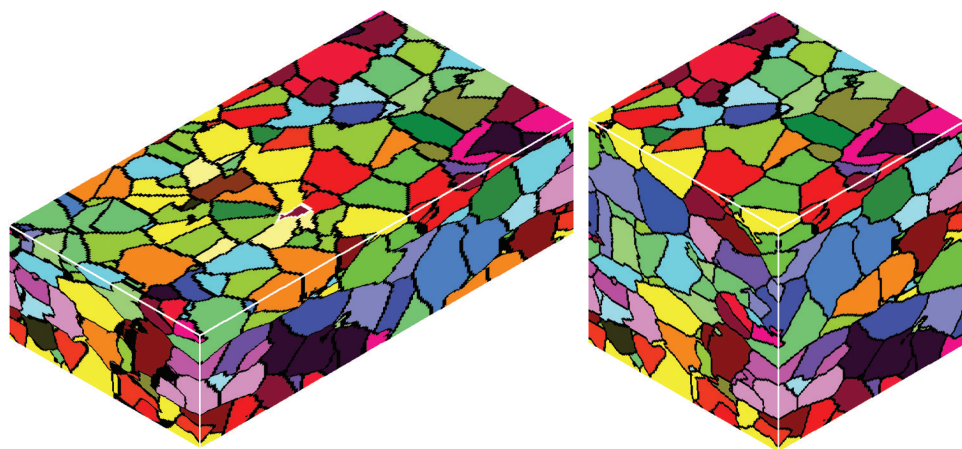
W tej metodzie przebudowy przestrzeni warunki początkowe nie mogą być dowolne. Ponieważ po przecięciu składamy ze sobą obydwie części, w miejscu sklekania mikrostruktura powinna być taka sama, ponadto przy sklepaniu powinny łączyć się części tych samych ziaren. Takie wymagania jednoznacznie narzucają zastosowanie warunków brzegowych przedstawionych na rysunku 3.14, na którym kolorem zaznaczono te

same brzegi przestrzeni przed przebudową i po przebudowie. Nie są to różne brzegi, są one przedłużeniem przestrzeni w tym czy innym kierunku. Wówczas do przebudowy w dwóch kierunkach stosujemy proste okresowe warunki brzegowe, natomiast w najkrótszym wymiarze przestrzeni w trzecim kierunku należy zastosować okresowe warunki z przesunięciem. Przesunięcie następuje w kierunku najdłuższego wymiaru przestrzeni i jest równe połowie jej długości (przesunięcie liczone w komórkach jest stałe). Po przebudowie warunki w jednym kierunku pozostają bez zmian (jest to kierunek prostopadły do rysunku 3.14). Warunki zadane w kierunku pierwotnie najdłuższego wymiaru zmieniają się z okresowych na okresowe z przesunięciem, natomiast w kierunku pierwotnie najkrótszego wymiaru – z okresowych z przesunięciem na proste okresowe. Natomiast takie warunki niekoniecznie powinny pozostać do końca symulacji i w zależności od potrzeb dalszego modelowania mogą ewoluować w warunki przed przebudową lub w warunki okresowe we wszystkich kierunkach. Wówczas, jak w metodzie pierwszej, warunki brzegowe przypisywane są do ziaren. Te ziarna, które pojawiły się na brzegach przed przebudową, po przebudowie zachowują odpowiednie warunki, które z tej przebudowy wynikają. Natomiast ziarnom, które osiągną brzegi przestrzeni po jej przebudowie, będą przypisane nowe warunki, które uwzględniają potrzeby wynikające z dalszego modelowania.



Rys. 3.14. Przecinanie i składanie przestrzeni modelowej oraz warunki brzegowe przed przekształceniem i po przekształceniu przestrzeni (Svyetlichnyy 2012)

Przykłady zastosowania tej metody przedstawiono na rysunku 3.15. Można na nim zobaczyć, że obie części idealnie do siebie pasują i w nowej, przebudowanej przestrzeni trudno znaleźć linię, wzdłuż której zostały połączone dwie części. Na rysunku tym można również zauważyć wyostrenie granic ziaren. Dodatkowo, algorytm wyboru ziarna w przypadku łączenia komórek należących do różnych ziaren powoduje wyeliminowanie ostrych kątów ziaren, czyli ich zaokrąglenie.



Rys. 3.15. Przebudowa przestrzeni w wyniku obcinania i składania
(Svyetlichnyy 2009)

Druga metoda przebudowy przestrzeni już na samym początku określa, w jaki sposób przestrzeń zostanie przecięta i jak będą składane jej części. Jednak czasem nie jest to wiadome na początku. Wówczas należy przeprowadzić modelowanie z zastosowaniem okresowych warunków brzegowych, a po wyznaczeniu sposobu przecięcia i składania powtórzyć obliczenia z warunkami okresowymi z przesunięciem.

Metody przebudowy różnią się od siebie warunkami brzegowymi stosowanymi od początku modelowania, więc metoda taka powinna być wybrana przed rozpoczęciem modelowania. Jednak nie zawsze jest to możliwe. Wiąże się to z faktem, że istnieje optymalna dla wybranych wymiarów przestrzeni liczba ziaren. Można przyjąć, że maksymalna liczba ziaren w przestrzeni trójwymiarowej jest równa liczbie komórek wzdłuż jednego wymiaru przestrzeni, a liczba minimalna waha się w przedziale 25–40 ziaren. Znaczne zwiększenie lub zmniejszenie liczby ziaren obniża skuteczność modelowania – wówczas jego wyniki wpływają na wybór metody przebudowy przestrzeni. Rozdrobnienie mikrostruktury prowadzi do znaczącego zwiększenia liczby ziaren, zmniejszenia ich objętości i zwiększenia niekorzystnego wpływu dyskretyzacji przestrzeni, więc w tym przypadku pożądany jest wybór pierwszej metody. Natomiast wybór tej metody przy zachowaniu średniego rozmiaru ziarna lub nieznacznym jego zmniejszeniu prowadzi do zmniejszenia liczby ziaren poniżej racjonalnej wartości, a nawet do sytuacji, kiedy ziarno wydłuży się tak bardzo, że osiągnie przeciwny brzeg przestrzeni.

Jednym z rozwiązań tego problemu jest powtórzenie obliczeń ze zmienionymi warunkami brzegowymi i zadaną odpowiednią kolejnością przebudowy przestrzeni. Należy również pamiętać, że pierwsza metoda zmienia stosunek wymiarów dwukrotnie, a druga – czterokrotnie i tam, gdzie w drugiej metodzie wystarczy jedna przebudowa, przy zastosowaniu metody pierwszej potrzebne są dwie przebudowy. Teoretycznie możliwa jest sytuacja, w której w jednej chwili w przestrzeni będą obecne warunki brzegowe otwarte oraz okresowe z przesunięciem i bez przesunięcia. Przykłady modelowa-

nia z zastosowaniem omówionych wyżej warunków brzegowych i metod przebudowy przestrzeni zostały opisane w kolejnych rozdziałach niniejszego opracowania.

Jednak bardziej racjonalnym rozwiązaniem wyżej opisanego problemu jest automatyczny wybór wariantu przebudowy. W podrozdziale 3.3.1 jako warunki brzegowe przed podzieleniem przestrzeni rozpatrywano głównie warunki periodyczne bez przesunięcia, ale można zastosować prawie każde inne. Gdy przebudowa przestrzeni polega na przecinaniu i sklejanu, warunki brzegowe przed przecinaniem powinny być periodyczne z przesunięciem i tylko takie. Jeżeli na początku założyć warunki periodyczne z przesunięciem, to wybór wariantu przebudowy można przeprowadzić automatycznie ze względu na wymiary przestrzeni i liczbę ziaren, czyli w zależności od bieżących warunków symulacji. Na przykład według następującej kolejności:

$$\begin{aligned}
 (n_g > n_{g \max}) \wedge (a_x = \max(a_x, a_y, a_z)) &\Rightarrow HX \\
 (n_g > n_{g \max}) \wedge (a_y = \max(a_x, a_y, a_z)) &\Rightarrow HY \\
 (n_g > n_{g \max}) \wedge (a_z = \max(a_x, a_y, a_z)) &\Rightarrow HZ \\
 (a_x > 3a_z) &\Rightarrow BXZ \\
 (a_x > 2a_z) \wedge (n_g > 2n_{g \min}) &\Rightarrow HX \\
 (a_y > 2a_z) \wedge (n_g > 2n_{g \min}) &\Rightarrow HY
 \end{aligned} \tag{3.1}$$

gdzie:

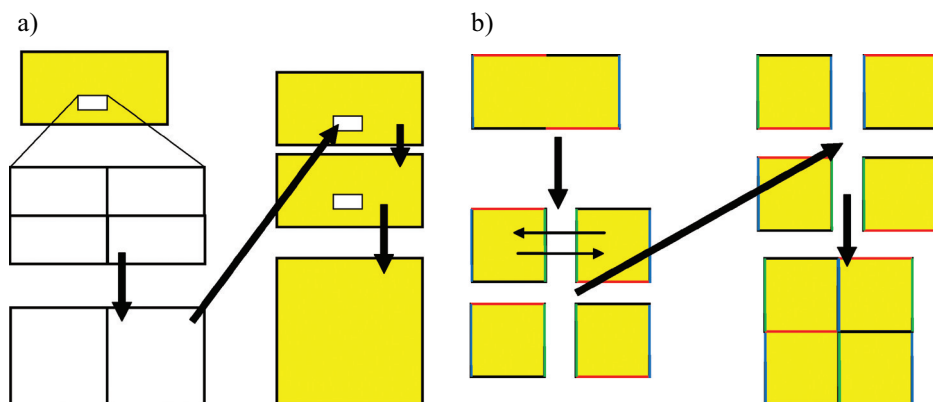
- $a_x, a_y, i a_z$ – wymiary modelowej przestrzeni,
- n_g – liczba ziaren,
- $n_{g \min}$ i $n_{g \max}$ – minimalna i maksymalna liczba ziaren (na przykład $n_{g \min} = 500$, $n_{g \max} = 4000$),
- BXZ – przecinanie w poprzek długości ($a_x/2$) i sklejanie wzdłuż grubości ($2a_z$),
- HX, HY i HZ – dzielenie w poprzek długości ($a_x/2$), szerokości ($a_y/2$) i grubości ($a_z/2$).

3.3.3. Podwojenie przestrzeni

W trzeciej metodzie przestrzeń jest podwajana. Ta metoda nie jest stosowana w modelach opisanych w niniejszej monografii, jednak w dalszych partiach tekstu zostaną podane pewne wskazówki dotyczące takiej transformacji przestrzeni. Aby można było połączyć dwie części przestrzeni, jej brzegi powinny się charakteryzować taką samą strukturą. Ta zasada była stosowana w drugiej metodzie i wymagała przyjęcia periodycznych warunków brzegowych z przesunięciem równym połowie długości przestrzeni.

Jeżeli należy podwoić przestrzeń, najłatwiej jest zastosować warunki periodyczne bez przesunięcia w taki sposób, jak pokazano na rysunku 3.4a. Działanie to można powtórzyć kilkakrotnie.

Po zwiększeniu przestrzeni liczba komórek powinna pozostać bez zmian. Najpierw więc należy połączyć komórki w kierunku najkrótszego wymiaru (rys. 3.16a), zmniejszając w ten sposób dwukrotnie ich liczbę i wypełniając nimi pierwszą część nowej, zwiększonej przestrzeni, a potem skopiować je do części drugiej.



Rys. 3.16. Podwojenie przestrzeni o periodycznych warunkach brzegowych bez przesunięcia (a) i z przesunięciem (b)

Dopasowując podwojenie przestrzeni do automatycznego wyboru przebudowy, należy zastosować periodyczne warunki brzegowe z przesunięciem. Podobnie jak poprzednio komórki łączono i wypełniano nimi pierwszą część nowej przestrzeni. Natomiast druga jej część była dzielona na pół, co pokazano na rysunku 3.16b, a lewa i prawa połowa zmieniły się miejscami. W efekcie powstała przestrzeń z periodycznymi warunkami brzegowymi bez przesunięcia (rys. 3.16b).

3.3.4. Wyprostowanie przestrzeni automatów komórkowych

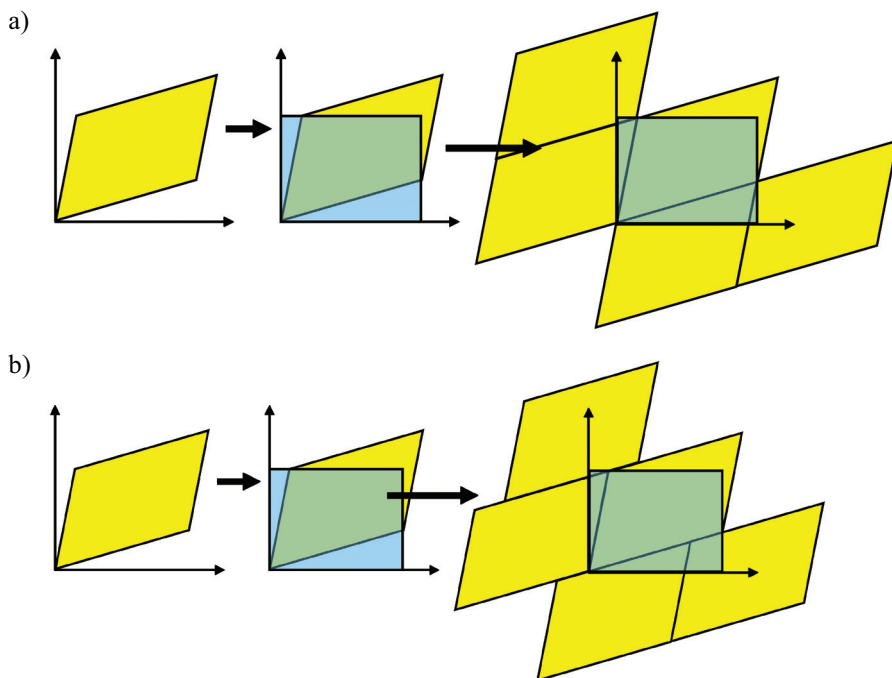
W poprzednich rozdziałach rozpatrywano przebudowę przestrzeni w przypadku, kiedy kierunki głównych odkształceń są zgodne z osiami przestrzeni automatów komórkowych. Jednak przy odkształceniach wieloetapowych nie zawsze jest możliwe zachowanie prostopadłościennego kształtu przestrzeni automatów. Niewielkie odkształcenia ścinające można pominąć – komórki i przestrzeń będą miały wówczas już kształt prostopadłościanu, a ściany i krawędzie będą pochylone. Przy dużych pochyleniach konieczne jest „wyprostowanie” równoległościanu do kształtu prostopadłościanu. Można to uzyskać tylko dzięki zadaniu periodycznych warunków brzegowych bez przesunięcia lub z przesunięciem (rys. 3.17).

Warunki periodyczne, zarówno bez przesunięcia, jak i z przesunięciem w jednym kierunku, pozwalają w prosty sposób replikować przestrzeń automatów komórkowych i two-

rzyć wielokrotnie powtarzający się wzór. W ten sposób można wypełnić przestrzeń bez żadnych ograniczeń. W tak utworzoną przestrzeń można wpisać nową przestrzeń o dowolnym kształcie i rozmiarach, na przykład, jak to pokazano na rysunku 3.17, prostokątną.

Stosując takie przekształcenie, należy pamiętać, iż wynikowa przestrzeń modelowa nie będzie miała okresowych warunków brzegowych. Wszystkim ziarnom na brzegach przestrzeni należy przypisać półotwarte warunki brzegowe, natomiast wobec pozostałych można stosować dowolne warunki brzegowe, według uznania lub w zależności od dalszych warunków symulacji.

W przypadku trójwymiarowych automatów komórkowych opisana powyżej procedura jest zasadniczo taka sama.



Rys. 3.17. Zastąpienie zniekształconej przestrzeni komórkowej z okresowymi warunkami brzegowymi bez przesunięcia (a) i z przesunięciem (b) przez przestrzeń prostokątną

3.4. Wskazówki dotyczące wyboru warunków brzegowych

W tym rozdziale rozpatrzono cztery podstawowe warunki brzegowe: półotwarte, okresowe (zamknięte), kombinowane i otwarte.

Przy nieziennej topologii jedynie zastosowanie półotwartych warunków brzegowych nie zawsze spełnia narzucone przez symulację wymagania (na przykład w stosunku do kinetyki procesów).

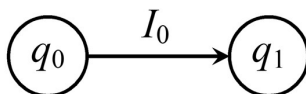
Przy zmiennej topologii, czyli zmianie wymiarów i kształtów automatów komórkowych, najbardziej przydatne, w szczególności do automatyzacji przebudowy (reorganizacji) przestrzeni, są warunki brzegowe periodyczne z przesunięciem w jednym kierunku o połowę długości przestrzeni, a ściśle mówiąc – warunki kombinowane o właściwościach periodycznych z przesunięciem. Takie warunki pozwalają na dowolną przebudowę: dzielenie, podwajanie, przecinanie i sklejanie oraz „prostowanie” przestrzeni. Kombinowane warunki brzegowe są dowolnym połączeniem innych warunków brzegowych. Przy czym mogą one być właściwością ziaren, a nie przestrzeni, kiedy sąsiadujące ziarna mają różne warunki brzegowe. Poza tym podczas symulacji warunki nie muszą pozostawać stałe, natomiast mogą ewoluować.

4. Opracowanie i zastosowanie frontalnych automatów komórkowych

Działanie automatów komórkowych wyznacza się za pomocą reguł przejścia. Reguła przejścia określa stan komórki w następnym kroku obliczeniowym na podstawie stanu komórek istniejącego w kroku bieżącym. Jednak te same reguły mogą być realizowane w różny sposób. Zastąpienie klasycznego, wymagającego dużych nakładów obliczeniowych automatu komórkowego innym, który może omijać duże obszary przestrzeni komórkowej, pozwala wielokrotnie skrócić czas obliczeń. W niniejszym rozdziale został opisany taki właśnie automat komórkowy, nazywany frontálním. Na początku omówiono przejście od automatu klasycznego do frontálního, opisano zasady działania i określono główne cechy automatu frontálního, które pozwoliły osiągnąć wielokrotne przyspieszenie obliczeń. Na końcu rozdziału przedstawiono uniwersalny automat, który został wykorzystany do modelowania zjawisk i procesów przedstawionych w części drugiej tego opracowania.

4.1. Konwencjonalne automaty komórkowe

Reguła przejścia automatu komórkowego jest jednym z głównych jego parametrów i na podstawie stanu komórek w bieżącym kroku obliczeniowym wyznacza stan komórki w kroku następnym. W najprostszym automacie komórka może się znajdować w dwóch podstawowych stanach – nazwijmy je stanem początkowym q_0 i stanem końcowym q_1 . Jako przykład działania automatów rozpatrzmy jeden z najprostszych algorytmów rozrostu. Podczas obliczeń wszystkie komórki, które na początku obliczeń znajdowały się w stanie początkowym q_0 , przechodzą przez zarodkowanie nowych ziaren lub ich rozrost w stan końcowy q_1 . Wówczas automat komórki można schematycznie przedstawić tak, jak to zrobiono na rysunku 4.1 (por. też rys. 2.2)

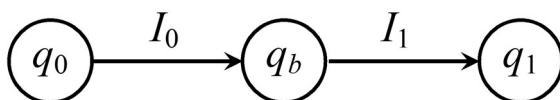


Rys. 4.1. Automat komórki oparty na algorytmie rozrostu ziaren z dwoma podstawowymi stanami (Malinowski *et al.* 2009)

Podstawą przedstawionego automatu w otoczeniu von Neumanna z oznaczeniem komórek, jak na rysunku 2.1 jest następująca reguła przejścia:

$$I_0 = p \vee c_2 \vee c_4 \vee c_5 \vee c_7 \quad (4.1)$$

Pierwszy człon prawej części równania przedstawia zarodkowanie, czyli p jest prawdopodobieństwem powstania zarodka w wybranej komórce. Pozostałe człony: c_2 , c_4 , c_5 i c_7 wyznaczają rozrost, czyli komórka przechodzi ze stanu q_0 w stan q_1 przy obecności w jej otoczeniu przynajmniej jednej komórki należącej do rosnącego ziarna, innymi słowy: znajdującej się w stanie q_1 . Jednak w rzeczywistości przejście z jednego stanu w inny nie następuje natychmiast, ponieważ warunkiem wyznaczenia następnego stanu wszystkich komórek jest posiadanie również informacji o stanie wszystkich komórek w bieżącym kroku obliczeniowym. Dlatego też najpierw następuje badanie stanu wszystkich komórek w celu wyznaczenia spełnienia warunków przejścia, a potem zmiana stanu komórek. Te dwa procesy powinny być rozdzielone w czasie, aby zapobiec możliwości wystąpienia zakłócenia obliczeń w wyniku zmiany stanu komórek w czasie obliczeń w bieżącym kroku. Nieuwzględnienie takiego rozdzielania spowoduje powstanie sytuacji, w której ta sama komórka może występować w różnych stanach względem różnych sąsiadów, a wyniki obliczeń będą zależały od kolejności badania komórek w całej przestrzeni automatów komórkowych. Automaty komórkowe, w których stosowano opisane wyżej rozdzielanie w czasie wyznaczenia nowego stanu i przejścia w nowy stan, nazywane są automatai synchronicznymi, a sam proces rozdzielania – synchronizacją stanu automatów. Schematycznie (rys. 4.2) można to przedstawić, wprowadzając dodatkowy stan buforowy q_b .

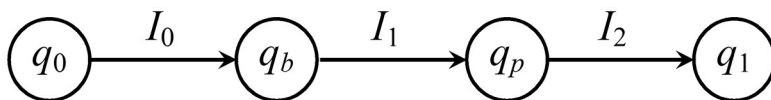


Rys. 4.2. Synchroniczny automat komórki oparty na algorytmie rozrostu ziaren z dwoma podstawowymi stanami i stanem buforowym q_b

W takim przypadku spełnienie warunku przejścia spowoduje natychmiastowe przejście w stan buforowy q_b i dopiero po przebadaniu wszystkich komórek (warunek I_1) ów stan buforowy zamieni się w stan końcowy q_1 . Wprowadzenie stanu buforowego ma i inny skutek: zamiast dwóch zmiennych odpowiadających za poprzedni i następny stan komórki występuje tylko jedna, która może określać zamiast dwóch – trzy stany. Poza tym w takiej strukturze kilka rodzajów asynchronicznych automatów sprowadza się do kategorii automatów synchronicznych (por. rozdz. 1 oraz część trzecia).

Zastosowanie bardziej zaawansowanego algorytmu rozrostu ziaren (por. rozdz. 2), który pozwala sterować kształtem rosnącego ziarna i jest oparty na wprowadzeniu opóźnienia przejścia z jednego stanu w inny w zależności od właściwości przestrzeni,

czasu lub innych parametrów, wymaga jeszcze jednego stanu. Nazwijmy go stanem przejściowym q_p . Schemat takiego automatu komórkowego przedstawiono na rysunku 4.3.



Rys. 4.3 Synchroniczny automat komórki oparty na zastosowaniu algorytmu rozrostu ziaren z dwoma podstawowymi stanami i dwoma stanami dodatkowymi

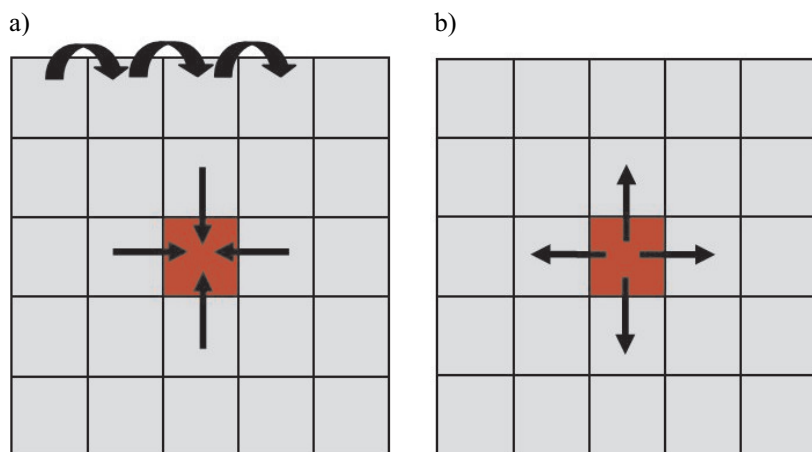
Opóźnienie przejścia w najprostszym przypadku może być wyrażone przez liczbę cykli obliczeń (lub wskazywać chwilę przejścia). W przedstawionym na rysunku 4.3 schemacie, po zbadaniu wszystkich komórek i przejściu części z nich ze stanu początkowego q_0 w stan buforowy q_b , komórki ze stanu q_b przechodzą w stan przejściowy q_p . Kiedy dla wybranej komórki w stanie przejściowym q_p zostanie obliczona liczba cykli lub chwila przejścia (warunek I_2 zostanie spełniony), komórka ta przyjmuje stan końcowy q_1 . Jeżeli istnieją komórki, których przejście może się odbyć bez opóźnienia, obydwa dodatkowe stany, q_b i q_p , są konieczne, natomiast jeżeli w automatach nie występuje możliwość przejścia bez opóźnienia, nie ma konieczności wykorzystania stanu buforowego q_b i schemat automatu jest podobny do schematu pokazanego na rysunku 4.2, z zamianą stanu q_b na q_p i warunku I_1 na I_2 .

4.2. Frontalne automaty komórkowe

Rozpatrzmy teraz działanie automatów komórkowych w warunkach stacjonarnych, kiedy w pobliżu rozpatrywanej komórki nie zachodzą żadne zmiany. Mówiąc precyzyjniej, interesuje nas tylko wyznaczone otoczenie wybranej komórki, ponieważ sprawia ono, że może ona zmienić swój stan. Jeżeli w otoczeniu komórki nie zachodzą żadne zmiany (zmiany stanu sąsiednich komórek), sama komórka nie znajduje się w stanie przejściowym oraz na jej stan nie wpływa czynnik losowy, zgodnie z zasadami działania automatów komórkowych nie ma żadnych podstaw, by oczekiwać jakichkolwiek zmian jej stanu. Więc jeżeli istnieją narzędzia, za pomocą których można zidentyfikować warunki stacjonarne w tym czy innym obszarze przestrzeni, można wyłączyć ten obszar z obliczeń w bieżącym cyklu obliczeniowym, co pozwoliłoby na ich przyspieszenie. Jednak stwierdzenie, czy obszar znajduje się w warunkach stacjonarnych, nie jest proste; okazuje się, że zadanie odwrotne, a mianowicie określenie obszaru, w którym zachodzą zmiany, jest znacznie łatwiejsze. Zakładamy, że rozpatrywany obszar i znajdujące się w nim komórki należą do stacjonarnych, jeżeli nie zostanie spełniony warunek odwrotny. Indykatorem zachodzących zmian jest zmiana stanu komórki. W takim przypadku główna zasada działania automatów komórkowych powinna zostać zmodyfikowana.

Działanie automatu komórkowego można opisać w następujący sposób: stan komórki w kolejnym kroku zależy od aktualnego stanu komórek w jej otoczeniu (oraz może zależeć od jej stanu bieżącego) (Svyetlichnyy 2010). Należy podkreślić, że nie ma mowy o zmianach ani stanu samej komórki, ani stanu jej sąsiadów, co oznaczałoby konieczność badania wszystkich komórek albo części komórek sąsiednich. Najistotniejszym elementem jest tu jednak zależność stanu komórki od stanu innych komórek. Wówczas powstaje pytanie, czy jeżeli w określeniu działania automatu komórkowego zamienimy miejscami komórkę i jej otoczenie, coś się zmieni? Taką zamianę można opisać w następujący sposób: stan sąsiednich komórek zależy od stanu danej komórki, więc rozpatrując otoczenie, można powiedzieć, iż działanie automatu będzie się opierało nie na otoczeniu wejściowym, lecz na otoczeniu wyjściowym. Wpływ odrębnej komórki na inną komórkę nadal się nie zmienia. W konwencjonalnych automatach komórkowych efekt działania całego otoczenia wejściowego liczony jest od razu, natomiast w proponowanym tu podejściu bierze się pod uwagę tylko oddziaływanie odrębnej komórki, ale na całe wyjściowe otoczenie. Efekt działania komórki może być kumulowany w komórkach wyjściowego otoczenia lub przejawiać się bezpośrednio, jak to ma miejsce w rozpatrywanym algorytmie rozrostu ziarna.

Na rysunku 4.4 pokazano zasady działania automatów klasycznych i frontalnych oraz kolejność badania przestrzeni i kierunku przekazywania informacji. W odniesieniu do komórki przedstawionej w centralnej części rysunku 4.4a sprawdzane są reguły przejścia i otrzymuje ona z sąsiednich komórek (przykład dotyczy wejściowego otoczenia von Neumanna) informacje o ich stanach. Na rysunku zostało to oznaczone skierowanymi ku komórce strzałkami. W efekcie w celu zrealizowania jednego kroku czasowego należy po kolei (strzałki w górnej części rysunku 4.4a) zbadać całą przestrzeń pod względem reguły przejścia, przy czym badane jest całe otoczenie wejściowe każdej komórki.



Rys. 4.4. Badanie przestrzeni i przekazywanie informacji w klasycznych (a) i frontalnych (b) automatach komórkowych (Svyetlichnyy, on-line)

We frontalnym automacie komórkowym kierunek przekazywania informacji jest odwrotny, co pokazano na rysunku 4.4b (strzałki skierowane na zewnątrz od centralnej komórki). Jest to pierwszy krok, który na razie jeszcze nie przynosi żadnych korzyści, ponieważ również w tym przypadku należy przebadać całą przestrzeń i nie ma różnicy, czy komórka będzie zbierała informacje, czy też ją rozsyłała.

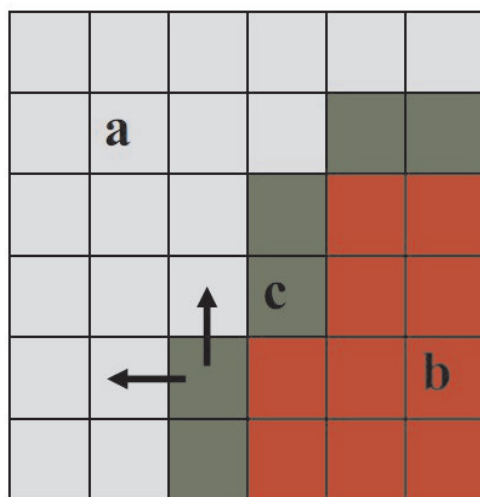
Różnica między tymi algorytmami powstanie wtedy, gdy będziemy rozpatrywać stany ustalone i przejściowe. Rozpatrzmy działanie zmodyfikowanego podejścia do algorytmu rozrostu ziarna. Jeżeli w otoczeniu komórki nie zachodzą żadne zmiany, pozostaje ona w tym samym stanie bez zmian – można zatem powiedzieć, że znajduje się w stanie ustalonym. Również jej wpływ na sąsiednie komórki się nie zmienia, więc nie ma sensu badać tej komórki i jej otoczenia. I odwrotnie, jeżeli w otoczeniu komórki następują zmiany, które spowodują zmianę jej stanu, będzie się ona znajdowała w stanie przejściowym. Już w tym momencie pojawiają się różnice między tymi dwoma algorytmami, ponieważ zbieraniu informacji z otoczenia wejściowego nie towarzyszą żadne okoliczności wskazujące na to, czy należy zbierać taką informację, czy też nie. Natomiast przy rozprowadzaniu informacji od komórki do jej wyjściowego otoczenia zmiana stanu komórki jest podstawą do podjęcia tej czy innej decyzji. Komórka, która nie zmienia swego stanu, nie będzie wysyłała żadnej informacji do swego otoczenia, natomiast gdy w jej stanie nastąpią zmiany, komórki w całym otoczeniu otrzymają taką wiadomość i na jej podstawie sprawdzą reguły przejścia, nie badając swego otoczenia. To jest drugi krok do automatu frontalnego.

Trzeci, a zarazem ostatni krok polega na wykorzystaniu do obliczeń tylko komórek znajdujących się w stanie przejściowym. Jeżeli drugi krok polegał na wysyłaniu sygnałów wyłącznie od komórek w stanie przejściowym, przy jednoczesnym zaniechaniu badania całej przestrzeni, to istotą trzeciego kroku modernizacji algorytmu jest ograniczenie badania przestrzeni wyłącznie do komórek w stanie przejściowym.

Jako przykład skutecznego zastosowania tych kroków można podać proces rozrostu ziaren. Fragment przestrzeni z rosnącym ziarnem pokazano na rysunku 4.5. W dowolnym procesie rozrostu kryształu lub ziarna można wyróżnić trzy strefy. W pierwszej strefie (a) nie następują jeszcze żadne zmiany stanu początkowego. W strefie drugiej (b) takie zmiany już nastąpiły i kolejne nie zajdą. Natomiast w trzeciej strefie (c) zachodzą zmiany w rozpatrywanej chwili czasu i dlatego też tylko ta trzecia strefa może być wykorzystana w obliczeniach. Zarówno pierwsza, jak i druga strefa powinny być wyłączone z obliczeń w bieżącym kroku, ponieważ zmiany stanu komórek w ich obrębie nie są oczekiwane i nie mogą nastąpić. W przypadku zaznaczonej na rysunku 4.5 komórki w stanie przejściowym strzałkami oznaczono przekazanie istotnej informacji do jej wyjściowego otoczenia. Tak więc w bieżącym kroku tylko sześć komórek w strefie (c) będzie branych pod uwagę w obliczeniach.

Komórki, które znajdują się w pewnej odległości od granicy rosnącego ziarna, pozostają w warunkach stacjonarnych niezależnie od tego, czy są w stanie początkowym q_0 czy w stanie końcowym q_1 . Nie zachodzą w nich żadne zmiany i mogą one być na pewnych stadiach obliczeń zupełnie z nich wyłączone. Zatem faktycznie interesuje nas

nie stan komórki, a jej przejście z jednego stanu w inny. Poniżej, opierając się na przykładzie automatu komórkowego przedstawionego na rysunku 4.3, opiszemy podejście stosowane we frontalnych automatach. W podejściu konwencjonalnym dopiero stan końcowy q_1 wyznacza zmiany zachodzące w kolejnych komórkach, czyli powoduje on pojawienie się warunku I_0 , który wywoła przeniesienie odpowiedniej komórki w stan buforowy q_b . Tymczasem wiadomo, że stan końcowy jest skutkiem zaistnienia odpowiedniego warunku I_2 . Innymi słowy, warunek I_2 dla jednej komórki jest przyczyną powstania warunku I_0 w komórkach sąsiednich. Wówczas w zaproponowanym podejściu warunek I_2 powoduje nie tylko zmianę stanu komórki, której warunek ten dotyczy, ale również zmienia stan wszystkich sąsiednich komórek z początkowego q_0 na buforowy q_b . W pozostałych swych częściach algorytm nie różni się od konwencjonalnego.



Rys. 4.5. Fragment przestrzeni z rosnącym ziarnem
(Svyetlichnyy, on-line)

Takie ujęcie zależności między komórką i jej otoczeniem w połączeniu z wyznaczeniem obszarów niestacjonarności pozwala osiągnąć na tyle zaskakujący skutek w postaci przyspieszenia obliczeń, że wydaje się, iż stosowanie podejścia konwencjonalnego jest zupełnie nieuzasadnione. Jednak zawsze należy zbadać, czy w wyniku zastosowania zaproponowanego tu podejścia nie zostaną utracone jakieś ważne właściwości automatów.

Dzięki takiej modyfikacji odpada konieczność badania każdej komórki pod kątem zachodzących w jej otoczeniu zmian. Informacje o tych zmianach docierają do komórki w postaci zewnętrznego sygnału, który automatycznie inicjuje w niej proces zmieniania stanu. Przypomina to zasadę domina: jedna padająca kostka powoduje upadek kolejnej, która przewraca następną, przy czym upadek powoduje nie stan sąsiedniej, leżącej już kostki (jak w konwencjonalnym automacie), lecz proces przejścia tej kostki

ze stanu stojącego w stan leżący. Jednak podczas badania całego procesu nie ma sensu badać kostek, które znajdują się daleko od ruchomego frontu, w którym właśnie zachodzą zmiany, przy czym należy pomijać zarówno kostki, które jeszcze stoją, jak i te, które już leżą. W tym sensie przechodzimy od badania całej przestrzeni do rozpatrywania bardzo ograniczonego obszaru, a mianowicie do badania ruchomego frontu zachodzących zmian. Stąd pochodzi nazwa takich automatów komórkowych – „frontalne”. W zastosowaniu do modelowania procesu rozrostu ziaren ten obszar lub ruchomy front odpowiada granicy rosnącego ziarna i przyległej do granicy, po jednej lub po obu jej stronach wąskiej warstwy komórek.

Porównując konwencjonalne automaty komórkowe z frontalnymi, można zauważyć jedną drobną, ale bardzo istotną między nimi różnicę: warunek przejścia komórki ze stanu początkowego w stan buforowy jest w nich wyznaczany w różny sposób. W konwencjonalnych automatach komórkowych należy zbadać całą przestrzeń i otoczenie komórek, natomiast w automatach frontalnych nie prowadzi się żadnych badań – warunek przejścia jest wyznaczany automatycznie przy przejściu frontu przez komórki. W rezultacie w automacie konwencjonalnym sąsiedztwo każdej komórki jest badane w każdym cyklu obliczeniowym, a w automacie frontalnym – tylko jeden raz podczas całych obliczeń. Już ten fakt wskazuje, o ile oszczędniejszy jest algorytm automatów frontalnych. Wynika z tego jeszcze jedna ciekawa własność frontalnych automatów komórkowych – czas obliczeń w znacznie mniejszej mierze zależy w nich od liczby cykli obliczeniowych.

Jednak, jak wiadomo, każda zaleta idzie w parze z jakąś wadą. Algorytm funkcjonowania automatu komplikuje się i jest to związane nie tylko z bardziej skomplikowanym sposobem modelowania zmiany stanu komórki. Należy w inny sposób zorganizować przestrzeń i wprowadzić dodatkowe struktury. Nadal pozostaje konieczne uwzględnienie stanu buforowego, który oddziela procesy zachodzące przed zmianą i po zmianie stanu. Pełni on funkcję zapory przy przejściu ze stanu początkowego w stan końcowy. Nie oznacza to jednak, że przejście ze stanu buforowego powinno koniecznie nastąpić w końcu cyklu. Można je również przeprowadzić na początku cyklu lub w jego środku, prowadząc obróbkę informacji dotyczącej komórek w różnych stanach. Koniecznym warunkiem prawidłowego funkcjonowania automatów jest rozdzielenie w czasie czynności związanych z obróbką informacji dotyczących komórek znajdujących się w różnych stanach, co oznacza, że nie można rozpatrywać komórek w dowolnej kolejności. Na przykład powinno się najpierw rozpatrywać wszystkie komórki znajdujące się w stanie przejściowym, a później te będące w stanie początkowym i w stanie buforowym. Przestrzenne rozmieszczenie komórek w tym nie pomaga, a kilkakrotnie (liczba bliska liczbie stanów komórki) skanowanie przestrzeni wydaje się dość rozrzutnym podejściem, powodującym stworzenie struktur, które będą łączyć komórki w tym samym stanie, czyli list komórek. Wówczas na początku obliczeń wszystkie komórki znajdowałyby się na jednej początkowej liście, w toku obliczeń byłyby przenoszone z jednej listy do kolejnej, na koniec trafiając na listę komórek w stanie końcowym. Oczywiście tworzenie list i ich obróbka ma poza zaletami również wady – wymaga dodatkowej pamięci i czasu.

Należy jednak zauważyć, że przenoszenie komórek z jednej listy do innej nie zawsze jest zadaniem bardziej skomplikowanym niż zmiana stanu komórki. Na przykład przenoszenie komórek będących w stanie buforowym polega na dołączeniu całej ich listy do listy komórek znajdujących się w stanie przejściowym. Tworzenie list komórek w konwencjonalnych automatach komórkowych również może zmniejszyć nakłady obliczeniowe, ale skuteczność takiego rozwiązania jest znacznie mniejsza.

4.3. Przykłady oszacowań nakładów obliczeniowych

Rozpatrzmy kilka prostych wariantów symulacji rozrostu ziaren (Svyetlichnyy, on-line):

- **Wariant pierwszy:** przestrzeń dwuwymiarowa 100×100 komórek, jeden zarodek. Obliczenia do wypełnienia całej przestrzeni z sąsiedztwem von Neumanna (czterech sąsiadów) bez opóźnienia.
- **Wariant drugi:** przestrzeń trójwymiarowa $100 \times 100 \times 100$ komórek, jeden zarodek. Obliczenia do wypełnienia całej przestrzeni z sąsiedztwem Moore’a (dwudziestu sześciu sąsiadów), rozrasta się ziarno mające kształt kuli ze średnim opóźnieniem przejścia równym 3.

Wyniki porównania nakładów obliczeniowych zebrano w tabeli 4.1. Nakłady, jak widać, różnią się o kilka rzędów. Im większa liczba cykli, tym większa jest różnica.

Tabela 4.1

Nakłady obliczeniowe w dwóch wariantach symulacji przy użyciu klasycznych i frontalnych automatów komórkowych (Svyetlichnyy, on-line)

Nakłady obliczeniowe	Klasyczny automat		Frontalny automat		Stosunek nakładów obliczeniowych w automacie klasycznym i frontalnym	
	Wariant 1	Wariant 2	Wariant 1	Wariant 2	Wariant 1	Wariant 2
Liczba cykli od zarodkowania do wypełnienia całej przestrzeni	50	184	50	184	1	1
Liczba badanych komórek we wszystkich cyklach	$5 \cdot 10^5$	$1,84 \cdot 10^8$	10^4	$3 \cdot 10^6$	50	61
Liczba interakcji między komórkami we wszystkich cyklach	$2 \cdot 10^6$	$4,78 \cdot 10^9$	$4 \cdot 10^4$	$2,6 \cdot 10^7$	50	184

Nakłady obliczeniowe w wariantach z większą liczbą zarodków mogą być oszacowane w przybliżeniu. Liczba zarodków wpływa jedynie na liczbę cykli koniecznych do

wypełnienia przestrzeni. W przestrzeni dwuwymiarowej liczba cykli średnio zmniejsza się odwrotnie proporcjonalnie do pierwiastka z liczby zarodków, w przestrzeni trójwymiarowej – odwrotnie proporcjonalnie do pierwiastka trzeciego stopnia z liczby zarodków.

Wszystkie podane wyżej liczby związane są wyłącznie z nakładami na badanie spełnienia warunku I_0 (rys. 4.2 i 4.3) i ze zmianą stanu z początkowego q_0 na buforowy q_b . Ogólna liczba badanych sąsiadów w całym procesie obliczeniowym jest proporcjonalna do liczby cykli, liczby komórek i liczby sąsiadów w wybranym otoczeniu.

Przejście ze stanu buforowego q_b w stan przejściowy q_p wymaga nakładów proporcjonalnych do liczby cykli i liczby komórek.

Nakłady na opóźnienie przejścia ze stanu q_p w stan końcowy q_1 są proporcjonalne do liczby komórek (ponieważ wszystkie komórki przechodzą w ten stan jednokrotnie) i średniej wartości opóźnienia (liczonej na przykład jako średnia liczba cykli opóźnienia).

Wykorzystanie FCA zamiast konwencjonalnych automatów komórkowych umożliwia skrócenie czasu obliczeń w modelu 2D, ale jest szczególnie spektakularne w modelu 3D, ponieważ istotne obszary przestrzeni są wyłączone z obliczeń w każdym kroku czasowym i tylko cienka warstwa komórek jest uwzględniana w obliczeniach. W klasycznych CA każdy krok wymaga praktycznie tych samych nakładów i czas jego wyznaczenia w ciągu całej symulacji pozostaje niezmienny. Natomiast czas obliczeń w jednym kroku FCA zależy od liczby uczestniczących w obliczeniach komórek i w dużym stopniu się zmienia, zawsze pozostając małym ułamkiem czasu obliczeń jednego kroku w klasycznych CA. Każda komórka FCA faktycznie jest brana pod uwagę jednokrotnie w całym procesie obliczeń, natomiast w klasycznym algorytmie jest poddawana obliczeniom w każdym kroku czasowym.

4.4. Uniwersalny frontalny automat komórkowy

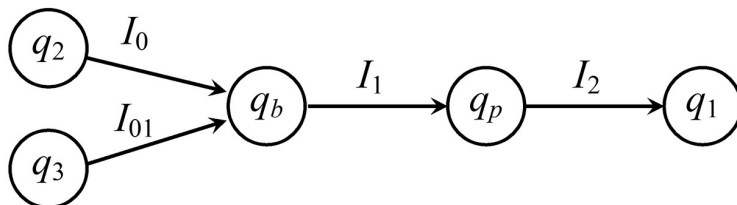
Wróćmy do schematu automatów komórkowych przedstawionego na rysunku 4.3 i reguły przejścia (4.1). Za pomocą takiego automatu można modelować różne procesy zachodzące w materiałach, na przykład krzepnięcie i krystalizację, rekrytalizację i przemiany fazowe, czyli procesy obejmujące dwa stadia: zarodkowanie i rozrost ziarna. Frontalne automaty komórkowe w opisanej wyżej postaci mogą być zastosowane tylko do modelowania rozrostu ziarna, natomiast czynnik losowy, który decyduje o przebiegu zarodkowania, powinien być wyznaczony oddzielnie.

Zastosowanie wzoru (4.1) w przedstawionej postaci nadal wiąże się z koniecznością skanowania całej przestrzeni w celu wyznaczenia warunków zarodkowania. Należy wyznaczyć wartość liczby losowej i porównać ją z ustalonym progiem prawdopodobieństwa zarodkowania. Jeżeli zarodkowanie następuje tylko przed rozrostem ziarna, można nie ingerować już w ten algorytm, pozostawiając w nim wszystko bez zmian. Natomiast zastosowanie tego algorytmu w odniesieniu do procesów, w których zarodkowanie trwa przez cały czas rozrostu ziaren znacząco wydłuży proces obliczeń.

Należy więc zastąpić go innym, bardziej skutecznym algorytmem. W algorytmie modelowania zarodkowania również można pozbyć się rozrzućnej metody skanowania całej przestrzeni. Jedno ze stosowanych przez autora niniejszej pracy podejść opisano poniżej.

Prawdopodobieństwo p pojawienia się zarodka nadal może pozostać parametrem charakteryzującym proces zarodkowania. Może ono być funkcją czasu lub mieć ograniczenia dotyczące lokalizacji miejsc powstawania zarodków (np. na granicy ziaren). Inne zależności, na przykład od współrzędnych, temperatury lub odkształcenia, ograniczają zastosowanie tego algorytmu. Wówczas, wyznaczając obszar, w którym może nastąpić zarodkowanie, a w szczególności liczbę przydatnych do zarodkowania komórek n , można wyznaczyć wartość oczekiwaną liczby zarodków n_z w wyznaczonych cyklach obliczeniowych $n_z = pn$. W tym podejściu parametrem charakteryzującym proces zarodkowania, czyli prawdopodobieństwo p , jest faktycznie oczekiwana liczba zarodków w odniesieniu do objętości całej przestrzeni (liczby komórek N). Dalej, zakładając na przykład normalny (lub inny) rozkład prawdopodobieństwa dla liczby zarodków, wyznaczamy liczbę zarodków w bieżącym cyklu obliczeniowym w dopuszczalnym obszarze. Wybór poszczególnych komórek z dopuszczalnego obszaru zarodkowania nie następuje żadnych trudności, w szczególności kiedy komórki te są zebrane na odpowiedniej liście. W przypadku opisanych powyżej frontalnych automatów komórkowych komórki są już zebrane na liście komórek w stanie początkowym. Nie dotyczy to jedynie algorytmu tworzenia początkowej struktury, kiedy zarodki mogą powstać w dowolnym miejscu przestrzeni.

Wprowadzenie ograniczenia dotyczącego miejsc zarodkowania wymaga stworzenia dodatkowej listy. W przypadku zarodkowania na granicy ziarna będzie to lista komórek znajdujących się na granicy ziarna. Wówczas sensownym działaniem będzie podzielenie wszystkich komórek początkowej mikrostruktury na komórki leżące na granicy lub wewnątrz ziarna. Komórkom leżącym na granicy ziarna przypiszemy stan q_2 , zaś komórkom leżącym wewnątrz ziarna – stan q_3 . Frontalny automat komórkowy przyjmie strukturę pokazaną na rysunku 4.6 (Malinowski *et al.* 2009).

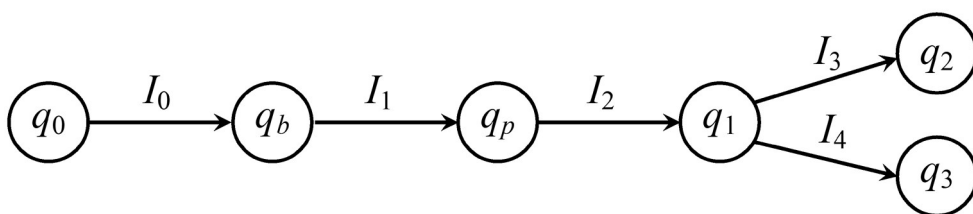


Rys. 4.6. Frontalny automat komórki rozróżniający początkowy stan komórki na granicy q_2 i wewnątrz ziarna q_3 (Malinowski *et al.* 2009)

Na powyższym rysunku warunek I_0 jest taki sam jak poprzednio (rys. 4.3), czyli uwzględnia zarodkowanie i rozrost ziarna, ale warunek I_{01} uwzględnia już tylko rozrost ziarna.

Wróćmy do algorytmu tworzenia początkowej struktury (rys. 4.3). Jest to na ogół algorytm wstępny, po którym stosowano już badanie zmian mikrostruktury: czy to rekrytalizacji, czy przemiany fazowej, czy innego procesu. Zatem stan końcowy q_1 i uzyskana mikrostruktura bez rozróżnienia komórek na granicy i wewnątrz ziarna nie mogą być wprost wykorzystane w kolejnym etapie modelowania.

Ze względu na dalsze jego wykorzystanie, algorytm tworzenia początkowej struktury (rys. 4.3) należy rozszerzyć, dodając stan komórek na granicy (q_2) i wewnątrz (q_3) ziarna, a wówczas stan q_1 będzie odpowiadał komórkom leżącym na froncie zmian zachodzących w mikrostrukturze, czyli na granicy rosnącego ziarna. W efekcie komórka może mieć dwa stany na granicy ziarna: q_1 i q_2 . Rozróżniamy je, nazywając stan q_1 stanem na ruchomej granicy ziarna lub stanem frontalnym, a stan q_2 – stanem na nieruchomej granicy ziarna lub po prostu na granicy ziarna. Po wprowadzeniu tych zmian schemat frontального automatu komórkowego przyjmuje postać pokazaną na rysunku 4.7.



Rys. 4.7. Frontalny automat komórki tworzenia początkowej struktury rozróżniający końcowy stan komórki na granicy q_2 i wewnątrz ziarna q_3

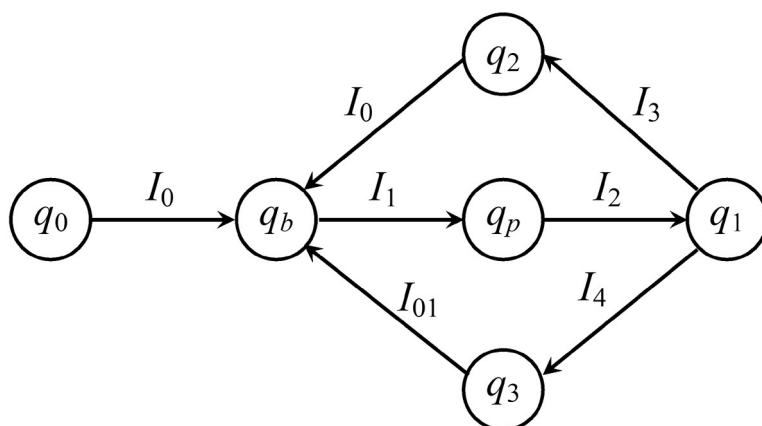
Wprowadzając dodatkowe badanie otoczenia podczas przejścia ze stanu przejściowego q_p , można by było pominąć stan frontalny q_1 , jednak bardziej przejrzyste wydaje się funkcjonowanie automatu przy zachowaniu stanu frontального, poza tym warunki są wówczas prostsze i sprawdzenie mniej skomplikowane. Warunkiem przejścia ze stanu frontального q_1 w stan wewnątrz ziarna q_3 jest przynależność wszystkich komórek w otoczeniu von Neumanna do tego samego ziarna. Niespełnienie tego warunku oznacza przejście w stan na nieruchomej granicy ziarna q_2 ($I_4 = \bar{I}_3$).

Porównajmy dwa warianty automatów: w pierwszym stan frontalny q_1 będzie pominięty, w drugim zostanie zastosowany. Stan przejściowy komórki q_p oznacza, że front przemian przechodzi właśnie przez tę komórkę i przed tym frontem przynależność do rosnącego ziarna może być jeszcze niewyznaczona. Zatem najpierw powinno nastąpić przejście sąsiednich komórek w stan buforowy z przypisaniem ich do rosnącego ziarna, a dopiero potem sprawdzanie całego otoczenia komórki w stanie przejściowym w celu ustalenia nowego stanu q_2 lub q_3 . Naruszenie tej kolejności w większości przypadków spowoduje nieuzasadnione przejście w stan na nieruchomej granicy ziarna q_2 . Stan frontalny q_1 oznacza faktycznie koniec przejścia frontu przemian przez komórkę.

Przy ustalaniu kolejności badania komórek, która zależy od ich stanu, a nie ich przestrzennej lokalizacji, należy pamiętać, że stan frontalny q_1 powinien być badany po przypisaniu wszystkich sąsiednich komórek do tego czy innego ziarna. Fakt ów wynika

z tego, że litera I_2 oznacza wykonanie dwóch podstawowych czynności: zmianę stanu komórki z q_p na q_1 oraz zmianę stanu wszystkich lub części sąsiednich komórek przed frontem przemian na stan buforowy q_b , z przypisaniem ich do odpowiednich ziaren. Zatem dzięki zastosowaniu stanu q_1 unikamy konieczności pilnowania kolejności prowadzonych czynności, ponieważ ich prawidłowa kolejność zachowuje się automatycznie. Natomiast pytanie: czy pozostawić stan q_1 , czy pominąć go, pozostawiamy do rozważenia programiście.

Po stworzeniu dwóch automatów komórkowych przedstawionych na rysunkach 4.6 i 4.7 dalszym naturalnym krokiem jest połączenie ich w jeden uniwersalny frontalny automat komórkowy (Svyetlichnyy 2007; Svyetlichnyy i Matachowski 2007). Schemat takiego automatu przedstawiono na rysunku 4.8.



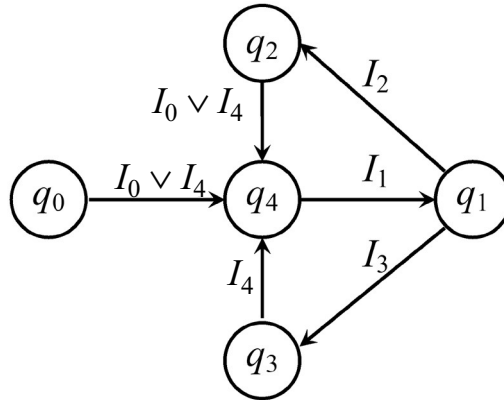
Rys. 4.8. Uniwersalny frontalny automat komórkowy (Svyetlichnyy 2010)

Kolejna modyfikacja polega na połączeniu stanu buforowego q_b ze stanem przejściowym q_p , jak to zostało zaproponowane w podrozdziale 4.1. Formalny opis opracowanego uniwersalnego frontального automatu wygląda tak, jak to przedstawiono poniżej.

Podstawą automatu komórkowego jest komórka, która może być opisana za pomocą szeregu parametrów. Owe parametry to na przykład położenie w przestrzeni, stany i inne. Opis automatu składa się z zestawu stanów Q , alfabetu Σ , reguły przejścia δ , stanów początkowych Q_{in} i końcowych Q_f i może wyglądać następująco: $M = (Q, \Sigma, \delta, Q_{in}, Q_f)$. Automat komórki schematycznie przedstawiono na rysunku 4.9.

Zbiór stanów $Q = \{q_0, q_1, q_2, q_3, q_4\}$, czyli stany, które może przyjmować komórka, określono w następujący sposób: stan początkowy q_0 , „komórka na froncie zmian” (stan frontalny) q_1 , „komórka na granicy ziarna” q_2 , „komórka w środku ziarna” q_3 , stan przejściowy q_4 . Stan przejściowy q_4 wprowadza opóźnienie, które pozwala kontrolować prędkość rozrostu ziarna, i jest stanem aktywnym podobnie jak stan frontalny q_1 . Natomiast stany q_0 , q_2 i q_3 są zasadniczo bierne i nie są brane pod uwagę w obliczeniach. Wejściowym stanem może być $Q_{in} = \{q_0\}$ w algorytmie kształtowania

początkowej mikrostruktury lub $Q_{in} = \{q_2, q_3\}$ w innych algorytmach modelowania mikrostruktury. Stanem końcowym formalnie zawsze jest $Q_f = \{q_2, q_3\}$, chociaż jeżeli proces nie jest kompletny, stan końcowy może być dowolny.



Rys. 4.9. Uniwersalny frontalny automat komórkowy do badań zmian mikrostruktury (Svyetlichnyy 2012a, 2012b)

Alfabet komórki zawiera „litery” $\Sigma = \{I_0, I_1, I_2, I_3, I_4\}$ i jest traktowany jako zestaw możliwych warunków przejścia pomiędzy poszczególnymi stanami, natomiast reguły przejścia δ w przypadku spełnienia warunku przejścia (pojawienia się odpowiedniej litery) I_k definiują stan bieżący q_i i kolejny stan q_{i+1} . Reguły δ zwykle przedstawiano albo w postaci tabeli, albo opisowo. Ograniczając się do pobieżnego omówienia pierwszego sposobu ich przedstawiania, można stwierdzić, że litery mają następujący sens: I_0 oznacza zarodkowanie, I_1 – czas opóźnienia przejścia ze stanu przejściowego w stan frontalny, I_2 i I_3 określają, czy komórka znajduje się na granicy ziarna, czy wewnątrz niego, odpowiednio ($I_3 = \bar{I}_2$), I_4 oznacza, iż granica rosnącego ziarna osiągnęła tę komórkę i została włączona w proces rozrostu owego ziarna.

Język stosowany do opisu automatu komórkowego składa się z kilku słów, które służą do objaśniania przejścia komórki ze stanu początkowego w stan końcowy, przykładem niech będzie fraza $(I_0 \vee I_4)I_1(I_2 \vee I_3)$, która w stanie początkowym q_0 opisuje symulację początkowej mikrostruktury.

Cechą szczególną automatów komórkowych jest zamknięty obwód przejść i stanów: $\dots q_1(q_2 \vee q_3)q_4\dots$ Stany, nie tylko stany końcowe, ale również początkowe, w kolejnych cyklach modelowania mogą się powtarzać kilkakrotnie. Taka właściwość jest pomocna przy modelowaniu niektórych zjawisk i procesów.

Automat komórkowy skuteczniej pracuje ze zmiennym krokiem czasowym. Głównym ograniczeniem kroku czasowego jest przemieszczenie się frontu rozrostu ziarna, który nie powinien przejść przez komórkę w jednym kroku. Zatem krok czasowy musi być krótszy niż czas opóźnienia przejścia każdej komórki ze stanu q_4 w stan q_1 .

Dlatego przy obliczaniu maksymalnej długości kroku czasowego analizuje się czas opóźnienia wszystkich komórek i bierze pod uwagę ten najkrótszy. W niektórych przypadkach istnieje również możliwość stosowania kroku czasowego dłuższego od wyznaczonego w ten sposób. Przykład taki został podany w części trzeciej niniejszej książki.

Uniwersalny frontalny automat komórkowy może być zastosowany praktycznie do modelowania wszystkich możliwych procesów, które składają się z zarodkowania i rozrostu ziaren. Obecność zamkniętych obwodów sprawia, że taki automat można zastosować do modelowania procesów wielokrotnego zarodkowania i rozrostu ziaren. Jako przykład można tu podać proces rekrytalizacji dynamicznej, w którym permanentnie trwa zarodkowanie, rozrost ziaren, ich „starzenie się” i stale powstają nowe nieruchome granice ziaren, które są źródłem nowych zarodków. Można też zasymulować wieloetapowe procesy z kilkoma odkształceniami, rekrytalizacjami.

5. Przygotowanie danych do tworzenia siatki metody elementów skończonych

W modelach wieloskalowych automaty komórkowe często stosowane są wraz z innymi metodami obliczeniowymi: elementów skończonych, różnic skończonych, elementów dyskretnych itp. Wówczas dane geometryczne ziaren uzyskane za pomocą automatów komórkowych należy przedstawić w postaci przydatnej do obliczeń innymi metodami. W niniejszym rozdziale zostanie rozpatrzony algorytm pozwalający stworzyć na podstawie wyników symulacji mikrostruktury automatami komórkowymi siatkę elementów skończonych. Powiązania automatów komórkowych z metodą różnic skończonych w nim nie omówiono, ponieważ w charakterze węzłów do obliczeń różnic zazwyczaj wykorzystywano środki poszczególnych komórek.

Najważniejszym etapem przygotowania danych jest przedstawienie głównych elementów ziarna (powierzchni, krawędzi, wierzchołków) w postaci jednoznacznej i dopasowanej do budowy elementów siatki MES.

5.1. Zasady przygotowania danych

Głównym celem przygotowania danych jest stworzenie plików zawierających dane o podstawowych elementach trójwymiarowej mikrostruktury. Tymi elementami są: ziarno, granice ziaren w postaci płaszczyzn styku dwóch ziaren, krawędzie, czyli linie styku przynajmniej trzech ziaren (trzech płaszczyzn), oraz wierzchołki, czyli punkty styku przynajmniej czterech ziaren. Informacja o ziarnie zawiera dane o wszystkich jego płaszczyznach, krawędziach i wierzchołkach, zaś informacja o płaszczyźnie zawiera dane o obu stykających się ze sobą ziarnach oraz o ograniczających ją liniach (krawędziach) i punktach (wierzchołkach). Informacja o linii zawiera dane o ziarnach, w których ona występuje jako krawędź, o płaszczyznach, które przecinają się wzdłuż niej, oraz o dwóch punktach ograniczających tę linię. Z kolei informacja o wierzchołkach zawiera dane o ziarnach, płaszczyznach i liniach.

Do uzyskania takiej informacji można stosować różne algorytmy, jednak tu wybrano jedną konkretną opcję, która pozwoliła uzyskać prosty, jednoznaczny wynik. Poniżej omówiono pokrótce inne algorytmy i niektóre z przyczyn ich odrzucenia (Svyetlichnyy *et al.* 2011).

Schemat jednego algorytmu był następujący: wyznaczenie komórek i ich ścian tworzących granicę ziaren, przedstawienie ich jako punktów na powierzchni, opisanie powierzchni równaniem, wyznaczenie linii ich przecięcia i punktów na podstawie równań powierzchni. Głównymi problemami były eliminacja linii powstałych w wyniku przecięcia powierzchni, które w rzeczywistości nie stykają się ze sobą i niedokładności określenia powierzchni, które powodują, że linie przecięcia nie zbiegają się w jednym punkcie. Zdecydowano wówczas, by powierzchnie, linie i punkty wyznaczać oddzielne.

W innym algorytmie po kolei wyznaczano powierzchnie ziaren, ich krawędzie i wierzchołki. Powierzchnie określano, biorąc pod uwagę wspólne ściany dwóch komórek należących do różnych ziaren. Krawędzie ziaren są wspólnymi krawędziami czterech komórek należących przynajmniej do trzech różnych ziaren. Wierzchołkami ziaren są wspólne wierzchołki ośmiu komórek, z których przynajmniej cztery należą do różnych ziaren. Dalej grupowano odpowiednie płaszczyzny, linie, punkty i ustalano ich współzależność. Głównymi problemami, które albo znacznie komplikowały algorytm, albo nie zostały rozwiązane, były m.in.: nieciągłość linii i powierzchni, istnienie krawędzi wspólnej dla czterech ziaren oraz wierzchołków, które są końcami tylko trzech krawędzi.

Zastosowano zatem trzeci algorytm, bazujący na zasadach topologii. Algorytm ten został szczegółowo opisany poniżej.

Ogólnie wspomniany algorytm można przedstawić w postaci ciągu po kolei wykonywanych procedur, z których żadna nie jest powtarzana. Procedury te obejmują następujące czynności:

- wczytywanie informacji o mikrostrukturze;
- sprawdzenie ciągłości ziaren;
- eliminacja zbyt drobnych ziaren;
- wyznaczenie komórek leżących na granicach ziaren (stykających się z innym ziarnem) i stworzenie listy tych komórek;
- stworzenie drzewa płaszczyzn styku ziaren i przekształcenie go w listę; każda płaszczyzna zawiera listę ścian tworzących ją komórek;
- uporządkowanie ścian komórek na powierzchni styku dwóch ziaren, sprawdzenie ich ciągłości i wyznaczenie konturu każdej płaszczyzny;
- uporządkowanie krawędzi komórek w konturze każdej płaszczyzny i włączenie tych krawędzi do linii kształtujących kontur płaszczyzny, stworzenie listy linii dla każdej płaszczyzny, przy czym linia zawiera listę krawędzi;
- stworzenie listy linii na podstawie listy linii każdej płaszczyzny i pomijanie powtarzających się linii;
- sprawdzanie linii w aspekcie podziału ich na kilka krótszych;
- stworzenie i wypełnienie listy linii dla każdego ziarna;
- stworzenie i wypełnienie listy wierzchołków wyznaczonych jako końce każdej linii z eliminacją powtórzeń;

- stworzenie i wypełnienie listy wierzchołków dla każdej płaszczyzny i ziarna oraz list płaszczyzn i ziaren dla każdego wierzchołka;
- zapisywanie wyników do plików.

5.1.1. Wczytywanie informacji o mikrostrukturze

Wejściowy plik zawiera informację o rozmiarach przestrzeni automatów komórkowych, liczbie ziaren oraz numer ziarna, do którego należy każda komórka. W podprogramie na podstawie danych o liczbie komórek i ziaren tworzą odpowiednią strukturę: tablicę komórek oraz dynamiczną listę ziaren. Podczas wczytywania numeru ziarna dla każdej komórki tworzą i wypełniają dynamiczne listy komórek należących do każdego ziarna. Informacja ta jest wykorzystywana w innych algorytmach. Tworzą również sześć urojonych ziaren o ujemnych numerach znajdujących się na zewnętrznych granicach przestrzeni automatów komórkowych.

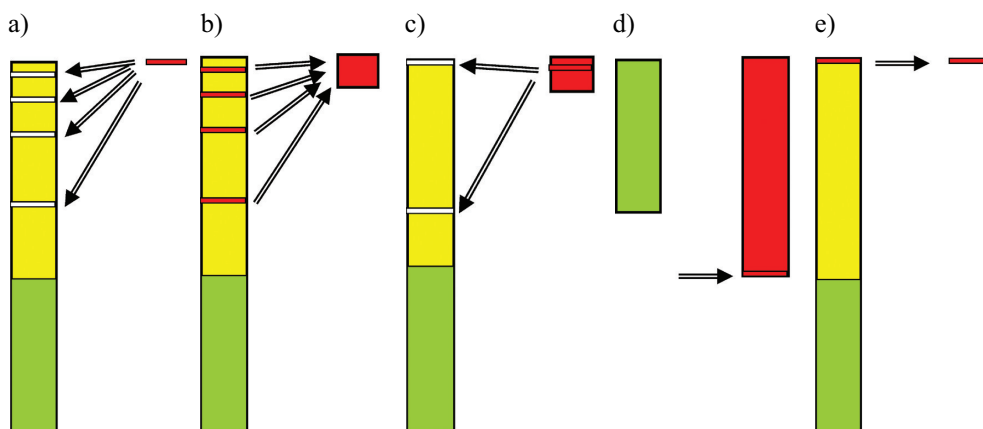
5.1.2. Sprawdzenie ciągłości ziaren

Przestrzeń automatów komórkowych często jest zamknięta, stosowano okresowe warunki brzegowe. Przyczyny takiego postępowania zostały opisane w poprzednich rozdziałach. Jednym z dostępnych wariantów jest zamknięcie przestrzeni w torus lub hipertorus. W wyniku zastosowania takiej przestrzeni powstają ziarna, które leżą na przeciwległych granicach. Dopóki modelowanie mieści się wyłącznie w ramach automatów komórkowych, fakt ten nie odgrywa istotnej roli i na ogół jest pomijany. Natomiast łączenie kilku metod w jednym modelu (na przykład automatów komórkowych z elementami skończonymi) zmienia tę sytuację zasadniczo. W metodzie elementów skończonych warunki brzegowe na przeciwległych granicach będą różne i spowodują powstanie nieciągłości w zamkniętej przestrzeni automatów komórkowych wykorzystanych do modelowania odkształcenia. Dlatego w modelu opartym na metodzie elementów skończonych przestrzeń powinna być otwarta, czyli jej przeciwległe granice nie mogą być połączone w torus (hipertorus). Takie nieciągłości mogą powstać również podczas modelowania procesu rekrytalizacji, kiedy stare ziarno zostaje podzielone rosnącymi nowymi ziarnami. Zatem model oparty na automatach komórkowych koniecznie powinien zawierać algorytm sprawdzania ziarna pod kątem ewentualnej nieciągłości i w razie jej stwierdzenia – przypisania części tego ziarna innym ziarnom.

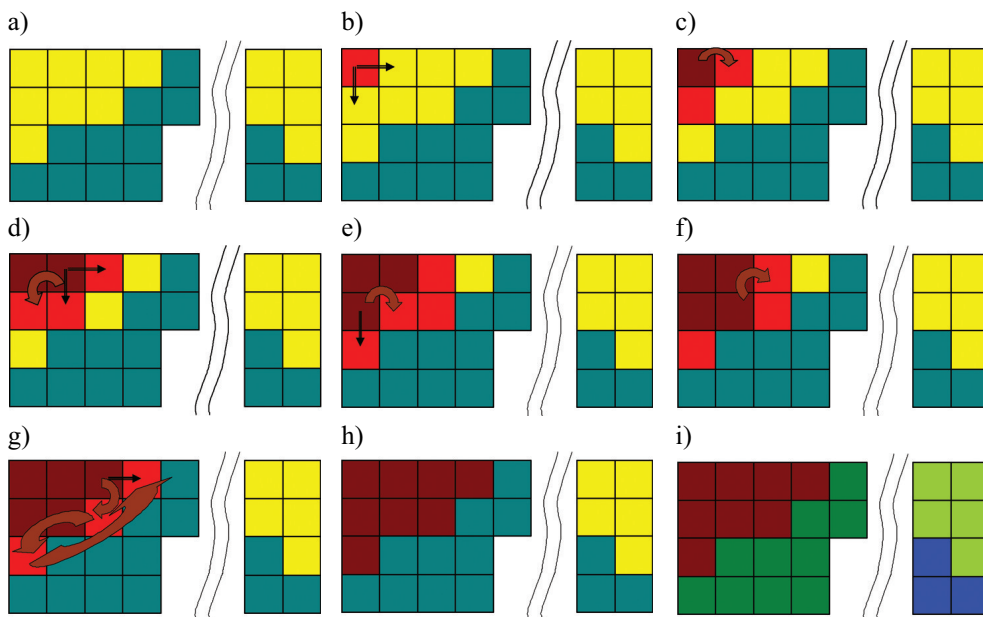
Dla wygody w podprogramie wczytywania danych stworzą również dynamiczną listę komórek należących do ziarna. Wówczas informacja o ziarnie zawiera wskaźniki odnoszące się do pierwszego i ostatniego elementu listy, natomiast element listy zawiera wskaźniki odnoszące się do poprzedniego i kolejnego elementu listy. Element listy można dołączyć do informacji dotyczącej komórki lub stworzyć oddzielną listę, która będzie zawierała te elementy. W drugim przypadku element listy powinien zawierać wskaźniki odnoszące się do komórki i odwrotnie – dane o komórce powinny zawierać wskaźniki odnoszące się do elementu listy.

Zasada działania algorytmu polega na stworzeniu nowej listy ziaren i przenoszeniu elementów (komórek) ze starej listy do nowej (rys. 5.1 i 5.2). Algorytm zrealizowano w następujący sposób:

1. Stworzyć nową listę ziaren NewGrainList.
2. Sprawdzić, czy stara lista ziaren OldGrainList jest pusta. Jeżeli lista jest pusta, oznacza to koniec algorytmu.
3. Wybrać jako analizowane ziarno (CurrentGrain) pierwsze ziarno ze starej listy ziaren (OldGrainList).
4. Sprawdzić, czy analizowane ziarno CurrentGrain zawiera komórki. Jeżeli ziarno nie zawiera żadnej komórki, usunąć ziarno CurrentGrain ze starej listy OldGrainList i przejść do pkt 2.
5. Stworzyć nowe ziarno NewCurrentGrain i dodać je do nowej listy ziaren NewGrainList.
6. Wybrać pierwszą komórkę z listy komórek ziarna CurrentGrain jako analizowaną CurrentCell (rys. 5.1a).
7. Przenieść CurrentCell z ziarna CurrentGrain do ziarna NewCurrentGrain (rys. 5.1a).
8. Wybrać pierwszą komórkę z listy NewCurrentGrain jako analizowaną NewCurrentCell (rys. 5.2a, b – czerwona komórka).
9. Wybrać wszystkie komórki sąsiadujące z NewCurrentCell należące do ziarna CurrentGrain (strzałki na rysunkach 5.1b i 5.2b) i przenieść je do ziarna NewCurrentGrain (czerwone komórki na rysunku 5.1c i 5.2c).
10. Sprawdzić, czy komórka NewCurrentCell nie jest ostatnia na liście ziarna NewCurrentGrain. Jeżeli nie jest, przejść do kolejnej komórki (rysunek 5.1d oraz brązowa strzałka na rysunku 5.2c) tej listy, która zostaje analizowaną NewCurrentCell i przejść do pkt 9.
11. Jeżeli analizowana komórka NewCurrentCell jest ostatnia (rys. 5.1e), przejść do pkt 4.



Rys. 5.1. Schemat przenoszenia komórek z listy na listę
(Svyetlichnyy *et al.* 2011)



Rys. 5.2. Kolejne etapy działania algorytmu sprawdzania nieciągłości ziaren (Svyetlichnyy *et al.* 2011)

Na rysunku 5.2 pokazano kolejne etapy wykonywania algorytmu. Ziarna znajdujące się na granicy przestrzeni wykazują nieciągłość, ponieważ w algorytmie zastosowano otwarte warunki brzegowe. W związku z tym, że do listy nowego ziarna przenoszone są tylko komórki, które sąsiadują z komórkami, które już na tej liście się znajdują, komórki niestykające się z przeniesionymi pozostają na starej liście. W wyniku takiego przenoszenia nieciągłość ziarna powoduje, że część komórek nie zostaje przeniesiona, więc z nich kształtowane jest dodatkowe ziarno i dla tego ziarna cała procedura jest powtarzana. W początkowym etapie działania algorytmu, na rysunku 5.2 można zobaczyć dwa ziarna (oznaczone dwoma kolorami). Dalej kolorem czerwonym oznaczono komórki przeniesione do nowego ziarna, dla których nie wyznaczono jeszcze wszystkich sąsiadów. Brązowym kolorem oznaczono komórki przeniesione do nowego ziarna, dla których wyznaczono wszystkich sąsiadów. Cienkie strzałki wskazują na sąsiadów analizowanej komórki, grube obrazują przejście do kolejnej komórki na liście nowego ziarna. Po wykonaniu algorytmu na rysunku 5.2 zamiast dwóch ziaren wyznaczono cztery.

Jeżeli struktura wejściowa miała zamknięte (periodyczne) warunki brzegowe lub ziarna zostały podzielone w procesie rekrytalizacji, wykonanie algorytmu prowadzi do zwiększenia liczby ziaren. W innych przypadkach liczba ziaren i cała mikrostruktura pozostaną bez zmian. Sąsiednie komórki w tym algorytmie wyznaczano, zakładając sąsiedztwo von Neumanna, czyli tworzą je tylko te komórki, które stykają się z komórką analizowaną bokami (ścianami). Wykorzystanie sąsiedztwa Moore'a może

doprowadzić do nierozróżnienia ziaren, które stykają się na przykład tylko jednym wierzchołkiem. Taka lub podobna sytuacja komplikuje wyznaczenie pozostałych elementów struktury, dlatego też tutaj wybrano sąsiedztwo von Neumanna.

5.1.3. Eliminacja zbyt drobnych ziaren

Ziarna składające się z małej liczby komórek (na przykład mniej niż 15) są uznawane przez program za zbyt drobne i zostają usunięte. Komórki, które należą do usuwanych ziaren, dołączano do ziaren sąsiednich. Jeżeli komórka sąsiadowała z kilkoma ziarnami, wybierano to, z którym miała ona większą liczbę wspólnych ścian, a jeżeli liczba wspólnych ścian była równa, ziarno wybierano losowo. Eliminacja drobnych ziaren jest opcjonalna, można zmienić najmniejszy rozmiar ziarna lub też pozostawić je wszystkie. Jednak obecność bardzo małych ziaren negatywnie wpływa na jakość siatki.

5.1.4. Stworzenie listy komórek leżących na granicach ziaren

Ten algorytm nie jest niezbędny i jeżeli można by było wyznaczyć płaszczyzny, biorąc pod uwagę wspólne ściany (boki) dwóch komórek należących do różnych ziaren, można by go było pominąć. Jednak ponieważ ów algorytm i podprogram zostały opracowane znacznie wcześniej niż frontalne automaty komórkowe, zostały wykorzystane i pozwoliły uprościć opisany niżej algorytm, który służy do wyznaczenia w początkowej strukturze komórek leżących na granicach ziaren, czyli zadania dla nich stanu q_2 „na granicy ziarna”. W algorytmie tym przeglądano po kolei wszystkie komórki i badano ich otoczenie. W przedstawionym tu algorytmie przekształcenia przestrzeni, w odróżnieniu od algorytmu wyznaczenia stanu komórki, jako otoczenie wybrano obcięte sąsiedztwo von Neumanna. Zamiast sześciu komórek w zwykłym otoczeniu von Neumanna badano zatem tylko trzy komórki, które stykają się bokami (ścianami) i mają większą współrzędną. Takie postępowanie prowadzi do prawie dwukrotnego zmniejszenia liczby badanych komórek i eliminuje dublowanie badanych ścian oraz nie wymaga dodatkowego podprogramu do usunięcia powtarzających się ścian. Każda komórka mająca przynajmniej jedną z trzech granicznych ścian jest wpisywana na listę komórek leżących na granicy ziarna.

5.1.5. Lista płaszczyzn styku ziaren

Algorytm tworzenia listy płaszczyzn z listą ścian należących do każdej płaszczyzny składa się z dwóch etapów. W pierwszym tworzone dynamiczne drzewo płaszczyzn i wypełniano listy ścian należących do każdej płaszczyzny, a w drugim przekształcano drzewo płaszczyzn w listę dynamiczną. Wykorzystanie drzewa pozwala uporządkować płaszczyzny jednocześnie z wypełnieniem list ścian, by uprościć poszukiwanie odpowiedniej płaszczyzny zarówno w tym algorytmie, jak i później, w innych. Informacją wyjściową jest lista komórek na granicy ziaren. Każda komórka wyznacza

od jednej do trzech ścian stykających się z innym ziarnem, każda z tych ścian otrzymuje numery dwóch ziaren w kolejności rosnącej oraz podawane są współrzędne jej środka. Wraz z pierwszą ścianą tworzone jest drzewo płaszczyzn i pierwsza płaszczyzna, do której listy jest przypisywana ta komórka. W odniesieniu do kolejnych ścian porównywano numery ziaren z numerami ziaren płaszczyzny w korzeniu drzewa, wybierano odpowiednią gałąź, przechodzono wzdłuż niej do kolejnej płaszczyzny. Jeżeli numer pierwszego ziarna ściany jest mniejszy od numeru pierwszego ziarna płaszczyzny, przechodzimy w lewo, jeżeli większy – w prawo, a jeżeli równy, sprawdza się numer drugiego ziarna. Jeżeli numer drugiego ziarna ściany jest mniejszy numeru drugiego ziarna płaszczyzny, przechodzimy w lewo, jeżeli większy – w prawo, jeżeli równy – dodajemy tę ścianę do analizowanej płaszczyzny. Jeśli po przejściu w lewo lub w prawo w tym węźle drzewa nie znajduje się żadna płaszczyzna, tworzymy w tym miejscu nową płaszczyznę, listę ścian w tej płaszczyźnie i dodajemy ścianę do tej listy. Powtarzamy tę procedurę dla każdej ściany aż do wyczerpania listy komórek na granicy ziaren. W efekcie uzyskujemy drzewo przygotowane do stworzenia uporządkowanej listy płaszczyzn, przy czym każda płaszczyzna już ma wypełnioną listę ścian.

Przekształcenie drzewa w listę obejmuje następujące czynności: przechodzimy po drzewie do skrajnego lewego węzła, który staje się pierwszym elementem listy. Dalej, jeżeli ten element ma prawą gałąź, wykonujemy wokół niego obrót w lewo i powtarzamy to, dopóki ów element nie będzie już miał prawej gałęzi. Wówczas przechodzimy od analizowanego węzła do węzła nadrzędnego. Po wielokrotnym powtórzeniu przejść i obrotów rozgałęzione drzewo przyjmuje postać drzewa bez rozgałęzień, czyli listy liniowej. Kolejność płaszczyzn na liście jest wyznaczona przez numery ziaren. Oczywiście, numer pierwszego ziarna wyznaczającego płaszczyznę zawsze jest mniejszy od numeru ziarna drugiego. Pierwsza płaszczyzna na liście ma najmniejszy numer pierwszego ziarna (-6) i najmniejszy numer drugiego ziarna. Płaszczyzny nie mogą mieć dwóch ziaren o ujemnych numerach, ponieważ ujemny numer oznacza zewnętrzne obszary i taka płaszczyzna będzie poza przestrzenią automatów komórkowych.

5.1.6. Uporządkowanie ścian komórek na powierzchni styku dwóch ziaren, sprawdzenie ich ciągłości i wyznaczenie konturu każdej płaszczyzny

Nieciągłość powierzchni styku dwóch ziaren może istnieć w strukturze początkowej lub powstać w procesie rekrytalizacji. Algorytm sprawdzania nieciągłości powierzchni jest podobny do algorytmu sprawdzania ciągłości ziaren, dlatego zostanie opisany mniej szczegółowo. Wybierano w nim pierwszy element z listy ścian i przyporządkowano go do nowej listy. Następnie na starej liście poszukiwano wszystkich sąsiednich ścian i przypisywano je do nowej listy. Powtarzano tę procedurę do wyczerpania listy lub do momentu, kiedy ściany pozostałe na starej liście nie stykały się z żadną ścianą z nowej listy (oznacza to nieciągłość powierzchni styku dwóch ziaren).

Algorytm ten, w odróżnieniu od algorytmu sprawdzania ciągłości ziaren, został wzbogacony o czynności związane z wyznaczeniem konturu płaszczyzny (rys. 5.3). Przenoszenie ścian z jednej listy na drugą związane ze zmianą konturu. Już pierwsza ściana wyznacza kontur składający się z czterech jej boków (krawędzi komórki). Wszystkie cztery krawędzie zostają wpisane na listę krawędzi wyznaczających kontur płaszczyzny. Głównymi parametrami krawędzi są jej współrzędne określone jako współrzędne środka krawędzi. Wykorzystywano je również do poszukiwania ściany przylegającej do konturu. Aby ściana przylegała do konturu płaszczyzny i została dołączona do płaszczyzny, odległość ściany od krawędzi płaszczyzny powinna być równa połowie wymiaru komórki. Po przyłączeniu ściany do płaszczyzny sprawdzano wszystkie cztery jej boki pod kątem ich przynależności do konturu przed przyłączeniem. Jeżeli ten czy inny bok już należał do konturu, zostawał z niego usunięty, natomiast jeżeli nie należał, był do konturu dodawany. Powtarzano to względem wszystkich ścian we wszystkich płaszczyznach. Jako wynik dla każdej płaszczyzny uzyskano opis konturu określonego przez nieuporządkowaną listę współrzędnych krawędzi, z których ten kontur się składał.



Rys. 5.3. Dołączanie ściany do płaszczyzny i zmiana konturu
(Svyetlichnyy *et al.* 2011)

Wynikiem zastosowania omawianego algorytmu jest nieuporządkowana lista krawędzi, co jest związane z faktem, że większość elementów wpisywanych na tę listę w końcu i tak zostanie z niej usunięta. Dlatego wydaje się, że nakłady na porządkowanie na bieżąco dynamicznie zmieniającej się listy przewyższają wynikające z niego korzyści. W związku z tym nowe elementy dodawano na koniec listy i sprawdzanie również rozpoczynano od końca, co pozwoliło uzyskać pewne przyspieszenie działania algorytmu.

Algorytm uporządkowania krawędzi tworzących kontur płaszczyzny jest podobny do algorytmu uporządkowania ścian. Wybiera się pierwszą krawędź i wyznacza się koniec linii konturu. Następnie sprawdza się, czy pozostałe krawędzie przylegają do jednego z końców linii, a jeżeli tak, to zostaną one dołączone do odpowiedniego końca linii, ze zmianą współrzędnych tego końca. Powtarza się to do wyczerpania listy. Ze względu na sposób tworzenia konturu zawsze jest on ciągły i zamknięty. Wynikiem algorytmu jest uporządkowany kontur każdej płaszczyzny.

5.1.7. Stworzenie listy krawędzi (linii) dla każdej płaszczyzny

Na podstawie uporządkowanego konturu każdej płaszczyzny można stworzyć i wypełnić listy jej krawędzi (linii) oraz listy krawędzi komórek dla każdej krawędzi

płaszczyzny. Krawędzie komórek, które należą do konturu płaszczyzny, mają współrzędne, ale nie zawierają informacji o ziarnach, do których należą. Poczynając od pierwszego elementu listy, obchodzimy cały kontur, dochodząc do elementu ostatniego. Podczas przejścia od krawędzi do krawędzi sprawdzano wszystkie cztery komórki, do których ona należy, i wyznaczano ziarna, do których należą te komórki. W ten sposób wyznaczano trzy lub cztery ziarna, których krawędzie tworzą rozpatrywaną linię. Ziarna zapisywano według numeracji rosnącej. Pierwsza krawędź rozpoczyna pierwszą linię, kolejne sprawdzano pod względem ich przynależności do rozpatrywanej linii. Kolejna krawędź należy do tych samych ziaren co linia, zostaje ona dołączona do tej linii, w przeciwnym razie rozpoczyna ona nową linię. Po obejściu konturu zostaje wypełniona lista linii, pozostaje tylko sprawdzić, czy pierwsza i ostatnia linia powinny być połączone, ponieważ kontur rozpoczyna się w przypadkowym miejscu i prawdopodobnie nie będzie to styk dwóch linii.

5.1.8. Stworzenie listy wszystkich linii

Na podstawie listy linii (krawędzi) każdej płaszczyzny można stworzyć ogólną listę linii. Oczywiście, każda linia w każdej płaszczyźnie jest wyznaczona niezależnie, więc należy zebrać wszystkie linie na jednej liście, odrzucając te, które się powtarzają.

Lista linii uporządkowywana jest według następujących kryteriów:

- numer pierwszego, drugiego, trzeciego (i czwartego, jeśli jest obecne) ziarna,
- współrzędna x , y , z końców linii (odpowiednio minimalna i maksymalna wartość każdej współrzędnej).

Ostatnie kryterium jest konieczne do odróżnienia linii, które są krawędziami tych samych ziaren. Może to być ta sama linia rozdzielona na odcinki, na przykład powstaniem na granicy nowego ziarna, bądź też linie zupełnie różne. Podczas tworzenia listy linii każdej nadaje się numer. Tworzy się również listy płaszczyzn przylegających do linii. Wynikiem działania algorytmu jest wypełniona lista linii.

5.1.9. Algorytm sprawdzania linii

Uzyskana lista linii może być niespójna. Większość utworzonych na różnych płaszczyznach linii jest taka sama, jednak niektóre mogą być na jednej płaszczyźnie krótsze niż na innych. Wówczas na liście będzie zapisana i krótsza, i dłuższa linia. Przyczyną rozbieżności w długości linii jest sposób ich wyznaczania. Linia może pozostać ciągłą, będąc krawędzią tych samych ziaren, ale granicą różnych płaszczyzn (rys. 5.4) – na przykład kiedy jedno ziarno (ziarno 1) ma dwie powierzchnie (pow. 1 i 4) styku z innymi ziarnami, a te powierzchnie stykają się w jednym punkcie. W takim przypadku w każdej z tych dwóch płaszczyzn (pow. 1 i 4) będą się znajdowały odcinki linii, która w innych płaszczyznach (pow. 3) będzie wyznaczona jako jedna linia ciągła. Właśnie wyeliminowaniu takiego przypadku służy omawiany algorytm. Główną oznaką zaistnienia takiej sytuacji jest liczba płaszczyzn mniejsza lub równa dwóm.

Pozostaje jeszcze znaleźć wszystkie linie, które mają wspólny odcinek, ale nie są tożsame. Linia dłuższa zostanie skrócona o krótszą. Dalej, jeżeli skróconej linii jest równa inna linia już istniejąca na liście, jedna linia z tej pary zostaje odrzucona. Po wykonaniu takiej operacji lista linii będzie spójna.



Rys. 5.4. Podział linii na dwie krótsze (Svyetlichnyy *et al.* 2011)

Algorytm ma postać głównej pętli, w której powtarza się czynności względem każdej linii z globalnej listy.

Informacja dotycząca każdej linii przed wykonaniem algorytmu zawiera nieuporządkowaną listę odcinków. Każdy odcinek jest krawędzią komórki, ma jednostkową długość, a jego położenie wyznaczono, podając współrzędne jego środka. Ponieważ współrzędne środka komórki są liczbami całkowitymi, tylko jedna ze współrzędnych odcinka będzie liczbą całkowitą, pozostałe dwie będą różnić się od całkowitej o 0,5. Końce odcinków odpowiadają wierzchołkom komórki i żadna z ich trzech współrzędnych nie będzie liczbą całkowitą – będą miały cyfrę pięć po przecinku. Ciągłość linii sprawdzano, opierając się na założeniu, że jeżeli dwa odcinki mają jeden wspólny koniec, to należą do jednej ciągłej linii. Natomiast jeżeli koniec danego odcinka nie jest wspólny z końcem żadnego innego odcinka, jest on końcem linii i odpowiednio wierzchołkiem ziarna.

Algorytm rozpoczyna się od wyboru pierwszego odcinka z listy odcinków wybranej linii i wyznaczenia jego końców. Sprawdza się również, czy nie jest to jedyny odcinek należący do tej linii. Jeżeli linia składa się z jednego odcinka, jego wyznaczone końce będą końcami linii i zostaną dodane do listy wierzchołków, o ile nie były na niej zapisane wcześniej. Jeżeli linia zawiera więcej niż jeden odcinek, pierwszy odcinek zostaje przeniesiony z bieżącej linii do linii tymczasowej, a jego końce zostają tymczasowymi końcami linii.

W drugim etapie pętli algorytmu sprawdza się po kolei odcinki pozostałe w rozpatrywanej linii i jeżeli koniec kolejnego odcinka jest tymczasowym końcem tymczasowej linii, zostaje on przeniesiony z listy rozpatrywanej linii do listy linii tymczasowej. Ten odcinek jest dodawany do początku lub końca listy z odpowiednią zmianą współrzędnych jednego z końców linii. Czynności te powtarza się do wyczerpania listy odcinków rozpatrywanej linii lub niespełnienia warunku ciągłości względem wszystkich

pozostałych na liście odcinków. W pierwszym przypadku rozpatrywana linia była ciągłą, więc tymczasowa linia będzie jedna, z uporządkowanymi odcinkami i wyznaczonymi końcami (wierzchołkami), które będą dodane do listy wierzchołków, jeżeli nie były zapisane w niej wcześniej. Uporządkowaną linię tymczasową zastępujemy linią rozpatrywaną i przechodzimy do kolejnej linii. Natomiast w drugim przypadku linia rozpatrywana nie była ciągła i dzieli się przynajmniej na dwie linie. Jedną z nich jest linia tymczasowa, już uporządkowana, która zostaje dodana do listy linii przed linią aktualnie rozpatrywaną. Z końcami linii postępujemy w ten sam sposób jak poprzednio. Natomiast analizowana lista linii pozostaje z nieuporządkowanymi odcinkami, dlatego wracamy do niej i rozpoczynamy procedurę od początku, od wyznaczenia pierwszego odcinka.

5.1.10. Stworzenie i wypełnienie listy linii dla każdego ziarna

W tej części programu po kolei przeglądano wszystkie listy linii. Zawierają one informację o ziarnach; do wszystkich tych ziaren dopisywano numery linii, tworząc odpowiednie listy dynamiczne.

5.1.11. Stworzenie i wypełnienie listy wierzchołków

Do tej pory jedynym niewyznaczonym elementem topologii mikrostruktury pozostaje wierzchołek. Nic nie stoi na przeszkodzie, aby wyznaczyć go podczas dzielenia konturu płaszczyzny na linie, jednak kilka kolejnych algorytmów podziału konturu na linie te modyfikuje, więc należałoby modyfikować również i informacje o wierzchołkach. Dlatego zdecydowano rozpocząć tworzenie listy wierzchołków dopiero wtedy, gdy wszystkie informacje o ziarnach, płaszczyznach i liniach zostały ukształtowane i nie będą już podlegały dalszym modyfikacjom.

Wierzchołki wyznaczano po kolei dla każdej linii, na podstawie współrzędnych pierwszej i ostatniej z krawędzi znajdujących się na uporządkowanej liście każdej linii. Na podstawie współrzędnych krawędzi jednoznacznie można wyznaczyć dwa jej końce. Aby wybrać wierzchołek z tych dwóch końców krawędzi, porównywano ich odległości od sąsiedniej krawędzi (drugiej lub przedostatniej). Koniec krawędzi znajdujący się dalej od sąsiedniej krawędzi jest wierzchołkiem ziarna i końcem linii. Jeżeli linia zawiera tylko jedną krawędź, obydwa końce tej krawędzi automatycznie zostają wierzchołkami.

Po wyznaczeniu wierzchołków sprawdzano ich obecność na liście wierzchołków. Lista wierzchołków była uporządkowana według współrzędnych x , y i z . W razie obecności wierzchołka na liście dodawano do niego informację o rozpatrywanej linii, natomiast do informacji o linii dodawano numer wierzchołka z listy. W razie braku wierzchołka na liście był on dodawany do tej listy w odpowiednim miejscu, nadawano mu kolejny numer oraz dodawano informacje o linii, a do informacji o linii dodawano dane o wierzchołku.

Wynikiem działania tego algorytmu jest uporządkowana według współrzędnych lista wierzchołków, z których każdy zawiera informację tylko o liniach, oraz uzupełniona informacja dotycząca linii, która teraz zawiera pełne dane dotyczące do związanych z nią ziaren, płaszczyzn i wierzchołków. Uzupełnieniu informacji o ziarnach, płaszczyznach i wierzchołkach służy kolejny algorytm.

5.1.12. Stworzenie i wypełnienie listy wierzchołków dla każdej płaszczyzny i ziarna oraz list płaszczyzn i ziaren dla każdego wierzchołka

Ogniwem, które łączy informację o ziarnach i płaszczyznach z informacją o wierzchołkach jest linia, która w tej chwili zawiera już pełną informację o pozostałych elementach.

Rozpatrywano po kolei wszystkie wierzchołki, w odniesieniu do każdego z nich przeglądano listę linii. Informacja o każdej linii wskazuje na trzy lub cztery płaszczyzny i tyle samo ziaren, zatem informacja o każdej z tych płaszczyzn i każdym ziarnie jest dodawana do informacji o odpowiednim wierzchołku. Przed dodaniem do tej listy płaszczyzn i ziaren sprawdzano ich obecność na liście, ponieważ każda płaszczyzna będzie wyznaczona dwukrotnie, a ziarno przynajmniej trzykrotnie. Jednocześnie informacja o wierzchołku jest dodawana do informacji o odpowiednim ziarnie i płaszczyźnie.

5.1.13. Zapisywanie wyników do plików

Informacja zapisywana jest w plikach w dwóch wersjach: pełnej i skróconej. Tworzono oddzielne pliki dla ziaren, płaszczyzn, linii i wierzchołków.

Pełna wersja pliku dla ziaren zawiera następujące informacje:

- liczbę ziaren;
- numer ziarna;
- liczbę komórek w ziarnie;
- liczbę płaszczyzn;
- numer każdej płaszczyzny, numery ziaren, do których należą, i liczbę ścian;
- liczbę linii;
- numer każdej linii, numery ziaren, dla których ona jest krawędzią, i liczbę odcinków;
- liczbę wierzchołków;
- dla każdego wierzchołka jego numer, współrzędne, liczbę ziaren i ich numery.

Skrócona wersja pliku dla ziaren zawiera informacje następujące:

- liczbę ziaren;
- numer ziarna;
- liczbę komórek w ziarnie;
- liczbę płaszczyzn i ich numery;
- liczbę linii i ich numery;
- liczbę wierzchołków i ich numery.

Pełna wersja pliku dla płaszczyzn zawiera następujące informacje:

- liczbę płaszczyzn;
- numer płaszczyzny;
- numery ziaren;
- liczbę linii i ich numery;
- liczbę wierzchołków i ich numery;
- liczbę ścian i ich współrzędne.

Skrócona wersja pliku dla płaszczyzn zawiera następujące informacje:

- liczbę płaszczyzn;
- numer płaszczyzny;
- numery ziaren;
- liczbę ścian i ich współrzędne.

Pełna wersja pliku dla linii zawiera następujące informacje:

- liczbę linii;
- numer linii;
- numery ziaren;
- liczbę płaszczyzn i ich numery;
- liczbę wierzchołków i ich numery;
- liczbę krawędzi i ich współrzędne.

Skrócona wersja pliku dla linii zawiera informacje następujące:

- liczbę linii;
- numer linii;
- numery wierzchołków;
- liczbę płaszczyzn i ich numery;
- liczbę krawędzi i ich współrzędne.

Pełna wersja pliku dla wierzchołków zawiera następujące informacje:

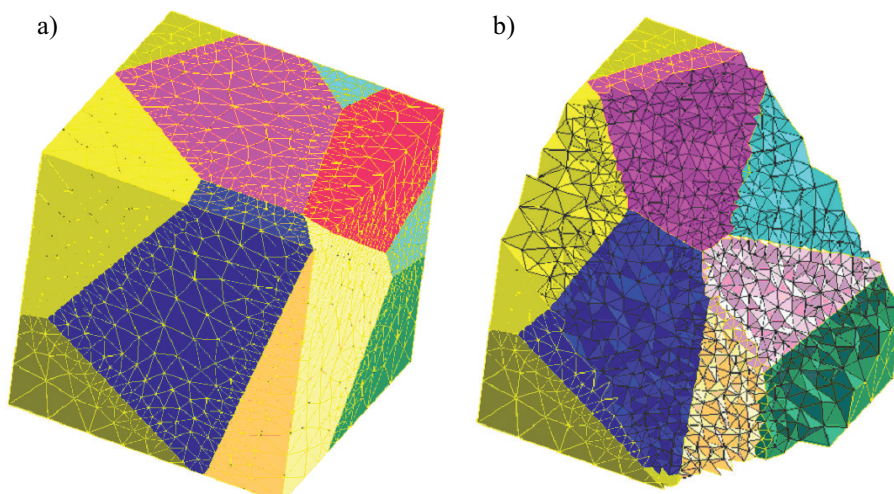
- liczbę wierzchołków;
- numer wierzchołka;
- współrzędne wierzchołka;
- numery ziaren;
- liczbę płaszczyzn i ich numery;
- liczbę linii i ich numery.

Skrócona wersja pliku dla wierzchołków zawiera informacje następujące:

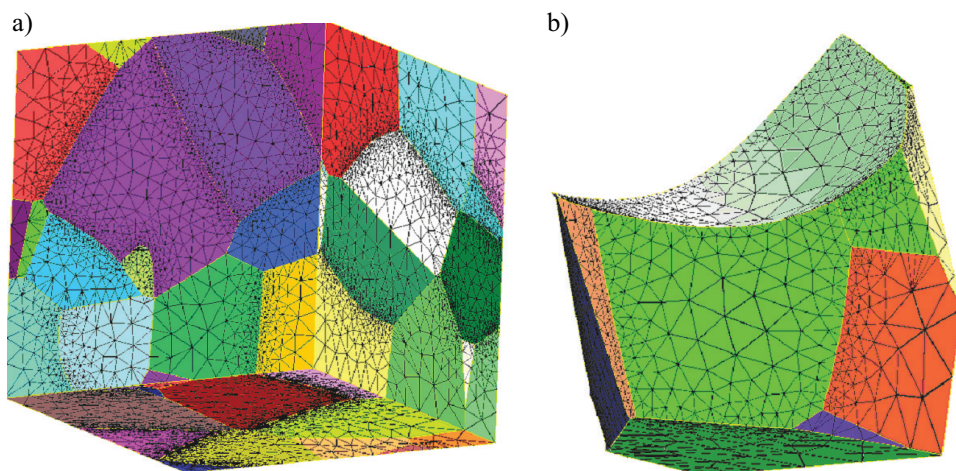
- liczbę wierzchołków;
- numer wierzchołka i jego współrzędne.

5.2. Przykłady siatek MES

Na podstawie danych zapisywanych w odpowiednich plikach Jurczyk (Svyetlichnyy *et al.* 2011) opracował algorytm tworzenia siatek MES. Niektóre z nich pokazano na rysunkach 5.5 i 5.6. Są to nierównomierne siatki zagęszczone w pobliżu granic ziaren. Granice ziaren mogą być zarówno płaskie (rys. 5.5a), jak i wygięte (rys. 5.5b). Więcej szczegółów na ten temat – por. Svyetlichnyy *et al.* (2011).



Rys. 5.5. Siatka objętościowa na powierzchni przestrzeni (a) i na jego przekroju (b) (Svyetlichnyy *et al.* 2011, oprac. Tomasz Jurczyk)



Rys. 5.6. Siatka objętościowa na powierzchni przestrzeni (a) i jednego ziarna (b) (Svyetlichnyy *et al.* 2011, oprac. Tomasz Jurczyk)

Część druga

Modelowanie zjawisk mikrostrukturalnych

Pierwsze rozdziały części drugiej niniejszej pracy zostały poświęcone zasadom i przykładom symulacji zjawisk mikrostrukturalnych. Niektóre z nich omówiono bardziej szczegółowo, inne jedynie pobieżnie. Jako pierwsze zagadnienie w rozdziale szóstym rozpatrzono stworzenie początkowej mikrostruktury, która często jest wykorzystywana jako stan wyjściowy do modelowania innych zjawisk. W rozdziale siódmym zostały zaprezentowane zasady modelowania krzepnięcia w maszynie ciągłego odlewania stali. W ósmym rozdziale omówiono zjawisko rekrytalizacji. Rozdział dziewiąty poświęcono przemianie fazowej, a w rozdziale dziesiątym, dotyczącym zjawisk mikrostrukturalnych, zademonstrowano możliwości automatów w zakresie modelowania rozdrobnienia ziarna oraz wyniki symulacji procesu walcowania pakietowego i wielokrotnego ściskania MAXStrain®.

Część druga książki przedstawia również kolejny, trzeci poziom układu (rys. 1.1) – modele ewolucji mikrostruktury w procesach technologicznych. Zagadnienia te są częściowo rozpatrywane w rozdziałach ósmym i dziesiątym oraz w rozdziale ostatnim, jedenastym, niniejszego opracowania. W rozdziale jedenastym przedstawiono też przykłady modelowania procesu walcowania w wykrojach oraz walcowania wyrobów płaskich.

6. Modelowanie początkowej mikrostruktury o zadanych parametrach

Zachodzące w materiałach procesy mikrostrukturalne ze względu na stan początkowy można podzielić na dwie grupy. Do pierwszej można zaliczyć procesy, które nie potrzebują początkowej mikrostruktury i rozpoczynają się od jednolitego pierwotnego stanu. Jednym z takich procesów jest krystalizacja. Procesy zaliczane do drugiej grupy potrzebują początkowej mikrostruktury. Mikrostruktura ta jest opisywana różnymi parametrami i modelowanie dowolnego procesu powinno się rozpoczynać od początkowej mikrostruktury najbardziej zbliżonej do rzeczywistej w określonych warunkach. Zasady, algorytm i oprogramowanie do stworzenia mikrostruktury o zadanych parametrach zostały omówione poniżej.

6.1. Zasady i struktura modelu kształtowania początkowej mikrostruktury

Kształtowanie początkowej mikrostruktury za pomocą automatów komórkowych można zrealizować, stosując opisany w rozdziale 4 uniwersalny frontalny automat komórkowy. Wówczas proces kształtowania można podzielić na dwa etapy: zarodkowanie i rozrost ziaren. Rozrost ziaren został szczegółowo opisany w części pierwszej niniejszego opracowania, natomiast proces zarodkowania zostanie przedstawiony poniżej.

Zgodnie z zasadami przedstawionymi w podrozdziale 1.4 należy określić podstawowe cechy automatów komórkowych, uwzględniając wymagania modelowania procesu – kształtowania początkowej mikrostruktury.

Geometria. Ponieważ początkowa mikrostruktura jest kształtowana głównie w celu wykorzystania w dalszych symulacjach, parametry geometrii są narzucone przez proces, na potrzeby którego jest przygotowywana. Przestrzeń jest trójwymiarowa. Rozmiary i kształt przestrzeni podczas symulacji nie zmienia się. Siatka jest skończona i przeważnie regularna.

Warunki brzegowe. Warunki brzegowe również powinny uwzględnić dalszą symulację. Można więc zastosować warunki okresowe z przesunięciem lub bez, kombinowane lub otwarte. Przy wyborze warunków można się stosować do zaleceń przedstawionych w rozdziale 3.

Stany komórek. Automat komórki w tym algorytmie może przyjmować pięć stanów: stan początkowy q_0 , „komórka na froncie zmian” (stan frontalny) q_1 , „komórka na granicy ziarna” q_2 , „komórka wewnątrz ziarna” q_3 oraz stan przejściowy q_4 .

Lokalna reguła. Lokalna reguła opiera się wyłącznie na stanach komórek, bez uwzględniania parametrów procesowych. Reguła ta wykorzystuje głównie otoczenie wyjściowe i działa jako frontalny automat komórkowy. Uniwersalny frontalny automat komórkowy opisano w rozdziale 4 i przedstawiono na rysunku 4.9.

Otoczenie. W przedstawionych przykładach stosowano dwa rodzaje otoczenia. Podczas rozrostu ziaren, do których należą komórki w stanie frontalnym, o tym, które komórki będą włączone w proces rozrostu, decyduje otoczenie Moore’a, natomiast po przejściu ruchomej granicy przez komórkę rosnącego ziarna do wyznaczenia, czy komórka znajduje się wewnątrz ziarna, czy na jego granicy, stosowano otoczenie von Neumanna.

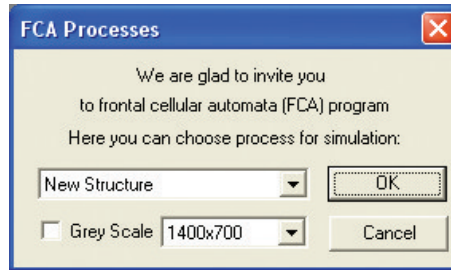
Stan początkowy. Kształtowanie początkowej mikrostruktury rozpoczyna się od stanu niezakłóconego, kiedy wszystkie komórki znajdują się w tym samym stanie początkowym.

Aktualizacja stanów. Ze względu na cechy formalne uniwersalne frontalne automaty komórkowe można zaliczyć do automatów z asynchroniczną aktualizacją stanów. Aktualizacja stanu następuje od razu po spełnieniu odpowiedniego warunku. Stan ten jest dostępny dla wszystkich komórek i może być przez nie wykorzystywany. Jednak warunki, które prowadzą do aktualizacji, są ściśle związane z czasem, krokiem obliczeniowym i kolejnością obliczeń. Długość kroku czasowego dobierano automatycznie. Prędkość rozrostu jest stała, jednostkowa i prowadzi do formalnie nierzeczywistego czasu.

Warunek zatrzymania. Kryterium końca obliczeń jest przejście wszystkich komórek w stan końcowy (q_2 lub q_3).

6.2. Wybór warunków symulacji i symulacja

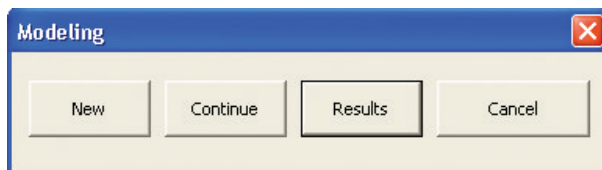
Zgodnie ze schematem przedstawionym na rysunku 1.1 opracowano program, który pozwala wybrać do symulacji proces lub zjawisko mikrostrukturalne. Jeden z możliwych do wybrania procesów to kształtowanie początkowej struktury o określonych parametrach. Po uruchomieniu oprogramowania pojawia się okno wyboru procesu modelowego, co pokazano na rysunku 6.1. W okienku tym można wybrać z listy kształtowanie początkowej mikrostruktury („New Structure”). Dodatkowo można wybrać rozdzielczość ekranu (na przykład 1400×700 pikseli), na którym będzie wizualizowana symulacja oraz zostaną przedstawione wyniki końcowe. Poza tym można wybrać wizualizację wielobarwną (domyślnie) lub utrzymaną w szarym kolorze.



Rys. 6.1. Wybór procesu i parametrów wizualizacji

Po określeniu parametrów wizualizacji i wyborze „New Structure” pojawia się okienko „Modeling” (rys. 6.2), w którym są dostępne do wyboru trzy opcje: kształtowanie nowej struktury („New”), kontynuacja już rozpoczętego procesu („Continue”) lub wizualizacja gotowej mikrostruktury („Results”).

Wybór przycisku „Continue” pozwala kontynuować dobór rozkładu wielkości ziaren. Warunkiem poprawnego działania programu jest prawidłowe przeprowadzenie poprzednich obliczeń i zachowanie ich wyników w odpowiednim pliku. Warunki poprzedniej symulacji zachowywano w pliku „ParametersIMS.txt” i domyślnym pliku „GrainsD0.txt”. Po zapisaniu tych plików w odpowiednich formatach proces symulacji rozpoczyna się po naciśnięciu przycisku „Continue”.

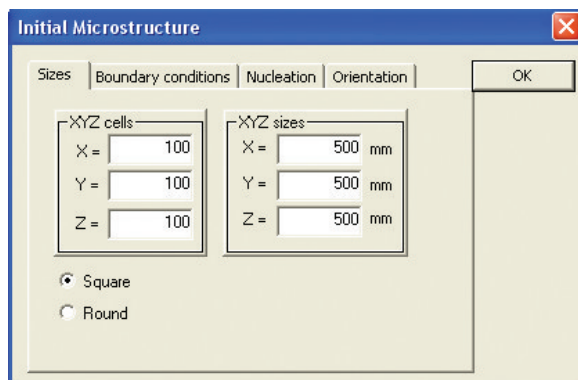


Rys. 6.2. Okno „Modeling”

Wybór opcji wizualizacji (przycisk „Results”) pozwala przejść do wyświetlenia wyników symulacji, czyli do uzyskanej mikrostruktury. Po naciśnięciu tego przycisku pojawia się opcja wyboru pliku z wynikami (domyślnie „Structure0.txt”).

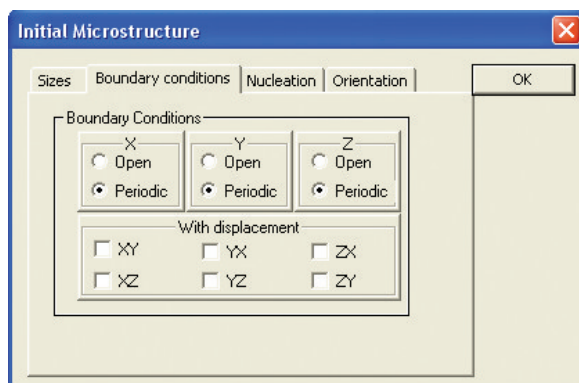
Natomiast najbardziej istotna jest opcja nowego modelowania. Przycisk „New” pozwala przejść do wyboru parametrów modelowania początkowej mikrostruktury i otwiera okno „Initial Microstructure”, w którym pojawiają się cztery zakładki.

Za pomocą pierwszej zakładki, „Sizes” (rys. 6.3), zadaje się geometrię automatów komórkowych. W niej należy określić wymiary przestrzeni trójwymiarowej – zarówno liczbę komórek wzdłuż każdej osi, jak i jej wymiary rzeczywiste. Dodatkowo można wybrać kształt modelowej przestrzeni. Do wyboru są dwie opcje kształtu określające przekrój YZ: prostokąt lub kwadrat („Square”) i elipsa lub koło („Round”). Wybór jednej z tych opcji skutkuje uzyskaniem przestrzeni o kształcie prostopadłościanu lub walca (w tym eliptycznego).



Rys. 6.3. Parametry przestrzeni modelowej

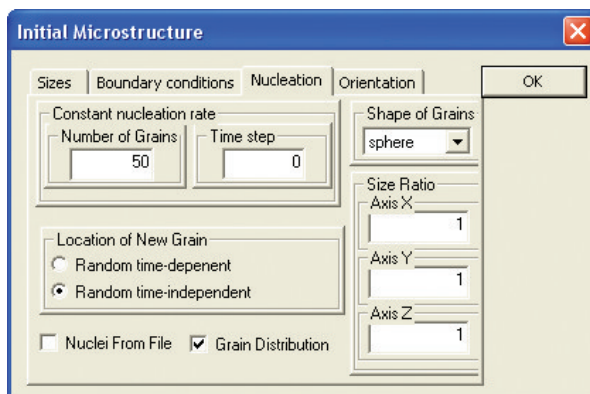
Zakładka druga, „Boundary conditions” (rys. 6.4), określa warunki brzegowe. Dla każdego kierunku można niezależnie wybrać warunki otwarte („Open”) lub periodyczne („Periodic”, domyślne), a faktycznie kombinowane warunki brzegowe. Wybór warunków periodycznych umożliwia określenie przesunięć przy łączeniu przeciwległych granic, które wówczas będą równe połowie odpowiedniego wymiaru przestrzeni. Można wybrać niezależnie dowolną kombinację sześciu możliwych przesunięć. W przypadku wyboru przestrzeni o kształcie walca dla kierunków *Y* i *Z* warunki brzegowe zostały narzucone jako półotwarte.



Rys. 6.4. Wybór warunków brzegowych

Zakładka trzecia, „Nucleation” (rys. 6.5), umożliwia wyznaczenie warunków zarodkowania oraz pozwala dobrać rozkład wielkości ziaren. Najpierw można określić kształt rosnących ziaren („Shape of Grains”), wybierając go z listy, a następnie – zaznaczyć opcję „Grain Distribution”, która określa, czy rozkład wielkości ziaren będzie liczony i zapisywany w pliku. Obliczanie rozkładu wielkości ziaren również umożliwia powtórzenie obliczeń w celu dopasowania uzyskanego rozkładu do zadanego. Zazna-

czenie tej opcji może także oznaczać dobór i optymalizację czasów zarodkowania w celu uzyskania zadanego rozkładu wielkości ziaren. Aby program działał poprawnie, w tym przypadku niezbędne są plik z teoretycznym rozkładem („TeorDistr.txt”) oraz zgodność liczby ziaren w teoretycznym rozkładzie i zadanym w polu „Number of Grains” (przy zerowej wartości w polu „Time step”). Niezgodność teoretycznego rozkładu z parametrami przestrzeni nie powoduje błędu działania programu, ale uniemożliwia dokładne dopasowanie rozkładu. Funkcja ta została bardziej szczegółowo opisana w kolejnym podrozdziale.



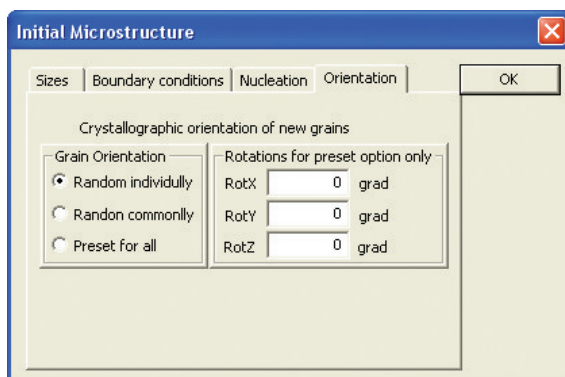
Rys. 6.5. Parametry zarodkowania

Istnieją dwa warianty określenia warunków zarodkowania. Pierwszy polega na określaniu ich na podstawie danych odczytanych z pliku. W odpowiednim pliku zapisane są wszystkie informacje o zarodkowaniu, czyli o czasie i miejscu powstania każdego zarodka (oraz jego przestrzennej orientacji). Informacje te są uporządkowane według czasu powstania zarodka.

Drugim wariantem jest zadanie parametrów zaczerpniętych bezpośrednio z opisywanej zakładki. Zarodkowanie przewidziano ze stałą prędkością względną (w stosunku do objętości przestrzeni wolnej od ziaren), która jest zadawana poprzez dwa parametry: „Number of Grains” (liczbę nowych ziaren) oraz „Time step” (odstęp czasu pomiędzy pojawieniem się kolejnej grupy zarodków). Równa zero wartość odstępu czasu oznacza, iż zarodkowanie następuje jedynie przed procesem rozrostu ziaren, natomiast wartość różna od zera oznacza, że zarodkowanie powtarza się w wyznaczonych odstępach czasu (jeżeli zarodek pojawia się w miejscu zajęтым przez ziarno, zostaje on odrzucony). Miejsce zarodkowania wybierano za pomocą liczb pseudolosowych w bloku „Location of New Grain” i możliwe są tu dwie opcje: zarodkowanie wyznaczone przez niezmienną kolejność liczb pseudolosowych („Random time-independent”, domyślnie) i uzależnienie punktu startowego od wewnętrznego zegarka systemowego komputera („Random time-dependent”). W pierwszym wariantie zarodkowanie będzie się odbywać w tym samym miejscu w tej samej chwili. W przypadku wyboru zarodkowania z pliku i wyboru

opcji „Random time-independent” proces odbywa się z pominięciem miejsc zarodkowania odczytywanych z pliku, ale z uwzględnieniem czasu powstania zarodków.

Ostatnia zakładka, „Orientation” (rys. 6.6), określa orientację rosnących ziaren.



Rys. 6.6. Parametry orientacji

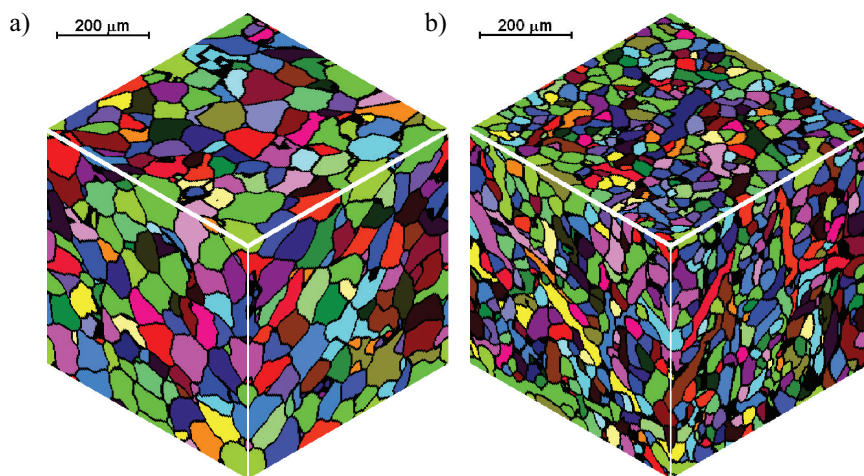
Można rozróżnić dwa rodzaje orientacji: przestrzenną orientację ziarna różnego od sferycznego i orientację krystalograficzną. W przypadku modelowania początkowej mikrostruktury te dwie orientacje łączy się ze sobą, czyli przestrzenna orientacja ziarna jest taka sama jak orientacja krystalograficzna. Orientacja jest zadawana jako obroty ziarna wokół trzech osi, w ten sposób można określić związek między globalnym i lokalnym układem współrzędnych (por. podrozdz. 2.5.1). Istnieją trzy opcje orientacji: orientacja każdego ziarna jest wybierana losowo („Random individually”) i orientacja wszystkich ziaren jest taka sama, ale wybierana albo losowo („Random commonly”), albo narzucona („Preset for all”). Jeżeli orientacja jest narzucona, w odpowiednie miejsca należy wpisać trzy kąty obrotów (w stopniach). Domyślnie wszystkie kąty obrotu ustawiono jako równe zeru. Jeżeli dane dotyczące zarodkowania są wczytywane z pliku wsadowego, orientacja również jest narzucona. W efekcie w przypadku ziaren sferycznych orientacja nie wpływa na końcowy wynik, natomiast w przypadku ziaren o pozostałych kształtach uzyskana mikrostruktura zależy od tego, czy ziarna są ułożone w sposób uporządkowany czy przypadkowo. Po uzyskaniu mikrostruktury orientacja krystalograficzna może być zmieniona innym podprogramem, który może wprowadzić zadany rozkład orientacji ziaren lub rozkład kątów dezorientacji granic ziaren.

Po wprowadzeniu danych (lub wykorzystaniu wartości domyślnych) można uruchomić symulację, a jej wynikiem będzie mikrostruktura, której przykłady można znaleźć w rozdziale 2 (rys. 2.20 i 2.22).

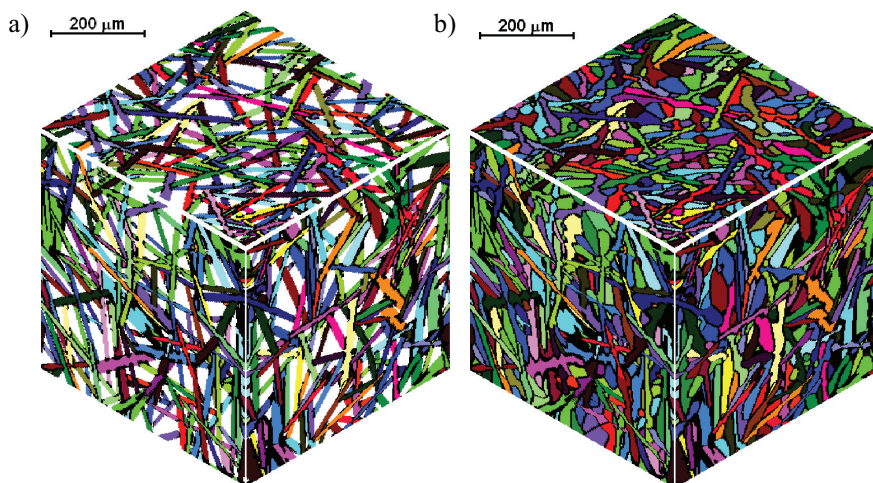
Kilka innych przykładów pokazano na rysunkach 6.7 i 6.8. Orientację we wszystkich wariantach wybierano losowo, liczba ziaren jest równa 500.

Na rysunku 6.7a uwidoczniło końcową mikrostrukturę w przypadku, kiedy kształt rosnących ziaren był ośmiościenny. Ziarna w kształcie wydłużonego walca ze stosunkiem długości do średnicy 40:1 przedstawiono na rysunku 6.7b. Tylko niektóre ziarna okazały się rozmieszczone i zorientowane tak, że oś walca leżała blisko powierzchni

przestrzeni i prawie do niej równoległe, co uwidoczniło ich wydłużony kształt. Pozostałe ziarna mają taki sam kształt, jednak nie jest to widoczne na rysunku, ponieważ ich osie przecinają brzegi przestrzeni pod dużym kątem.



Rys. 6.7. Wynikowa mikrostruktura uzyskana przy rozroście ziaren w kształcie ośmiościanu (a) i wydłużonych walców (b)

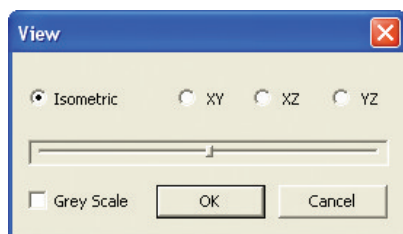


Rys. 6.8. Mikrostruktura przy rozroście ziaren w kształcie kwadratowych płytek (a) i okrągłych dysków (b)

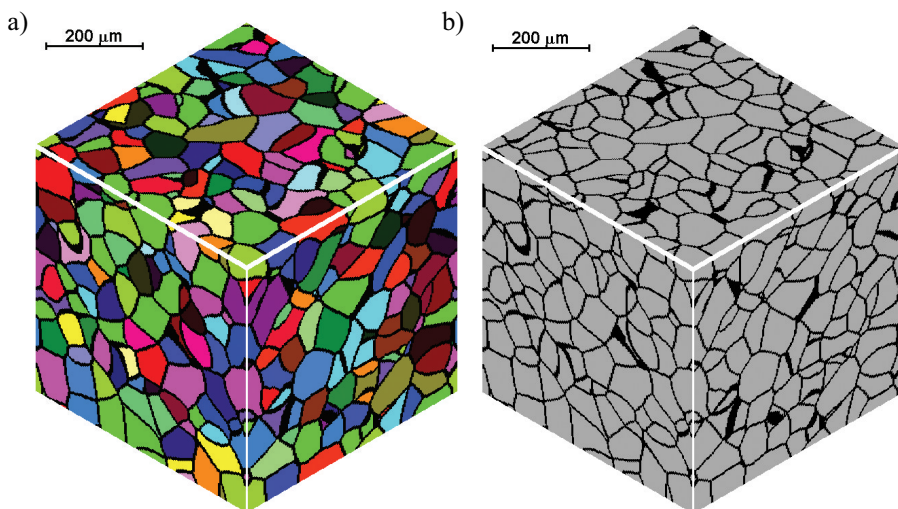
Na rysunku 6.8a jest pokazana mikrostruktura w trakcie kształtowania, gdy rosące ziarna mają kształt kwadratowych płytek o zmiennej grubości. Przekrój takiej płytki był elipsą o stosunku półosi 50:1. Na powierzchni sześcianu w przestrzeni automatów

komórkowych przeważnie widać różne przekroje tych płytek. Na rysunku 6.8b przedstawiono końcową mikrostrukturę złożoną z ziaren o kształcie okrągłych dysków ze stosunkiem ich grubości do średnicy równym 1:25. Końcowa mikrostruktura w dwóch ostatnich wariantach jest zbliżona.

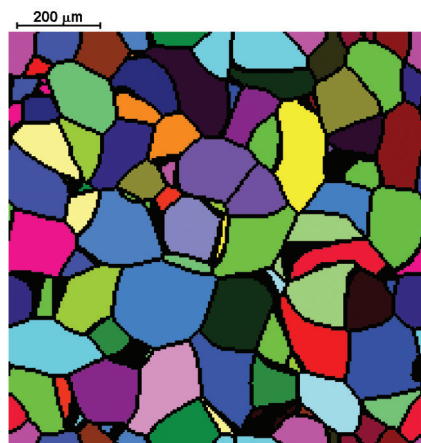
Po zakończeniu obliczeń pojawia się kolejne okienko, pokazane na rysunku 6.9, które pozwala przedstawić uzyskaną mikrostrukturę w różny sposób. Wybierając widok izometryczny, otrzymamy wynik w tej samej postaci jak podczas modelowania. Zaznaczając odpowiednie pole („Grey Scale”), podobnie jak przed symulacją, można przedstawić wyniki przy użyciu kolorów lub wyszarzone. Przykład struktury z różną skalą kolorową pokazano na rysunku 6.10. Oprócz widoku izometrycznego można zobaczyć dowolny przekrój w trzech ortogonalnych płaszczyznach XY , XZ i YZ . W tym celu należy zaznaczyć odpowiednią opcję na pasku wyboru, wybrać pozycję przekroju i nacisnąć przycisk „OK”. Przykład mikrostruktury w przekroju XY w środku przestrzeni przedstawiono na rysunku 6.11. Każdy wariant wizualizacji może być zachowany w pliku jako mapa bitowa. Po rozpatrzeniu wybranych wariantów należy nacisnąć „Cancel”, aby przejść do kolejnego etapu.



Rys. 6.9. Wybór wariantu wizualizacji wyników symulacji początkowej mikrostruktury

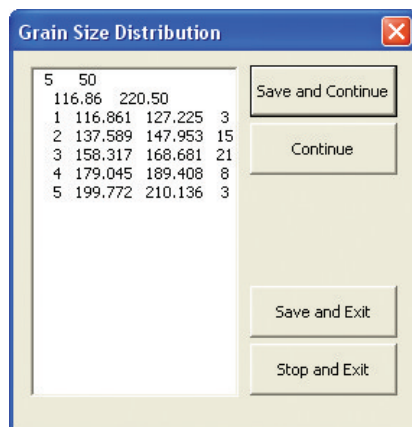


Rys. 6.10. Wizualizacja mikrostruktury w kolorze (a) i wyszarzonej (b)



Rys. 6.11. Mikrostruktura w jednym przekroju XY

Jeżeli podczas wprowadzania danych zaznaczono pole „Distribution” w zakładce „Nucleation”, pojawi się okno „Grain Size Distribution” (rys. 6.12), które przedstawia rozkład wielkości ziaren w uzyskanej mikrostrukturze. W pierwszym wierszu jest wypisana liczba klas w szeregu rozdzielczym oraz liczba ziaren. Liczba klas jest wybierana na podstawie liczby ziaren lub, gdy celem modelowania jest uzyskanie wymaganego rozkładu wielkości ziaren, narzucona. W kolejnym wierszu jest podana wielkość najmniejszego i największego ziarna. Dalej zostały zapisane wszystkie klasy: numer, dolna granica, środek i liczebność. Istnieją cztery warianty dalszego działania programu. Można zachować wyniki obliczeń rozkładu lub je pominąć. Można powtórzyć obliczenia w celu uzyskania zadanego rozkładu (wówczas powinno się skorzystać z pliku z zadanym rozkładem) lub skończyć symulację.



Rys. 6.12. Rozkład wielkości ziarna w uzyskanej mikrostrukturze

6.3. Uzyskanie mikrostruktury o zadanym rozkładzie wielkości ziaren

Do uzyskania wymaganego rozkładu wielkości ziaren można zastosować dwa podejścia. Pierwszym jest korekta istniejącej struktury, która będzie opisana w podrozdziale 6.3.6, drugim – modelowanie nowej struktury z dopasowaniem prędkości zarodkowania i rozrostu ziaren, które omówiono poniżej (Svyetlichnyy i Łach 2012).

Algorytm kształtowania struktury o zadanym rozkładzie wielkości ziaren składa się z następujących etapów:

1. Przedstawienie rozkładu teoretycznego wielkości ziaren za pomocą szeregu rozdzielczego.
2. Wyznaczenie liczby ziaren, reprezentatywnej objętości modelowej i sprawdzenie rozmiarów przestrzeni komórkowej.
3. Zadanie warunków zarodkowania.
4. Modelowanie mikrostruktury i wyznaczenie rozkładu empirycznego dla uzyskanej mikrostruktury.
5. Porównanie rozkładu empirycznego z teoretycznym, wyznaczenie błędu dopasowania.
6. W razie niespełnienia kryterium dopasowania – korekta warunków zarodkowania i powrót do punktu 3.

Spełnienie warunku dopasowania oznacza uzyskanie cyfrowej reprezentacji mikrostruktury o zadanym rozkładzie wielkości ziaren oraz określenie warunków zarodkowania. Poszczególne etapy algorytmu zostały opisane w kolejnych podrozdziałach. Algorytm korekty istniejącej struktury zawiera tylko pierwsze trzy etapy podanego wyżej algorytmu.

6.3.1. Przedstawienie rozkładu teoretycznego w postaci szeregu rozdzielczego

Głównym celem opisywanych tu działań jest uzyskanie zadanego rozkładu, przy czym danymi wejściowymi może być rozkład teoretyczny wyrażony w postaci równania lub za pomocą parametrów rozkładu, a uzyskane wyniki zawsze przedstawiane są w postaci rozkładu empirycznego. Należy zatem przedstawić te dwa rozkłady w postaci nadającej się do porównania. Rozkład empiryczny ma często postać szeregu rozdzielczego. Wówczas ciągły teoretyczny rozkład gęstości prawdopodobieństwa należy zastąpić szeregiem rozdzielczym z zadaną liczbą klas i wyznaczeniem prawdopodobieństwa występowania każdej klasy.

Najpierw należy wyznaczyć zakres. W przypadku normalnego rozkładu jest on równy trzem odchyleniom standardowym (3σ), odpowiada wartości gęstości prawdopodobieństwa $p \approx 0,46\%$, a poza obszarem $\xi_0 \pm 3\sigma$ prawdopodobieństwo $p \approx 0,27\%$.

Na podstawie tych danych dla rozkładów innych niż normalny do wyznaczenia najmniejszej d_{\min} i największej wartości d_{\max} określających zakres wielkości ziarna wybrano wartość gęstości prawdopodobieństwa $p = 0,5\%$.

Po wyznaczeniu zakresu w zależności od liczby klas k szeregu rozdzielczego (maksymalnie 15 klas) wyznaczano szerokość klasy: $\Delta d = \frac{d_{\max} - d_{\min}}{k}$; granice klas, czyli minimalną wartość dla każdej i -tej klasy: $d_{i\min} = d_{\min} + \Delta d (i - 1)$, środki klas: $\bar{d}_i = d_{i\min} + 0,5\Delta d$ oraz prawdopodobieństwo występowania każdej klasy p_i . Prawdopodobieństwo obliczano przez całkowanie numeryczne metodą parabol (wzór Simpsona) dla każdej klasy:

$$p_i = \frac{\Delta d}{6} \left[p(d_{i\min}) + 4p(\bar{d}_i) + p(d_{i\max}) \right] \quad (6.1)$$

Po wyznaczeniu prawdopodobieństw normalizowano je tak, żeby sumaryczne prawdopodobieństwo było równo jedności: $\sum_{i=1}^k p_i = 1$.

6.3.2. Wyznaczenie liczby ziaren i reprezentatywnej objętości modelowej oraz sprawdzenie rozmiarów przestrzeni komórkowej

Do obliczenia liczby ziaren oraz objętości modelowej przestrzeni konieczne jest przyjęcie kilku podstawowych założeń:

- Najmniejsza liczebność klasy n_{\min} (na przykład jedno ziarno, $n_{\min} = 1$).
- Najmniejsza średnica ziarna d_{\min} (na przykład 4–5 komórek, $d_{\min} = 4$).
- Maksymalna dopuszczalna wielkość modelowej przestrzeni, która zależy od możliwości sprzętu komputerowego. Wielkość przestrzeni jest liczona w komórkach: $N_{\max} = n_x \times n_y \times n_z$, gdzie n_x , n_y i n_z – liczba komórek na odpowiednich krawędziach przestrzeni (na przykład $500 \times 500 \times 500$).

Przy uwzględnieniu najmniejszego prawdopodobieństwa występowania klasy $p_{\min} = \min(p_1, p_2, \dots, p_k)$ oraz przyjętej najmniejszej liczebności klasy n_{\min} wstępnie oblicza się liczbę wszystkich ziaren: $n_g = n_{\min} / p_{\min}$. Na podstawie uzyskanej liczby ziaren, stosując wzór Sturgesa: $k = 1 + \log_2 n_g$, można obliczyć lub sprawdzić liczbę klas. Po uzyskaniu znacznie mniejszej liczby klas obliczenia należy powtórzyć, przyjmując zmniejszoną liczbę klas lub zwiększoną najmniejszą liczebność klasy n_{\min} .

Następnie szereg rozdzielczy w postaci prawdopodobieństw występowania każdej klasy powinien być przekształcony w szereg w postaci liczebności klas, przy uwzględnieniu liczby wszystkich ziaren n_g . Wówczas liczebność każdej klasy będzie równa: $n_{gi} = n_g p_i = \frac{n_{\min}}{p_{\min}} p_i$. Uzyskane liczebności klas nie wyrażają się w liczbach całkowitych, więc należy je wszystkie zaokrąglić do liczb całkowitych tak, żeby średni błąd

kwadratowy odchylek był minimalny oraz całkowita liczba ziaren pozostawała bez zmian.

Aby obliczyć objętość modelowej przestrzeni, konieczne jest przejście od rozkładu wielkości ziaren do rozkładu ich objętości. Wówczas oblicza się środki klas objętościowych na podstawie środków klas wielkościowych: $\bar{V}_i = \frac{\pi \bar{d}_i^3}{6}$ oraz średnią objętość ziarna: $\bar{V} = \sum_{i=1}^k \bar{V}_i p_i$ i objętość przestrzeni $V = \sum_{i=1}^k \bar{V}_i n_{gi}$.

Zakładając, że najmniejsza średnica ziarna ma 4–5 komórek lub że objętość ziaren wynosi 35–70 komórek, można uzyskać wymiar lub objętość jednej komórki. Dalej można obliczyć minimalną liczbę komórek niezbędnych do zamodelowania wymaganego rozkładu N . Jeżeli uzyskana liczba komórek jest większa od maksymalnej dopuszczalnej wielkości modelowej przestrzeni ($N > N_{\max}$), należy zmniejszyć liczbę klas i powtórzyć obliczenia. W razie uzyskania liczby komórek mniejszej niż maksymalna dopuszczalna wielkość modelowej przestrzeni ($N < N_{\max}$) można postąpić na trzy sposoby. Pierwszy to zmniejszenie modelowej przestrzeni N przez zmniejszenie liczby komórek n_x , n_y i n_z , drugi – zwiększenie liczby wszystkich ziaren n_g , a trzeci – zmniejszenie rozmiaru komórki.

Bezczelowe jest znaczne zmniejszenie modelowej przestrzeni, rozmiarów komórki również lepiej istotnie nie zmniejszać, ale obydwa sposoby można stosować jako metody docelowe, ponieważ nie wpływają one na reprezentatywną objętość modelową ani nie wymagają powtórzenia obliczeń. Najbardziej korzystne jest zwiększenie liczby ziaren n_g , co powoduje powiększenie liczebności klas (lub liczby klas).

Wynikiem obliczeń są liczba ziaren n_g , liczba klas k i ich liczebność n_{gi} , wymiary przestrzeni komórkowej $N = n_x \times n_y \times n_z$, reprezentatywna objętość modelowa V oraz wymiary liniowe modelowanej przestrzeni $a_x = n_x l$, $a_y = n_y l$, $a_z = n_z l$.

6.3.3. Zadanie warunków zarodkowania

W odniesieniu do pierwszej symulacji założono, że zarodkowanie następuje wyłącznie przed rozrostem ziaren lub że informacja o zarodkowaniu może być wczytana z pliku, pod warunkiem że wcześniej były przeprowadzone jakiekolwiek obliczenia. Faktycznie, rozkład wielkości ziaren po symulacji jest w takim przypadku zbliżony do rozkładu normalnego z wartością odchylenia średniokwadratowego, która jest proporcjonalna do średniej wielkości ziarna (wartości oczekiwanej).

6.3.4. Modelowanie mikrostruktury i wyznaczenie empirycznego rozkładu wielkości ziarna

Dane wejściowe zależą od celu i etapu modelowania. W pierwszej symulacji, kiedy nie została jeszcze utworzona żadna struktura, takimi danymi są wymiary przestrzeni komórkowej $N = n_x \times n_y \times n_z$, objętość reprezentatywna V [mm³], kształt, orientacja

i liczba ziaren n_g , obliczone w drugim etapie algorytmu. Miejsce zarodkowania wybierano losowo. Zarodkowanie wszystkich ziaren następuje przed rozrostem.

W kolejnych symulacjach zarodki powstają w tych samych miejscach co poprzednio, natomiast czas zarodkowania jest wyznaczany w etapie 6. algorytmu. To znaczy, że ziarna pojawiają się także podczas tworzenia struktury.

W trzecim wariancie symulacji czas jest zdefiniowany w 6. etapie algorytmu, a miejsce wybierane losowo.

Wynikiem symulacji jest cyfrowa reprezentacja mikrostruktury, liczba ziaren, ich wielkość d_{gi} oraz empiryczny rozkład ich wielkości w uzyskanej mikrostrukturze (rys. 6.12).

6.3.5. Porównanie rozkładu empirycznego z teoretycznym i wyznaczenie błędu dopasowania

Ponieważ każdy rozkład może być dopasowany idealnie do teoretycznego (pod warunkiem że obliczenia wykonane w etapie 2. będą prawidłowe), jako moment końca obliczeń można przyjąć uzyskanie błędu dopasowania równego zero. Jednak aby uniknąć zbyt dużej liczby powtórzeń, liczba cykli może być ograniczona, można też wprowadzić niezerowy błąd dopasowania.

Najpierw rozkład empiryczny jest porównywany z teoretycznym. Definiowane są dwa parametry: liczba ziaren nie w swojej klasie (sumaryczna odległość ziarna od swojej klasy L) i średni błąd kwadratowy s , obliczany jako różnica między wielkością ziarna a środkiem klasy, w której ziarno powinno się znajdować. Dlatego należy zdefiniować klasę rozkładu teoretycznego dla każdego ziarna. Aby to osiągnąć, stosuje się dwie metody.

Pierwsza polega na wykorzystaniu rozkładu empirycznego, co prowadzi do częściowego (niepełnego) sortowania ziaren. Klasy rozkładu teoretycznego są wypełnione ziarnami z klas rozkładu empirycznego, począwszy od tych największych. Jest to nie tyle wypełnienie klas, ile przypisywanie do nich ziaren.

W drugiej metodzie ziarna są przypisywane do klas po kolei, zgodnie z ich miejscem na liście ziaren, również począwszy od klasy z największymi ziarnami. Z wyjątkiem pierwszej symulacji lista ziaren jest uporządkowana według czasu powstania zarodków. W związku z tym ziarna mają więcej czasu na rozrost, więc są grubsze.

W ten sposób wyznaczano, do której klasy i powinno należeć ziarno j . Obliczona może być także odległość każdego ziarna od jego klasy (podawana w liczbie klas $|\Delta k_j|$), a także różnica między jego wielkością a środkiem jego klasy ($\Delta d_{gj} = d_{gj} - \bar{d}_i$). Odległość jest wykorzystywana do obliczenia kryterium zgodności dwóch rozkładów (błąd dopasowania) L :

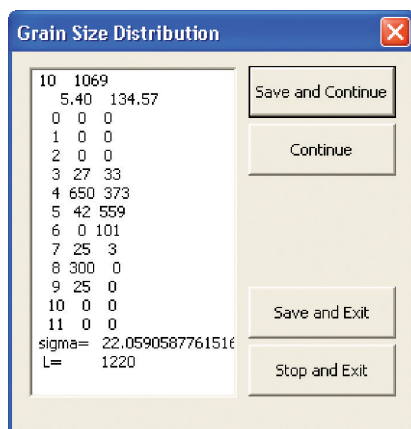
$$L = \sum_{j=1}^{n_g} |\Delta k_j| \quad (6.2)$$

Różnica natomiast służy do korekty czasu zarodkowania i obliczenia średniego błędu kwadratowego s :

$$s = \sqrt{\frac{1}{n_g} \sum_{j=1}^{n_g} (\Delta d_{gj})^2} \quad (6.3)$$

6.3.6. Sprawdzenie kryterium dopasowania i korekta warunków zarodkowania

Jak napisano w podrozdziale 6.2, po uzyskaniu mikrostruktury pojawia się pierwsze okno „Grain Size Distribution” (rys. 6.12). Drugie okno „Grain Size Distribution” (rys. 6.13) pojawi się po wybraniu opcji „Grain Distribution” w oknie „Nucleation” oraz naciśnięciu przycisku „Save and Continue” lub „Continue” w poprzednim oknie. Okno to pozwala oszacować jakość dopasowania rozkładu. W pierwszych dwóch wierszach drugiego okna również wyświetla się liczba klas i ziaren oraz minimalna i maksymalna wielkości ziarna w klasach teoretycznych. Dalej znajdują się wiersze z numerami klas oraz ich teoretyczną i faktyczną liczebnością. Dwie klasy zostały dodane: zerowa, która zawiera ziarna o wielkości mniejszej, niż wynosi minimalna wielkość ziaren w klasach teoretycznych, i ostatnia, z ziarnami większymi niż ziarna o wielkości maksymalnej. Ostatnie wiersze zawierają średni błąd kwadratowy s dopasowania rozmiarów ziaren do środków odpowiednich klas teoretycznych oraz sumaryczną odległość ziaren od ich klasy. Za pomocą przycisku „Save” można zapisać wyniki porównania w pliku „Distribution.txt”. „Exit” umożliwi przerwanie programu. Wciskając przycisk „Continue”, zapiszemy w pliku „GrainsD0.txt” (może on być zmieniony) informacje o ziarnach (w tym skorygowanych czasach zarodkowania), a następnie rozpocznie się nowy cykl obliczeń, mający na celu zbliżenie rzeczywistego rozkładu wielkości ziaren do ich rozkładu teoretycznego.



Rys. 6.13. Zadany i uzyskany rozkład wielkości ziarna

Tak więc drugie okno „Grain Size Distribution” (rys. 6.13) służy do podejmowania decyzji o kontynuacji dopasowywania. Jeżeli wybrana zostaje kontynuacja, należy skorygować warunki zarodkowania. W kolejnych symulacjach zarodki powstają w tych samych miejscach co poprzednio, natomiast czas zarodkowania jest korygowany. Aby wybrać inny wariant, na przykład symulację, podczas której skorygowany czas jest pobierany z pliku, a miejsce wybierane jest losowo, należy przerwać działanie programu i uruchomić go ponownie, zaznaczając odpowiednie pola.

Korekta warunków zarodkowania (lub ich wyznaczenie) opiera się na zasadach, które zostały omówione poniżej. Porównywano rozkład uzyskany z rozkładem zadany. Ziarna w strukturze uzyskanej najpierw szeregowano od największego do najmniejszego, następnie zaliczano je do klas zgodnie z rozkładem zadany. Dla każdego (j -tego) ziarna obliczano różnicę między wielkością uzyskaną d_{gj} a środkiem klasy (i -tej), do której zostało ono zaliczone \bar{d}_i : $\Delta d_{gj} = d_{gj} - \bar{d}_i$. Różnica ta wskazuje, czy ziarno powinno być większe, czy mniejsze.

Korekta istniejącej struktury jest bardziej skuteczna, ale trudniejsza do przeprowadzenia. Zasady jej działania są następujące: dopasować ziarno do tej czy innej klasy można, zwiększając lub zmniejszając jego objętość, przy czym przynajmniej jedno z sąsiednich ziaren również zmieni swoją objętość. Wówczas sąsiednie ziarna także mogą zmieniać swoje klasy. Jeżeli ograniczyć się tylko do jednego sąsiedniego ziarna, czyli przemieszczać tylko jedną granicę międzyziarnową, to objętość dwóch ziaren będzie się zmieniała o tę samą wartość i przemieszczanie się ziaren z klasy do klasy będzie zachodzić z podobnym natężeniem, ale nie identycznym, ponieważ szerokości klas objętościowych nie są równe. Przemieszczenie pojedynczej granicy nie jest łatwe, ponieważ spowoduje ono wytworzenie się w miejscu styku kilku ziaren granic o nienaturalnych kształtach. Tak więc krawędzie i wierzchołki ziaren powinny się znajdować w tym samym miejscu, a zwiększenie objętości będzie uzyskane przez zmianę kształtu granicy (ścian ziaren), co również jest niekorzystne. Lepszym więc rozwiązaniem jest przemieszczenie nie jednej granicy (ściany), a na przykład wierzchołka, w którym stykają się ziarna (najczęściej cztery ziarna). Wówczas jednocześnie zmieniałyby się trzy granice i objętości trzech sąsiednich ziaren, ale granice poruszałyby się nie równolegle, lecz obracałyby się. Jednak najkorzystniejszy jest ruch bez zmiany orientacji granicy. Można go uzyskać, jeżeli założy się go jednocześnie w odniesieniu do wszystkich granic ziaren. W takim przypadku można mówić o rozroście lub zmniejszeniu jednego ziarna w istniejącej strukturze. Realizacja tej opcji nie jest zbyt trudnym zadaniem. Inna wynikająca z takich zmian wielkości pojedynczego ziarna korzyść polega na tym, iż objętości sąsiednich ziaren zmieniają się o wartości kilkakrotnie mniejsze niż objętość wybranego ziarna. Większość sąsiednich ziaren prawdopodobnie pozostanie więc w swoich klasach lub też dystans do pokonania przy przechodzeniu przez nie z klasy do klasy będzie krótszy.

Po wyznaczeniu głównej zasady zmian objętości pojedynczego ziarna można opisać zasadę działania algorytmu dopasowania do teoretycznego szeregu. Dopasowanie większych ziaren, ze względu na nierówną szerokość klas objętościowych, powoduje

znacznie większe zmiany w strukturze niż dopasowanie ziaren mniejszych. Tak więc należałoby rozpoczynać algorytm od dopasowania największych ziaren, a kończyć dopasowaniem mniejszych, jednak niekoniecznie najmniejszych. Na przykład po dopasowaniu dwóch klas z większymi ziarnami można dopasowywać jedną z mniejszymi.

Na początku algorytmu należałoby uporządkować wszystkie ziarna według ich objętości od największego do najmniejszego (lub odwrotnie). Następnie trzeba obliczyć rzeczywisty rozkład objętości ziaren i przejść do dopasowania go do rozkładu wymaganego. Wybierając największe ziarno, doprowadzamy go do klasy z największymi ziarnami i o niezerowej liczebności. W zależności od tego, czy należy ono do jednej z poprzednich lub kolejnych klas, realizujemy jego rozrost lub zmniejszenie, dopóki nie osiągnie ono wymaganej klasy. Podczas rozrostu ziarna należy pilnować, aby nie zniknęło żadne z ziaren sąsiednich. W razie istotnego zmniejszania sąsiedniego ziarna należy rozrost przerwać i dokonać rozrostu tego najmniejszego ziarna. Powtarzamy opisaną procedurę względem kolejnych ziaren, dopóki cała klasa nie zostanie wypełniona. Po wypełnieniu klasy przechodzimy do wypełnienia kolejnej klasy, a po wypełnieniu wszystkich klas rozkład zwykle jest dopasowany, ale może być potrzeba powtórne wykonanie algorytmu wypełnienia klas.

Istotną częścią procedury jest algorytm rozrostu lub zmniejszenia ziarna. Proponuje się następującą procedurę lub cykl zmiany objętości: podczas rozrostu ziarna komórki, które wyznaczają granicę dzielącą rozpatrywane ziarno od ziarna sąsiedniego, przyłączają się do rosnącego ziarna oraz wyznaczane są nowe graniczne komórki dla sąsiedniego ziarna. Podczas zmniejszenia ziarna jego graniczne komórki przyłączają się do sąsiedniego ziarna i wyznacza się nowe komórki graniczne. Oczywiście część komórek ziarna zwiększającego swoją objętość przechodzi do wewnątrz ziarna. Wielkość ziarna w obydwu przypadkach zmienia się w zakresie bliskim wymiarowi dwóch komórek. Po wykonaniu każdego cyklu sprawdza się uzyskaną objętość i, jeśli to konieczne, cykl jest powtarzany. Dopasowywanie prowadzi się do środka klasy.

Iteracyjna korekta warunków zarodkowania nie opiera się na dokładnym badaniu całej mikrostruktury i otoczenia każdego ziarna, dlatego nie pozwala precyzyjnie skorygować czasu zarodkowania każdego ziarna. Jednak przybliżone oszacowanie chwili zarodkowania można przeprowadzić na podstawie różnicy między wymaganą a uzyskaną wielkością ziarna Δd_{gi} . Znając prędkość rozrostu ziarna v_g (a jest ona zadawana i pozostaje niezmiennym parametrem podczas wszystkich obliczeń), można przeprowadzić korektę czasu zarodkowania, czyli chwili, w której powstaje zarodek każdego ziarna: $\Delta t_j = \Delta d_{gi} / v_g$. Trudność polega na tym, że zmiany warunków zarodkowania sąsiednich ziaren są nieznane, więc faktyczna zmiana objętości na poszczególnych granicach może być większa (znacznie większa) lub mniejsza (a nawet mieć odwrotny znak) od oczekiwanej zgodnie z obliczonym Δt_j . Szczególną ostrożność należy zachowywać w stosunku do najmniejszych ziaren, aby nie pozwolić im zaniknąć. W tym celu należy wprowadzać mniejsze zmiany chwil powstania zarodków dla większych ziaren $\Delta t_j < \Delta d_{gi} / v_g$:

$$\Delta t_j = k \Delta d_{gi} / v_g \quad (6.4)$$

lub rozpocząć dopasowywanie od mniejszych ziaren, w kolejnych iteracjach włączając ziarna coraz większe.

Po wyznaczeniu chwil powstania zarodków wyniki zapisuje się do plików. Jeden z nich zawiera nieuporządkowaną informację o współrzędnych zarodków i czasie ich powstania, a drugi – tylko uporządkowane czasy powstania zarodków w kolejności rosnącej. Pliki służą do odpowiedniego wariantu symulacji (por. podrozdz. 6.3.4). Kryterium końca obliczeń może być $L = 0$.

6.3.7. Obliczenia końcowe prędkości zarodkowania oraz modelowanie sprawdzające

Poza cyfrową prezentacją mikrostruktury o zadanym rozkładzie wielkości ziarna bardzo istotne jest wyznaczenie warunków zarodkowania. Można określić trzy jego prędkości: bezwzględną, względną i rzeczywistą.

Bezwzględna prędkość zarodkowania może być uzyskana bezpośrednio z danych dotyczących czasów powstania zarodków, czyli jest funkcją liczby zarodków $\Delta n_g(t_m)$ powstałych w jednostce objętości V w przyroście czasu modelowania Δt_m :

$$v_{nb}(t_m) = \frac{\Delta n_g(t_m)}{V \Delta t_m} \quad (6.5)$$

Względna prędkość zarodkowania różni się od bezwzględnej tym, iż jest odniesiona nie do pełnej objętości V , lecz do wolnej od ziaren przestrzeni V_m , która również jest funkcją czasu modelowania:

$$v_{nw}(t_m) = \frac{\Delta n_g(t_m)}{V_m(t_m) \Delta t_m} \quad (6.6)$$

Rzeczywista prędkość zarodkowania może być obliczona tylko w odniesieniu do rzeczywistego procesu (rekrytalizacji, przemiany fazowej i innych). Różni się od względnej tym, że wyznaczone funkcje są funkcjami czasu rzeczywistego t :

$$v_{nr}(t) = \frac{\Delta n_g(t)}{V_m(t) \Delta t} \quad (6.7)$$

Wyznaczona w ten czy inny sposób prędkość zarodkowania może być poddana modelowaniu testowemu lub sprawdzającemu. W takiej symulacji wystarczy zadać dowolną przestrzeń komórkową, odtworzyć obliczone warunki zarodkowania i uzyskać empiryczny rozkład wielkości ziaren, który można porównać z rozkładem zadanym. Zbieżność wyników będzie wskazywała na poprawność uzyskanych rezultatów.

Innym sposobem sprawdzenia może być symulacja rozkładu z określonymi warunkami zarodkowania i wyznaczenie tych warunków na podstawie mikrostruktury.

6.3.8. Określenie warunków zarodkowania

W wyniku obliczeń końcowych można uzyskać różne prędkości zarodkowania przy stałej prędkości rozrostu według wzorów (6.5)–(6.7). Uzyskanie takich prędkości rodzi następujące pytania:

- czy uzyskane prędkości zarodkowania odpowiadają rzeczywistej prędkości zarodkowania?
- czy na podstawie mikrostruktury lub rozkładu wielkości ziarna można określić warunki zarodkowania i rozrostu ziaren?

W związku z powyższym można sformułować cel badań: będzie nim określenie na podstawie znanego rozkładu wielkości ziarna warunków zarodkowania i (lub) rozrostu ziaren.

Oczywiście jeden wariant modelowania jest wynikiem tylko jednej próby i uzyskany rozkład wielkości ziarna będzie rozkładem próbnym, a zbiór próbny będzie zawierał tylko jeden element. Zbiór generalny będzie natomiast zawierał olbrzymią liczbę elementów. Jego liczebność w niektórych wariantach można obliczyć. Jeżeli zarodkowanie zachodzi przed rozrostem w przestrzeni modelowej z N_c komórek i powstaje n_g zarodków, liczebność zbioru generalnego N_n będzie równa liczbie możliwych wariantów wyboru n_g elementów z N_c :

$$N_n = \frac{N_c!}{(N_c - n_g)!n_g!} \quad (6.8)$$

Zbiór generalny w tym przypadku jest jednorodny. Ponieważ jest to zmienna losowa, każdy element różni się od zbioru generalnego i stanowi tylko jedną jego reprezentację. W celu oszacowania zbioru generalnego należy przeprowadzić kilka badań, aby liczebność próby była wystarczająco duża. Oczywiście jest, że ponieważ żadne dwa elementy zbioru generalnego nie są identyczne, uzyskana mikrostruktura i rozkład wielkości ziaren będą różne. Przy czym w tych dwóch wariantach nie istnieje możliwości uzyskania tej samej mikrostruktury, nawet jeśli się będzie sterować czasem zarodkowania. Natomiast można uzyskać bardzo zbliżone rozkłady wielkości ziaren w owych dwóch wariantach, przynajmniej pod względem szerokości klas, co zostanie zilustrowane przykładami w podrozdziale 6.4. Ten sam rozkład wielkości ziaren można zatem uzyskać przy różnym rozmieszczeniu zarodków, dobierając czasy ich powstania. Oznacza to, że dwa podobne rozkłady będą uzyskane w nieco różnych warunkach zarodkowania, w zależności od miejsca powstania zarodków. Odpowiedź na pytanie, czy uzyskane prędkości zarodkowania odpowiadają rzeczywistej prędkości zarodkowania, będzie zatem negatywna, ponieważ prędkości zarodkowania w dwóch dowolnych wariantach rozmieszczenia zarodków będą nieco różne. Natomiast identyczny rozkład z identyczną prędkością zarodkowania można uzyskać tylko w przypadku, kiedy zarodki będą rozmieszczone we właściwych miejscach i zostanie uzyskana identyczna mikrostruktura.

Skutkiem nieidentyczności jest to, że jeżeli w rzeczywistości zarodki powstały w jednej chwili, to zarodkowanie przy próbie uzyskania rozkładu będzie rozciągnięte w czasie. W rzeczywistości zarodkowanie zawsze będzie zachodzić w krótkim czasie, natomiast w modelu – w dłuższym.

Tak więc należy rozpatrywać oddzielnie dwa zagadnienia: uzyskanie mikrostruktury z zadany rozkładem wielkości ziaren i wyznaczenie warunków zarodkowania na podstawie uzyskanej mikrostruktury. Przykłady dotyczące pierwszego zagadnienia zostały omówione w podrozdziale 6.4, natomiast drugie z nich wiąże się z koniecznością udzielenia odpowiedzi na kilka pytań. Pierwsze z nich brzmi: czy parametry zbioru generalnego są zależne od parametrów przestrzeni i warunków zarodkowania, a jeśli są, to w jaki sposób?

Żeby odpowiedzieć na to pytanie, należy rozwiązać kilka prostszych zadań. Można zbadać trzy warianty zarodkowania i rozrostu ziaren:

- 1) pełne zarodkowanie przed rozrostem ziaren;
- 2) zarodkowanie ze stałą prędkością podczas całego procesu rozrostu przebiegającego ze stałą prędkością;
- 3) dowolny proces (zarodkowanie zarówno przed rozrostem, jak i podczas rozrostu ziaren; zmienne prędkości zarodkowania i (lub) rozrostu ziaren).

Rozpatrując pierwszy wariant, można sformułować pierwszą hipotezę.

Hipoteza pierwsza. W procesie zarodkowania przed rozpoczęciem rozrostu istnieje znormalizowany rozkład wielkości ziaren, który nie zależy od rozmiarów przestrzeni, liczby ziaren, ich kształtu ani prędkości rozrostu.

Należy podkreślić, że wymiary przestrzeni komórkowej rozpatrywane jako liczba komórek, a nie jako długość jej trzech wymiarów liniowych, wpływają jedynie na rozdzielczość wyników i dokładność wyznaczenia wielkości ziaren, więc przy wystarczającej rozdzielczości ich wpływ można pominąć.

Niezależność od prędkości rozrostu pod warunkiem, iż jest ona taka sama we wszystkich punktach przestrzeni, wynika z faktu, że czas modelowy można przeskalować w dowolny sposób, zmieniając rzeczywisty czas i tym samym rzeczywistą prędkość rozrostu ziaren. Współczynnik przeskalowania może się przy tym zmieniać, co spowoduje, że prędkość rozrostu ziaren również jest zmienna. Można przeprowadzić i odwrotną operację.

Jeżeli więc założyć stałą prędkość rozrostu ziaren w modelu przy zmiennej prędkości rzeczywistej, właśnie rzeczywista prędkość rozrostu ziaren v_g pozwala wprowadzić do modelu czas rzeczywisty. Wówczas zmiany prędkości rozrostu można zastosować do skalowania czasu i w konsekwencji do skalowania prędkości zarodkowania. Jeżeli w symulacji rozpatrywano proces ze stałą prędkością rozrostu i rzeczywista prędkość rozrostu jest znana w każdej chwili czasu, to skala czasu rzeczywistego może być wyznaczona przez stosunek prędkości modelowej v_{gm} do rzeczywistej v_{gr} oraz czas modelowy t_m :

$$\Delta t(t) = \frac{v_{gm} \Delta t_m}{v_{gr}(t)} \quad (6.9)$$

Jest to bardzo istotny fakt, do którego będziemy się odwoływać w dalszej części tej pracy.

Jeżeli rozpatrywać dwa dowolne warianty rozmieszczenia zarodków o tej samej liczbie ziaren, oczywiście jest, że należą one do tego samego zbioru generalnego. Jeżeli połączyć je w jedną przestrzeń, dwukrotnie większą od pierwotnej, będzie ona miała ten sam rozkład wielkości ziaren, więc można ją zaliczyć do tego samego zbioru generalnego. Tak więc każdą wielokrotną (większą lub mniejszą) przestrzeń zaliczamy do zbioru generalnego. A jeżeli tak, to ponieważ liczba komórek zawsze jest liczbą całkowitą, dowolną przestrzeń z dowolnym stosunkiem wymiarów również można zaliczyć do tego samego zbioru. Wprowadzając pojęcie objętościowej gęstości zarodkowania ρ_n , która jest równa liczbie ziaren n_g w jednostkowej objętości $\rho_n = n_g/V$, można stwierdzić, że ta sama gęstość zarodkowania ρ_n zawsze daje taki sam rozkład wielkości ziaren (w zbiorze generalnym) i może służyć jako liczba podobieństwa.

Kolejne pytanie dotyczy zależności rozkładu wielkości ziaren od gęstości zarodkowania. Przestrzeń z mniejszą liczbą ziaren można rozpatrywać jako wycinek większej przestrzeni, w której znajdowałyby się taka sama liczba ziaren. Wówczas można by było zaliczyć je do jednego zbioru generalnego, pod warunkiem uwzględnienia różnych wymiarów przestrzeni, czyli w takiej sytuacji, w której rozkłady byłyby proporcjonalne.

Otwarte pozostaje pytanie o niezależność rozkładu od kształtu ziaren, w szczególności ziaren, których poszczególne wymiary liniowe różnią się od siebie (wydłużone cylindry, elipsoidy, dyski).

Podsumowując, można zatem stwierdzić, że hipoteza pierwsza jest słuszna przynajmniej w odniesieniu do ziaren o jednakowych kształtach. Istnieje więc znormalizowany rozkład gęstości prawdopodobieństwa wielkości ziaren. Pod pojęciem normalizacji należy rozumieć wielkość ziarna d odniesioną do jej wartości oczekiwanej \bar{d} (średnią wielkość ziarna), która może być wyznaczona jako odwrotność pierwiastka sześciennego z gęstości zarodkowania z uwzględnieniem współczynnika kształtu k : $\bar{d} = \frac{1}{k\sqrt[3]{\rho_n}}$.

Zatem znając znormalizowany rozkład $p(d/\bar{d})$ i gęstość zarodkowania, można obliczyć rzeczywisty rozkład wielkości ziaren w procesie, w którym zarodkowanie nastąpiło przed ich rozrostem. Porównanie rozkładu rzeczywistego ze znormalizowanym pozwala sprawdzić, czy rzeczywisty proces można zaliczyć do takich, w których zarodkowanie nastąpiło przed rozrostem ziaren.

Rozpatrując drugi wariant zarodkowania i rozrostu ziaren, można sformułować kolejną hipotezę.

Hipoteza druga. W warunkach stałych (niezmiennych podczas procesu) prędkości zarodkowania i rozrostu ziaren istnieje znormalizowany rozkład wielkości ziaren. Wielkość ziarna zależy od stosunku prędkości zarodkowania i prędkości rozrostu.

Jak pokazano wyżej, prędkość rozrostu ziaren można rozpatrywać jako czynnik skalujący model, który to czynnik pozostaje niezmienny w procesie.

Stosując rozumowanie takie samo jak poprzednio, można pokazać, że wpływ wymiarów przestrzeni, liczby ziaren oraz prędkości rozrostu jest taki sam również w procesie ze stałą prędkością zarodkowania. Należy tylko wprowadzić zamiast gęstości zarodkowania ρ_n , prędkość zarodkowania $v_n = d\rho_n/dt$. Również wielkość ziarna zależy nie od gęstości zarodkowania ρ_n , lecz od prędkości zarodkowania v_n . Uwzględniając fakt, że z jednej strony prędkość zarodkowania jest funkcją liczby ziaren w jednostce czasu, a z drugiej strony prędkość rozrostu ziaren v_g może być wykorzystana jako czynnik skalujący, można stwierdzić, że liczba podobieństwa w takim zagadnieniu będzie stosunkiem dwóch prędkości: v_n/v_g .

I tak, jak wyżej, można otrzymać rozkład gęstości prawdopodobieństwa znormalizowanej wielkości ziaren $p(d/\bar{d})$, tylko że w tym przypadku wymiar ziarna zależy od dwóch prędkości: $\bar{d} \propto \sqrt[4]{v_g/v_n}$.

Rozpatrując trzeci wariant, można sformułować kolejną hipotezę.

Hipoteza trzecia. Przy znanej prędkości rozrostu ziaren można ustalić zależność prędkości zarodkowania od czasu, czyli wyznaczyć dowolne warunki zarodkowania.

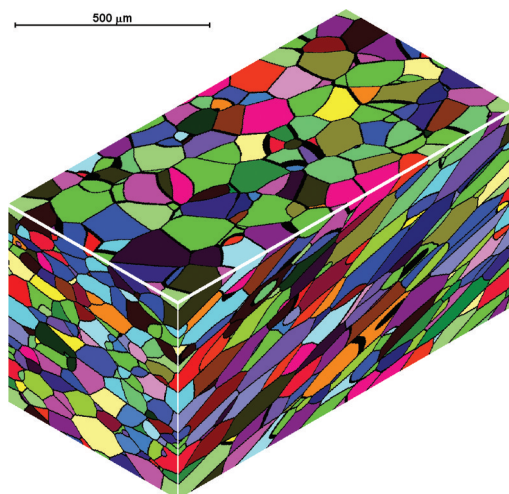
Uzasadnienie tej hipotezy jest jeszcze bardziej skomplikowane i nie zostanie tu przytoczone. Jednak przyjęcie tej hipotezy umożliwia ustalenie warunków zarodkowania na podstawie uzyskanej mikrostruktury i rozkładu wielkości ziarna.

Należy jednak przypomnieć, że wszystko to ma zastosowanie w odniesieniu do zbioru generalnego, jednak zarówno w procesach rzeczywistych, jak i podczas modelowania mamy do czynienia z jednokrotną reprezentacją lub próbą o małej liczebności, co wprowadza błąd przypadkowy i nie pozwala dokładnie wyznaczyć warunków zarodkowania.

6.4. Przykład symulacji mikrostruktury i jej wyniki

W tym rozdziale zaprezentowano trzy warianty symulacji mikrostruktury. Wszystkie przedstawiają mikrostrukturę z tą samą liczbą ziaren (1069), z tym samym średnim stosunkiem poszczególnych wymiarów (4:2:1) i pochyleniem najdłuższej osi ziarna do osi x wynoszącym 20 stopni. Wymiary reprezentatywnej objętości: $x \times y \times z = 1000 \times 500 \times 500 \mu\text{m}$.

W pierwszym wariantie strukturę otrzymano za pomocą symulacji z ciągłym zarodkowaniem podczas całego procesu modelowania struktury na automatach komórkowych o wymiarach $n_x \times n_y \times n_z = 800 \times 400 \times 400$ komórek. Wzorcową mikrostrukturę pierwszego wariantu przedstawiono na rysunku 6.14. W drugim wariantie wybrano sztuczny unimodalny rozkład z prawie równymi ziarnami. Trzecim wariantem jest rozkład bimodalny. Wszystkie rozkłady wyznaczono za pomocą dziesięciu klas (z dwoma dodatkowymi nieograniczonymi klasami zewnętrznymi). Parametry zadanych rozkładów zebrano w tabeli 6.1.



Rys. 6.14. Wzorcową mikrostruktura w pierwszym wariantie symulacji (Svyetlichnyy i Łach 2012)

Tabela 6.1

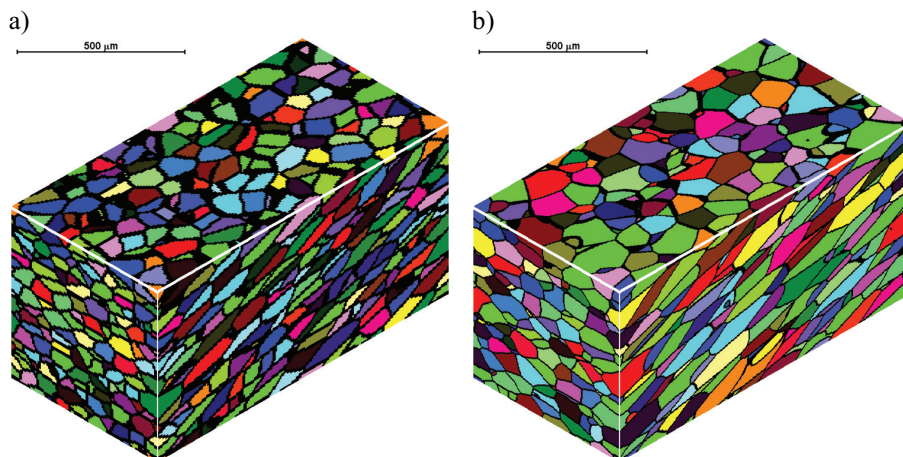
Zadane rozkłady wielkości ziarna (Svyetlichnyy i Łach 2012)

Klasa	Środek klasy $\bar{d}_i, \mu\text{m}$	Wariant 1	Wariant 2	Wariant 3
0	–	0	0	0
1	10,8	61	0	0
2	23,7	116	0	0
3	36,7	115	0	33
4	49,6	152	0	650
5	62,5	133	30	36
6	75,5	147	1000	0
7	88,4	136	39	25
8	101,4	130	0	300
9	114,3	65	0	25
10	127,2	14	0	0
11	–	0	0	0

Pierwszy wariant symulacji rozpoczyna się od procesu, w którym zarodkowanie następuje przed rozrostem ziarna, drugi i trzeci wariant – od wynikowej struktury i warunków zarodkowania uzyskanych w poprzednim wariantcie.

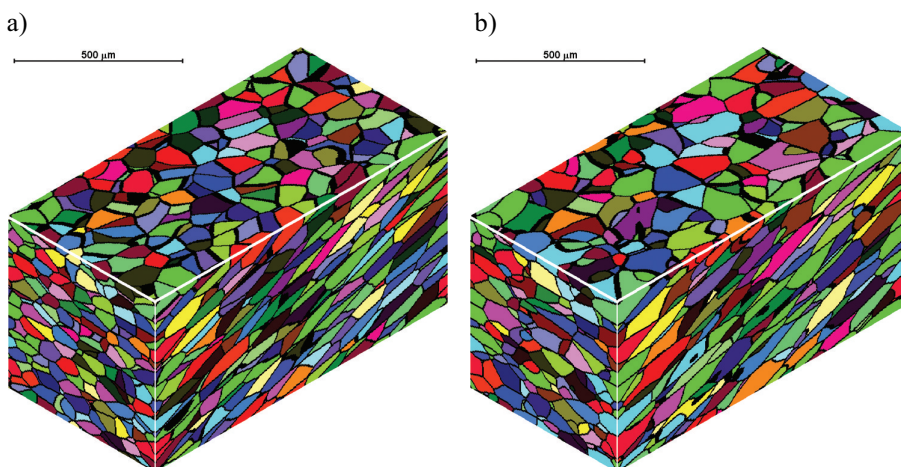
Pierwszą symulację w pierwszym wariantcie przeprowadzono na automatach komórkowych o wymiarach $n_x \times n_y \times n_z = 200 \times 100 \times 100$ komórek. Następnie, po dopasowaniu współczynnika k (6.4), który wpływa na wartość korekty chwili powstania

zarodków Δt_j i prędkość dopasowania rozkładu, przestrzeń zmieniono na $n_x \times n_y \times n_z = 400 \times 200 \times 200$ komórek. Dalej przestrzeń pozostaje bez zmian we wszystkich pozostałych symulacjach. Wzorcowa mikrostruktura dla pierwszego wariantu została przedstawiona na rysunku 6.14, natomiast początkowa i wynikowa na rysunku 6.15. Wzorcowa i wynikowa mikrostruktura są identyczne, ze względu na rozkład wielkości ziarna, natomiast mikrostruktura początkowa istotnie się od nich różni.



Rys. 6.15. Pierwszy wariant symulacji mikrostruktury: a) po pierwszej symulacji; b) po ostatniej symulacji (po dopasowaniu) (Svyetlichnyy i Łach 2012)

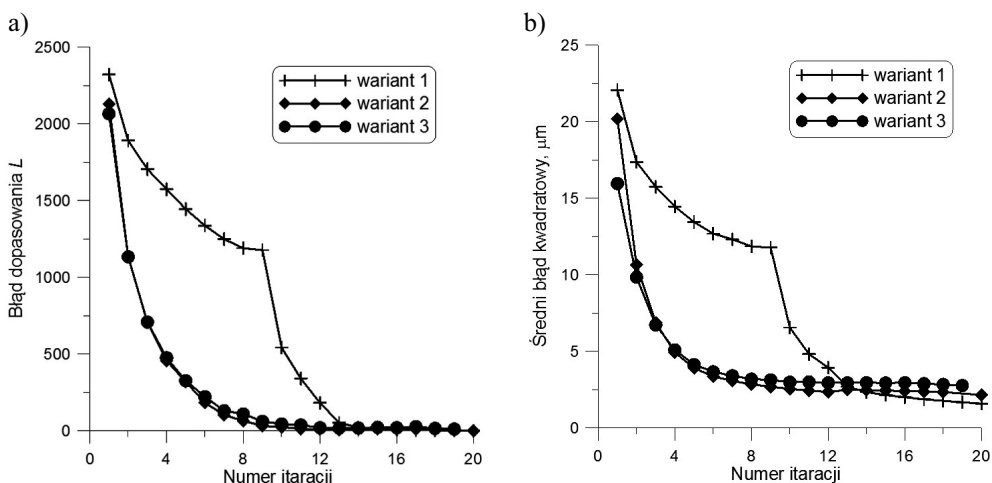
Wynikowa mikrostruktura w drugim i trzecim wariantcie symulacji została przedstawiona na rysunku 6.16.



Rys. 6.16. Wynikowa mikrostruktura w drugim (a) i trzecim (b) wariantcie symulacji (Svyetlichnyy i Łach 2012)

Porównanie wynikowej struktury w wariancie drugim ze strukturą początkową w wariancie pierwszym pokazuje, iż w wyniku dopasowywania w drugim wariancie uzyskano ziarna o bardziej zbliżonych rozmiarach niż możliwe do uzyskania w rzeczywistym procesie. Za pomocą powyżej omówionego algorytmu można uzyskać ziarna o identycznej wielkości, co jest niemożliwe w jakimkolwiek rzeczywistym procesie. Stosując tę metodę, można otrzymać rozkład: równomierny (zbliżony do tego w wariancie 1), unimodalny (wariant 2), bimodalny (wariant 3), zbliżony do rzeczywistego i inne.

Zmieniając współczynnik k (6.4) od zera do wartości bliskiej 1, można sterować szybkością dopasowania. Jego wpływ widoczny jest na rysunku 6.17 przedstawiającym sytuację, w której na początku symulacji w wariancie 1 wynosił on $k = 0,4$, a potem jego wartość wzrosła do 1, po czym zwiększyła się szybkość dopasowania. Faktycznie w programie zamiast jednego współczynnika wykorzystywano trzy, z których każdy odpowiada za inny rodzaj zmian. Pierwszy, największy, jest stosowany w sytuacjach, kiedy należy zmniejszyć wielkość ziarna. Drugi wykorzystywano, kiedy ziarno powinno być większe. Współczynnik ten może mieć taką samą wartość jak pierwszy lub nieco mniejszą. Trzeci współczynnik dotyczy ziaren, które znajdują się w jednej klasie; jest on bliski do zera (jest to ważne, aby osiągnąć błąd dopasowania równy zero: $L = 0$). Pod koniec symulacji wartość współczynników należałoby zmniejszać.

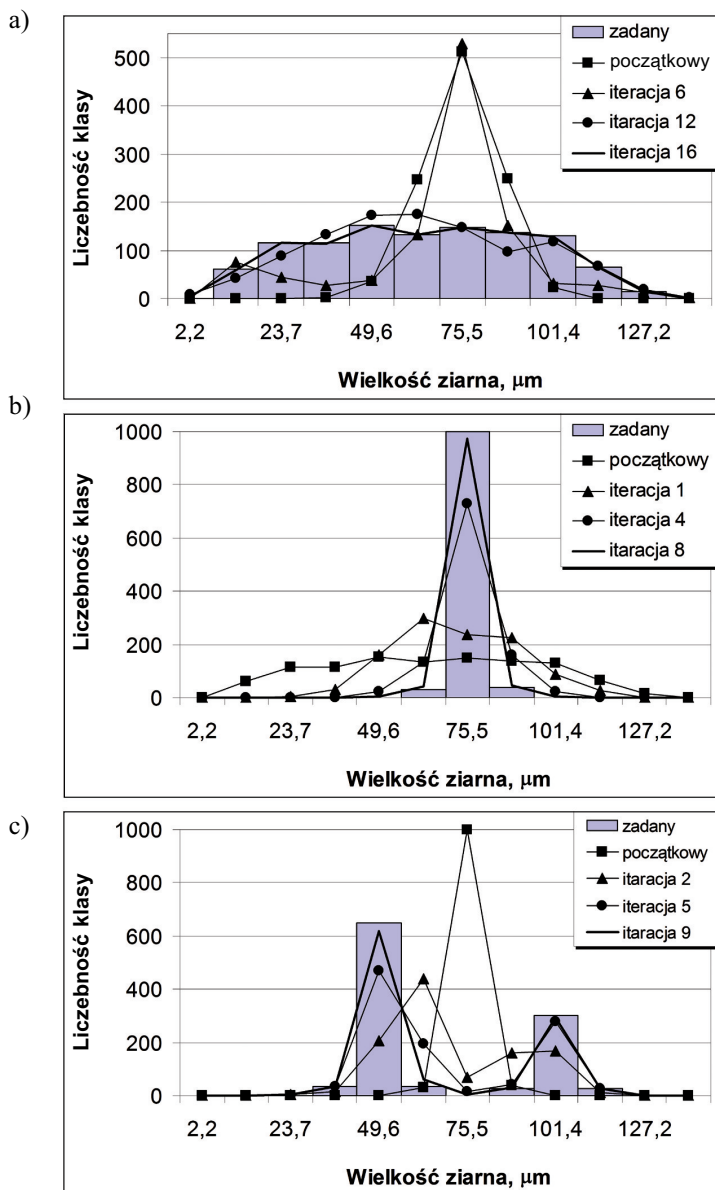


Rys. 6.17. Zmiany błędu dopasowania L (a) i średniego błędu kwadratowego s (b) (Svyetlichnyy i Łach 2012)

Rozkład wielkości ziaren: początkowy, zadany i uzyskany przy różnych krokach dopasowywania, pokazano na rysunku 6.18.

Analizując uzyskane wyniki, należy stwierdzić, że omówione wyżej symulacje pokazały, iż opracowany algorytm i program pozwalają na uzyskanie dowolnej wymaganej topologii. Prawidłowy dobór parametrów modelowania przy wstępnych obliczeniach –

wymiary przestrzeni i liczba ziaren wraz ze skutecznym algorytmem doboru warunków zarodkowania – gwarantuje perfekcyjne dopasowanie rozkładu wielkości ziarna do wymaganej. Struktura wynikowa może być wykorzystana do badań własności materiału lub do określenia warunków jej powstania.



Rys. 6.18. Zmiany rozkładu wielkości ziarna podczas dopasowywania:
a) wariant 1; b) wariant 2; c) wariant 3 (Svyetlichnyy i Łach 2012)

6.5. Orientacja krystalograficzna ziaren

Jak napisano w podrozdziale 6.2, istnieją dwa rodzaje orientacji ziarna: przestrzenna i krystalograficzna. Przy modelowaniu początkowej mikrostruktury nie rozróżniano ich. W wyniku symulacji orientacja wszystkich ziaren w końcowej mikrostrukturze jest albo taka sama, albo losowa z rozkładem zbliżonym do równomiernego. Obydwa warianty nie są naturalne i należy to poprawić, tak więc po uzyskaniu mikrostruktury orientacja krystalograficzna powinna być zmieniona innym podprogramem, który wprowadzi zadany rozkład orientacji ziaren lub rozkład kątów dezorientacji granic ziaren.

Jako rozkład wzorcowy może być wybrany zarówno rozkład teoretyczny, jak i doświadczalny. Uzyskanie wymaganego rozkładu zawsze rozpoczyna się od dopasowania parametrów rozkładu do uzyskanej mikrostruktury, czyli od wyznaczenia liczby klas i ich liczebności w sposób podobny do opisanego w podrozdziale 6.3.1, tylko z pominięciem skomplikowanego wyboru parametrów przestrzeni. Liczba ziaren lub granic jest narzucona przez uzyskaną strukturę.

Jeżeli rozkład jest zadany wzorem, należy przekształcić rozkład zmiennej losowej ciągłej w rozkład zmiennej skokowej (szereg rozdzielczy). Wówczas ciągły rozkład gęstości prawdopodobieństwa zostaje zastąpiony szeregiem rozdzielczym zadaną liczbą klas i wyznaczeniem prawdopodobieństwa występowania każdej klasy.

Szereg rozdzielczy powinien być przekształcony w histogram liczebności elementów każdej klasy. W tym celu należy wyznaczyć teoretyczną liczebność każdej klasy, jako iloczyn prawdopodobieństwa wystąpienia klasy i liczby wszystkich elementów w niej zawartych. Potem należy zastąpić uzyskaną liczebność liczbą całkowitą, minimalizując sumę kwadratów odchyłek liczby całkowitej od teoretycznej liczebności. Jako wynik takiego działania uzyskamy gotowy rozkład, w którym będzie wyznaczona optymalna liczba elementów w każdej klasie.

W zależności od wymagań co do mikrostruktury możliwe są trzy warianty: dopasowanie rozkładu orientacji ziaren, dopasowanie rozkładu dezorientacji granic ziaren lub dopasowanie obydwu tych rozkładów. Warianty te rozpatrzono poniżej.

6.5.1. Dopasowanie rozkładu orientacji ziaren

Jest to dość prosty algorytm, który faktycznie polega tylko na tym, że należy wybrać ziarno i przypisać je do odpowiedniej klasy. Może to być algorytm całkowicie deterministyczny lub z elementami losowymi. W niektórych przypadkach, głównie kiedy zarodkowanie odbywa się przed rozrostem ziaren, rozmieszczenie ziaren w przestrzeni jest przypadkowe i uzyskana ich lista nie jest uporządkowana ani przestrzenią, ani czasowo. Wówczas można by było zastosować algorytm deterministyczny. Jednak jeżeli zarodkowanie odbywa się również podczas rozrostu ziaren (w tym przy dopasowywaniu rozkładu wielkości ziaren), uzyskana lista ziaren, pozostając losową względem przestrzeni, staje się uporządkowaną względem czasu i wielkości ziaren. Wynika

więc z tego, że niezależnie od sposobu uzyskania mikrostruktury, lepiej jest zastosować algorytm z elementami losowymi

Algorytm polega na przypisaniu każdego ziarna do odpowiedniej klasy rozkładu orientacji ziarna. Wówczas z pierwotnej listy ziaren, w odniesieniu do której zakłada się, że znajdujące się na niej ziarna nie posiadają orientacji, jest losowo wybierany element i przenosi się go do nowej listy ziaren z przypisaniem do odpowiedniej klasy. Wypełniać klasy można w dowolnej kolejności, począwszy od wartości najmniejszych, od największych, od najbardziej liczebnej klasy itp. W programie wybrany został pierwszy wariant. Po wypełnieniu pierwszej klasy program przechodzi do kolejnej i tak do wyczerpania pierwotnej listy ziaren i wypełnienia wszystkich klas.

Jeżeli przewidywano dopasowanie rozkładu dezorientacji granic, algorytm ten kończy pracę, w przeciwnym razie jest precyzowana orientacja ziaren wewnątrz każdej klasy. Rozkład gęstości prawdopodobieństwa wewnątrz klasy jest aproksymowany funkcją kwadratową na podstawie szerokości tej klasy, jej liczebności oraz liczebności dwóch klas sąsiednich. Dalej zgodnie z tym rozkładem losuje się orientację każdego ziarna. W ten sposób każde ziarno uzyskuje swoją unikatową orientację krystalograficzną.

6.5.2. Dopasowanie rozkładu kąta dezorientacji granic ziaren

Dopasowanie rozkładu dezorientacji ziaren wymaga wstępnego przygotowania, ponieważ pliki z danymi o uzyskanej mikrostrukturze nie zawierają bezpośredniej informacji nie tylko o kątach dezorientacji, ale nawet o tym, które ziarno z którym się styka i jaka jest ogólna liczba granic. W tym celu jest uruchamiany podprogram, który uzupełnia informacje dotyczące każdego ziarna o listę jego granic oraz tworzy listę wszystkich granic zawierającą informację o stykających się ziarnach i kącie dezorientacji. Podprogram pracuje według algorytmu podobnego do opisanego w podrozdziale 5.1, jest jednak w stosunku do niego nieco uproszczony, pomija kilka etapów, za to dodatkowo oblicza kąt dezorientacji.

Podobnie jak w innych algorytmach rozkład jest przekształcany w histogram liczebności klas. Lista ziaren jest uporządkowana nie wzrastająco według liczby ziaren, z którymi rozpatrywane ziarno z listy ma wspólną granicę. Kolejny element listy ma więc równą lub mniejszą liczbę sąsiednich ziaren w porównaniu z poprzednim elementem listy.

Przed rozpoczęciem procesu dopasowania rozkładu orientację ziarnom nadaje się losowo lub rozpoczyna się od struktury z ziarnami o tej samej orientacji. Drugi wariant można wybrać, kiedy mamy do czynienia z małymi kątami dezorientacji (na przykład kiedy jest to związane z rozwojem struktur dyslokacyjnych w podziarna i oddzielne ziarna – zob. rozdz. 10). Wówczas uzyskany rozkład można rozpatrywać jako charakterystyczny dla struktury statycznej lub jako jeden etap w procesie rozwoju struktury. W pozostałych przypadkach zalecany jest wybór orientacji losowej.

Jeśli przy dopasowaniu rozkładu wielkości ziarna jako kryterium wykorzystywano sumaryczną odległość L (6.2) ziarna od swojej klasy, decyzja o przerwaniu procesu

należy do użytkownika. Przy dopasowywaniu rozkładu orientacji ziarna kryterium to było niepotrzebne, ponieważ klasy były wypełniane wprost. Natomiast przy dopasowywaniu rozkładu kąta dezorientacji granic ziaren zastosowano dwa kryteria. Pierwsze z nich, F_{b1} , jest sumą kwadratów odchylenia liczebności klasy od zadanej wartości i ma następującą postać:

$$F_{b1} = \sum_{i=1}^{k_b} (n_{bi} - n_{bi0})^2 \quad (6.10)$$

gdzie:

- i – numer klasy kąta dezorientacji granic ziaren,
- k_b – liczba klas kąta dezorientacji,
- n_{bi}, n_{bi0} – liczebność i -tej klasy, odpowiednio uzyskana i zadana.

Drugie kryterium, F_{b2} , jest zbliżone do kryterium L (6.2), również opiera się na odległości. Jednak w tym przypadku nie można *a priori* przypisać każdej granicy wprost do tej czy innej klasy, dlatego najpierw wyznacza się klasy z nadwyżką k^+ i z niedoborem k^- liczebności, a następnie – odpowiedniość tych klas i kryterium F_{b2} , które jest sumą kwadratów odległości klas z nadwyżką k^+ od klas z niedoborem k^- liczebności:

$$F_{b2} = \sum_{j=1}^{n_k} (k_j^+ - k_j^-)^2 \quad (6.11)$$

Pierwsze kryterium, F_{b1} , jest nadrzędne, uprzywilejowane, drugie F_{b2} – podrzędne, pomocnicze. Więc algorytm określa cykliczne przejście po całej liście ziaren, poczynając od ziaren z największą liczbą sąsiadów, aż zostanie spełniony warunek zatrzymania procedury. Następuje to, gdy brak jest zmian kryteriów F_{b1} i F_{b2} w cyklu.

Podczas wykonywania procedury cyklu dla każdego ziarna przeprowadzano najpierw minimalizację kryterium F_{b1} , a potem dla minimalnej wartości kryterium F_{b1} poszukiwano minimum kryterium F_{b2} .

Poszukiwanie minimum polega na obrocie wybranego ziarna po kolei wokół trzech lokalnych osi.

Inne podejście, które sprawia wrażenie bardziej obiecującego, polega na analitycznym wyznaczeniu orientacji ziarna, która prowadzi do zmniejszenia wartości F_{b1} i F_{b2} .

Niestety, funkcja, którą określają omówione kryteria, ma dużą liczbę lokalnych minimów, co pociąga za sobą konieczność wielokrotnego powtórzenia obliczeń. Poza tym duża, a wręcz olbrzymia liczba lokalnych minimów uniemożliwia nawet uzyskanie odpowiedzi na pytanie, czy możliwe jest pełne dopasowanie rozkładów, dopóki program przypadkowo nie trafi na dokładne dopasowanie rozkładów.

W programie zastosowano metodę potrojenia prób (optymalnie liczba prób powinna być zwiększona o liczbę e). Początkowo zakładano dziesięć prób. Jeżeli wartość minimalną kryterium osiągnięto w próbie o numerze n , nową liczbę prób wyznaczano

jako $3n$ (lub en). Jeżeli w wyniku wykonywania dodatkowych prób nie da się uzyskać mniejszej wartości, obliczenia się kończą. Jeżeli zostanie uzyskana wartość mniejsza od poprzedniej, wykonuje się kolejne próby. Ogólną liczbę prób ograniczono do 100.

6.5.3. Dopasowanie rozkładu orientacji ziaren i kąta dezorientacji granic

Jak wykazano w poprzednich podrozdziałach, rozkład orientacji ziaren jest osiągany zawsze, natomiast nie ma gwarancji uzyskania zadanego rozkładu kąta dezorientacji granic. Tak więc dopasowanie dwóch rozkładów ma na ogół dwa warianty.

Pierwszy polega na dokładnym zachowaniu rozkładu orientacji ziaren i maksymalnym dopasowaniu rozkładu ich dezorientacji do rozkładu zadanego. Drugi, kompromisowy, polega na dwukryterialnej minimalizacji. W niniejszym podrozdziale zostanie rozpatrzone tylko pierwszy wariant.

W pierwszym etapie dopasowania należy uzyskać dokładne dopasowanie rozkładu orientacji ziaren (por. podrozdz. 6.5.1). W etapie drugim stosuje się metodę Monte Carlo. Rozwiązanie polega na losowaniu. Wybiera się dwa dowolne ziarna, dla których następuje wymiana orientacji krystalograficznej oraz sprawdzenie kryteriów F_{b1} (6.10) i F_{b2} (6.11). W razie wzrostu wartości F_{b1} i F_{b2} wymiana jest odwoływana, a w razie zmniejszenia – zatwierdzana. Główną wadą metody Monte Carlo w zastosowaniu do tego zagadnienia jest niewyznaczalność lub trudność w określeniu warunku zatrzymania obliczeń. Ponieważ para ziaren jest wybierana losowo, przy zbliżeniu do optimum prawdopodobieństwo wyboru pary, która doprowadzi do poprawy dopasowania, istotnie maleje. Jeżeli taka para istnieje, prawdopodobieństwo jej pominięcia jest bardzo wysokie. Dlatego kiedy liczba nieudanych wymian wzrasta do wartości n_b^2 , należy albo przyjąć ten wynik, albo przejść do etapu trzeciego.

Trzeci etap wykorzystuje algorytm pełnego wyboru, kiedy sprawdza się po kolei wszystkie możliwe pary ziaren. Jeżeli to nie prowadzi do poprawy dopasowania rozkładu, oznacza to, że znaleziono minimum, w innym przypadku należy ten etap powtórzyć.

7. Krzepnięcie

W niniejszym rozdziale zostaną przedstawione moduły dwóch poziomów hierarchicznego układu modelu, służące do modelowania zjawisk zmiany stanu skupienia podczas przejścia materii ze stanu ciekłego do stanu stałego. Takim zjawiskiem jest krzepnięcie lub krystalizację i w przedstawionym modelu zostało ono potraktowane jako jeden z dwóch modeli drugiego poziomu, które są źródłem przestrzennej reprezentacji struktury (mikro- lub makro-). Najpierw, na podstawie automatów komórkowych, które również zostały zmodyfikowane, zostaną przedstawione zasady modelowania zjawiska krzepnięcia. Dalej, zaczynając od podstaw, omówimy abstrakcyjny proces krzepnięcia, by następnie przejść do modelowania rzeczywistego procesu technologicznego zachodzącego w maszynie ciągłego odlewania stali (COS) z wykorzystaniem wyników obliczeń metodą elementów skończonych.

Za pomocą automatów komórkowych często modeluje się charakter rozrostu dendrytów podczas krzepnięcia. Jednak takie podejście można zastosować tylko w przypadku, kiedy mamy do czynienia z bardzo niewielką objętością lub małą liczbą dendrytów. W bieżącym rozdziale zaprezentowano inne rozwiązanie. Bazuje ono na hipotetycznym kształcie ziaren i pozwala symulować strukturę w przestrzeni o znacznie większych wymiarach.

W niniejszej pracy zastosowano określenie „makrostruktura” do określenia struktury na całym przekroju wlewka lub dużej jego części bez detalizacji wewnętrznej struktury dendrytów, aby odróżnić ją od „mikrostruktury” (struktury) uzyskiwanej w modelowaniu na poziomie jednego lub kilku dendrytów.

7.1. Model FCA do modelowania makrostruktury podczas krzepnięcia

Uniwersalny frontalny CA (rys. 4.9) charakteryzuje się zamkniętym obwodem stanów. Stany q_2 i q_3 mogą być nie tylko stanami końcowymi, ale również stanami początkowymi w kolejnym cyklu symulacji. Własności te są użyteczne podczas modelowania niektórych procesów, które zostaną omówione w kolejnych rozdziałach, jednak w symulacji kształtowania się makrostruktury podczas krzepnięcia nie znajdują za-

Początkowy stan może być tylko jeden: $Q_{in} = \{q_0\}$. Stan końcowy to $Q_f = \{q_2, q_3\}$. Alfabet zawiera „litery” lub warunki przejścia $\Sigma = \{I_0, I_1, I_2, I_3, I_4\}$.

Sens warunków przejścia jest taki, jak opisano w rozdziale 4, i przedstawia się następująco: I_0 wyznacza warunki zarodkowania. I_1 służył do synchronizacji i został przejęty z poprzednich wersji CA, powinien on ograniczyć rozpowszechnienie wpływu komórki poza jej otoczenie. W FCA z organizacją odpowiednich list i ustaloną kolejnością ich obróbki niebezpieczeństwo takiego rozpowszechnienia jest wyeliminowane, warunek I_1 nie jest więc konieczny, można go zatem w nowej wersji FCA wyłączyć. Teraz I_1 wprowadza opóźnienie i pozwala na sterowanie ruchem granicy rosnącego ziarna. Warunki I_2 i I_3 wyznaczają, czy komórka będzie się znajdowała na granicy ziarna (I_2), czy wewnątrz niego (I_3). Warunek I_4 jest funkcją warunku I_1 lub stanu q_1 w jednej z komórek w otoczeniu rozpatrywanej komórki i zmienia stan bieżącej komórki na q_4 . „Język” automatów komórkowych zawiera „słowa”, które składają się z kilku liter i oznaczają przejście komórki ze stanu początkowego Q_{in} w stan końcowy Q_f (czyli wyznaczają tor przejścia). Na przykład cztery słowa $(I_0 \vee I_4 I_1)(I_2 \vee I_3)$ w FCA oznaczają kształtowanie makrostruktury podczas krzepnięcia.

W rozdziale 3 rozpatrzono większość możliwych warunków brzegowych dla przestrzeni, w której nie zachodzą zmiany lub zmiany zachodzące nie wpływają w istotny sposób na podstawowe parametry przestrzeni.

Łączenie kilku metod w jednym modelu (na przykład automatów komórkowych z elementami skończonymi) wpływa na wybór warunków brzegowych. Nie tylko nieciągłość w przestrzeni automatów komórkowych przeszkadza w zastosowaniu periodycznych warunków brzegowych. Przy wykorzystaniu obliczonego przez MES pola temperatury występowanie dużych różnic jej wartości na granicach również uniemożliwia zamknięcia przestrzeni. Dlatego w modelu łączącym metodę elementów skończonych z automatami komórkowymi przestrzeń nie zawsze będzie periodyczna.

Musimy jednak pamiętać o zaletach zastosowania warunków periodycznych. Dzięki możliwości zamknięcia przestrzeni jej wymiary mogą być znacznie skrócone. Według niektórych źródeł obecność ponad trzech ziaren wzdłuż najkrótszego wymiaru przestrzeni jest wystarczająca do adekwatnej symulacji kinetyki zjawisk mikrostrukturalnych. Rozpatrując warunki krzepnięcia metalu w maszynie ciągłego odlewania, należy podkreślić, że strumień cieplny jest skierowany głównie na powierzchnie boczne wlewka, natomiast gradient wzdłuż wlewka jest znacznie mniejszy. Dlatego na przekroju poprzecznym nie można zastosować warunków periodycznych i to właśnie gradient temperatury jest głównym źródłem powstania struktury dendrytycznej. Natomiast różnica temperatury na dość cienkim wycinku lub reprezentatywnej objętości będzie niewielka, dlatego można ją pominąć, zastosować periodyczne warunki brzegowe i ograniczyć się do trzech, pięciu dendrytów wzdłuż najkrótszego wymiaru wycinka.

Algorytm modelowania struktury odzwierciedla dwa etapy procesu krzepnięcia, z których każdy składa się z dwóch podprocesów: zarodkowania nowych ziaren i ich rozrostu.

Pierwszy etap odpowiada początkowi krystalizacji na ściankach krystalizatora i charakteryzuje się heterogenicznym zarodkowaniem ziaren na powierzchni styku cie-

kłego metalu z przechłodzonym krystalizatorem. Proces zarodkowania trwa, dopóki miejsce na powierzchni styku pozostaje swobodne. Równocześnie następuje rozrost ziaren. Po wypełnieniu całej powierzchni zarodkowanie ustaje i dalsze modelowanie mikrostruktury przebiega wyłącznie na podstawie rozrostu ziaren. Każde nowe ziarno podczas zarodkowania otrzymuje losowo wybraną orientację krystalograficzną, która w sposób bezpośredni wpływa później na rozrost ziaren. Próbowano sterować kształtem rosnących ziaren bez uwzględnienia i z uwzględnieniem warunków chemiczno-termicznych. Izotropowy rozrost ziaren zarówno bez uwzględnienia, jak i z uwzględnieniem tych warunków doprowadził do powstania jednorodnej niedendrytycznej struktury, a swobodny rozrost skutkował kulistym kształtem rozrastających się ziaren. W warunkach izotropowego rozrostu orientacja krystalograficzna nie odgrywa żadnej roli, powstała struktura składa się wyłącznie ze słupów, które mają ten sam przekrój na całej swojej długości. Wprowadzenie warunków chemiczno-termicznych niemal nie wpływa na uzyskaną strukturę. Anizotropowy rozrost ziaren, niezależnie od ich wybranego kształtu (sześciennego, ośmiościennego, elipsoidalnego czy innego), daje mikrostrukturę zbliżoną do rzeczywistej, w której oprócz struktury słupowej widoczna jest strefa ziaren równoosiowych przy powierzchni wlewka. Spośród możliwych wariantów kształtu rosnących ziaren wybrano kształt ośmiościenny, jako odpowiadający siatce krystalicznej i kształtowi kryształu.

Drugi etap zarodkowania następuje pod koniec krystalizacji. Wówczas w przestrzeni ciekłej fazy metalu następuje zarodkowanie homogeniczne. Rozrost tych ziaren prowadzi do powstania strefy równoosiowych ziaren przy osi wlewka.

Opracowane dla tego procesu algorytmy i programy uwzględniają pole temperatury, skład chemiczny i warunki mieszania elektromagnetycznego. Przeprowadzono testowe symulacje z narzuconymi warunkami.

7.2. Założenia wstępne

Dokładne modelowanie kształtowania makrostruktury podczas krzepnięcia jest skomplikowanym zadaniem, którego prawidłowe wykonanie zależy od wyboru odpowiedniej metody. Wybór ten zależy z kolei od oczekiwanego wyniku, jego dokładności i nakładów obliczeniowych. Główną przewagą metod opartych na CA jest możliwość dokładniejszej symulacji mikrostruktury zarówno w skalach mikro i mezo, jak i w skali makro. Połączenie metody automatów komórkowych z metodą elementów skończonych zwiększa dokładność symulacji złożonych zjawisk zachodzących w materiałach.

Wiele wysiłku włożono w modelowanie charakteru rozrostu dendrytów podczas krzepnięcia za pomocą automatów komórkowych. Do modelowania rozrostu dendrytów stosowano dwa główne typy CA. Pierwszy typ bazuje na numerycznym rozwiązaniu rozrostu dendrytów lub na numerycznym rozwiązaniu równania transportu. Jednak zastosowanie takich rozwiązań jest możliwe tylko w przypadku bardzo małej modelowej przestrzeni lub małej liczby dendrytów. W niniejszej pracy wykorzystano inne

podejście, bazujące na hipotetycznym kształcie ziaren. Rozpatrywano przy tym zmniejszenie anizotropii przestrzeni automatów i anizotropowego rozrostu. Zakładając, że wewnętrzna struktura każdego dendrytu może być uzyskana innymi metodami, główny wysiłek skierowano na kształtowanie makrostruktury, która może być otrzymana w przestrzeni o stosunkowo dużych wymiarach. Tak więc zjawisko krzepnięcia symulowano z wyznaczonym kształtem dendrytów. Takie symulacje można przeprowadzić wyłącznie w ramach opracowanego modelu.

Wykorzystanie CA w połączeniu z innymi metodami pozwala uzyskać bardziej dokładne i wiarygodne wyniki. Często łączono CA z metodą różnic skończonych lub elementów skończonych (MES). Schematy połączenia metod CA i MES można ze względu na ich rolę w procesie modelowania podzielić na dwie grupy. Pierwsza grupa zawiera rozwiązania, w których symulacje są prowadzone niezależnie, bez sprzężenia zwrotnego. Układy ze ścisłą współpracą mogą być zaliczone do drugiej grupy. Często jeden z ich komponentów jest pierwotny, inny zaś, który służy do dostarczenia parametrów i zmiennych potrzebnych do prawidłowego modelowania metodą pierwotną – wtórny. W przypadku gdy CA mają prostą strukturę i nie wymagają wielu nakładów obliczeniowych, czyli pamięci i czasu obliczeń, mogą stanowić metodę wtórną. Trójwymiarowe automaty komórkowe do modelowania rozwoju makrostruktury, ze względu na wymagania obliczeniowe, nie mogą być stosowane jako metoda wtórna. Takie CA odgrywają rolę pierwotnej metody w badaniach teoretycznych, z nierównomiernym odkształceniem na poziomie ziaren lub struktury polikrystalicznej.

Jednak w kształtowaniu makrostruktury CA pełnią ważną rolę, mogą bowiem działać równolegle do MES lub jako dodatkowe, post-processingowe moduły.

Na potrzeby modelowania rzeczywistego procesu technologicznego w skali makro połączono metodę CA z metodą elementów skończonych. Wybrano rozwiązanie niesprężone symulacji metodami MES i CA. Proces krzepnięcia był modelowany za pomocą MES w skali makro, a wyniki symulacji MES zostały wykorzystane w automatach komórkowych. CA wykorzystano w schemacie post-processingu, w którym na podstawie odpowiedniej informacji modelowano kształtowanie makrostruktury. W takim schemacie może być wykorzystany dowolny kod MES. Wystarczy spełnić wymagania co do sposobu przekazywania danych do obliczeń metodą CA.

Głównym celem dopasowania modelu było uzyskanie jak najprostszego rozwiązania bez wprowadzenia sztucznych warunków pozwalających na uzyskanie prawdopodobnej struktury wlewka. Dlatego ograniczono się do trzech podstawowych założeń:

- 1) głównym czynnikiem nierównomiernego rozrostu kryształów jest ich kształt, ograniczony przez krystaliczną strukturę metalu;
- 2) wielkość dendrytów i kryształów równoosiowych zależy głównie od warunków zarodkowania i rozrostu, zarówno na początku procesu, jak i podczas jego trwania;
- 3) podstawowym czynnikiem wpływającym na ruch frontu krystalizacji jest strumień ciepła.

Symulację rozwoju makrostruktury i jej walidację przeprowadzono w kilku etapach:

1. Wstępne modelowanie struktury krystalicznej na równomiernie chłodzonej ścian-ce formy – realizacja pierwszego założenia (podrozdz. 7.3).
2. Wprowadzenie do modelu trzeciego założenia podstawowego, jakim jest uwzględnienie rzeczywistej temperatury wlewka (podrozdz. 7.4).
3. Opracowanie oprogramowania (podrozdz. 7.5).
4. Modelowanie makrostruktury, walidacja modelu, oszacowanie parametrów zarodkowania – realizacja drugiego założenia (podrozdz. 7.6).
5. Modyfikacja modelu i oprogramowania (podrozdz. 7.7).
6. Weryfikacja modelu i jego optymalizacja ze względu na nakłady obliczeniowe (podrozdz. 7.8).

Przewiduje się, że kolejnym etapem będzie połączenie w modelu metody automatów komórkowych i metody elementów skończonych.

7.3. Modelowanie zjawiska zmiany stanu skupienia materiału (krystalizacja)

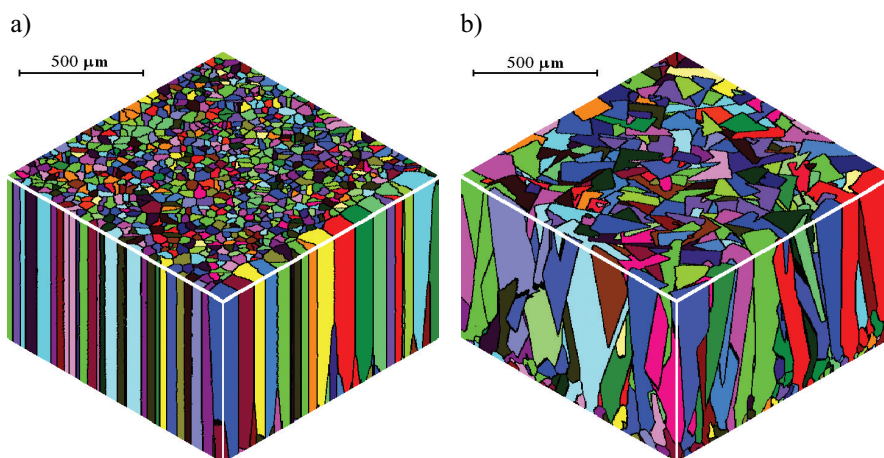
Główną różnicą między opisanym w rozdziale 6 kształtowaniem początkowej mikrostruktury a krystalizacją ze względu na przebieg procesu jest lokalizacja zarodków. Nie wchodząc w tej chwili w szczegóły różnic między zarodkowaniem homo- a heterogenicznym, można tylko zasygnalizować podstawowy skutek tej różnicy. Zarodkowanie na chłodniejszej powierzchni jest zdecydowanie bardziej pod względem energetycznym preferowane, wobec tego na początku krystalizacji można rozpatrywać tylko ten przypadek. Zarodkowanie w algorytmie kształtowania początkowej struktury można wybierać losowo w całej objętości materiału, natomiast w przypadku krystalizacji – wyłącznie na powierzchni wlewka (z wyjątkiem ostatniego etapu).

Przy kształtowaniu początkowej mikrostruktury stosowano przeważnie okrągły kształt ziaren lub ziarna o innym kształcie, ale zorientowane konforemnie, jednak pierwsze próby symulacji krzepnięcia wykazały niecelowość takiego podejścia. Główną tego przyczyną była izotropowość rosnących ziaren lub jednorodność anizotropii.

Opracowane i opisane w rozdziale 2 próby modelowania krzepnięcia, po uzyskaniu globalnej izotropii modelowej przestrzeni i wprowadzeniu lokalnej anizotropii rozrostu ziaren, okazały się bardzo skuteczne. Można jednoznacznie stwierdzić, że w klasycznych CA, z anizotropią przestrzeni wprowadzoną przez powszechnie stosowane sąsiedztwa, realistyczne modelowanie krzepnięcia jest niemożliwe, wymaga stosowania nieuzasadnionych podejść lub olbrzymich nakładów obliczeniowych.

Dendrytyczna struktura metalu po krzepnięciu wykazuje dużą niejednorodność, którą można wytłumaczyć różnymi prędkościami rozrostu. Wprowadzenie różnych prędkości rozrostu poszczególnych okrągłych ziaren trudno uzasadnić, więc próbowano zamodelować krzepnięcie z ziarnami mającymi ten sam kształt, rosnącymi z tą samą prędkością, ale zorientowanymi w różny sposób. Takie podejście jest często stosowane do teoretycznego uzasadnienia różnej prędkości rozrostu ziaren, wyjaśniającego, dlaczego preferowany rozrost ma miejsce tylko w przypadku niewielu ziaren, a pozostałe przestają rosnąć ze względu na ograniczenie przestrzeni ich rozrostu.

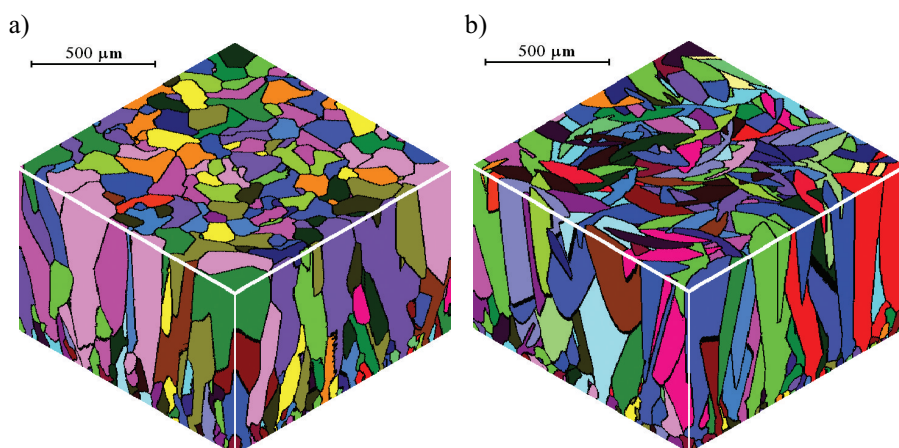
Na początku zbadano więc wpływ kształtu rosnących kryształów na uzyskaną makrostrukturę. Swobodny rozrost ziaren o wybranych kształtach przedstawiono w rozdziale 2. Modelowano ziarna o następujących kształtach: kule, elipsoidy, walce (również w postaci nici oraz dysków), sześciany i ośmiościany foremne oraz równoległosciany (w tym wydłużone płytki). Im więcej komórek znajdowało się w modelowej przestrzeni, tym lepszą można było uzyskać rozdzielczość i tym bardziej izotropowa była przestrzeń. Jednak zawsze pozostaje pytanie, w jaki sposób można efektywniej zastosować ograniczoną przestrzeń. Czy należy zwiększyć rozdzielczość, czy też rozmiary modelowanej objętości? Trzeba więc również wyznaczyć dopuszczalną rozdzielczość, czyli maksymalne dopuszczalne wymiary pojedynczej komórki. Do kształtowania makrostruktury z zarodkowaniem na jednej powierzchni przestrzeni wypróbowano kilka kształtów w celu sprawdzenia wpływu kształtu na końcowy wynik. Niektóre wyniki przedstawiono na rysunkach 7.2–7.4.



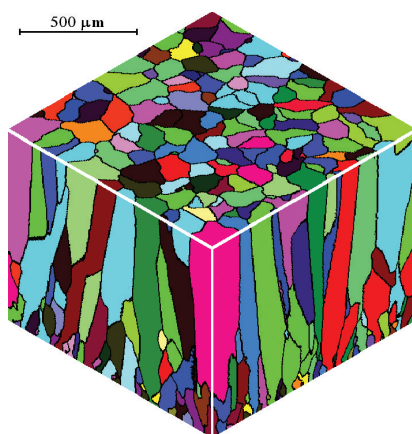
Rys. 7.2. Końcowa struktura kryształów w kształcie kul (a) i prostopadłościaków (b) (Svyetlichnyy 2011)

W przypadku rosnących ziaren w kształcie kuli z uwagi na ich symetrię żaden kierunek rozrostu nie jest uprzywilejowany, dlatego wszystkie kryształy rosną w ten sam sposób i z tą samą prędkością. Z tego powodu liczba rosnących ziaren pozostaje bez zmian w ciągu całego procesu kształtowania struktury. Powstała w ten sposób struktura, która cechuje się granicami równoległymi do kierunku rozrostu, została przedstawiona na rysunku 7.2a. W przeprowadzonych symulacjach zadano półotwarte warunki brzegowe i właśnie efekt brzegowy jest łatwo zauważalny w przypadku modelowania kulistych ziaren. Wszystkie brzegowe ziarna są większe od ziaren znajdujących się w środkowej części przestrzeni. W szczególności jest to widoczne na przedniej powierzchni bocznej przestrzeni, dlatego że obliczenia rozpoczynano właśnie od zarodkowania w tym miejscu, natomiast kolejność obliczeń wzmacniała efekt brzegowy. W przypadku pozostałych wynikowych struktur efekt brzegowy jest mniej widoczny, jednak nadal pozostaje obecny.

Wprowadzenie dowolnej asymetrii w rozroście ziaren prowadzi do tego, że niektóre z nich rosną szybciej niż inne. Zależy to głównie od krystalograficznej orientacji ziaren, którą wybiera się losowo dla każdego zarodka. W wyniku anizotropowego rozrostu ziaren przewagę zyskują ziarna, których krystalograficzna orientacja jest najbardziej zbliżona do kierunku rozrostu. Ziarna mające inną orientację, mniej korzystną ze względu na rozrost, zostają stopniowo opóźnione w rozwoju, tracą swobodną przestrzeń do dalszego rozrostu i w efekcie przestają rosnąć. Na skutek tego liczba rosnących ziaren w procesie krzepnięcia zmniejsza się. Ponadto kształt rosnących kryształów wpływa na końcową strukturę. Te różnice (rys. 7.2–7.4) są widoczne zarówno w przekroju wzdłużnym do kierunku rozrostu, jak i w przekroju poprzecznym. Bardzo istotnym czynnikiem jest również kształt rosnących ziaren – od jego wyboru zależy prawidłowość uzyskanej końcowej struktury.

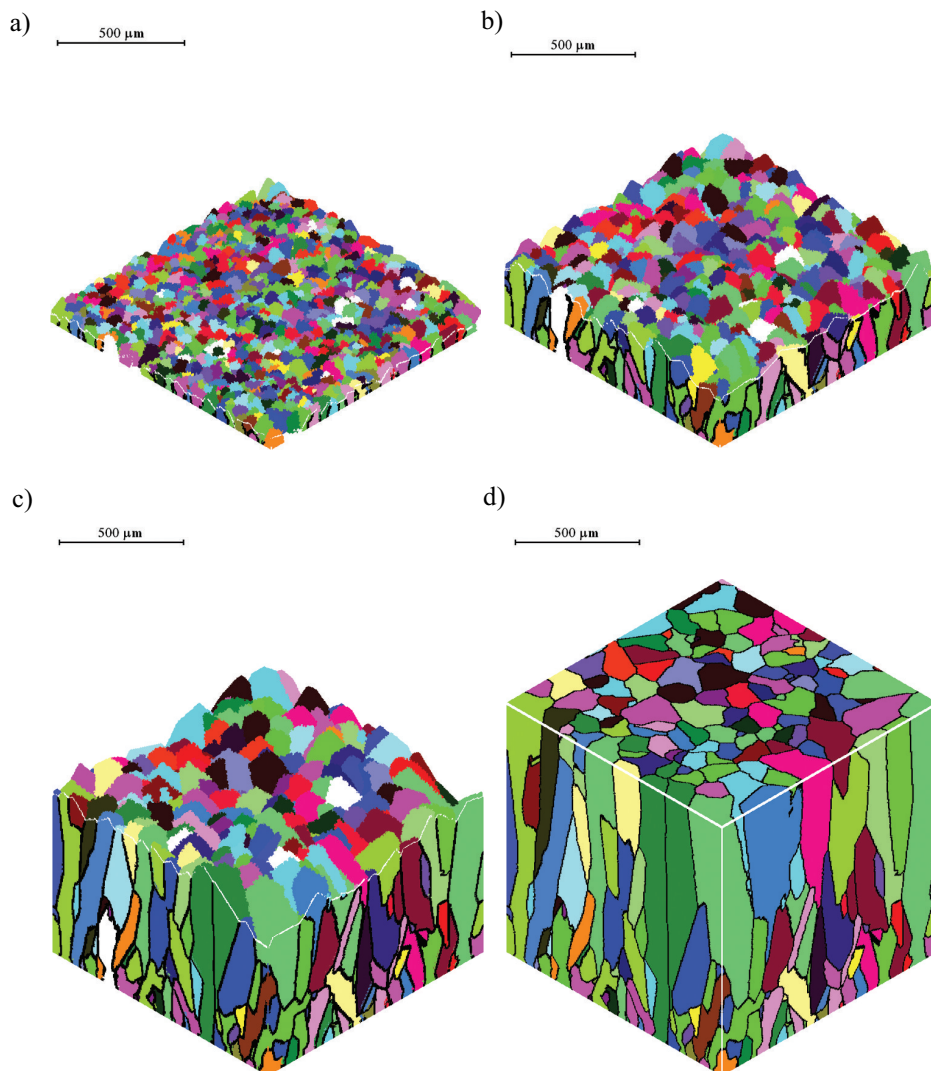


Rys. 7.3. Końcowa struktura zbudowana z kryształów w kształcie sześcianów (a) i walców (b)



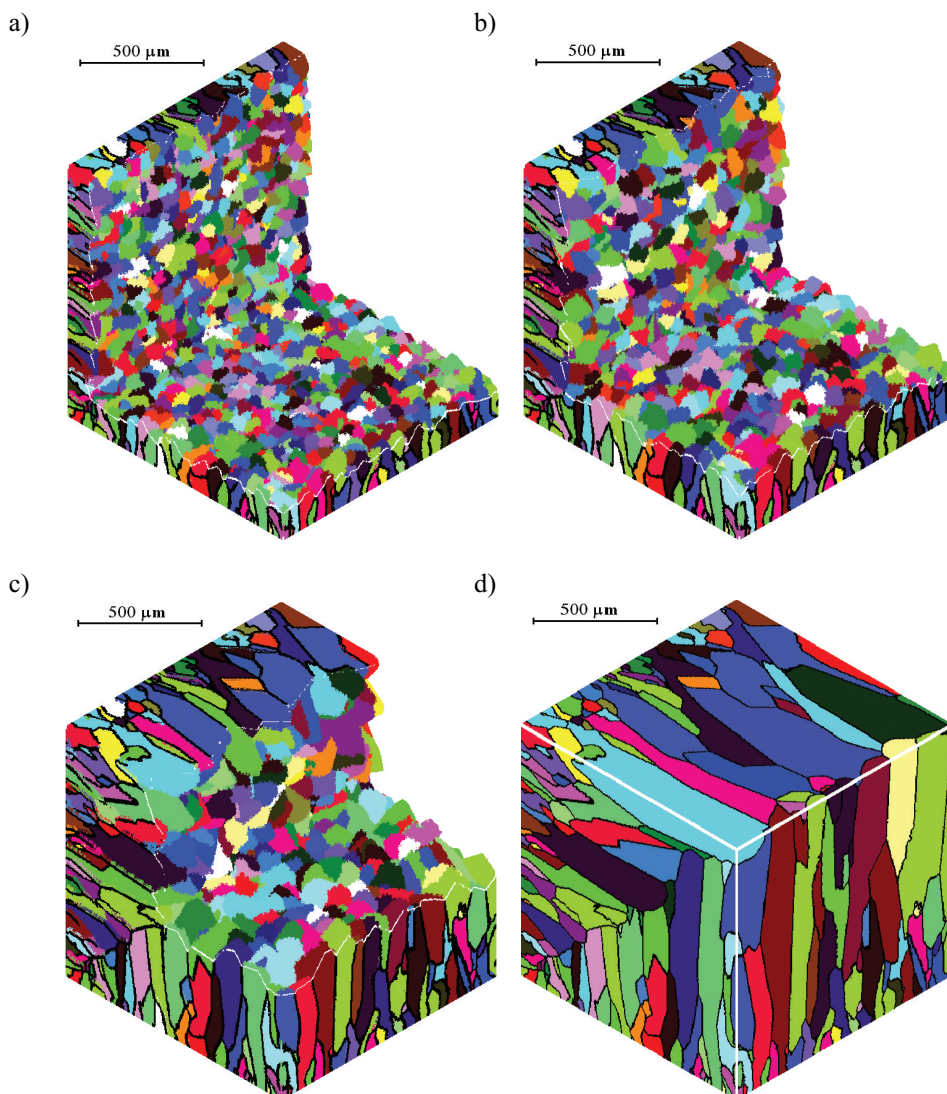
Rys. 7.4. Końcowa struktura zbudowana z kryształów w kształcie ośmiościanów foremnych (Svyetlichnyy 2011)

Porównanie końcowych struktur z makrostrukturą odlewanej stali wykazało ich największą zgodność ze strukturą, która powstała w wyniku modelowania z kryształami w kształcie ośmiościanu. Zgodnie z krystaliczną strukturą austenitu ośmiościan jest najbardziej odpowiednim kształtem swobodnie rosnącego kryształu. Potwierdza to słuszność pierwszego z przyjętych założeń podstawowych. Dlatego pozostałe symulacje dotyczyły wyłącznie rozrostu kryształów w kształcie ośmiościanów. Na rysunku 7.5 pokazano niektóre etapy procesu krzepnięcia struktury utworzonej z ziaren o tym kształcie.



Rys. 7.5. Kształtowanie makrostruktury zbudowanej z kryształów w kształcie ośmiościanów foremnych (Svyetlichnyy 2011)

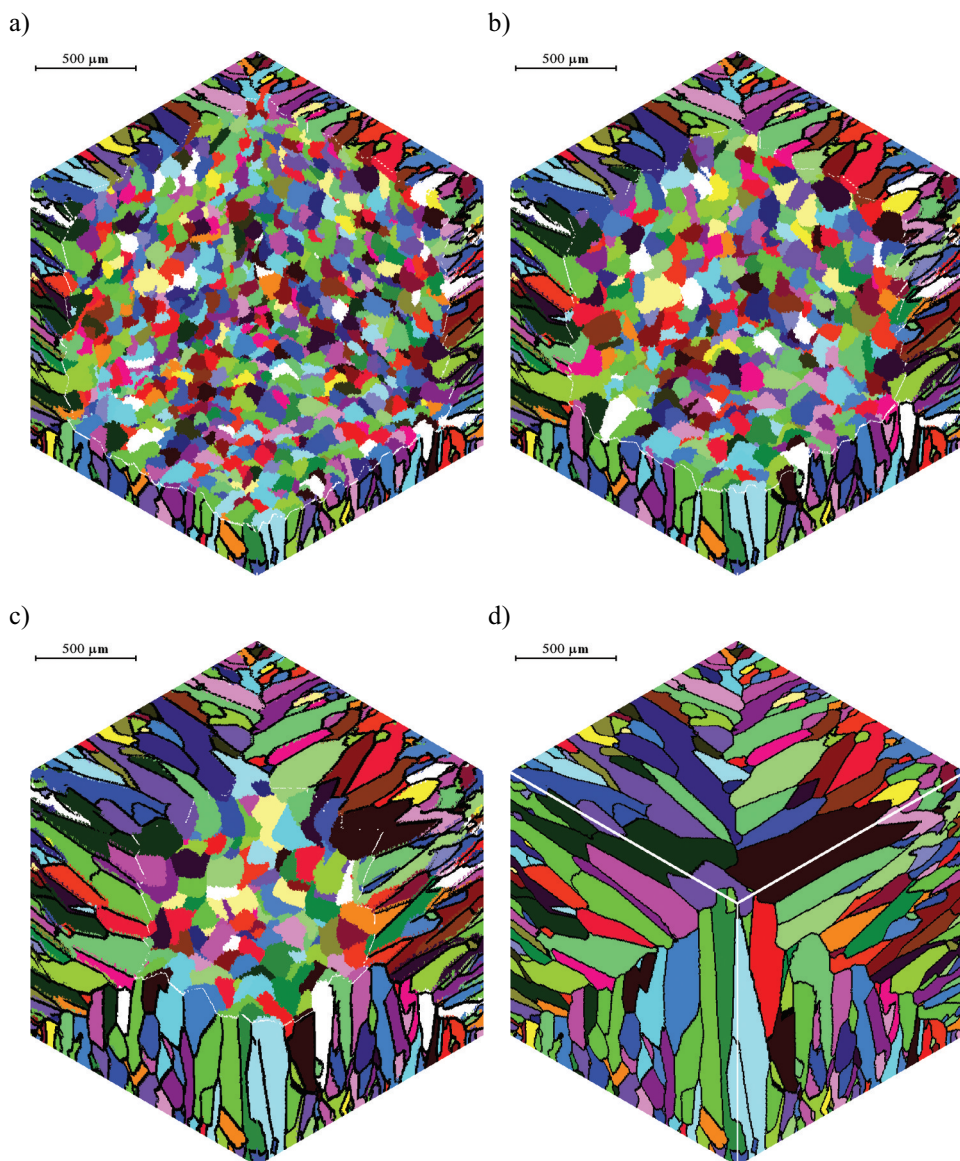
Warianty kształtowania makrostruktury, w sytuacji, gdy zarodkowanie następuje na dwóch i trzech powierzchniach sześcienniej przestrzeni komórek, pokazano odpowiednio na rysunkach 7.6 i 7.7.



Rys. 7.6. Kształtowanie makrostruktury podczas krzepnięcia z zarodkowaniem na dwóch ścianach (Svyetlichnyy 2011)

Pierwsze symulacje przeprowadzono bez uwzględnienia rzeczywistych warunków krzepnięcia. Zakładano, że powierzchnia lub front krzepnięcia jest równoległa do powierzchni, na której zachodzi zarodkowanie, lub prostopadła do kierunku rozrostu,

czyli prędkość rozrostu jest stała w dowolnym punkcie przestrzeni (w tych symulacjach nie wprowadzono żadnej zależności prędkości rozrostu od temperatury). Kolejnym etapem modelowania kształtowania mikrostruktury powinno być zatem połączenie metod CA i MES oraz wprowadzenie zależności rozrostu ziaren od temperatury. Wprowadzeniu odpowiednich zależności będzie poświęcony następny podrozdział pracy.



Rys. 7.7. Kształtowanie mikrostruktury podczas krzepnięcia z zarodkowaniem na trzech ścianach (Svyetlichnyy 2011)

7.4. Uwzględnienie pola temperatury podczas modelowania krzepnięcia

W rzeczywistym procesie krzepnięcia lokalizacja frontu krystalizacji określona jest rozkładem temperatury w metalu. Dlatego wyniki obliczeń MES wykorzystuje się w symulacji kształtowania makrostruktury podczas procesu krzepnięcia.

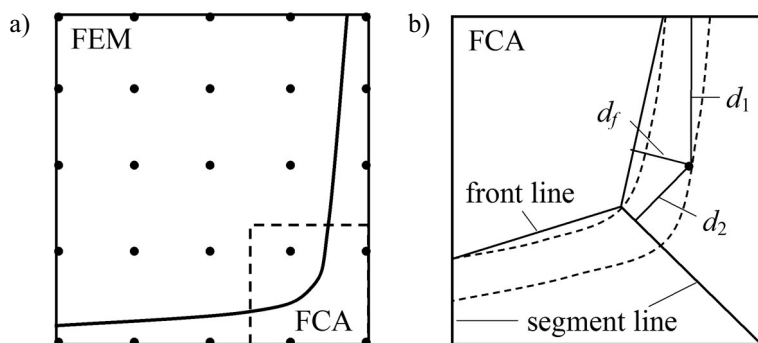
Jak wspomniano wyżej, zastosowano wariant niezależnych obliczeń MES i CA bez sprzężenia zwrotnego. Proces został zamodelowany programem MES (Hadała i Malinowski 2009) opracowanym przez prof. Z. Malinowskiego. Za pomocą tego programu można uzyskać rozwiązanie w skali makro. Rozwiązaniem MES jest trójwymiarowy stacjonarny rozkład temperatury we wlewkę poruszającym się ze stałą prędkością, z uwzględnieniem konstrukcyjnych i technologicznych warunków.

Jako interfejs między programem MES a programem CA wykorzystywano dwa zewnętrzne pliki. Pierwszy z nich zawierał informacje o podstawowych parametrach modelowanego procesu, takich jak kształt i rozmiary wlewka, liczba elementów, współrzędne węzłów, temperatury solidus i likwidus, prędkość odlewania i inne. W drugim pliku zawarto informacje o rozkładzie temperatury we wlewkę.

Rozkład temperatury powinien być dopasowany do przestrzeni automatów komórkowych. Chociaż rozwiązaniem MES jest stacjonarny rozkład temperatury, FCA rozpatruje setki lokalizacji poruszającego się przekroju poprzecznego wlewka w linii ciągłego odlewania. Zamiast położenia przekroju wlewka FCA wykorzystuje czas, w którym przekrój znajduje się w tym miejscu z uwzględnieniem prędkości odlewania.

Każda lokalizacja przekroju poprzecznego jest sprawdzana przed obliczeniami FCA. Symulacja kształtowania makrostruktury rozpoczyna się, kiedy temperatura przynajmniej w jednym węźle MES, który jednocześnie należy do przestrzeni komórkowej FCA, spadnie poniżej temperatury solidus. Kiedy temperatura spadnie poniżej temperatury solidus w całej przestrzeni komórkowej, symulacja dobiega końca. Symulację prowadzi się od jednego przekroju do kolejnego. Ułatwia to uwzględnienie w obliczeniach czasu rzeczywistego, ponieważ położenie przekroju jest jednoznacznie związane z czasem, natomiast krok obliczeniowy wykorzystuje czas względny, a nie bezwzględny.

Najpierw, przed obliczeniami FCA dotyczącymi aktualnie rozpatrywanej lokalizacji, należy przekształcić dyskretny rozkład temperatury w dwie ciągłe linie, które będą reprezentować izotermy likwidus i solidus. Przekrój poprzeczny wlewka, węzły MES zawierające wartości temperatury, przestrzeń FCA oraz izotermy przedstawiono na rysunku 7.8a. Po uzyskaniu izoterm na przekroju poprzecznym przenosi się je do przestrzeni komórkowej FCA. Na rysunku 7.8b uwidoczniło dalsze obliczenia dla jednej linii. Linie przerywane reprezentują nową lokalizację izoterm solidus przeniesionej z MES do FCA i front krystalizacji otrzymany w poprzednim kroku przy poprzedniej lokalizacji izoterm. Nowa izoterma jest zastąpiona kilkoma (do ośmiu) odcinkami prostej, które na rysunku zaznaczone są jako *front line*. Dalej przestrzeń komórkową dzielono na segmenty przylegające do odpowiedniej linii frontalnej. *Segment line* są granicami, które oddzielają jeden segment od drugiego.



Rys. 7.8. Połączenie metod MES i FCA (a) oraz schemat obliczeń współczynnika prędkości rozrostu ziarna (b) (Svyetlichnyy 2011)

Segmentację przestrzeni komórkowej stosuje się w celu dyferencjacji prędkości rozrostu w różnych strefach przestrzeni, uwzględniając rzeczywiste warunki krzepnięcia i wprowadzając w niejajny sposób zależność prędkości rozrostu od temperatury.

Prędkością krystalizacji steruje się przez wprowadzenie współczynnika prędkości rozrostu, który zależy od lokalizacji komórki w odpowiednim segmencie. Współczynnik ten nie jest stały wewnątrz jednego segmentu, natomiast jest funkcją liniową względnej współrzędnej x . Współrzedną x wyznacza się ze wzoru: $x = d_1/(d_1 + d_2)$ i jej wartość mieści się w zakresie od 0 do 1. W tym wzorze d_1 i d_2 są odległościami komórki od odpowiednich linii segmentowych.

Ponieważ front krystalizacji powinien dotrzeć do wyznaczonej przez frontalne linie nowej lokalizacji jednocześnie na całej długości, prędkość rozrostu powinna zależeć od początkowej odległości frontu krystalizacji od linii frontalnej. Zatem dla każdej komórki na froncie krystalizacji (w stanie q_4) oblicza się trzy odległości: d_f (od linii frontalnej) oraz d_1 i d_2 (od linii segmentowych). Otrzymujemy wówczas zależność odległości d_f od względnej współrzędnej x dla każdego segmentu. W ogólnym przypadku dla całej przestrzeni komórkowej otrzymana zależność jest funkcją nieciągłą na granicach segmentów i nieciągłość ta powinna być wyeliminowana. Poza tym zależność jest normalizowana względem maksymalnej odległości od *front line* d_f , zostało także wprowadzone ograniczenie minimalnej wartości. W wyniku odpowiednich przekształceń otrzymywany jest współczynnik prędkości rozrostu ziaren dla każdej komórki na froncie krystalizacji. Nowe komórki włączane do procesu krystalizacji kopiują od sąsiedniej komórki numer segmentu, oblicza się ich lokalne (względne) współrzedne i czas opóźnienia stanu q_4 z uwzględnieniem odpowiedniego do lokalizacji komórki współczynnika prędkości rozrostu.

Po dotarciu frontu krystalizacji do frontalnej linii aktualnie rozpatrywanego przekroju poprzecznego obliczenia w tym kroku się kończą. Następnie wybiera się nową lokalizację i obliczenia są powtarzane w nowym kroku.

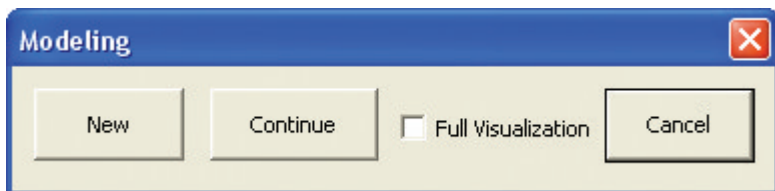
7.5. Opis modułu FCA

Opracowany został program, który jest wciąż modyfikowany i może nieco odbiegać od opisanego poniżej.

Po wyborze opcji modelowania makrostruktury podczas krzepnięcia w maszynie COS (rys. 6.1) uruchomiany jest odpowiedni moduł FCA. Użytkownik może wybrać jedną z dwóch opcji dalszej pracy z modułem (rys. 7.9). Możliwe jest modelowanie procesu kształtowania makrostruktury lub przegląd uzyskanych wyników. W przypadku gdy zostanie wybrane modelowanie procesu, informacja o procesie będzie zapisywana w kilku plikach. Pierwszy z nich zawiera podstawowe informacje: typ modelowanego procesu, wymiary przestrzeni komórkowej (zarówno w komórkach, jak i w jednostkach długości), początkową liczbę zarodków, numer modelowanej części wlewka i inne. Drugi plik zawiera początkową informację o kryształach: numer, krystalograficzną orientację i lokalizację zarodka. Kolejne pliki zawierają listy komórek, które przeszły ze stanu likwidus w stan solidus. Są to listy współrzędnych (w komórkach) i numery kryształów, do których przyłączana jest komórka, zapisywana w kolejności przechodzenia komórek w stan końcowy. Takie pliki pozwalają zaprezentować cały zamodelowany proces lub kontynuować przerwane obliczenia (rys. 7.10).



Rys. 7.9. Wybór modelowania lub przeglądu wyników

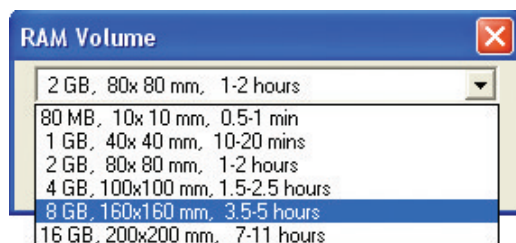


Rys. 7.10. Menu do wyboru modelowania nowego procesu lub kontynuacji przerwanych obliczeń

Wybór opcji „Modeling” daje dwie możliwości: nowe modelowanie („New”) lub kontynuację przerwanych obliczeń („Continue”) (rys. 7.10). Nowe modelowanie może być rozpoczęte w dowolnej chwili, a przerwanie obliczenia można kontynuować wówczas, gdy przynajmniej jeden proces obliczeń został uruchomiony, niezależnie do tego, czy uruchomione procesy zostały skończone, czy nie. Faktycznie informacja zachowana

w plikach jest zdjęciem migawkowym ostatniego stanu obliczeń, wykonanym w momencie, kiedy zostały one przerwane i ten stan obliczeń może być wznowiony na podstawie danych z owego ostatniego zachowanego punktu. Jednocześnie z zapisywaniem informacji w plikach, wyniki obliczeń mogą być wyświetlane na displayu, co będzie stanowić wizualizację całych obliczeń. Tę opcję można wybrać, zaznaczając odpowiednie pole. Jednak pełna wizualizacja wielokrotnie wydłuża obliczenia i nie jest zalecana. Jeżeli ta opcja nie została zaznaczona, wizualizacja podczas obliczeń ograniczona jest do kilku – kilkunastu (maks. 20) zdjęć, które również są zapisywane w odrębnych plikach. Po wykonaniu modelowania końcowa makrostruktura uzyskana w procesie symulacji krzepnięcia może być wykorzystana w innych FCA do dalszego modelowania.

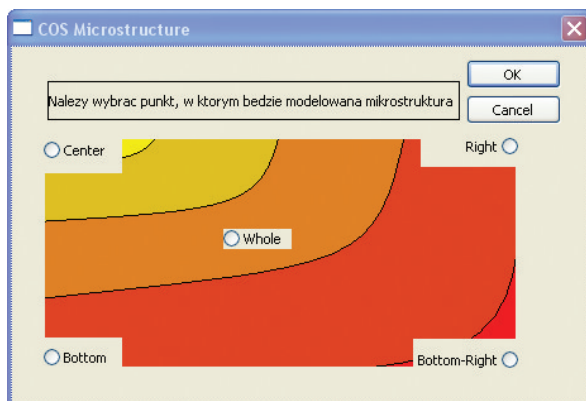
Użytkownik może w pewnym stopniu sterować modelowaniem w zależności od możliwości komputera (głównie pamięci RAM) lub od innych czynników. Ponieważ istnieje ograniczenie dotyczące wielkości komórki, które jest związane z rozdzielczością, pojemność pamięci RAM wyznacza wymiar modelowanej przestrzeni. Nie zawsze można zamodelować cały przekrój – w niektórych przypadkach tylko jego część może być wybrana do obliczeń. Na rysunku 7.11 pokazano menu do wyboru pamięci („RAM Volume”). Przy założeniu, że sześcienna komórka ma długość krawędzi 0,015 mm, program pozwala na obliczenie kształtowania makrostruktury w przestrzeni o kształcie prostopadłościanu o wymiarach od $3,5 \times 3,5 \times 0,875$ mm (1 GB RAM) do $15,0 \times 15,0 \times 1,8$ mm (16 GB RAM). FCA w takich obliczeniach wykorzystuje od 3,5 do 115 milionów komórek. Odpowiednio czas obliczeń zmienia się w szerokim zakresie – od dwóch minut do 12 godzin.



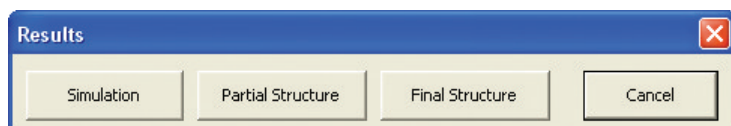
Rys. 7.11. Menu do wyboru możliwej pamięci

W przypadku gdy niemożliwe jest modelowanie całego przekroju za pomocą FCA, użytkownik powinien ograniczyć modelowany obszar, dokonując wyboru (rys. 7.12) spośród czterech możliwości: prawa lub dolna powierzchnia, prawa dolna krawędź lub środek wlewka. Przykładowe obliczenia odnoszące się do odpowiedniej części wlewka przedstawiono w podrozdziale 7.6.

Po wykonaniu obliczeń użytkownik może zobaczyć wyniki modelowania (wybór opcji „Results” – rys. 7.9). Wówczas wybierając opcję „Simulation” (rys. 7.13), można obserwować symulację całego procesu krzepnięcia.



Rys. 7.12. Menu do wyboru części wlewka, która będzie modelowana



Rys. 7.13. Menu do demonstracji wyników

Wizualizacja tego procesu polega na sczytywaniu informacji z plików i przenoszeniu ich na ekran bez powtarzania obliczeń. Taka wizualizacja może jednak trwać nawet dłużej niż same obliczenia (dlatego też zaleca się wyłączać pełną wizualizację podczas obliczeń). Inną możliwość oglądania wyników daje wyświetlenie pełnej makrostruktury („Final Structure”) lub jej fragmentu („Partial Structure”). Jak już wspomniano, podczas modelowania (w tym bez pełnej wizualizacji) w równych odstępach wyświetla się bieżący stan procesu krzepnięcia i jednocześnie jest on zapisywany do odpowiednich plików. Należy tylko dokonać wyboru odpowiedniego pliku z wynikami obliczeń, wpisując numer zrzutu ekranu w otwierającym się okienku. Każda makrostruktura jest prezentowana jako rysunek izometryczny. Końcowa makrostruktura prezentowana jest na trzech powierzchniach bocznych prostopadłościanu przestrzeni komórkowej lub na wybranym przekroju. Na rysunkach jest pokazany również front krystalizacji.

7.6. Wyniki modelowania

7.6.1. Modelowanie wlewków za pomocą FCA o dużej liczbie komórek

Maksymalne wymiary przestrzeni, którą można modelować za pomocą przedstawionego automatu komórkowego, wynosiły około $15,0 \times 15,0 \times 1,8$ mm (przy liniowych wymiarach komórki $0,015 \times 0,015 \times 0,015$ mm i wykorzystaniu 16 GB RAM).

Gdy wymiary wlewka są większe, tylko jego część może być zamodelowana. Zaokrąglona krawędź również może być uwzględniona przy projektowaniu przestrzeni komórkowej. W niżej omówionych przypadkach, żeby zaprezentować możliwości FCA, próbowano modelować wlewek o dużych wymiarach.

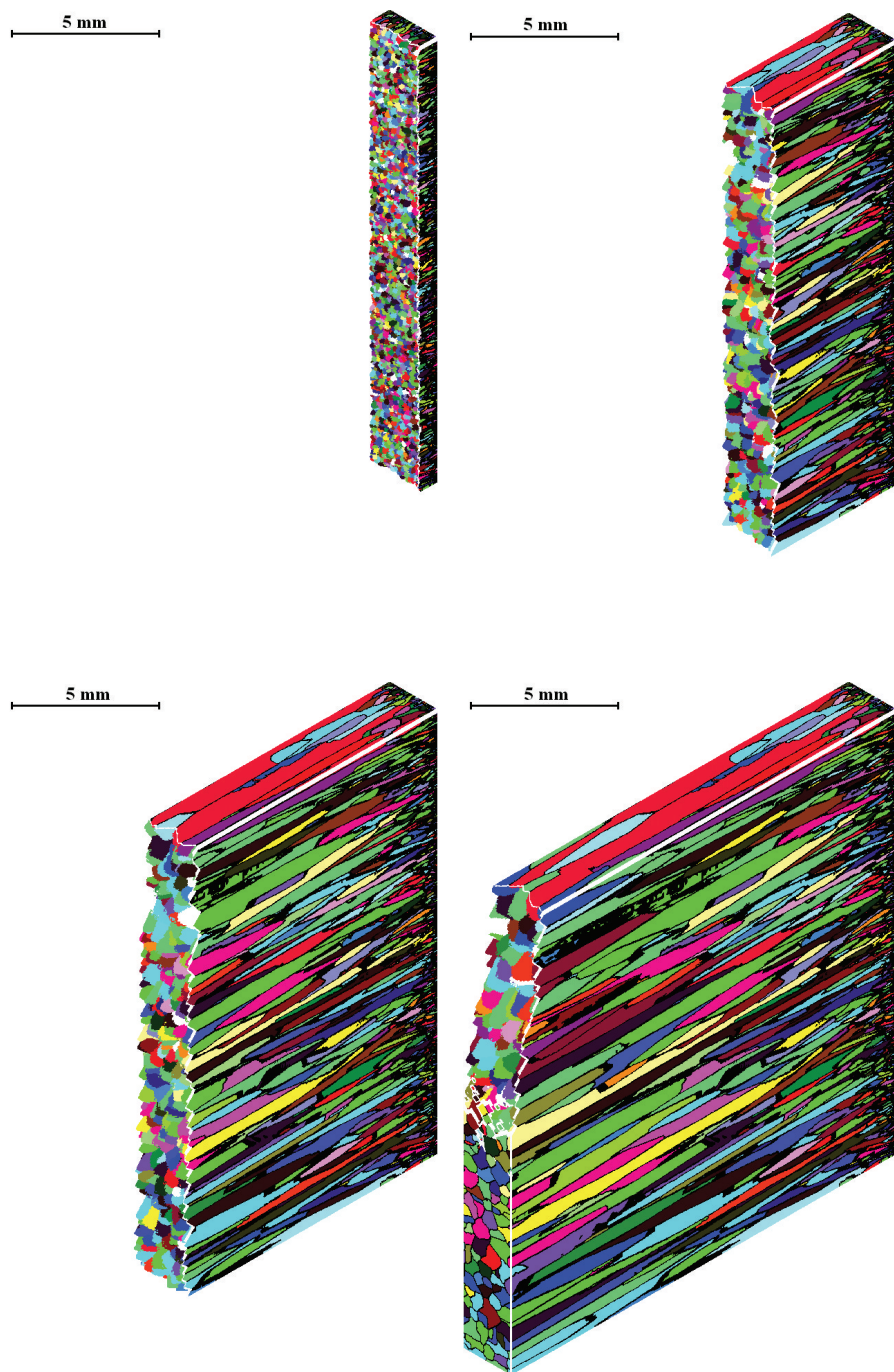
Modelowano proces ciągłego odlewania wlewka o przekroju 100×100 mm, promień zaokrąglenia wynosił 5 mm. Rozwiązanie MES uzyskano dla ćwiartki przekroju na całej długości linii odlewania. Całe rozwiązanie ma 313 przekrojów wzdłuż wlewka z 17×17 węzłami na każdym przekroju. Nie wszystkie przekroje zostały wykorzystane przez FCA, niektóre z pierwszych mają temperaturę powyżej temperatury solidus, natomiast kilka ostatnich ma temperaturę poniżej temperatury solidus w całym przekroju. Wymiary komórki wynoszą $0,015 \times 0,015 \times 0,015$ mm. Ze względu na wymiary wlewka przy takich założeniach rozdzielczości tylko jego część może być modelowana za pomocą FCA. Wówczas są do wyboru cztery opcje: prawa lub dolna część, prawa dolna krawędź i środek wlewka.

Na rysunku 7.14 pokazano cztery stadia kształtowania makrostruktury podczas modelowania prawej strony wlewka. W takim przypadku heterogeniczne zarodkowanie następuje tylko na jednej powierzchni przestrzeni automatów komórkowych. Wybiera się maksymalną gęstość zarodkowania, która wynosi jedną komórkę na dziesięć. Po wykonaniu pierwszego kroku obliczeniowego drobne ziarna pokrywają prawie 90% bocznej powierzchni. Dlatego zwiększenie liczby zarodków nie powinno znacząco wpłynąć na końcowy wynik. Krystalograficzną orientację zarodków wybiera się losowo. Konkurencja pomiędzy rosnącymi kryształami prowadzi do uprzywilejowanego rozrostu dendrytów mających krystalograficzną orientację zbliżoną do kierunku rozrostu. Liczba rosnących dendrytów zmniejsza się, a ich wymiary poprzeczne wzrastają. Front krystalizacji na początku jest równoległy do powierzchni wlewka, a później stopniowo staje się nierównoległy do bocznej powierzchni, zgodnie z linią solidus.

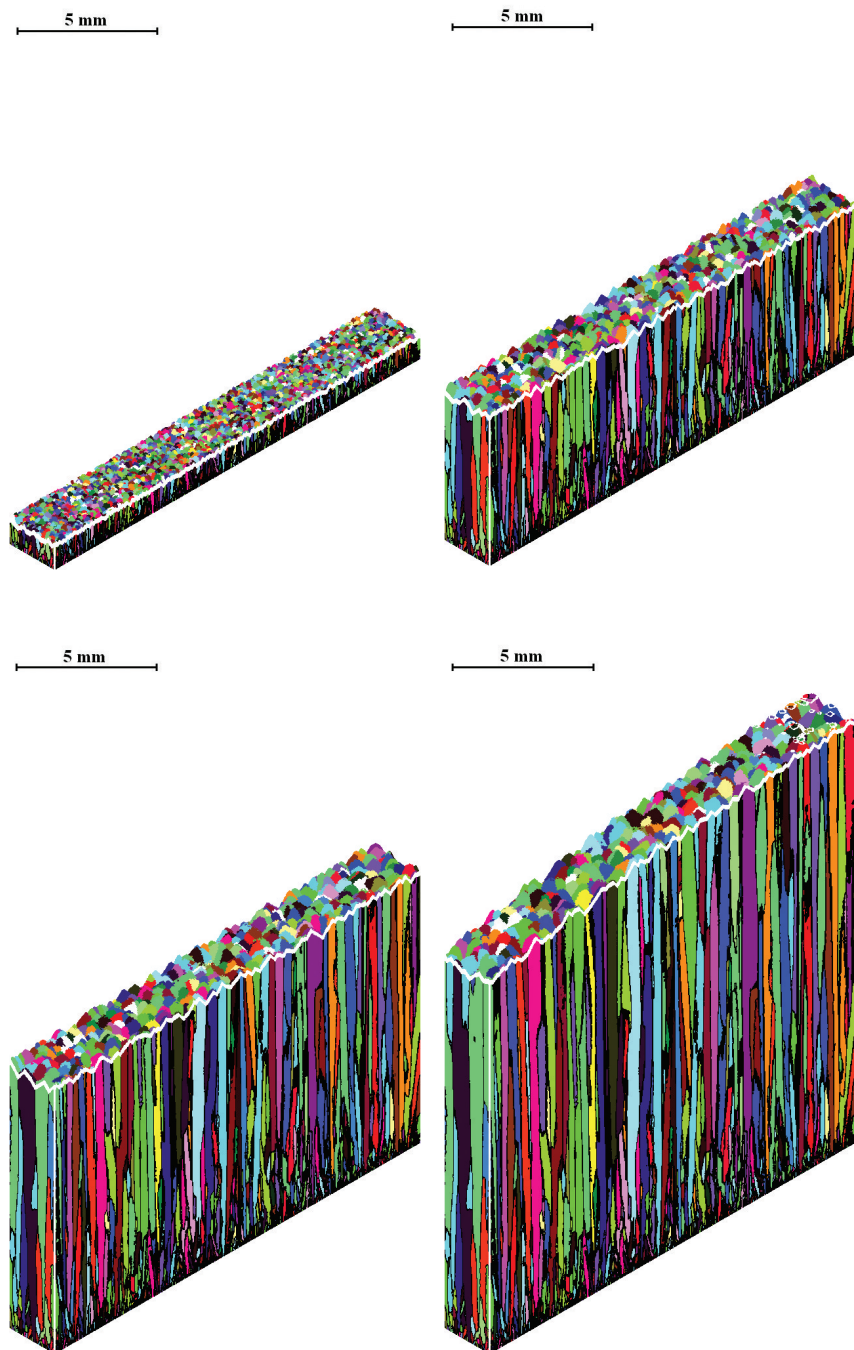
Proces ten przebiega podobnie, kiedy modeluje się dolną część wlewka. Niektóre rzuty procesu kształtowania makrostruktury od dolnej powierzchni przedstawiono na rysunku 7.15.

Kolejne cztery zdjęcia ilustrujące przebieg procesu przedstawiono na rysunku 7.16. Pokazano na nim prawą dolną krawędź wlewka. Kryształy również rosną prostopadłe do powierzchni wlewka. Później strefa obok krawędzi zamyka się i pozostają dwa główne kierunki rozrostu.

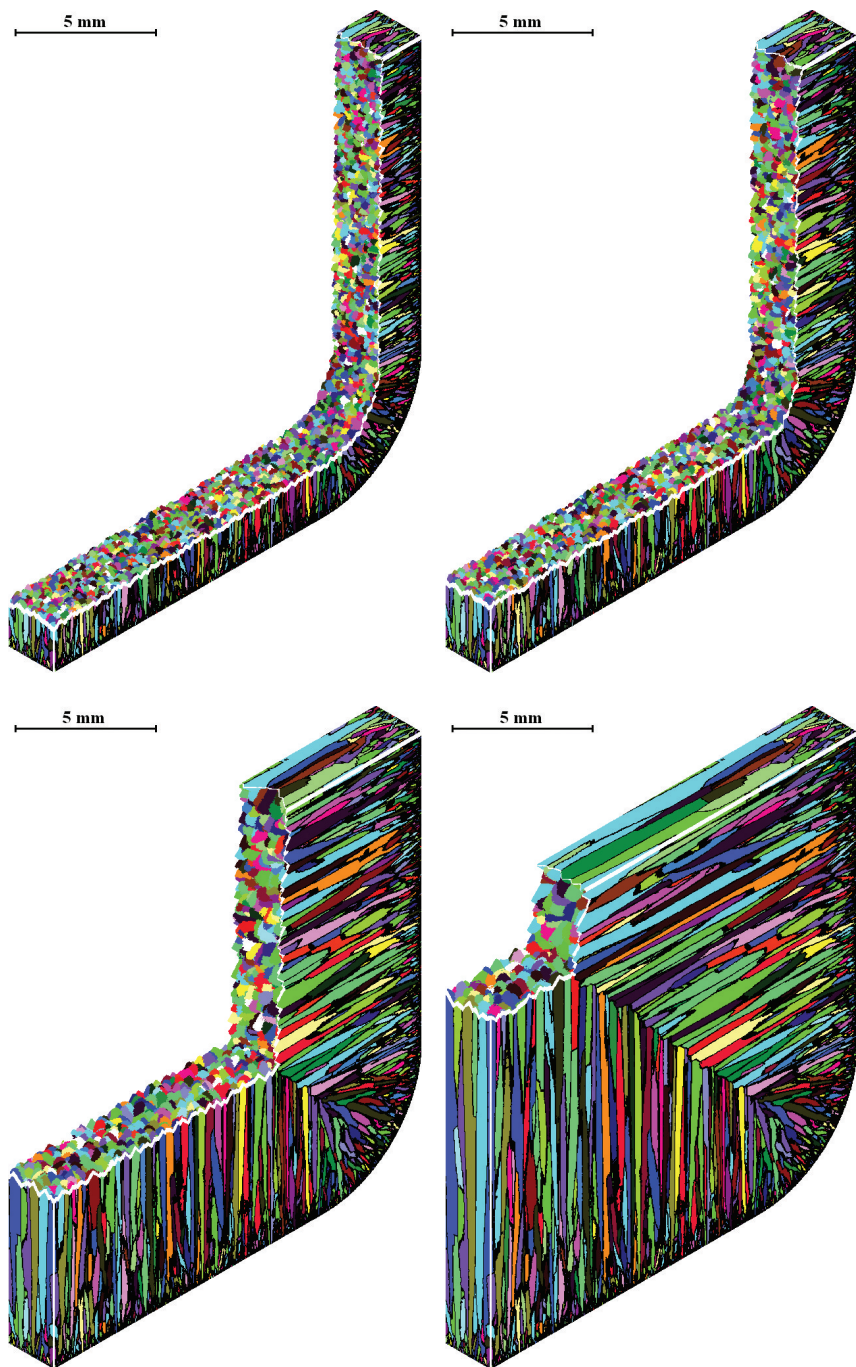
Kształtowanie makrostruktury w środku wlewka pokazano na rysunku 7.17. Kiedy proces krzepnięcia dobiega końca, tworzy się strefa równoosiowych kryształów. Powodem powstawania takich kryształów jest odbywające się w ciągu całego procesu krzepnięcia homogeniczne zarodkowanie. Kryształy na początkowych etapach krzepnięcia albo stykają się z dendrytami, albo pozostają zanurzone w cieczy i opadają w dół. Gdy stykają się z pozostałą strukturą, są włączane do struktury dendrytycznej, uzyskują preferowany kierunek rozrostu (zgodnie ze strumieniem ciepła) i ich odróżnienie od pozostałych kryształów staje się niemożliwe. Rozrost swobodnych kryształów jest hamowany aż do końca procesu krzepnięcia, kiedy następuje przyspieszenie rozrostu równoosiowych kryształów. Kilka etapów takiego rozrostu pokazano na rysunku 7.17.



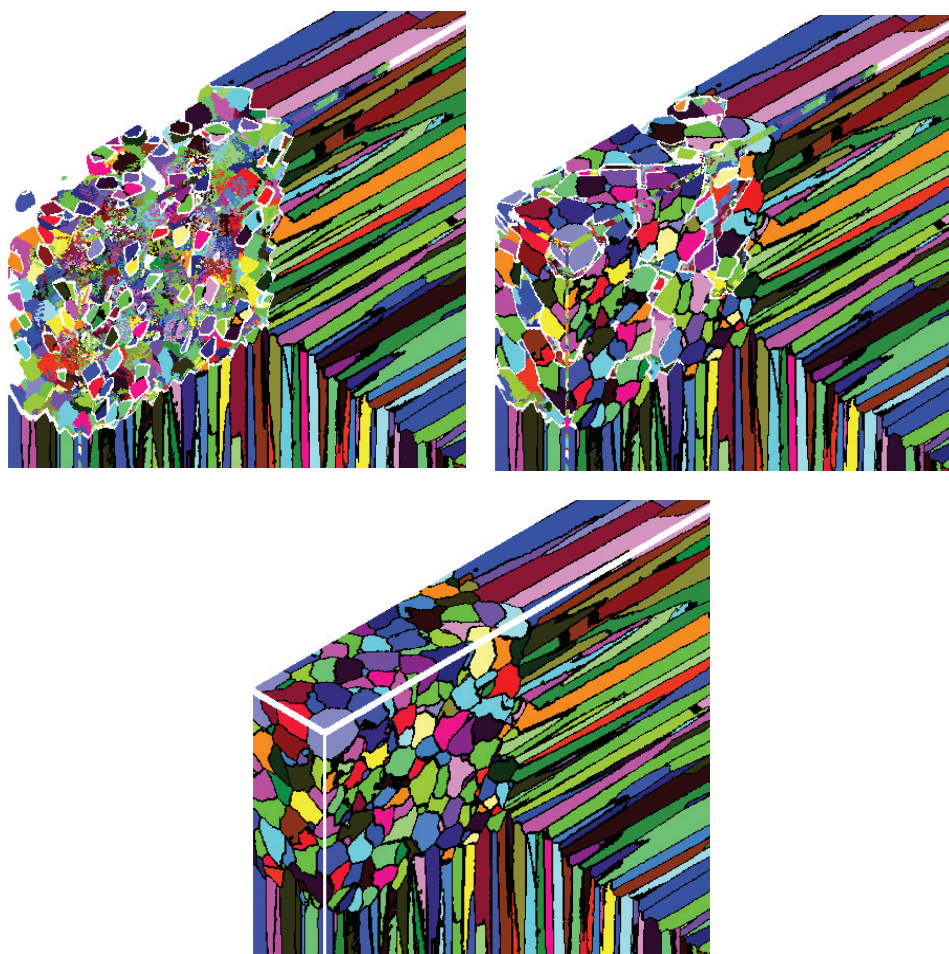
Rys. 7.14. Kształtowanie makrostruktury przy zarodkowaniu na prawej powierzchni wlewka (Svyetlichnyy 2011)



Rys. 7.15. Kształtowanie makrostruktury przy zarodkowaniu na dolnej powierzchni wlewka
(Svyetlichnyy 2011)



Rys. 7.16. Kształtowanie makrostruktury na krawędzi wlewka
(Svyetlichnyy 2011)



Rys. 7.17. Kształtowanie makrostruktury w środku wlewka
(Svyetlichnyy 2011)

7.6.2. Wyznaczenie parametrów zarodkowania

Celem tego etapu jest realizacja drugiego podstawowego zadania, a polega ono na skalowaniu parametrów modelu i wyznaczeniu wpływu warunków zarodkowania na średnicę dendrytów w środkowej części wlewka, gdzie ich wielkość się stabilizuje. Testy symulacyjne potwierdziły prawie odwrotną proporcjonalność średniego poprzecznego przekroju dendrytów do gęstości zarodkowania na ścianie krystalizatora. Gęstość zarodkowania dobrano tak, aby wyniki symulacji odpowiadały założonej wielkości dendrytów.

Możliwości opracowanego w tym etapie modelu są ograniczone głównie pojemnością pamięci RAM komputera. Większe wymiary wymagają większej przestrzeni

i większej ilości pamięci, przy zachowaniu niezmienniej rozdzielczości. Wyrażone w liczbie komórek wymiary ziarna, kryształu lub dendrytu zależą głównie od warunków zarodkowania. Im mniej zarodków, tym większe są kryształy. Jednak kiedy gęstość zarodków osiąga wartość krytyczną, następuje nasycenie i dalsze zwiększenie liczby zarodków nie prowadzi do zmian wymiarów większych kryształów. Tak więc najdrobniejsza struktura, która może być uzyskana za pomocą FCA, jest skutkiem zastosowania największej gęstości zarodkowania. Wówczas na podstawie uzyskanej struktury można określić wymiary komórki.

Na rysunku 7.18 przedstawiono przykładowy wygląd struktury w przekroju poprzecznym wlewka. Pomiary wskazują, że dendryty w odległości 10–20 mm od powierzchni osiągają wielkość 2–2,5 mm. Opierając się na takim wskaźniku, po kilku wstępnych symulacjach ustalono, że najmniejsza rozdzielczość będzie osiągnięta, gdy przyjęte wymiary komórki będą wynosić 0,15–0,2 mm, to jest prawie dziesięciokrotnie więcej niż poprzednio, co pozwala zastosować większą modelowaną przestrzeń. Oczywiście w przypadku bardziej szczegółowego modelowania należy wybierać wyższą rozdzielczość, komórki o mniejszych wymiarach i należy wówczas zachować odpowiednią gęstość zarodkowania.



Rys. 7.18. Przykładowy wygląd przekroju poprzecznego wlewka
(Svyetlichnyy *et. al.* 2012)

Na rysunku 7.19 przedstawiono wyniki wstępnego modelowania makrostruktury wycinka modelowanego wlewka o przekroju 100×100 mm z komórkami o wymiarach 0,1 mm. W tym przykładzie modelowano wycinek ćwiartki przekroju o grubości 10 mm z zastosowaniem przestrzeni automatów komórkowych o $500 \times 500 \times 100$ komórek. Uzyskane wyniki są zbliżone do doświadczalnych (zarodkowania homogenicznego nie modelowano).



Rys. 7.19. Przykład modelowanej struktury wlewka
(Svyetlichnyy *et al.* 2012)

7.7. Modyfikacje programu FCA

Jednym z kierunków rozwoju programu FCA do modelowania procesu krzepnięcia było zwiększenie przestrzeni komórkowej. W opisanej wyżej wersji głównym ograniczeniem rozszerzenia przestrzeni jest pojemność pamięci RAM, a w przypadku 32-bitowego systemu operacyjnego dodatkowo możliwości adresowania. Maksymalna liczba komórek w przypadku zastosowania procesora 32-bitowego wynosi 25 milionów, co pozwala przy rozdzielczości 0,1 mm na komórkę zamodelować przekrój 50×50 mm o grubości 10 mm. Do modelowania wlewków o większych przekrojach lub prowadzenia modelowania z wyższą rozdzielczością niezbędne są komputery z 64-bitowymi procesorami lub też modelowaniu poddaje się odpowiednie fragmenty wlewka, co opisano w poprzednim podrozdziale.

Opracowano warianty rozszerzenia przestrzeni komórkowej, które pozwoliły rozwiązać ten problem i modelować prawie dowolne wymiary z dowolną rozdzielczością. Jednak kiedy liczba komórek sięga miliardów, powstają niedogodności związane z czasem obliczeń, ich przerywaniem i kontynuacją, z wizualizacją oraz przedstawieniem wyników.

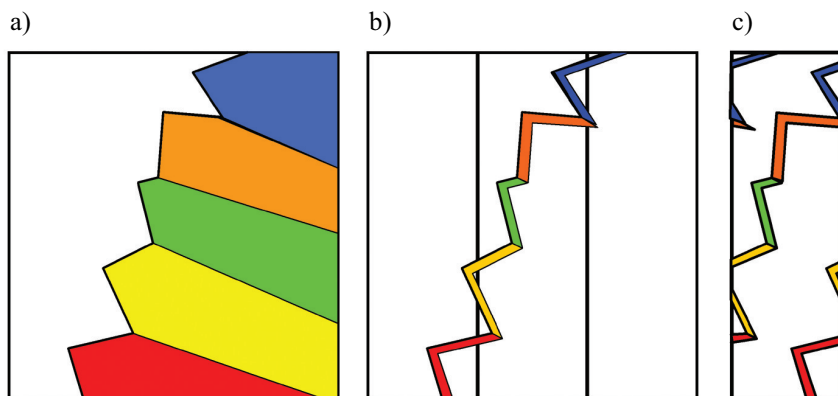
Aby zrozumieć istotę wprowadzonych zmian, należy przeprowadzić analizę struktury wykorzystywanych w programie zmiennych. Podstawową jednostką jest komórka,

która zawiera informacje o przestrzeni i zmiennych wykorzystywanych w procesie obliczeń. W miarę rozszerzania zastosowań FCA do komórki przypisywano coraz większą liczbę zmiennych. Oprócz pierwotnych danych o współrzędnych komórki i jej stanie są to parametry ziarna i elementy związane z listami komórek, czas przejścia w stan przejściowy i frontalny, kierunek rozrostu, numer segmentu, współczynnik prędkości rozrostu i inne. Modyfikacja klasycznych automatów komórkowych w kierunku automatów frontalnych doprowadziła do powstania nowej, nadrzędnej struktury – listy komórek. Ponieważ obróbkę komórek w FCA prowadzono sekwencyjnie, zgodnie nie z przestrzennym rozmieszczeniem komórek, a ze stanem komórek, w FCA stworzono listy komórek znajdujących się w tym samym stanie. Tak więc w FCA przejściu komórki z jednego stanu w inny towarzyszy przeniesienie komórki z jednej do drugiej listy. Listy komórek zostały zorganizowane w taki sposób, że zawierają nie bezpośrednio komórki, lecz dodatkowe pośrednie elementy, nazwane elementami listy. Komórka może się znajdować na kilku listach, więc może wskazywać na kilka elementów list. Aby dołączyć komórkę do listy, wystarczy stworzyć dodatkowy element listy, bez tworzenia komórki. Jednak w większości algorytmów komórka znajduje się tylko na jednej liście i wówczas powstaje sztywna zależność niezmiennie łącząca komórkę i element listy komórek. Listy komórek są dynamiczne, składają się z elementów, które pierwotnie zawierały tylko trzy wskaźniki (informację o liście oraz poprzednim i następnym elemencie listy) oraz współrzędne komórki. Pojedynczy element listy wskazuje tylko na jedną komórkę. Przejście od elementu listy do komórki i odwrotnie odbywa się w różny sposób. Element listy podaje współrzędne komórki, natomiast komórka zawiera numer elementu listy.

W chwili modyfikacji liczba zmiennych komórki wynosiła 24. Sześć dodatkowych zmiennych zawierał element listy (przynajmniej jeden) związany z komórką.

Przedstawione w poprzednim rozdziale wyniki zostały uzyskane w opisanym układzie, który można określić, jako stacjonarną przestrzeń komórkową, w której każda komórka ma sztywno ustalone przestrzenne współrzędne, jest sztywno przypisana do swojego elementu i tylko elementy są przenoszone z listy na listę. Każda komórka jest tworzona na początku modelowania i istnieje w ciągu całego procesu, więc zajmuje odpowiednią część pamięci. Jest to uzasadnione przy modelowaniu procesów cyklicznych, na przykład rekrytalizacji dynamicznej, lub procesów wieloetapowego odkształcenia, na potrzeby modelowania których FCA (rys. 4.9) był pierwotnie opracowywany.

Na rysunku 7.20 pokazano fragment przestrzeni modelowej i komórkowej. Dwie strefy komórek w stanach początkowych i końcowych we frontalnych automatach komórkowych, w odróżnieniu od klasycznych, nie uczestniczą w obliczeniach i głównie to przynosi efekt w postaci przyspieszenia obliczeń. Jednak komórki tych stref nadal znajdują się na odpowiednich listach i zajmują pamięć w pełnym wymiarze (około 30 zmiennych na komórkę). Właśnie liczba zmiennych charakteryzujących komórki i objętość pamięci wyznaczają liczbę komórek i w konsekwencji maksymalny wymiar modelowej przestrzeni.



Rys. 7.20. Przestrzeń w nowych frontalnych automatach komórkowych (Svyetlichnyy 2011)

Na tej podstawie można więc określić sposoby zwiększenia wymiarów przestrzeni modelowej. Pierwszym, najbardziej oczywistym jest zmniejszenie liczby zmiennych przypisanych do komórki i elementu listy, na przykład wykorzystanie zamiast trzech współrzędnych jednego wskaźnika. Uporządkowanie zmiennych przez ich przenoszenie do innych struktur danych lub rezygnacja z nich na korzyść dodatkowych rozwiązań programowych pozwala zmniejszyć ogólną liczbę zmiennych z 30 do 25, czyli zwiększyć przestrzeń modelową o około 16%. Jednak istnieje jeszcze inne podejście, pozwalające zwiększyć przestrzeń jeszcze bardziej. Polega ono na zmianie organizacji zmiennych.

Dwie strefy komórek w stanach początkowych i końcowych (rys. 7.20) mogą być nie tylko wyłączone z obliczeń, ale też mogą nie być uwzględnione na żadnej liście. Właściwie tylko cienka warstwa zaznaczona na rysunku 7.20b kolorem jest niezbędna do prowadzenia obliczeń. Przed frontem krystalizacji wszystkie komórki znajdują się w tym samym pierwotnym stanie i nie wymagają żadnej szczegółowej informacji, więc mogą nie być połączone z żadnym elementem listy. Po przejściu frontu nie są już potrzebne, informacja jest zachowywana w komórce, natomiast elementy listy mogą być z listy usunięte.

Na rysunku 7.20a przedstawiono przestrzeń komórkową z nałożoną na nią adresacją do elementów listy. Przestrzeń komórkowa oraz lista komórek przez cały czas fizycznie znajdują się w pamięci podręcznej, ze względu na nieefektywność wykorzystania pamięci wirtualnej lub dyskowej.

Tylko bardzo mała część komórek w każdym kroku znajduje się w stanie aktywnym (rys. 7.1), do którego mogą być zaliczone stany q_1 , q_4 i q_5 . W obliczeniach należy również uwzględnić przylegające do nich komórki w stanach q_2 , q_3 i q_{-1} . Najwięcej komórek znajduje się w stanach q_0 , q_2 , q_3 i q_{-1} – na listach zawierających te komórki może się znajdować 95–99% wszystkich komórek. Wyłączenie tych list z obliczeń pozwala zaoszczędzić około 95% pamięci przeznaczonej na przechowywanie informacji o listach komórek.

Na rysunku 7.20b pokazano całą przestrzeń komórkową i cienkie kolorowe pasmo, które reprezentuje adresację komórek uczestniczących w bieżącym kroku obliczeń do elementów list. Można więc zmniejszyć objętość pamięci wykorzystywaną do przechowywania informacji o elementach listy. Faktycznie zamiast dużej przestrzeni komórkowej obejmującej całą przestrzeń modelowaną tworzona jest przestrzeń elementów list, którą jest dość cienka warstwa komórek (rys. 7.20c) o grubości 20 (maksymalnie), 10 lub nawet 6 komórek, na podstawie których jest możliwe modelowanie całego procesu.

W porównaniu z rozwiązaniem poprzednim, w którym 25 milionów elementów list wskazywało na komórki ułożone w sześcian o wymiarach $500 \times 500 \times 100$ komórek, w proponowanym podejściu wystarczą 2 miliony elementów list ($20 \times 500 \times 100$) lub 25 milionów elementów list można wykorzystać na przykład do modelowania przestrzeni o wymiarach $20 \times 6250 \times 200$ komórek.

Jednak w wyniku zastosowania tego podejścia przestrzeń nie zwiększa się radykalnie. Jest to rozwiązanie połowiczne, jego efekt jest nieznaczny, gdyż najwięcej pamięci zajmuje informacja o komórkach (przed opracowaniem 24 zmiennych przypisanych do komórki w przeciwieństwie do 6 przypisanych do elementu listy). Skuteczność tego rozwiązania można zwiększyć, zmieniając liczbę zmiennych przypisanych do komórki i elementu listy.

Po przekształceniu struktury zmiennych do komórki zostają przypisane tylko trzy zmienne: stan, numer elementu listy i numer ziarna. Natomiast element listy zawiera 16 zmiennych. Komórki nadal pozostają sztywno związane z przestrzenią, natomiast elementy listy dynamicznie łączą się z komórkami. W wyniku takiego przekształcenia wykorzystanie pamięci zmniejsza się około siedmiokrotnie. Otrzymanie wyniku przedstawionego na rysunku 7.19 wymaga teraz tylko 500 MB pamięci RAM zamiast prawie maksymalnej dopuszczalnej dla 32-bitowego procesora 3,5 GB.

Takie podejście można zastosować do modelowania prawie dowolnego zjawiska lub procesu przedstawionego w niniejszej pracy: początkowej mikrostruktury, rekrytalizacji, czy też przemiany fazowej. Wymiary przestrzeni można zwiększyć jeszcze bardziej, jednak opisane niżej podejście można zastosować tylko w odniesieniu do krzepnięcia i początkowej mikrostruktury, natomiast w przypadku modelowania pozostałych zjawisk jest to nie do zaakceptowania, co również zostanie wytłumaczone.

Wracając do rysunku 4.20b – przestrzeń komórkowa jest na nim znacznie większa od wykorzystywanej w bieżącym cyklu obliczeniowym (kolorowe pasmo), nadal informacja o niej znajduje się w pamięci podręcznej przeznaczonej na przechowywanie informacji o zmiennych na początku obliczeń. Jest to kwestia niepodważalna, jeśli chodzi o modelowanie z wykorzystaniem istniejącej mikrostruktury. Bardzo niekorzystne jest wykorzystywanie pamięci innej niż podręczna, ponieważ odczytywanie i zapisywanie informacji na dysk podczas obliczeń wielokrotnie je wydłuża. Związane jest to z koniecznością posiadania nieuporządkowanego dowolnego dostępu do pamięci.

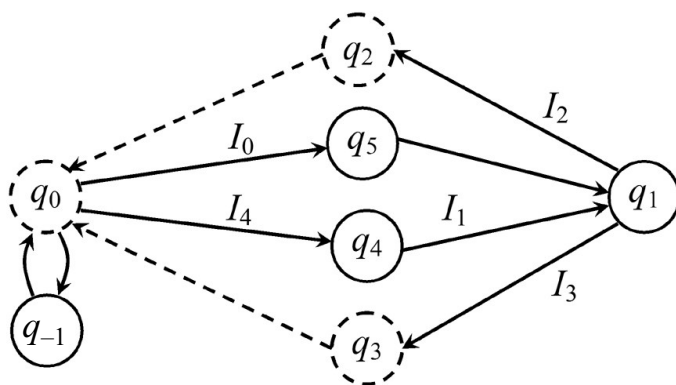
Z drugiej strony zapisywanie końcowej mikrostruktury jest operacją niezbędną, która może wpłynąć na ogólny czas symulacji. W przypadkach, w których komórka może zmieniać swój stan, wielokrotnie powtarzając sekwencje w kilku cyklach, jedynym

rozwiązaniem jest zapisywanie końcowej struktury po ostatnim cyklu, po ostatniej iteracji, natomiast podczas modelowania procesów, w których przejście następuje jednokrotnie, zapisywanie można prowadzić jednocześnie z obliczeniami. Do takich procesów należą krzepnięcie i tworzenie początkowej struktury – zapisywanie następuje w nich równoległe z obliczeniami, co jest korzystniejsze niż zapisywanie po skończeniu obliczeń. Jednak korzyści te, jeśli chodzi o przyspieszenie obliczeń, nie są warte uwagi, jeżeli zapisywanie końcowej struktury nie zwiększa rozmiarów modelowej przestrzeni w bardzo dużym stopniu.

Wracając do rysunku 4.20b, można powiedzieć, że przestrzeń na prawo od kolorowego pasma, czyli po tej samej stronie co front krystalizacji, zawiera końcową strukturę, jest zapisana i nie będzie wykorzystana w dalszym modelowaniu. Wszystkie informacje jej dotyczące mogą być usunięte z pamięci podręcznej. Innymi słowy, ta część rysunku obrazuje część przestrzeni, w której nie istnieją już komórki ani elementy list na nią nie wskazują. Natomiast przed frontem krystalizacji wszystkie komórki znajdują się w stanie początkowym, nie są przypisane ani do ziaren ani do list, czyli nie zawierają żadnej unikalnej informacji, którą należałoby zachować. A skoro tak, to one również nie muszą być uwzględniane w obliczeniach i nie wymagają żadnych zasobów pamięci.

Kolorowe pasmo na rysunku 4.20b symbolizuje zarówno automaty komórkowe istniejące w danej chwili, jak i adresację do nich elementów aktywnych list. Na rysunku 4.20b cała przestrzeń zawiera komórki i elementy list możliwe do wykorzystania w obliczeniach.

Taka zmiana obliczeń prowadzi również do modyfikacji automatu komórkowego, który przyjmuje postać przedstawioną na rysunku 7.21. Komórki w stanie końcowym q_2 i q_3 oraz w stanie q_{-1} są przenoszone w stan początkowy q_0 wówczas, gdy w ich otoczeniu Moore'a nie ma już komórek w stanach q_1 , q_4 lub q_5 .



Rys. 7.21. Frontalny automat komórkowy do modelowania kształtowania makrostruktury podczas krzepnięcia ze zmienną przestrzenią komórkową

Obliczenia rozpoczynają się od zarodkowania. Wybiera się współrzędne według założonych warunków zarodkowania i w tym miejscu tworzy się automat komórki, który przechodzi w stan zarodkowania q_5 . Jednocześnie wybierany jest dowolny element listy, łączy się z nim komórkę i dołącza się go do odpowiedniej listy. Następnie automat przechodzi w stan frontalny q_1 , a element listy jest przenoszony na inną listę. Komórka w stanie frontalnym q_1 bada swoje otoczenie i jeżeli w żadnym jego miejscu nie istnieją inne komórki, oznacza to, że front krystalizacji jeszcze nie dotarł do tego miejsca i automat komórki nie został utworzony, więc automat komórki powinien się znajdować w stanie początkowym q_0 . Tworzy się w tym miejscu automat, wprowadza się go w stan q_4 (lub q_{-1}), wylicza się czas opóźnienia (czas przejścia) i łączy się go ze swobodnym elementem listy, tak samo jak w przypadku zarodkowania. Komórka w stanie q_4 po spełnieniu warunku I_1 zmienia swój stan i jest przenoszona do innej listy.

Automat komórkowy w stanie q_1 , po zbadaniu całego otoczenia i wymuszeniu przejścia wszystkich sąsiednich komórek do stanu q_4 , sprawdza warunki I_2 i I_3 i zmienia swój stan na granicy ziarna q_2 lub wewnątrz ziarna q_3 , jednocześnie zapisując informację do pliku z końcową strukturą. Komórki w stanach końcowych q_2 i q_3 (oraz w stanie q_{-1}), gdy w ich otoczeniu nie ma już komórek w stanach q_1 , q_4 lub q_5 , wprowadza się w stan początkowy, odłącza się je od ziarna i od elementu listy. Element listy również się zeruje i przenosi do listy elementów swobodnych.

W przypadku zastosowania takiego podejścia 25 milionów komórek, bez ograniczeń jak poprzednio, można ułożyć w przestrzeń dynamiczną o rozmiarach $20 \times 6250 \times 200$ komórek, co pozwala zamodelować przestrzeń o przekroju 625×20 mm, dowolnej długości. Przy rozdzielczości 0,1 mm na komórkę, zamiast wycinka o wymiarach $50 \times 50 \times 10$ mm, można zatem uzyskać makrostrukturę praktycznie wszystkich możliwych przekrojów rzeczywistych wlewków. Natomiast dla tego samego wycinka o wymiarach $50 \times 50 \times 10$ mm można uzyskać rozdzielczość 0,02 mm na komórkę (i wyższą), czyli można zwiększyć rozdzielczość kilkakrotnie. Oczywiście każde zwiększenie przestrzeni (liczby komórek) proporcjonalnie wydłuża czas obliczeń.

7.8. Weryfikacja modelu i jego optymalizacja ze względu na nakłady obliczeniowe

Wprowadzenie wyżej opisanych modyfikacji pozwala istotnie zwiększyć wymiary modelowej przestrzeni, zmniejszyć wymagania co do obszaru pamięci podręcznej i przeprowadzić symulację krzepnięcia całego wlewka w maszynie COS.

Powiększenie modelowej przestrzeni do całego przekroju wlewka stworzyło możliwość przeprowadzenia bardziej szczegółowego porównania uzyskanych wyników kształtowania struktury z wynikami modelowania metodą elementów skończonych.

Teraz można porównać już nie mały wycinek, a cały przekrój. W tym celu przeprowadzono trzy warianty symulacji makrostruktury. W dwóch pierwszych symulacji poddano wlewek kwadratowy i okrągły odlewany zgodnie z rzeczywistymi warunkami. Trzecia, dodatkowa symulacja dotyczyła niesymetrycznych warunków chłodzenia – miejsca intensywnego chłodzenia wlewka okrągłego znajdowały się na pionowej i poziomej osi przekroju poprzecznego oraz na jego przekątnej. W ten sposób celowo zaburzono pole temperatury wlewka, co doprowadziło do istotnego zniekształcenia frontu krzepnięcia.

Modelowanie procesu odlewania zostało wykonane przez Beatę Hadałę (Svyetlichnyy *et al.* 2012) za pomocą programu MES opracowanego przez prof. Zbigniewa Malinowskiego (Hadała i Malinowski 2009). Obliczeniom poddano proces ciągłego odlewania pasma o przekroju kwadratowym 160×160 mm z promieniem zaokrąglenia naroża 16 mm, odlewane z prędkością 27 mm/s oraz pasma o przekroju kołowym o średnicy 160 mm, odlewane z prędkością 25 mm/s. W obu przypadkach zastosowano stal o następującym składzie chemicznym: C – 0,82%, Mn – 2%, Si – 1,8%, Co – 1,5%. Obliczenia numeryczne miały na celu wyznaczenie pola temperatury, które uwzględniono przy modelowaniu struktury krzepnącego wlewka.

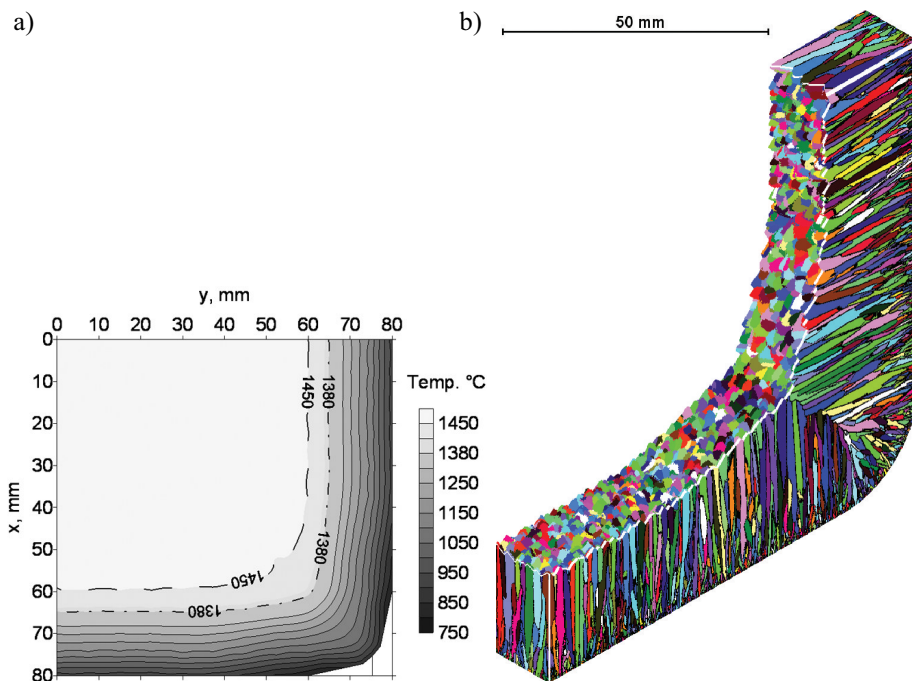
W obliczeniach zmiany temperatury wlewka podczas ciągłego odlewania przyjęto następujące parametry:

- temperatura zalewania ciekłej stali – 1475°C ,
- temperatura krzepnięcia zastosowanej zasyпки – 1150°C ,
- temperatura solidus – 1380°C , temperatura likwidus – 1450°C ,
- temperatura początku i końca przemiany w stanie stałym – odpowiednio: 810°C i 750°C ,
- temperatura otoczenia – 20°C .

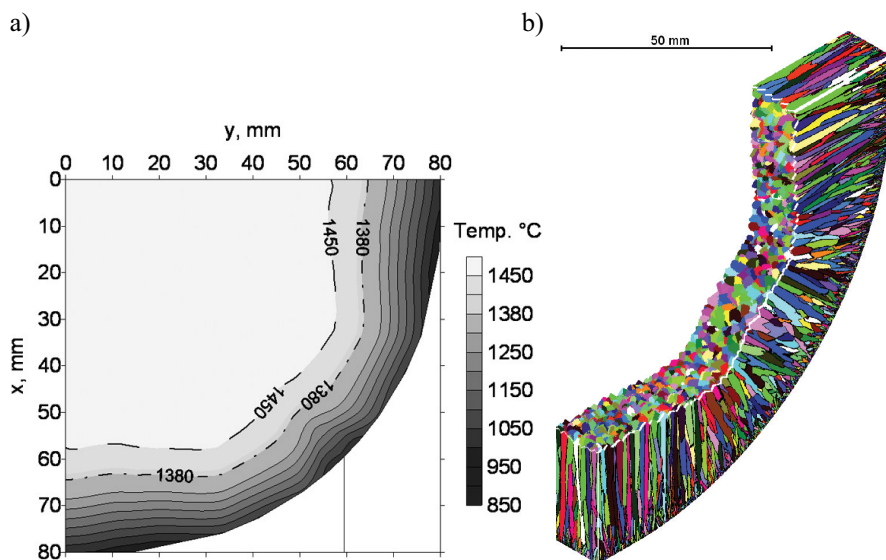
Parametry chłodzenia w krystalizatorze dobrano odpowiednio do krystalizatora o długości 800 mm, grubości ścianki 14 mm i średnicy hydraulicznej kanału chłodzenia wodnego 4 mm. Prędkość wody w kanale wynosiła 10 m/s.

Na rysunkach 7.22 i 7.23 przedstawiono wyniki rozkładu temperatury (rys. 7.22a i 7.23a) oraz makrostrukturę w tym samym przekroju (rys. 7.22b i 7.23b) wlewka kwadratowego (rys. 7.22) i okrągłego z nierównomiernym chłodzeniem (rys. 7.23). Porównanie kształtu linii solidus z uzyskanym frontem krzepnięcia wskazuje na bardzo dobrą zbieżność, co pozwala stwierdzić, iż uzyskiwany bieżący front krzepnięcia odpowiada wynikom modelowania pola temperatury, i potwierdza, że metodę tę zastosowano słusznie oraz że jest ona wysoce skuteczna.

Ostatni etap modelowania dotyczył optymalizacji obliczeń i polegał na badaniu zależności między rozdzielczością przestrzeni automatów komórkowych a wynikiem symulacji. Wpływa to w istotny sposób na wykorzystanie pamięci operacyjnej komputera oraz na czas obliczeń. Makrostrukturę symulowano w czterech wariantach przestrzeni komórkowej – zostały one przedstawione w tabeli 7.1, w której podano również czas obliczeń.



Rys. 7.22. Rozkład temperatury (a) i struktura (b) w przekroju poprzecznym wlewka kwadratowego 160×160 mm (Svyetlichnyy *et al.* 2012)



Rys. 7.23. Rozkład temperatury (a) i struktura (b) w przekroju poprzecznym wlewka okrągłego o średnicy 160 mm (Svyetlichnyy *et al.* 2012)

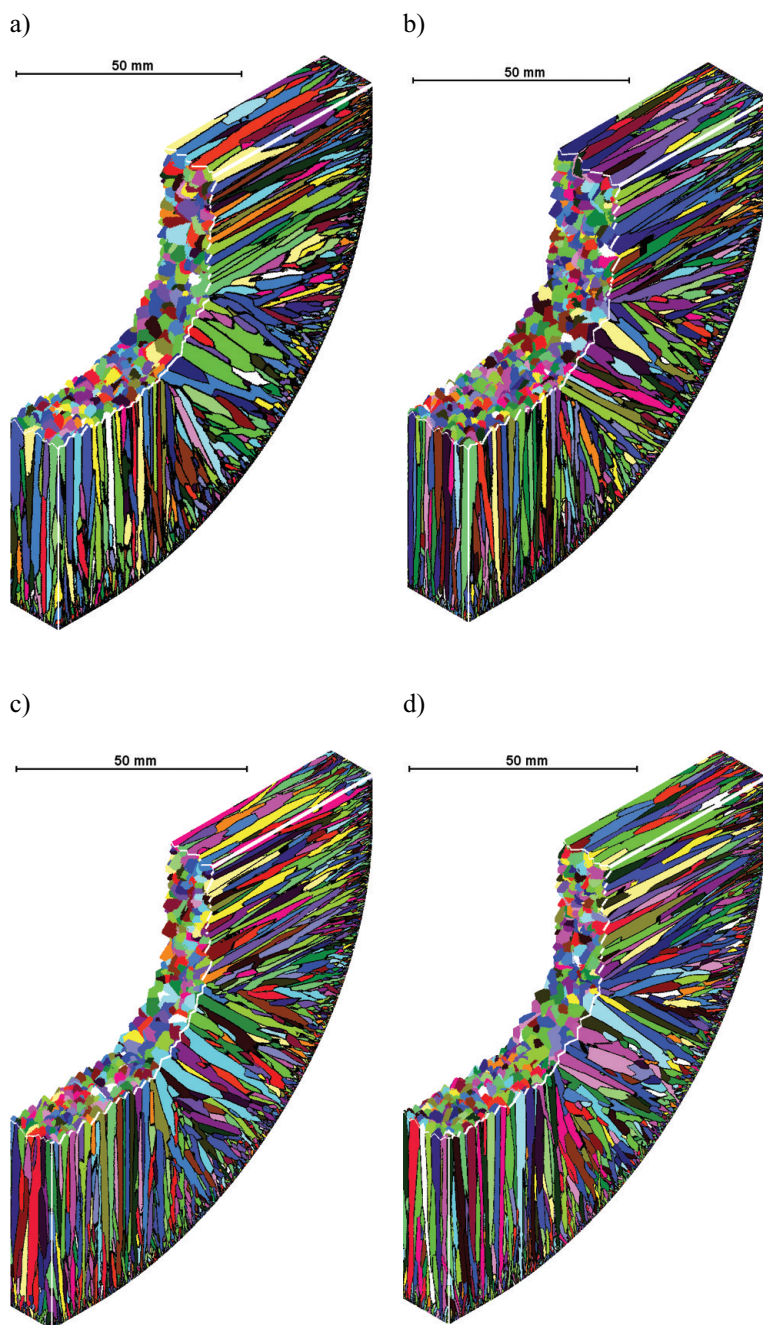
Tabela 7.1Warianty symulacji makrostruktury (Svyetlichnyy *et al.* 2012)

Parametry	Wariant 1	Wariant 2	Wariant 3	Wariant 4
Wymiary przestrzeni, liczba komórek	800 × 120 × 800	1000 × 200 × 1000	1600 × 200 × 1600	2000 × 280 × 2000
Liczba komórek, w mln	76,8	200	512	1120
Rozmiar komórki, μm	100	80	50	40
Pamięć RAM, GB	2	4	8	16
Czas symulacji, godz:min				
Okrąg	0:53	1:47	3:36	6:52
Kwadrat	0:59	2:19	4:52	11:09

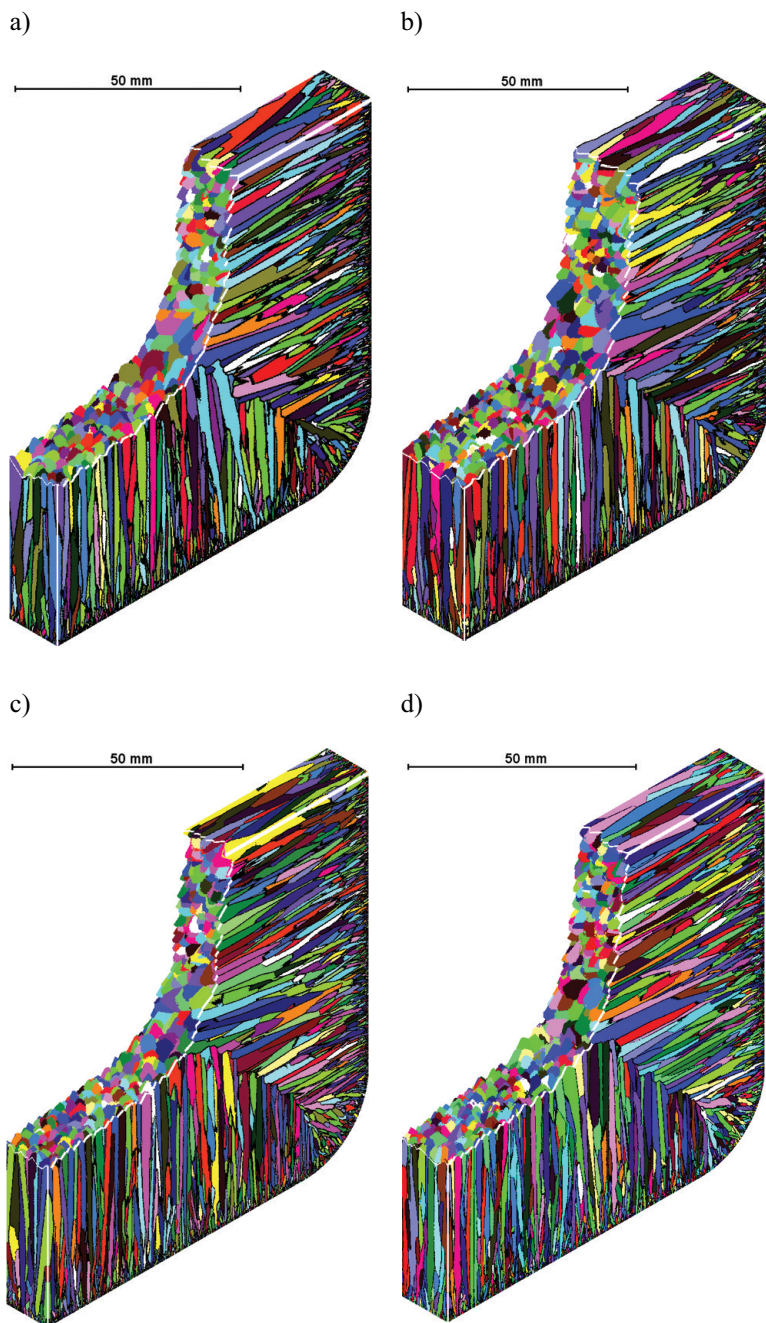
Przykładowe makrostruktury pokazano na rysunkach 7.24 i 7.25. Poniżej przedstawiano też wyniki symulacji w tym samym przekroju we wszystkich wariantach przestrzeni komórkowej dla wlewka okrągłego (rys. 7.24) i kwadratowego (rys. 7.25). Uzyskana makrostruktura we wszystkich wariantach jest bardzo podobna i różni się wyłącznie nieistotnymi szczegółami. Uzyskane wyniki wskazują na to, że wymiar komórki mniejszy od maksymalnego dopuszczalnego (około 150 μm) nie wpływa na uzyskiwaną makrostrukturę. Jednak zwiększenie rozdzielczości przestrzeni istotnie rzutuje na wymagane zasoby pamięci komputera oraz na czas obliczeń. W tabeli 7.1 przedstawiono czasy obliczeń numerycznych przy wykorzystaniu stacji roboczej Xeon CPU E5420, 2.50 GHz.

Na rysunkach 7.26 i 7.27 pokazano odpowiednio przebieg zmian makrostruktury w wariantcie 4 podczas krzepnięcia wlewka okrągłego (z zakłóconym chłodzeniem) i kwadratowego.

Na rysunku 7.28 przedstawiono wyniki symulacji według najprostszego wariantu (wariant 1), niewymagającego dużych nakładów obliczeniowych, dla przypadku, w którym warunki symulacji doprowadziły do prawie osiowosymetrycznego rozkładu temperatury na przekroju wlewka. Powoduje to wytworzenie się prawie osiowosymetrycznej struktury, w której dendryty skierowane są w kierunku osi wlewka, natomiast front krzepnięcia jest przez cały czas trwania procesu zbliżony do okrągłego. Jest to dobrze widoczne w szczególności w porównaniu z rysunkiem 7.22 lub 7.27, na którym można zauważyć, że linia solidus i odpowiednio front krystalizacji składa się z trzech odcinków oraz we wlewku można wydzielić odpowiednie strefy orientacji głównych osi dendrytów. Wyniki przedstawione na rysunku 7.28 potwierdzają możliwości skutecznego modelowania nie tylko wlewków o kwadratowym kształcie, ale również okrągłych, z wysokim stopniem osiowej symetrii.



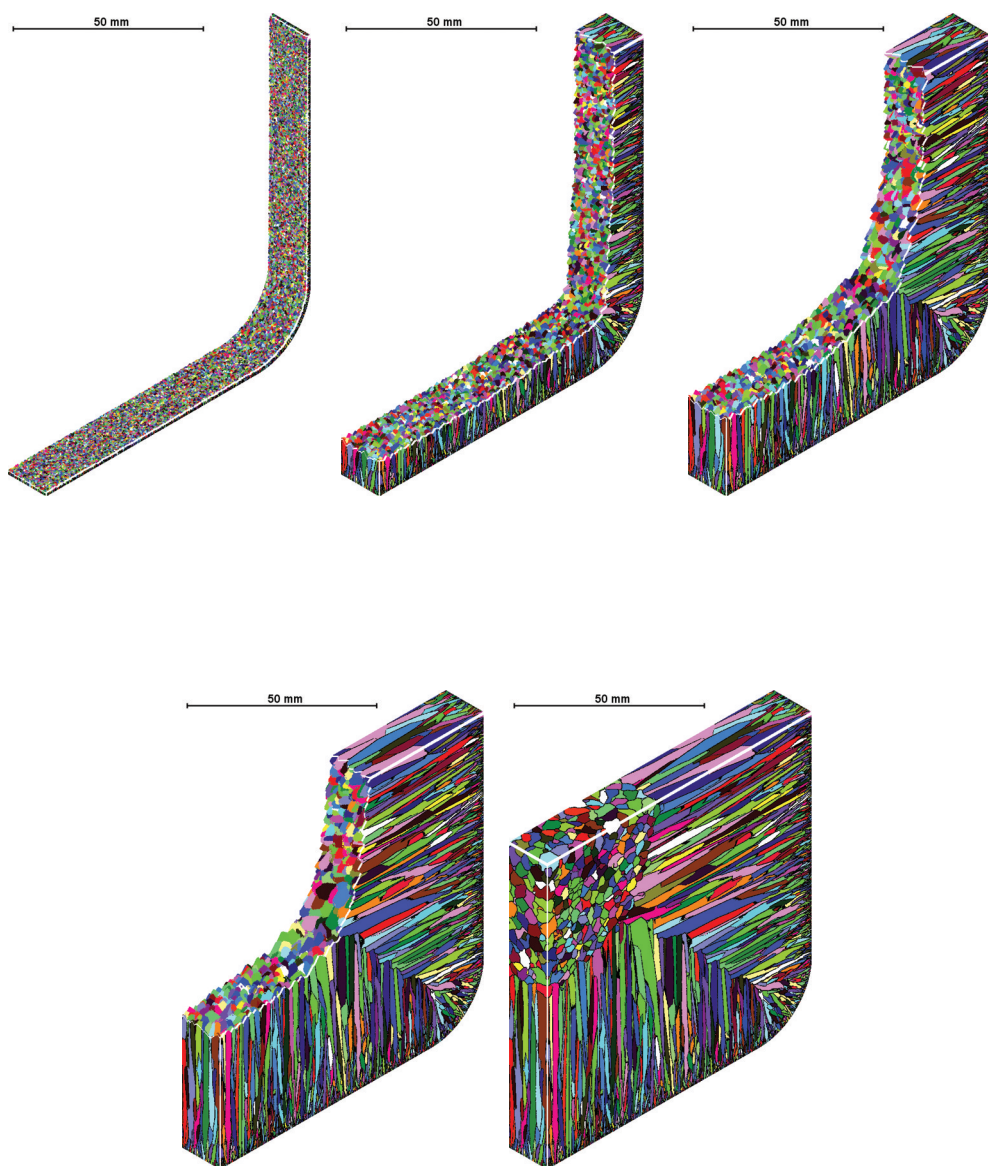
Rys. 7.24. Makrostruktura przekroju wlewką okrągłego
w czterech wariantach symulacji:
a) wariant 1; b) wariant 2; c) wariant 3; d) wariant 4



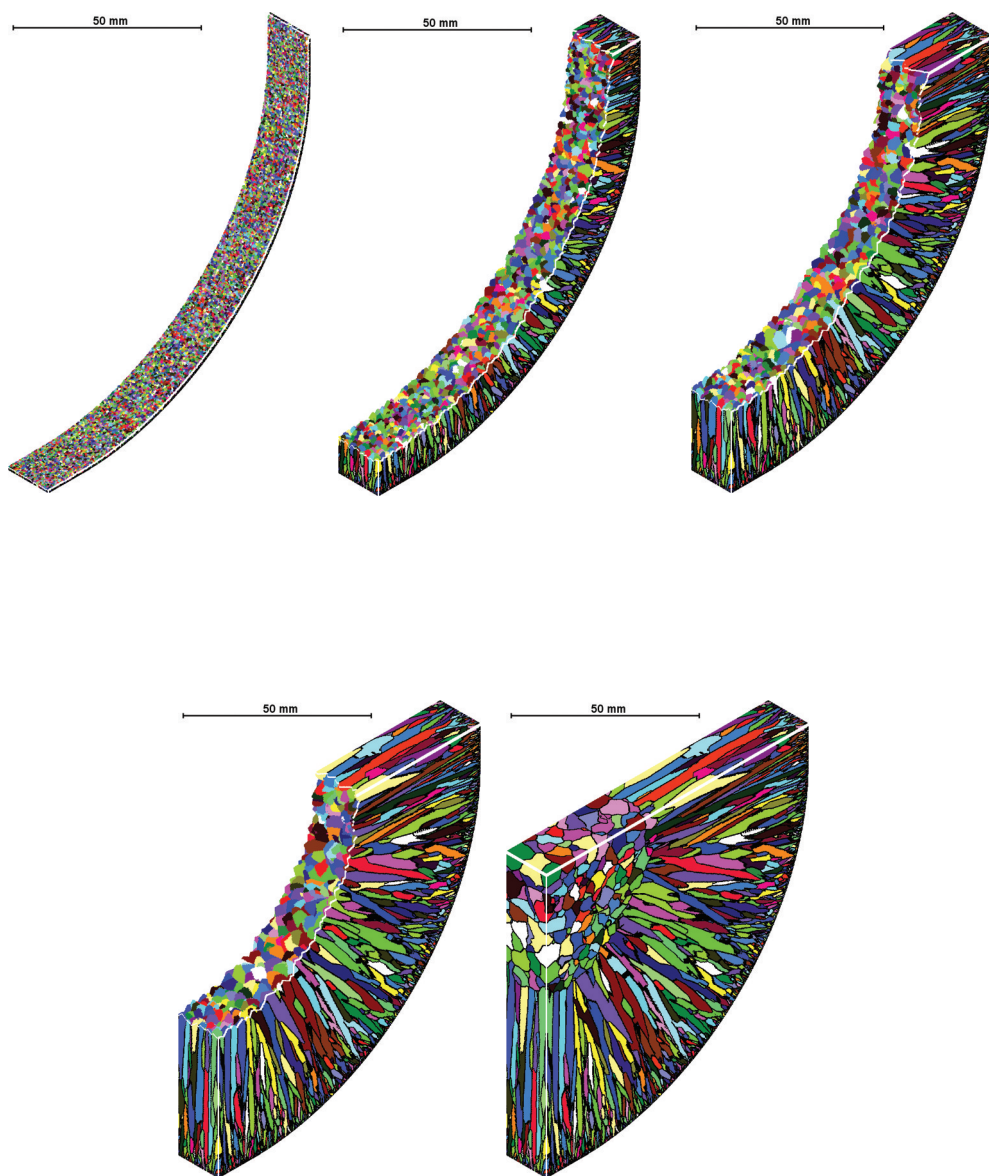
Rys. 7.25. Makrostruktura przekroju wlewnka kwadratowego
w czterech wariantach symulacji:
a) wariant 1; b) wariant 2; c) wariant 3; d) wariant 4



Rys. 7.26. Wyniki symulacji rozwoju makrostruktury podczas krzepnięcia wlewka okrągłego z zakłóconym chłodzeniem w wariantie 4



Rys. 7.27. Wyniki symulacji rozwój makrostruktury podczas krzepnięcia wlewka kwadratowego w wariancie 4



Rys. 7.28. Wyniki symulacji rozwoju makrostruktury podczas krzepnięcia wlewka okrągłego w wariantie 1

W niniejszym rozdziale przedstawiono wyniki symulacji procesu krzepnięcia w maszynie ciągłego odlewania stali z zastosowaniem modelu opartego na automatach komórkowych z wykorzystaniem danych obliczonych metodą elementów skończonych. Obliczenia MES pozwoliły uzyskać rzeczywisty rozkład temperatury z uwzględnieniem technologicznych warunków odlewania stali. Na podstawie tak wyznaczonego pola temperatury można było wykorzystać automaty komórkowe do symulacji kształtowania struktury wlewka ciągłego. Przedstawione wyniki potwierdziły wysoką skuteczność połączonego działania MES i FCA. Opracowane oprogramowanie może być z powodzeniem stosowane w rzeczywistych warunkach ciągłego odlewania stali.

8. Rekrytalizacja

Niniejszy rozdział został poświęcony zjawisku rekrytalizacji, które jest istotnym czynnikiem wpływającym na właściwości materiału i bez którego uwzględnienia niemożliwe jest ani poprawne przeprowadzenie procesu plastycznej przeróbki metalu, ani jego zamodelowanie czy zaprojektowanie. Na początku (podrozdz. 8.1) opisano rozwój metod badania i modelowania zjawiska rekrytalizacji. Dalej określono główne założenia modelu wykorzystującego frontalne automaty komórkowe (podrozdz. 8.2). Zarodkowanie i rozrost ziaren są podstawowymi etapami rekrytalizacji, które mogą być poprawnie zasymulowane z wykorzystaniem modeli rozwoju gęstości dyslokacji. Modele te zostały opisane w podrozdziale 8.3. W podrozdziale 8.4 zbadano wpływ warunków brzegowych na poprawność modelowania kinetyki rekrytalizacji oraz wpływ, jaki mają na kinetykę wymiary przestrzeni i kształt ziaren. Następnie rozpatrzono rekrytalizację: taką, którą można uznać za statyczną (podrozdz. 8.5), i taką, którą zalicza się do dynamicznej (podrozdz. 8.6). W rozdziale przedstawiono również metodę oszacowania wielkości ziaren na podstawie liczby komórek na granicy ziarna (podrozdz. 8.7) oraz określono możliwy kierunek rozwoju badań zjawiska rekrytalizacji za pomocą FCA (podrozdz. 8.8).

8.1. Rozwój metod badania i modelowania rekrytalizacji

Zjawisko rekrytalizacji jest znane od ponad 130 lat, kiedy to w 1886 roku angielski naukowiec H.C. Sorby przy użyciu mikroskopu odkrył, że w wyniku kucia na zimno ziarna w żelazie stają się wydłużone, a po dłuższym rozżarzaniu do czerwoności – ponownie równoosiowe. Zjawisko to nazwał powtórna krystalizacja – rekrytalizacja. Podstawy kinetyki transformacji (krystalizacji, rekrytalizacji, przemian fazowych itp.) zostały opisane w końcu lat 30. XX wieku w pracach Kołmogorowa (1937), Johnsona i Mehla (1938) oraz Avramiego (1939, 1940 i 1941), którzy opracowali teorię (oznaczoną od nazwisk twórców akronimem KJMA) i zaproponowali wzory, nadal stosowane w wielu modelach teoretycznych i empirycznych. Zbadali oni kilka szczegółowych przypadków i doszli do podobnych wniosków. Za najważniejsze osiągnięcia tej teorii

można uznać kilka podobnych do siebie równań, które powszechnie nazywa się równaniem Avramiego. Opisują one kinetykę rekrytalizacji (zmiany ułamku rekrytalizacji χ) jako funkcję czasu.

Kolejne podejście do zagadnienia rekrytalizacji zostało zaproponowane przez Cahna (1996). Ze względu na układ geometryczny w prezentacji graficznej zostało ono nazwane metodą Cahna stożków czasowych. Metoda ta pozwala obliczyć kinetykę rekrytalizacji z uwzględnieniem różnych prędkości zarodkowania/wzrostu czy anizotropii materiału. Model Cahna umożliwia też zastosowanie wyników rozwiązania w przypadku ograniczonego obszaru próbki. Opiera się on na prawdopodobieństwie zajścia transformacji w przestrzeni w wybranym punkcie testowym. Dla przypadku przestrzeni jednowymiarowej można łatwo przedstawić metodę Cahna w sposób graficzny. Wielkość ułamka zrekrystalizowanego jest determinowana przez wzajemny wpływ poszczególnych punktów zarodkowania oraz obszarów wzrostu (obszarów stożkowych) na siebie. Model Cahna uwzględnia trzy podstawowe przypadki zajścia transformacji (rekrytalizacji) w wybranym punkcie, w zależności od rozmiarów próbki i położenia punktu w stosunku do granic materiału. Jednak w przestrzeni dwuwymiarowej jest to trudne, a w trójwymiarowej – niemożliwe.

Taki sam wynik, w tym prezentację graficzną, można uzyskać, stosując jednowymiarowe automaty komórkowe lub metodę Monte Carlo, w której jedna oś jest przestrzenna, a druga – czasowa. Jednak metody te mogą być łatwo zastosowane w odniesieniu do przetrzeni dwu- i trójwymiarowej. Właśnie zadanie warunków zarodkowania i rozrostu ziaren jest jednym z najczęściej stosowanym podejść do modelowania kinetyki rekrytalizacji i przemian fazowych za pomocą automatów komórkowych (lub bliższych im metod geometryczno-topologicznych).

Jedną z głównych zalet metody automatów komórkowych w porównaniu z teoriami KJMA i Cahna jest możliwość uwzględnienia większej liczby czynników, w tym lokalnych (na przykład krystalograficzną orientację lub gęstość dyslokacji). Inną jej zaletą jest możliwość uzyskania realistycznej mikrostruktury, która jest ściśle związana z rzeczywistymi warunkami zarodkowania i rozrostu ziaren i która umożliwia stworzenie cyfrowej reprezentacji materiału oraz określenie jego własności.

W modelach empirycznych często rozpatrywano różne rodzaje rekrytalizacji (statyczną, dynamiczną, metadynamiczną), chociaż w istocie jest to jeden proces, który odbywa się w różnych warunkach.

Niewiele modeli pozwala na wprowadzenie teoretycznych zależności zarodkowania i rozrostu ziaren od bieżących warunków. Modele oparte na automatach komórkowych pozwalają na to – przede wszystkim umożliwiają uwzględnianie kilku jednocześnie zachodzących zjawisk, zbliżając modelowanie do rzeczywistych procesów. Modelowanie i symulacja powinny się opierać na badaniach doświadczalnych, które pozwolą wyznaczyć i bardziej dokładnie opisać te zjawiska, dobrać odpowiednie parametry i dzięki porównaniu wyników symulacji i badań doświadczalnych pogłębić wiedzę o zjawisku rekrytalizacji.

Rozwój wiedzy z tej dziedziny nauki można opisać jako przejście od wzorów teorii KJMA, poprzez graficzną prezentację metodą Cahn'a, do cyfrowej reprezentacji materiałów z wykorzystaniem głównie automatów komórkowych.

Bardzo szczegółowy przegląd stanu wiedzy na temat rekrytalizacji przedstawiony jest w artykułach Doherty'ego *et al.* (1997) oraz Riosa *et al.* (2005). Dlatego poniżej zostaną tylko krótko omówione główne osiągnięcia, które również będą uwzględnione w opracowanym modelu i podczas symulacji. Po pierwsze, należy podkreślić, że rekrytalizacja jest nieodwracalnym, zachodzącym w wyższych temperaturach zjawiskiem, które składa się z zarodkowania i rozrostu ziaren. Charakteryzuje się następującymi cechami (Rios *et al.* 2005):

- Siła napędowa rekrytalizacji (energia odkształcenia przejawiająca się defektami krystalicznej struktury) jest bardzo mała w porównaniu z innymi zjawiskami zachodzącymi w ciałach stałych i prawie nie zależy od temperatury.
- Klasyczna teoria zarodkowania, oparta na fluktuacjach termicznych, nie ma w jej przypadku zastosowania, dlatego, że ukształtowanie dużego zarodka jest zdarzeniem nieprawdopodobnym, w szczególności przy zbyt małej sile napędowej i dość dużej energii granic pomiędzy zrekrystalizowanymi i niezrekrystalizowanymi ziarnami.
- Zarodkowanie polega na kształtowaniu i inicjalizacji ruchu wysokokątowych granic.
- Udowodniono doświadczalnie istnienie przynajmniej trzech mechanizmów zarodkowania.
- Zarodkowanie następuje wyłącznie w heterogeniczny sposób, przy niejednorodności poprzedzającego je odkształcenia.
- Stosunkowo mała jest liczba zarodków w jednostce objętości – zarodek powstaje w jednym z miliona potencjalnych miejsc zarodkowania (komórki dyslokacyjne lub podziarna).
- Po początkowym zarodkowaniu zachodzi tylko rozrost, czyli następuje wysycenie miejsc zarodkowania.
- Generalnie prędkość rozrostu ziaren z czasem maleje. Związane jest to z gradientem odkształcenia i jest wynikiem dwóch konkurujących ze sobą procesów: odkształcenia i zdrowienia.
- Zrekrystalizowane ziarno jest bardzo czułe na odkształcenie i nieco mniej na temperaturę.

Historia symulacji rekrytalizacji za pomocą automatów komórkowych rozpoczęła się wraz z opublikowaniem pracy Hesselbartha i Göbla (1991). Później Davies (1997) zbadał wpływ zarodkowania i rozrostu zarodków na rekrytalizację, następnie Ding i Guo (2001) zasymulowali proces rekrytalizacji dynamicznej z wyznaczeniem naprężenia uplastyczniającego przy wykorzystaniu modelu rozwoju dyslokacji. W ostatnim czasie zostało opublikowanych kilka obszernych przeglądów na temat zastosowania automatów komórkowych w inżynierii materiałowej: Raabe (2002), Janssens (2010), Yang *et al.* (2011).

Mukhopadhyay (2007) przedstawił wyniki symulacji rekrytalizacji statycznej z uwzględnieniem różnego rodzaju zarodkowania, a także wpływ gęstości dyslokacji, temperatury i dezorientacji granic na mobilność granic ziaren.

Podsumowując ten krótki przegląd metod badań i modelowania mikrostruktury podczas rekrytalizacji, należy zaznaczyć, że problemami, którym poświęcono zbyt mało uwagi, są:

- brak lub ograniczone możliwości wyboru kształtu rosnących ziaren,
- nieuwzględnienie rzeczywistego odkształcenia,
- niewprowadzenie zmiennej prędkości lub brak jej uzasadnienia,
- brak kompleksowych badań dotyczących związku między końcową mikrostrukturą a rzeczywistymi warunkami jej kształtowania.

Niektóre z tych problemów zostały rozwiązane i opisane w rozdziale 2 tej książki. Opracowano model frontálnego automatu komórkowego, który pozwala symulować rozrost ziarna o dowolnym kształcie w zależności od struktury materiału i warunków rozrostu.

Głównym mankamentem innych opartych na automatach komórkowych modeli jest modelowanie z zastosowaniem siatki, której kształt i rozmiary pozostają niezmiennie przez cały czas trwania symulacji. W niektórych procesach (takich jak na przykład przemiany fazowe i rekrytalizacja statyczna) jest to dopuszczalne, natomiast w przypadku innych (takich jak rekrytalizacja dynamiczna), w których odkształcenie i rozwój mikrostruktury odbywają się jednocześnie, wyników modelowania nie można uznać za adekwatne. Odkształcenie jest uwzględniane w wielu metodach (elementów skończonych, różnic skończonych, pól fazowych i innych), ale nie w automatach komórkowych. Natomiast przedstawiony w rozdziale 2 model oparty na frontalnych automatach komórkowych uwzględnia rzeczywiste odkształcenie podczas symulacji, co pozwala uzyskać dokładniejsze wyniki.

Zmienna prędkość rozrostu ziaren wpływa głównie na kinetykę i jest uwzględniana przy modelowaniu rekrytalizacji dynamicznej – bierze się wówczas pod uwagę zmiany gęstości dyslokacji jako siły napędowej rozrostu ziaren. Natomiast przy modelowaniu rekrytalizacji statycznej prędkość rozrostu ziaren we wszystkich znanych modelach jest stała.

Moim zdaniem dotychczasowe badania były skierowane na osiągnięcie celów cząstkowych, określenie oddzielnych cech zjawiska. Niektórzy z zajmujących się tym tematem badali kinetykę, inni uzyskaną strukturę, jeszcze inni rozwój dyslokacji lub napężenie uplastyczniające. Poza tym rekrytalizację dynamiczną badano bez uwzględnienia odkształcenia, a rekrytalizację statyczną – bez uwzględnienia zdrowienia.

Dlatego też podjąłem w tym rozdziale próbę rozwikłania nierozwiązanych do tej pory problemów, których rozwiązanie pozwoli stworzyć narzędzie do kompleksowych badań i adekwatnego modelowania rozwoju mikrostruktury. Nie opracowano jeszcze ostatecznego wariantu takiego narzędzia, poczyniono jedynie pierwsze ku temu kroki.

W rozdziale niniejszym zostaną przedstawione główne założenia modelu rekryształizacji, model zarodkowania, rozrostu ziaren, rozwoju dyslokacji, a także wstępne wyniki modelowania rekryształizacji po odkształceniu (statyczna-metadynamiczna) i podczas odkształcenia (dynamiczna). Zostanie także wskazany kierunek dalszych prac.

8.2. Główne założenia do modelowania rekryształizacji

Podstawą opisywanego modelu jest frontalny automat komórkowy (rys. 4.9) przedstawiony w rozdziale 4. W FCA słowo $(I_0 \vee I_4)I_1(I_2 \vee I_3)$ z początkowymi stanami q_2 i q_3 opisuje rekryształizację. FCA działa w ten sam sposób jak podczas tworzenia początkowej mikrostruktury, ale wykorzystując inny stan początkowy oraz inny warunek zarodkowania i rozrostu zrekrystalizowanych ziaren. Początkowa mikrostruktura powinna spełniać wymagania co do wielkości ziaren, ich kształtu i orientacji. Bardzo ważne przy modelowaniu rekryształizacji są zarodkowanie (warunek I_0) i prędkość migracji granic ziaren (prędkość rozrostu – warunek I_1).

Zgodnie z zasadami przedstawionymi w podrozdziale 1.4 należy określić podstawowe cechy (niektóre zostaną pominięte) automatów komórkowych w odniesieniu do wybranego procesu rekryształizacji.

Geometria. Główną różnicą w porównaniu z poprzednio omówionymi procesami (kształtowanie początkowej mikrostruktury, krzepnięcie) jest to, że wymiary i kształt przestrzeni podczas symulacji nie są stałe, lecz zmieniają się.

Otoczenie. W przedstawionych przykładach stosowano dwa rodzaje otoczenia. Podczas rozrostu ziaren dla komórek w stanie frontalnym otoczenie Moore’a decyduje o tym, które komórki będą uczestniczyć w procesie rozrostu, natomiast po przejściu przez komórkę ruchomej granicy rosnącego ziarna do wyznaczenia, czy komórka znajduje się wewnątrz ziarna, czy na jego granicy, stosowano otoczenie von Neumanna.

Stan początkowy. Struktura początkowa powinna zostać uzyskana przed rozpoczęciem modelowania; do tego celu służy program opisany w rozdziale 6.

Warunek zatrzymania. Kryterium końca obliczeń jest zadany czas modelowanego procesu lub koniec rekryształizacji: przejście wszystkich komórek w stan końcowy (q_2 lub q_3).

Model, umownie nazywany modelem rekryształizacji, faktycznie składa się z kilku części: modelu odkształcenia automatów komórkowych, modelu rozwoju dyslokacji, modelu zarodkowania i modelu prędkości rozrostu ziaren.

Uwzględnienie w automatach komórkowych geometrii rzeczywistego odkształcenia jest ważne, ponieważ pozwala modelować ewolucję mikrostruktury, kiedy jej kształtowanie przebiega w kilku procesach (odkształcenie, zarodkowanie i rozrost ziaren), zarówno jednocześnie (jak przy rekryształizacji dynamicznej), jak i sekwencyjnie (jak w procesach wieloetapowych z odkształceniem i rekryształizacją statyczną).

Model rozwoju dyslokacji wnosi do modelu rekrytalizacji informację niezbędną do wyznaczenia warunków zarodkowania oraz rozrostu ziaren zarówno podczas odkształcenia, jak i po nim. Model naprężenia uplastyczniającego jest ogniwem łączącym procesy zachodzące w skali mikro (dyslokacji) i skutki obserwowalne, globalne (naprężenie). Naprężenie, które łatwo wyznacza się w badaniach doświadczalnych, służy do weryfikacji poprawności działania modeli.

Model rekrytalizacji dynamicznej składa się z dwóch podstawowych części: zarodkowania nowych ziaren i ich rozrostu. Rozwój mikrostruktury zależy głównie od prędkości zarodkowania ziaren i prędkości ich rozrostu. Dodatkowym czynnikiem jest krytyczna gęstość dyslokacji, od której wartości zależy rozpoczęcie procesu rekrytalizacji.

Główne założenia modelu wymieniono poniżej:

- Wszystkie komórki jednego ziarna mają zbliżoną gęstość dyslokacji. Innymi słowy, gęstość dyslokacji przypisana jest do ziarna, a w odniesieniu do odrębnej komórki może być wprowadzony niewielki czynnik losowy.
- Początkowa gęstość dyslokacji wszystkich ziaren jest taka sama, równa minimalnej wartości w aktualnych warunkach termicznych.
- Gęstość dyslokacji wzrasta zgodnie z równaniami teorii dyslokacji, które uwzględniają umocnienie i zdrowienie.
- Zarodkowanie rozpoczyna się, gdy gęstość dyslokacji w ziarnie osiągnie wartość krytyczną ρ_c .
- Zarodki powstają wyłącznie na granicach ziaren, głównie podczas odkształcenia.
- Prędkość zarodkowania zależy od gęstości dyslokacji, odkształcenia, prędkości odkształcenia i temperatury.
- Gęstość dyslokacji nowego, zrekrystalizowanego ziarna (zarodka) spada do wartości minimalnej w aktualnych warunkach termicznych.
- Prędkość migracji granic ziaren jest funkcją temperatury, gęstości dyslokacji, krystalograficznej orientacji i innych czynników.

Modele gęstości dyslokacji, zarodkowania i rozrostu ziaren zostaną opisane w kolejnych podrozdziałach.

8.3. Podstawowe modele

8.3.1. Rozwój dyslokacji

Jako model rozwoju gęstości dyslokacji często stosuje się równanie różniczkowe, które uwzględnia umocnienie materiału U i zdrowienie dynamiczne Ω :

$$\frac{d\rho}{d\varepsilon} = U\dot{\varepsilon}^n - \Omega\rho \quad (8.1)$$

Niestety, równanie (8.1) niedostatecznie dokładnie odzwierciedla zmienne warunki odkształcenia oraz nie bierze pod uwagę zmiany gęstości dyslokacji po odkształceniu. Dlatego opracowano model rozwoju dyslokacji, który uwzględnia zarówno zmienne warunki odkształcenia, jak i jego brak, wprowadzając zdrowienie statyczne. Ponieważ większość omówionych w tym rozdziale badań dotyczy modelowania rozwoju mikrostruktury stali na gorąco, której ziarna mają strukturę kryształu typu FCC, zastosowano w nich multiplikatywny model gęstości dyslokacji:

$$\rho = k_v \rho_c \quad (8.2)$$

Wzór (8.1) zawiera dwa czynniki, jeden z nich, ρ_c , odpowiada zmianom wolniejszym, ale widocznym zarówno w procesach ze stałymi, jak i zmiennymi warunkami odkształcenia, natomiast czynnik k_v pozwala uwzględnić zmiany warunków odkształcenia.

Podstawą do wyznaczenia czynnika ρ_c jest wzór (8.1), w którym pochodna odkształcenia została zastąpiona pochodną czasu, wykonano linearyzację umocnienia oraz dodano człon R , odpowiadający za zdrowienie statyczne. W wyniku tych modyfikacji równanie na ρ_c można zapisać w następujący sposób:

$$\frac{d\rho_c}{dt} = (U - \Omega\rho_c)\dot{\epsilon} - R\rho_c \quad \forall \quad \rho_c(\epsilon = 0) = \rho_{c0} \quad (8.3)$$

gdzie: $U = 1$ odpowiada generacji oraz gromadzeniu się dyslokacji (umocnienie materiału) i jest wielkością względną, a Ω odpowiada za anihilację dyslokacji (zdrowienie dynamiczne). Ostatni człon wzoru (często pomijany), R , odzwierciedla zdrowienie statyczne, które wraz z rekrytalizacją prowadzi do mięknięcia materiału. Podczas odkształcenia zachodzą wszystkie trzy procesy, a w przerwach tylko ostatni, czyli zdrowienie statyczne.

Dla czynnika k_v , odzwierciedlającego zmiany warunków odkształcenia, zastosowano następujący wzór:

$$\epsilon_v \frac{dk_v}{d\epsilon} + k_v = AZ^n \quad (8.4)$$

gdzie:

Z – parametr Zenera–Hollomona,

A i n – współczynniki materiałowe.

W przedstawionym tu modelu czynnik k_v w stanie ustalonym osiąga wartość AZ^n , w innych powszechnych modelach zmienia się on skokowo i po wprowadzeniu do równania (8.3) daje bezwzględną wartość umocnienia $U(\epsilon, T) \neq 1$.

8.3.2. Naprężenie uplastyczniające

Model gęstości dyslokacji zazwyczaj jest częścią modelu naprężenia uplastyczniającego i spełnia wobec niego wtórną, pomocniczą rolę. W automatach komórkowych bardziej istotna jest znajomość gęstości dyslokacji, ponieważ ona decyduje o całym procesie rekrytalizacji, wpływając zarówno na zarodkowanie, jak i na rozrost ziaren. Natomiast naprężenie uplastyczniające rzadko stanowi cel badań, najczęściej jest jedynie czynnikiem pozwalającym zweryfikować działanie modelu gęstości dyslokacji w różnych warunkach.

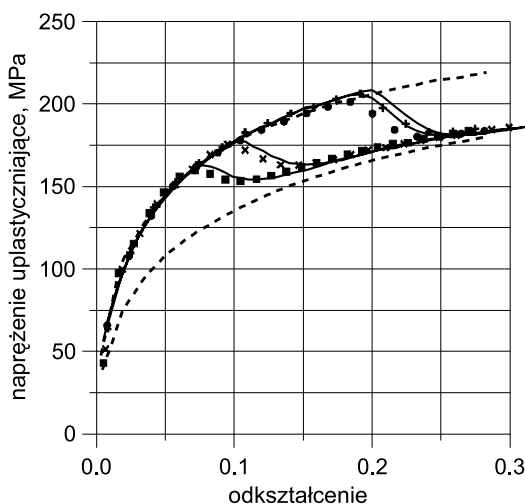
Tak więc naprężenie uplastyczniające modelowane za pomocą automatów komórkowych określa wzór, w którym zamiast makroskopowej gęstości dyslokacji ρ wykorzystywano średnią gęstość dyslokacji ρ_{av} w reprezentatywnej modelowej objętości:

$$\sigma = \sigma_0 + \alpha G b \sqrt{\rho_{av}} \quad (8.5)$$

gdzie:

- σ_0 – naprężenie bazowe;
- G – moduł ścinania;
- α – współczynnik uwzględniający współdziałanie dyslokacji;
- b – długość wektora Burgersa (wartość rzędu 10^{-11} m).

Gęstość dyslokacji ρ_{av} jest liczona jako średnia dla wszystkich komórek przestrzeni lub dla wszystkich ziaren z uwzględnieniem ich objętości. Rysunek 8.1 przedstawia porównanie wyników obliczeń modelem z uzyskanymi za pomocą prób plastometrycznych przy zmniejszeniu prędkości odkształcenia. Symbole i linie przerywane reprezentują badania plastometryczne, linie ciągłe – model.



Rys. 8.1. Zmiany naprężenia uplastyczniającego przy zmiennych warunkach odkształcenia
 $T = 800^\circ\text{C}$, $\dot{\epsilon} = 1 \rightarrow 0,1 \text{ s}^{-1}$

8.3.3. Zarodkowanie

Jak wspomniano powyżej, siła napędowa rekrytalizacji i dość wysoka energia granic wysokokątowych prowadzą głównie do zarodkowania heterogenicznego. Tak więc głównym warunkiem zarodkowania jest uzyskanie dość dużego obszaru o niskiej gęstości defektów struktury krystalicznej oraz obecność granic wysokokątowych.

Udowodniono doświadczalnie istnienie przynajmniej trzech mechanizmów zarodkowania:

- 1) migracja istniejących wysokokątowych granic,
- 2) migracja niskokątowych granic (komórki dyslokacyjne, podziarna),
- 3) koalescencja podziaren.

Pierwszy mechanizm jest bardzo ważny przy odkształceniach do 0,4 – występuje wówczas jako mechanizm główny, a czasem jedyny. Drugi mechanizm jest związany z dużym odkształceniem i dużym rozrzutem wielkości podziaren, a występuje przy wyższych temperaturach i głównie w metalach o małej energii błędu ułożenia. Natomiast trzeci mechanizm, związany z pasmami ścinania, z dużym rozrzutem kątów dezorientacji granic podziaren, ze średnimi odkształceniami, występuje w obszarach w pobliżu granic ziaren, przy stosunkowo niskich temperaturach i w metalach o wysokiej energii błędu ułożenia.

Obecnie stosowane są dwa podejścia do modelowania zarodkowania podczas odkształcenia za pomocą automatów komórkowych. W pierwszym z nich, zaproponowanym przez Dinga i Guo (2001), praktycznie nie rozpatruje się mechanizmu zarodkowania, chociaż wykorzystuje się heterogeniczność pierwszego mechanizmu. Zgodnie z tą propozycją zarodkowanie zachodzi przy granicach ziaren (granicach pierwotnych lub nowych granicach zrekrystalizowanych ziaren). Innymi słowy, tylko te komórki, które należą do granic ziaren (nieruchomych), mogą stać się miejscami zarodkowania. Prędkość zarodkowania na jednostkę powierzchni granicy ziarna (długości w modelach dwuwymiarowych) w procesie rekrytalizacji dynamicznej jest zależna od temperatury T oraz prędkości odkształcenia $\dot{\epsilon}$ i oblicza się ją następującym wzorem:

$$\dot{N}(\dot{\epsilon}, T) = C \dot{\epsilon}^m \exp\left(-\frac{Q_n}{RT}\right) \quad (8.6)$$

gdzie:

C – stała;

m – wykładnik o wartości bliskiej jeden ($m = 0,9$);

Q_n – energia aktywacji zarodkowania.

Krytyczną gęstość dyslokacji w przypadku rekrytalizacji zachodzącej na granicach ziaren można obliczyć na podstawie energii swobodnej, jak to zaproponowali Roberts i Ahlblom (1978).

Krytyczną wartość gęstości dyslokacji ρ_c wyznacza się za pomocą wzoru:

$$\rho_c = \frac{8\gamma}{\tau l} = \sqrt[3]{\frac{20\gamma\dot{\epsilon}}{3blM\tau^2}} \quad (8.7)$$

gdzie:

γ – energia granicy ziarna;

τ – energia linii dyslokacji $\tau = Gb^2/2$;

l – swobodna droga dyslokacji mobilnych;

M – współczynnik kinetyczny ruchu granic ziaren (mobilności granic).

Model zarodkowania jest stochastyczny, tzn. przewiduje, że jeżeli w istniejącej komórce (ziarnie) gęstość dyslokacji osiągnie wartość krytyczną ($\rho \geq \rho_c$), to w tej komórce powstanie nowe ziarno z prawdopodobieństwem P_N , które przyjmuje stałą wartość. W takich warunkach prędkość zarodkowania zależy od powierzchni granic, a więc od wielkości ziarna (w tym nowych, zrekrytalizowanych ziaren). Należy podkreślić, że model ten nie uwzględnia powstawania nowych ziaren wewnątrz już istniejących (na przykład na granicach podziaren, bliźniaków lub będących częstkami innej fazy i wydzieleń). W przypadku każdej komórki znajdującej się na granicy ziarna z gęstością dyslokacji większą od krytycznej prawdopodobieństwo zarodkowania nowego ziarna P_N można zapisać w następujący sposób:

$$P_N = \frac{\dot{N}\Delta t}{N_c} \quad (8.8)$$

gdzie:

Δt – krok czasowy;

N_c – liczba komórek na granicach ziaren na jednostkę powierzchni dla 3D CA $N_c = 1/F_{CA}$ lub na jednostkę długości dla 2D CA $N_c = 1/L_{CA}$;

F_{CA} i L_{CA} – odpowiednio: powierzchnia ściany (jeśli komórka 3D) i długość boku komórki (jeśli komórka 2D).

Zaproponowane przez Jina i Cui (2010) nieco inne podejście opiera się bezpośrednio na pierwszym mechanizmie zarodkowania. Zakładając, że prędkość zarodkowania prawdopodobnie zależy od odkształcenia, a ono może być skutecznie przedstawione za pomocą poziomu gęstości dyslokacji, opracowano model zarodkowania zależnego od dyslokacji, w którym prędkość zarodkowania na granicach ziaren \dot{N} jest proporcjonalna do $\rho^{1/2}$. Idea ta bazuje na modelu rekrytalizacji dynamicznej opracowanym przez Derby'ego i Ashby'ego (1987) dla mechanizmu zarodkowania opartego na migracji wysokokątowych granic ziaren. W modelu tym, kiedy materiał zbudowany z ziaren o średniej wielkości d_{av} jest poddawany odkształceniu na gorąco, wewnątrz ziaren jest kształtowana struktura komórkowa (lub powstają podziarna) o wielkości komórek δ_{av} . Granice podziaren (lub ściany komórek) losowo przecinają granice ziaren

i dzielą je na segmenty o średniej długości δ_{av} . Właśnie różnica w liczbie granic podziaren po obu stronach segmentu granicy jest siłą napędową zarodkowania.

Jeżeli długość segmentu l z jednej strony można wyrazić rozkładem wykładniczym, a liczbę przecięć tego samego segmentu z innej strony n – rozkładem Poissona, wówczas liczba segmentów z n przecięciami z jednej strony można wyrazić następującym wzorem:

$$N(n) = \frac{1}{2^{n+1}l} \quad (8.9)$$

Derby i Ashby (1987) stwierdzili, że segment granicy będzie się wyrzyszał, kształtując zarodek, gdy $n \geq n_{cr}$. Przy założeniu, że długość segmentu l będzie równa wielkości komórki (podziarna) δ_{av} , liczba zarodków generowanych na jednostkę długości granicy ziaren będzie następująca:

$$N = \sum_{n_{cr}}^{\infty} N(n) = \frac{1}{2^{n_{cr}} \delta_{av}} \quad (8.10)$$

Ponieważ średnia wielkość struktury komórkowej (podziaren) jest odwrotnie proporcjonalna do $\delta^{1/2}$, liczbę zarodków można obliczyć za pomocą następującego wzoru:

$$N = B\sqrt{\rho} + C \quad (8.11)$$

Na segmencie granicy, gdzie gęstość dyslokacji osiągnęła wartość krytyczną ρ_{cr} , następuje zarodkowanie z idealnym nasyceniem, które rzadko obserwuje się w rzeczywistych materiałach. Zatem generowano N zarodków na początkowych granicach i więcej zarodków nie powstanie z wyjątkiem sytuacji, kiedy gęstość dyslokacji się zmieni. Tak więc zakłada się tu (podobnie jak w poprzednim podejściu) zarodkowanie ze stałą prędkością. Prędkość zarodkowania na granicach ziaren \dot{N} będzie więc stała i proporcjonalna do liczby zarodków N , która zależy od gęstości dyslokacji ρ . Prędkość zarodkowania wyraża się następującym wzorem:

$$\dot{N} = B'\sqrt{\rho} + C' \quad (8.12)$$

Jeżeli założyć ustaloną wartość I_s prędkości zarodkowania przy pełnej rekrytalizacji dynamicznej, kiedy gęstość dyslokacji również osiąga wartość ustaloną $\rho_s = k_{vs} = AZ^n$ (8.4), to graniczne prędkości zarodkowania można opisać następującym układem równań:

$$\begin{cases} \dot{N} = 0 & \forall \rho = \rho_{cr} \\ \dot{N} = I_s & \forall \rho = \rho_s \end{cases} \quad (8.13)$$

Podstawiając (8.13) do (8.12), można uzyskać:

$$\dot{N} = \frac{\sqrt{\rho} - \sqrt{\rho_{cr}}}{\sqrt{\rho_s} - \sqrt{\rho_{cr}}} I_s \quad (8.14)$$

Prawdopodobieństwo zarodkowania P_N można obliczyć w ten sam sposób jak poprzednio, według wzoru (8.8).

8.3.4. Rozrost ziaren

Migracja wysokokątowych granic ziaren jest podstawowym mechanizmem ich rekrytalizacji i rozrostu. Główna różnica w rozroście ziarna podczas rekrytalizacji i przy jej braku dotyczy siły napędowej tego rozrostu. Siłą napędową rozrostu ziaren jest energia ich wysokokątowych granic, natomiast siłą napędową rekrytalizacji jest ujawniająca się w postaci defektów krystalicznych (głównie dyslokacji) energia odkształcenia.

Kiedy wysokokątowa granica porusza się, strumień atomów przechodzących przez granicę w jednym kierunku jest mniejszy niż w przeciwnym. Rozpatrując model, w którym każdy atom przekracza barierę aktywacji niezależnie od innych, przechodząc z ziarna odkształconego do zrekrystalizowanego, należy zaznaczyć przy założeniu $\Delta P \ll kT$:

$$v = \frac{ALf}{kT} \exp\left(\frac{-Q_b}{RT}\right) \Delta P \quad (8.15)$$

gdzie:

- v – prędkość migracji;
- ΔP – siła napędowa rekrytalizacji;
- Q_b – energia aktywacji granic;
- A – współczynnik materiałowy;
- L – grubość granicy;
- f – charakterystyczna częstotliwość przeskoku atomów z ziarna do ziarna;
- T – temperatura;
- k – stała Boltzmanna;
- R – stała gazowa.

Prędkość migracji granicy ziarna często przedstawiana jest w uproszczonej postaci, jako iloczyn dwóch czynników – mobilności granic m i siły napędowej rekrytalizacji ΔP :

$$v = m\Delta P \quad (8.16)$$

W rzeczywistości ΔP jest sumą sił napędzających i hamujących ΣF , które działają na granicy i które zostaną krótko omówione poniżej.

Część energii odkształcenia jest gromadzona w materiale w postaci defektów struktury, głównie w postaci dyslokacji. Jeżeli oznaczyć gęstość dyslokacji przed odkształceniem i po nim odpowiednio symbolami ρ_0 i ρ , to siła napędowa F_n będzie następująca (gęstość dyslokacji początkowa ρ_0 , ze względu na jej wartość, może być pominięta):

$$F_n = Gb^2 (\rho - \rho_0) \quad \text{lub} \quad F_n = Gb^2 \rho \quad (8.17)$$

gdzie:

G – moduł ścinana;

b – długość wektora Burgersa.

Związaną z krzywizną powierzchni siłą napędową rozrostu ziaren F_{gg} można wyrazić przez energię granicy γ i wielkość ziarna D :

$$F_{gg} = \frac{2\gamma}{D} \quad (8.18)$$

Siła napędowa rozrostu ziaren F_{gg} podczas rekrytalizacji często przeciwdziała rozrostowi nowych (w szczególności na początku ich rozrostu), ale jest przynajmniej o rząd mniejsza od F_n i dlatego często nie uwzględnia się jej przy modelowaniu rekrytalizacji.

Siła związana z wydzieleniami pierwiastków lub związków pojawia się, kiedy jednocześnie z rekrytalizacją następuje zmniejszenie przesylenia pierwiastków i ich wydzielenie. Może ona być znacznie większa od siły napędowej rekrytalizacji, która jednak powstaje tylko w warunkach przesyconego roztworu i nie będzie tu rozpatrywana.

Małe cząstki mogą hamować proces rekrytalizacji. W takim przypadku siła napędowa musi być na tyle duża, by granica mogła cząstki ominąć. Siła może zatem powodować, że granica przypina się do cząstki i przeciwdziała jej ruchowi. Rozróżnia się dwa mechanizmy: wchodzenie granicy do cząstek i ich przejście lub omijanie. Pierwszy mechanizm został opisany przez C. Zenera i wedle jego ustaleń siła hamująca F_p ma w nim następującą postać:

$$F_p = \frac{3\gamma V_p}{2r} \quad (8.19)$$

gdzie:

γ – energia granicy;

V_p – ułamek objętościowy wydzielen (cząstek);

r – średnia wielkość cząstek.

Drugi mechanizm wymaga dwukrotnie większej siły, ale jest ona i tak znacznie mniejsza od siły napędowej rekrytalizacji i porównywalna z siłą napędową rozrostu ziaren, a więc przy rozpatrywaniu rekrytalizacji można ją pominąć i uwzględnić

jedynie podczas modelowania rozrostu ziarna (na przykład po rekrytalizacji). Wpływ na siłę rozpuszczonych pierwiastków jest jeszcze mniejszy i w ogóle się go nie rozpatruje.

Tak więc przy modelowaniu konieczne jest, aby siły napędowe były większe od sił hamujących.

Drugim czynnikiem wpływającym na prędkość rozrostu zrekrytalizowanych ziaren (8.16) jest mobilność granicy m . Obecnie nie istnieje przekonująca teoria, która tłumaczyłaby zależność mobilności od różnych czynników, głównie od kąta dezorientacji. Ani równanie (8.15), ani (8.16) nie formułuje wprost zależności od kąta dezorientacji, chociaż wpływ tego kąta na prędkość rozrostu ziaren jest znany od dawna. Gdy kąty dezorientacji są małe, energia aktywacji jest bliska energii aktywacji w procesie samodyfuzji, uważano więc, że niskokątowe granice są kontrolowane przez dyfuzję i dlatego pozostają prawie nieruchome. Natomiast energia wysokokątowych granic ziaren jest prawie o połowę mniejsza, a o ich ruchu decyduje inny mechanizm, a mianowicie wymiana atomów przez granice (8.15). W wyniku wpływu orientacji granic na mobilność i energię aktywacji prędkość granic wysokokątowych ($\theta > 15^\circ$) wzrasta setki, a czasami nawet tysiące razy w porównaniu z granicami niskokątowymi ($\theta = 2-5^\circ$). Zgodnie z danymi otrzymanymi przez Ferry'ego i Humphreysa (1996), dla Al-0,05%Si mobilność granic podziaren wzrosła 14 razy w wyniku zmiany kąta z 2° do 5° i 2500 razy przy przejściu do granic wysokokątowych ($\theta > 15^\circ$). Dlatego wzór (8.16) jest rozszerzony o dodatkowy czynnik $f(\theta)$, wprowadzający zależność od kąta dezorientacji θ :

$$v = mf(\theta)\Delta P \quad (8.20)$$

Mobilność (niezależna od kąta) wysokokątowych granic można zapisać w następującej postaci:

$$m = \frac{A}{kT} \exp\left(\frac{-Q_b}{RT}\right) \quad (8.21)$$

Natomiast w przypadku granic niskokątowych należy uwzględnić zależność $f(\theta)$, którą wyraża następujący wzór aproksymujący:

$$f = 3,72 \cdot 10^{-4} \exp\left\{8 \left[1 - \exp\left(-\frac{\theta}{2,2}\right)\right]^{10}\right\} \quad (8.22)$$

Wówczas mobilność granicy o kącie 7° stanowi około 7% mobilności granic wysokokątowych i dlatego ruch granic o mniejszych kątach nie jest rozpatrywany. W tej sytuacji zarodki powstające w wyniku wybrzuszenia wysokokątowej granicy rosną tylko wewnątrz „obcego” ziarna, natomiast wewnątrz „swojego” ziarna nie rosną, ponieważ pomiędzy nimi wciąż istnieje granica niskokątowa.

8.4. Kinetyka rekrytalizacji

Kinetyka rekrytalizacji zależy od kilku czynników: warunków zarodkowania i rozrostu, parametrów przestrzeni i kształtu ziaren. Ponieważ są to symulacje wstępne, nie są one odniesione do rzeczywistych warunków. Rozpatrywano tylko dwa warianty zarodkowania: wysycone, następujące przed rozrostem ziaren i ciągle zarodkowanie ze stałą prędkością. Rozrost ziaren zadawano ze stałą prędkością. Symulacje rozpoczynano od badania parametrów przestrzeni: zarówno jej geometrii, jak i szczegółów dotyczących bezpośrednio automatów komórkowych. Następnie badano wpływ kształtu ziaren na kinetykę.

Kinetyka przemian, w tym rekrytalizacji, jako zmiana ułamka przemiany χ w funkcji czasu t , jest często opisywana równaniem Avramiego:

$$\chi = 1 - \exp(-at^n) \quad (8.23)$$

gdzie:

n – wykładnik Avramiego;

a – współczynnik.

W takim przypadku przyjmuje się, że najważniejszym parametrem jest wykładnik Avramiego n . W celu określenia i porównania kilku wariantów symulacji z wykładnikiem Avramiego poniższe wykresy dotyczące kinetyki rekrytalizacji przedstawiano w tak zwanym układzie Avramiego, w którym zastosowano anamorficzne przekształcenie osi. Oś odciętych przedstawia logarytm czasu $\ln(t)$, natomiast oś rzędnych jest bardziej skomplikowaną funkcją ułamka rekrytalizacji χ : $\ln(-\ln(1-\chi))$. W wyniku takiego przekształcenia równanie Avramiego przyjmuje postać linii prostej, której nachylenie odpowiada wykładnikowi n , natomiast współczynnik a można wyznaczyć dla wartości czasu $t = 1$ ($\ln(t) = 0$), wówczas $\ln(a)$ jest równy wartości odciętej dla tej chwili czasu. W celu parametryzacji kinetyki rekrytalizacji na osi rzędnych można zaznaczyć najbardziej charakterystyczne wartości ułamka rekrytalizacji, tzn. $\chi = 0,05$ odpowiada $-2,97$, $\chi = 0,5$ odpowiada $-0,366$, a $\chi = 0,95$ odpowiada $1,097$. Wynika z tego, że wartości mniejszych od $-3,0$ i większych od $1,0$ można nie uwzględniać.

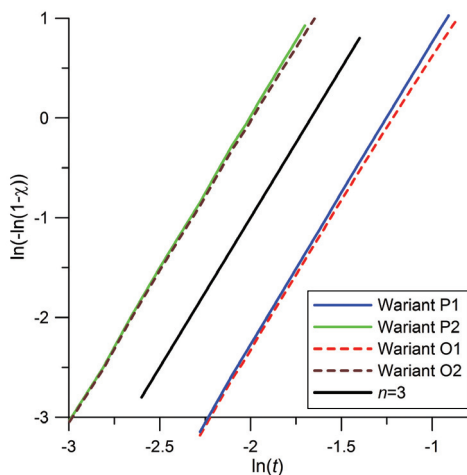
8.4.1. Warunki brzegowe

Celem badania wpływu warunków brzegowych na kinetykę rekrytalizacji jest potwierdzenie przewagi warunków brzegowych zamkniętych nad półotwartymi. Zgodnie z klasyfikacją przedstawioną w rozdziale 3 równoważnymi do zamkniętych warunków brzegowych ze względu na kinetykę są warunki otwarte i kombinowane. Tak więc porównanie wpływu różnych warunków brzegowych na kinetykę będzie tu ograniczone tylko do warunków periodycznych i półotwartych. Na rysunku 8.2 przedstawiono dwa warianty zarodkowania ziarn. Symulację przeprowadzono przy wykorzystaniu

trójwymiarowych automatów komórkowych o następujących parametrach: $n_x \times n_y \times n_z = 400 \times 400 \times 400$ komórek, $l_x \times l_y \times l_z = 400 \times 400 \times 400 \mu\text{m}$. Liczba zarodków wynosi 500 i 5000.

Półotwarte warunki brzegowe powodują pewne wydłużenie procesu, ponieważ wypełnienie niezrekrytalizowanej przestrzeni w pobliżu brzegów, ze względu na brak ziaren wrastających do przestrzeni z zewnątrz, następuje wolniej. Efekt ten jest tym wyraźniejszy, im mniejsza liczba ziaren jest wykorzystywana do symulacji, a ponieważ przy mniejszych rozmiarach przestrzeni stosowano mniejszą liczbę ziaren, to i na przestrzeniach również ten efekt jest bardziej widoczny. Powoduje to również niewielkie zmniejszenie się wartości wykładnika Avramiego pod koniec przemiany. W przypadku ciągłego zarodkowania różnice są nieco mniejsze, ponieważ zarodkowanie przy brzegach zachodzi nieznacznie szybciej, ze względu na większą objętość niezrekrytalizowanych ziaren w tych obszarach.

Chociaż różnice są niewielkie, wyniki potwierdzają słuszność stosowania zamkniętych warunków brzegowych. Nawet jeżeli różnice te mogą być pominięte, dalsze symulacje były prowadzone z zamkniętymi warunkami brzegowymi.



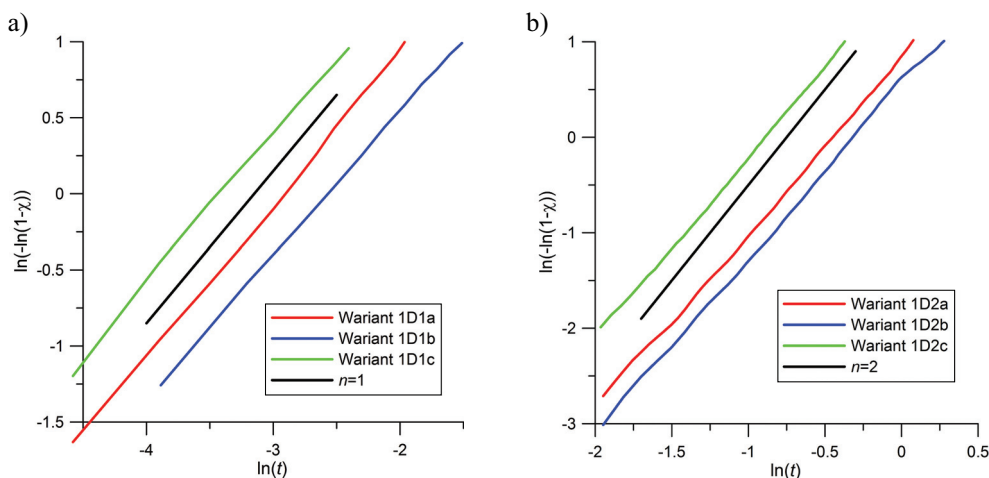
Rys. 8.2. Kinetyka przemiany modelowanej z periodycznymi (linie ciągłe) i półotwartymi (linie przerywane) warunkami brzegowymi: P – warunki periodyczne, O – warunki półotwarte

8.4.2. Wymiar przestrzeni

Ponieważ modelowanie prowadzono za pomocą trójwymiarowych automatów komórkowych, w przypadku modelowania przestrzeni jedno- i dwuwymiarowej liczbę komórek w dwóch lub jednym kierunku ograniczono do dziesięciu. W ten sposób wpływ tych kierunków został ograniczony do kilku pierwszych iteracji. Należy również zauważyć,

że ze względu na nakłady obliczeniowe modelowanie trójwymiarowymi automatami komórkowymi przestrzeni jedno- i dwuwymiarowych jest bardzo niekorzystne i zostało przeprowadzone jedynie dlatego, że opracowanie osobnego oprogramowania do modelowania tego typu zjawisk wymagałoby jeszcze większych nakładów. Symulacje przeprowadzono po kolei przy wykorzystaniu automatów jedno-, dwu- i trójwymiarowych z zarodkowaniem wysyconym i ciągłym. Ich wyniki przedstawiono na rysunkach.

Na rysunku 8.3 zaprezentowano wyniki symulacji za pomocą automatu jednowymiarowego, który jest wydłużonym prętem o przekroju kwadratowym z $n_y \times n_z = 10 \times 10$ komórek. Trzeci wymiar jest wielokrotnie większy, co powinno wyeliminować wpływ dwóch pozostałych wymiarów. Dodatkowym ograniczeniem jest uzyskanie w końcowej strukturze takich ziaren, w których rozmiar ziarna wzdłuż pręta będzie znacznie większy od rozmiarów w przekroju pręta. Wynika z tego, że liczba ziaren nie powinna być zbyt duża. Jako komórkę do symulacji wybrano sześciąt o długości krawędzi $1 \mu\text{m}$. A zatem na rysunku 8.3a przedstawiono trzy warianty symulacji ze stałą prędkością rozrostu i zarodkowaniem przed rozrostem.



Rys. 8.3. Kinetyka przemiany uzyskana z wykorzystaniem jednowymiarowych automatów komórkowych z zarodkowaniem przed rozrostem (a) i z ciągłym zarodkowaniem podczas rozrostu ziaren (b)

Warianty te różnią się liczbą komórek wzdłuż pręta n_x i liczbą zarodków N_n :

- wariant 1D1a: $n_x = 5000$, $N_n = 50$;
- wariant 1D1b: $n_x = 10000$, $N_n = 70$;
- wariant 1D1c: $n_x = 10000$, $N_n = 150$.

Na rysunku 8.3a również jest widoczna linia o nachyleniu odpowiadającym wykładnikowi Avramiego $n = 1$, która stanowi dowód na to, że wyniki symulacji odpowiadają właśnie takiemu wykładnikowi Avramiego.

Kolejne trzy warianty symulacji z wykorzystaniem jednowymiarowych automatów komórkowych przedstawiono na rysunku 8.3b. Różnią się one między sobą długością pręta n_x i prędkością zarodkowania \dot{N} , która jest wyznaczana jako początkowa prędkość zarodkowania w całej modelowej przestrzeni w jednostce czasu. Część niezrekrytalizowana podczas rekrytalizacji zmniejsza się, proporcjonalnie zmniejsza się również liczba zarodków w jednostce czasu, czyli prędkość jest odniesiona do niezrekrytalizowanej objętości:

- wariant 1D2a: $n_x = 5000$, $\dot{N} = 100 \text{ s}^{-1}$;
- wariant 1D2b: $n_x = 10000$, $\dot{N} = 150 \text{ s}^{-1}$;
- wariant 1D2c: $n_x = 10000$, $\dot{N} = 500 \text{ s}^{-1}$.

W przypadku symulacji z ciągłym zarodkowaniem niezerowe rozmiary przekroju pręta wpływają na kinetykę w początkowej fazie modelowania i dlatego odcinki te nie zostały uwidocznione na wykresach. Warto też podkreślić że nachylenie prostej ma wartość zbliżoną do teoretycznej wartości $n = 2$.

Symulację przy użyciu dwuwymiarowych automatów komórkowych przedstawiono na rysunku 8.4. Tym razem modelowano cienką kwadratową płytkę o tych samych rozmiarach zarówno pod względem liczby komórek: $n_x \times n_y \times n_z = 1000 \times 1000 \times 10$, jak i w jednostkach długości: $l_x \times l_y \times l_z = 1000 \times 1000 \times 1 \text{ }\mu\text{m}$. Należy zwrócić uwagę, że w wyniku dziesięciokrotnego zmniejszenia grubości każdej komórki dodatkowo zmniejszono grubość płytki. Pierwsza seria wyników (rys. 8.4a) dotyczy zarodkowania przed rozrostem. Warianty różnią się od siebie jedynie liczbą zarodków:

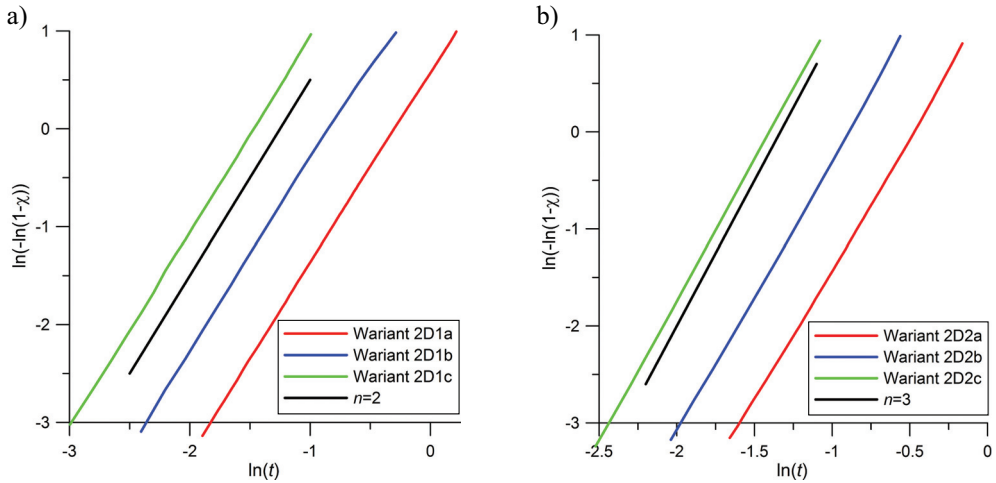
- wariant 2D1a: $N_n = 50$;
- wariant 2D1b: $N_n = 150$;
- wariant 2D1c: $N_n = 500$.

Druga seria wyników (rys. 8.4b) dotyczy zarodkowania ciągłego. Warianty różnią się prędkością zarodkowania:

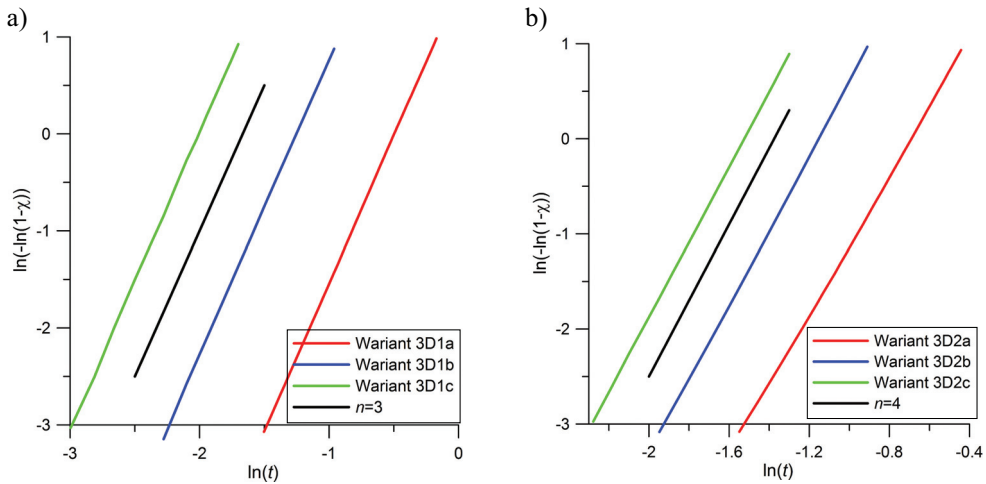
- wariant 2D2a: $\dot{N} = 250 \text{ s}^{-1}$;
- wariant 2D2b: $\dot{N} = 1000 \text{ s}^{-1}$;
- wariant 2D2c: $\dot{N} = 5000 \text{ s}^{-1}$.

W wyniku modelowania dwuwymiarowymi automatami komórkowymi również uzyskano kinetykę zbliżoną do teoretycznej, opisaną równaniem Avramiego z wykładnikiem $n = 2$ i $n = 3$ odpowiednio przy zarodkowaniu przed rozrostem i z ciągłym zarodkowaniem podczas przemiany.

Ostatnią serię badań wpływu wymiarowości automatów komórkowych na wyniki symulacji przeprowadzono na automatach trójwymiarowych. Wykorzystano taką samą jak w poprzednich badaniach przestrzeń, która zawierała $n_x \times n_y \times n_z = 400 \times 400 \times 400$ komórek o wymiarach $l_x \times l_y \times l_z = 400 \times 400 \times 400 \text{ }\mu\text{m}$. Wyniki owych badań przedstawiono na rysunku 8.5.



Rys. 8.4. Kinetyka przemiany otrzymana za pomocą dwuwymiarowych automatów komórkowych z zarodkowaniem przed rozrostem (a) i z ciągłym zarodkowaniem podczas rozrostu ziaren (b)



Rys. 8.5. Kinetyka przemiany otrzymana przy użyciu trójwymiarowych automatów komórkowych z zarodkowaniem przed rozrostem ziaren (a) i z ciągłym zarodkowaniem podczas ich rozrostu (b)

Na rysunku 8.5a przedstawiono wyniki symulacji trzech wariantów zarodkowania przed rozrostem N_n :

- wariant 3D1a: $N_n = 50$;
- wariant 3D1b: $N_n = 500$;
- wariant 3D1c: $N_n = 5000$.

Z kolei na rysunku 8.5b porównano kinetykę przy ciągłym zarodkowaniu podczas całego procesu rekrytalizacji dla następujących prędkości zarodkowania \dot{N} :

- wariant 3D2a: $\dot{N} = 500 \text{ s}^{-1}$;
- wariant 3D2b: $\dot{N} = 4000 \text{ s}^{-1}$;
- wariant 3D2c: $\dot{N} = 20\,000 \text{ s}^{-1}$.

Wyniki przedstawione na rysunkach 8.3–8.5 umożliwiają wyciągnięcie pewnych wniosków – wyniki te są całkowicie zgodne z teoretyczną wartością wykładnika Avramiego dla każdego wariantu przestrzeni i warunków zarodkowania. Świadczy to o tym, że główne założenia modelu są słuszne, a także o tym, że automaty komórkowe pozwalają z dużą dokładnością symulować procesy przemiany. Zatem rezultat testowania za pomocą automatów komórkowych kinetyki w ustalonych warunkach teoretycznych można uznać za zadowalający.

8.4.3. Kształt ziaren

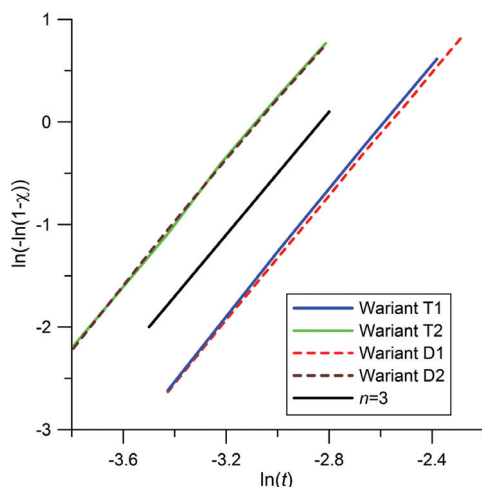
W literaturze przedmiotu często można spotkać się z opinią, iż kinetyka zależy od kształtu rosnących ziaren. Nie podważając wpływu różnych prędkości rozrostu w różnych kierunkach na współczynnik a we wzorze Avramiego (8.23), należałoby jednak odrzucić błędne stwierdzenie o zależności wykładnika n od kształtu ziaren. W celu udowodnienia tej tezy przeprowadzono symulacje zbliżone do wariantu 3D1c, czyli w przestrzeni trójwymiarowej zbudowanej z $n_x \times n_y \times n_z = 400 \times 400 \times 400$ komórek i o wymiarach $l_x \times l_y \times l_z = 400 \times 400 \times 400 \text{ }\mu\text{m}$. Rozmieszczono w tej przestrzeni 5000 zarodków. Jednak kształt rosnących ziaren w tej symulacji był inny niż w wariancie 3D1c – tam były to kule, natomiast tym razem wybrano cylindry, przy czym w wariancie T miały one postać nici o wysokości pięćdziesięciokrotnie większej od promienia, natomiast w wariancie D przybrały postać płaskich tarczy, których promień był pięćdziesięciokrotnie większy od grubości (wysokości). W wariancie T był więc preferowany jeden kierunek rozrostu, w wariancie D – dwa, natomiast w poprzednich symulacjach wszystkie trzy kierunki były równoważne. Ponieważ nici i tarcze mogą być w różny sposób zorientowane w przestrzeni, rozpatrywano dwa warianty ich orientacji: pierwszy (1) odpowiada sytuacji, w której wszystkie cylindry są zorientowane tak samo, drugi (2) – kiedy przestrzenna orientacja każdego ziarna jest wybierana losowo. Wyniki symulacji tych czterech wariantów (T1, T2, D1 i D2) w postaci wykresów Avramiego przedstawiono na rysunku 8.6.

Na podstawie analizy rysunku można wysnuć dwa wnioski:

- kształt ziarna nie wpływa na wykładnik Avramiego n ,
- przestrzenna orientacja ziarna nie wpływa na kinetykę rekrytalizacji.

Brak wpływu kształtu na wykładnik Avramiego można wytłumaczyć tym, że mimo anizotropii w rozroście ziarna podczas tego rozrostu proporcje między wymiarami ziarna pozostają bez zmian i objętość ziarna nadal wzrasta w funkcji sześciannu czasu.

Jeżeliby ziarna rosły tylko w jednym lub dwóch kierunkach, to przy niezmiennych pozostałych wymiarach takie ziarna nie mogłyby wypełnić trójwymiarowej przestrzeni. Niektóre drobne różnice w kinetyce w zależności od orientacji ziaren być może były spowodowane czynnikiem losowym.



Rys. 8.6. Kinetyka przemiany otrzymana za pomocą trójwymiarowych automatów komórkowych z zarodkowaniem przed rozrostem (a) i z ciągłym zarodkowaniem podczas rozrostu ziaren (b)

Reasumując to, co tutaj napisano i uwzględniając treść poprzedniego podrozdziału, można stwierdzić, iż wykładnik n zależy od właściwości przestrzeni, a nie od charakteru rozrostu (czyli od kształtu) ziaren. Zatem wykładnik o mniejszej wartości jest przeważnie wynikiem trójwymiarowego rozrostu ziaren w ograniczonej przestrzeni, a nie ukierunkowanego (ze względu na ich kształt) rozrostu ziaren w przestrzeni trójwymiarowej. Jednak ukierunkowany rozrost ziaren ze względu na wysycenie miejsc zarodkowania może spowodować zmniejszenie wartości wykładnika n . I tak, na przykład w przypadku kinetyki krystalizacji wyznaczonej w rozdziale 7 wykładnik w warunkach przedstawionych na rysunkach 7.2–7.5 wynosi $n = 1$, a w warunkach zilustrowanych na rysunku 7.6 i w większości przypadków dotyczących krzepnięcia w maszynie COS jest on równy lub zbliżony do dwóch: $n \approx 2$. W warunkach rekrystalizacji lub przemian fazowych wysycenie miejsc zarodkowania nie doprowadza do tak istotnych zmian wykładnika, lecz może zaburzyć przemiany, zakłócając i chwilowo zmieniając wykładnik podczas procesu.

Należy zwrócić uwagę, że w rzeczywistych procesach rekrystalizacji dynamicznej i statycznej wykładnik n nigdy nie przyjmuje wartości teoretycznej uzyskanej w procesach opisanych w tym i poprzednich podrozdziałach (wątek ten będzie rozwinięty dalej). Główną tego przyczyną jest zmienna prędkość rozrostu ziaren. To zagadnienie zostanie omówione w dalszej części pracy.

Najbardziej doniosły wniosek z przeprowadzonych w tym podrozdziale badań kinetyki przy użyciu automatów komórkowych brzmi tak: za pomocą jedno- lub dwuwymiarowych automatów niemożliwe jest uzyskanie kinetyki właściwej dla większości rzeczywistych procesów. Nie ma również żadnych podstaw do uogólniania lub przenoszenia wyników uzyskanych automatami dwuwymiarowymi na automaty trójwymiarowe lub procesy rzeczywiste. Potwierdza to tezę, że modelowanie większości procesów należy przeprowadzić trójwymiarowymi automatami komórkowymi.

8.5. Rekrystalizacja statyczna

Zwykle kiedy jest mowa o rekrystalizacji w połączeniu z przeróbką plastyczną na gorąco, rozpatruje się dwa przypadki: rekrystalizację dynamiczną i statyczną. Pierwsza z nich zachodzi podczas odkształcenia i jej kontynuację po odkształceniu nazywano rekrystalizacją metadynamiczną, druga charakteryzuje się tym, że jej skutki są odczuwalne po odkształceniu. Moim zdaniem jest to ten sam proces, który zachodzi w różnych warunkach. Jednak przebieg i efekty rekrystalizacji dynamicznej i statycznej na tyle się od siebie różnią, że taki podział jest zupełnie uzasadniony. To jednak nie przeczy potrzebie kompleksowego opisu tego zjawiska przy uwzględnieniu każdego warunku. Rozpatrywanie modelu kompleksowego należy rozpocząć od dopasowania warunków symulacji do tych dwóch szczegółowych przypadków. Jako pierwszy wybrano model rekrystalizacji statycznej.

Model rekrystalizacji statycznej powinien spełnić dwa zadania, tzn. umożliwić:

- uzyskanie mikrostruktury, która będzie odpowiadała warunkom odkształcenia;
- uzyskanie kinetyki mięknięcia zgodnej z warunkami odkształcenia.

Spełnienie pierwszego zadania zależy wyłącznie od określenia warunków zarodkowania, ponieważ wielkość ziarna, która pozostaje prawie jedynym znanym i uwzględnianym parametrem, jest w jednoznaczny sposób związana z liczbą zarodków. Natomiast rozwiązanie drugiego, bardziej skomplikowanego zadania wymaga określenia prędkości rozrostu ziaren, jej zależności od czasu i parametrów procesowych.

Obydwa zadania w tej chwili nie zostały jeszcze ostatecznie rozwiązane, dlatego poniżej przedstawiono jedynie wstępne ich wyniki.

8.5.1. Zarodkowanie podczas rekrystalizacji statycznej

Podstawy zarodkowania podczas rekrystalizacji statycznej zostały przedstawione w podrozdziale 8.2.3. Jednak do rozważenia pozostaje jeszcze kilka kwestii.

Pierwsze zasadnicze pytanie dotyczy czasu zarodkowania. W literaturze przedmiotu nie znajdujemy odpowiedzi na pytanie, czy zarodki powstają podczas odkształ-

cenia, czy po nim. Jeżeli rozpatrywać wzór Robertsa i Ahlbloma (8.7), który pozwala wyznaczyć krytyczną dla zajścia rekrytalizacji dynamicznej wartość gęstości dyslokacji ρ_c , można zauważyć, że jest w nim bezpośrednio uwzględniony parametr procesowy, a mianowicie prędkość odkształcenia. Więc nadal pozostają otwarte pytania, czy przed osiągnięciem wartości krytycznej gęstości dyslokacji powstają zarodki i czy w razie braku odkształcenia (przy zerowej prędkości odkształcenia) wartość krytyczna spada do zera. Kolejne związane z tym zagadnieniem pytanie brzmi: czy proces odkształcenia wspomaga zarodkowanie, czy je hamuje? Czy ruch dyslokacji przeciwdziała, czy wspiera zarodkowanie? Jeżeli hamuje, to po odkształceniu warunki mogą być lepsze, a jeżeli wspiera, to możliwości zarodkowania po odkształceniu znacznie się zmniejszają. Dlatego powszechnie uznawano, że zarodkowanie podczas rekrytalizacji statycznej zachodzi po odkształceniu, a w przypadku dynamicznej (metadynamicznej) tylko w czasie odkształcenia. Taki tok rozumowania jest jednak niekonsekwentny. Należy trzymać się albo wersji o zarodkowaniu wyłącznie podczas odkształcenia, albo uznać, że po odkształceniu z rekrytalizacją dynamiczną, podczas rekrytalizacji metadynamicznej, również powstają zarodki. Na korzyść zarodkowania przy braku odkształcenia (po odkształceniu) jednoznacznie świadczy zarodkowanie i rekrytalizacja w procesie wyżarzania.

Przyjęcie jednej lub drugiej z wymienionych koncepcji zarodkowania poza tym, że jest kwestią teoretyczną o niewielkim wpływie na kinetykę rekrytalizacji, rzutuje na strukturę modelu. Przyjęcie koncepcji, że zarodkowanie zachodzi wyłącznie podczas odkształcenia, nieco komplikuje model, ponieważ wymaga zadania odpowiedniej prędkości zarodkowania podczas odkształcenia z uwzględnieniem parametrów procesu, a także nieco utrudnia uzyskanie odpowiedniej kinetyki, ponieważ nie pozwala uwzględnić w modelu okresu inkubacyjnego poprzedzającego zarodkowanie. Koncepcja ta pozwala jednak na płynne przejście od modelowania rekrytalizacji statycznej do modelowania rekrytalizacji dynamicznej i metadynamicznej.

Przyjęcie koncepcji zarodkowania po odkształceniu prowadzi do nieuwzględniania zarodkowania podczas odkształcenia, dopóki gęstość dyslokacji nie osiągnie wartości krytycznej. Nie bierze się w niej także pod uwagę żadnych danych o zarodkowaniu po odkształceniu z rekrytalizacją dynamiczną. A zatem modelowanie rekrytalizacji dynamicznej i metadynamicznej komplikuje się, natomiast modelowanie rekrytalizacji statycznej ulega uproszczeniu. W tym ostatnim przypadku zarodkowanie powinno nastąpić od razu po odkształceniu, kiedy są najlepsze dla niego warunki (największa gęstość dyslokacji, która jest siłą napędową tego procesu). Szybko następuje wysycenie miejsc zarodkowania, dlatego zarodkowanie w tych miejscach można rozpatrywać jako proces zachodzący jednocześnie bez opóźnienia.

W rozpatrywanym tu modelu zastosowano pierwszą koncepcję zarodkowania podczas odkształcenia. Chociaż przyjęcie innej koncepcji wymaga odpowiedniej jego przebudowy, zmiany te nie są zbyt skomplikowane. Dlatego większość opisanych niżej działań może być zastosowana również w przypadku przyjęcia innej niż rozpatrywana koncepcji zarodkowania.

Zrealizować model zarodkowania można na dwa sposoby: z wykorzystaniem wzorów empirycznych na wielkość zrekrystalizowanego ziarna i z wykorzystaniem prawdopodobieństwa zarodkowania obliczonego na podstawie gęstości dyslokacji. Drugie podejście jest z powodzeniem stosowane do modelowania rekryształizacji dynamicznej, jednak rekryształizacja statyczna nadal czeka na swoje opracowanie. Dlatego poniżej zostanie pokrótce opisany pierwszy sposób.

Można założyć, że prędkość zarodkowania zależy od odkształcenia ϵ , prędkości odkształcenia $\dot{\epsilon}$, temperatury T i początkowej wielkości ziarna D_0 . Równania, które opisują wielkość ziarna zrekrystalizowanego statycznie, mogą być przedstawione w następującej ogólnej postaci:

$$D_{srx} = AD_0^k \epsilon^{-l} \dot{\epsilon}^{-m} \left[\exp\left(-\frac{Q}{RT}\right) \right]^n \quad (8.24)$$

gdzie:

D_0 i D_{srx} – wielkości ziarna przed rekryształizacją i po rekryształizacji statycznej;

A, k, l, m, n – współczynniki materiałowe.

Współczynnik A zmienia się w szerokim zakresie, $k = 0,37 \div 0,67$, $l = 0,5 \div 1$, m rzadko różni się od zera, $n = 0$ lub $n = 1$ przy $Q = 17\,540 \div 45\,000$ J/mol.

W kolejnym równaniu liczba ziaren jest związana z ich wielkością:

$$A_V N_V D_{srx}^3 = 1 \quad (8.25)$$

gdzie A_V – czynnik geometryczny (często $A_V = \pi/6$ lub $A_V \approx 0,75$, w zależności od przyjętych założeń).

Podstawiając wzór (8.24) do równania (8.25), można obliczyć liczbę zarodków N_V w jednostce objętości:

$$N_V = a_N D_0^{-k_N} \epsilon^{l_N} \exp\left(\frac{Q_{NS}}{RT}\right) \quad (8.26)$$

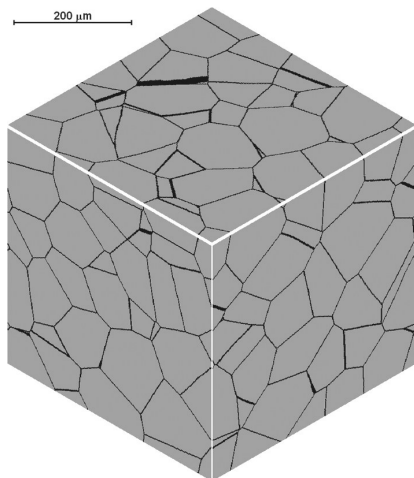
Następnie można obliczyć prędkość zarodkowania \dot{N}_V :

$$\dot{N}_V = a_N D_0^{-k_N} \epsilon^{l_N-1} \exp\left(\frac{Q_{NS}}{RT}\right) \dot{\epsilon} \quad (8.27)$$

Współczynniki wzoru (8.27) można obliczyć ze wzoru (8.24) $k_N = 3k = 1 \div 2$, $l_N = 3l = 1,5 \div 3$, $Q_{NS} = 3Q$.

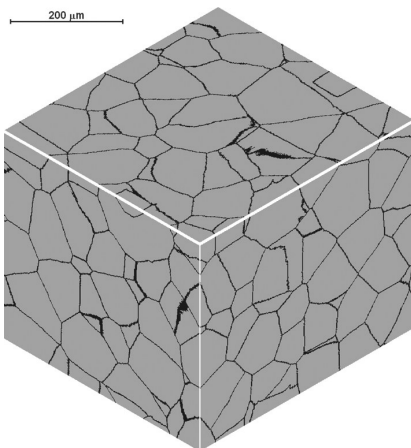
Do zarodkowania po odkształceniu odnosi się wzór (8.26), do zarodkowania podczas odkształcenia – wzór (8.27).

Na rysunkach 8.7 i 8.8 pokazano początkową mikrostrukturę oraz dwa przykłady mikrostruktury końcowej po odkształceniu w temperaturze $T = 1100^{\circ}\text{C}$ i pełnej rekrytalizacji statycznej. Obliczenia przeprowadzono dla stali 45. Pierwszy przykład (rys. 8.8a) dotyczy procesu z prędkością odkształcenia $\dot{\epsilon} = 0,1 \text{ s}^{-1}$ i odkształceniem $\epsilon = 0,094$. Średnia wielkość ziarna $d_{av} = 92 \text{ }\mu\text{m}$ (zmierzona $92,4 \text{ }\mu\text{m}$). W drugim przykładzie (rys. 8.8b) warunki powstawania mikrostruktury były następujące: $\dot{\epsilon} = 1 \text{ s}^{-1}$, $\epsilon = 0,188$. Średnia wielkość ziarna $d_{av} = 75 \text{ }\mu\text{m}$ (zmierzona $78,5 \text{ }\mu\text{m}$).

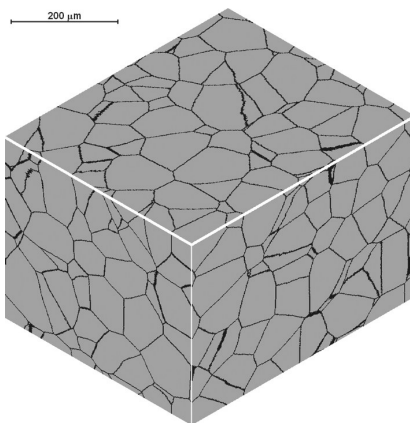


Rys. 8.7. Początkowa mikrostruktura ze średnią wielkością ziarna $d_{av} = 100 \text{ }\mu\text{m}$

a)



b)



Rys. 8.8. Końcowa mikrostruktura po odkształceniu w temperaturze $T = 1100^{\circ}\text{C}$ i pełnej rekrytalizacji statycznej: a) $\dot{\epsilon} = 1 \text{ s}^{-1}$, $\epsilon = 0,18$, $d_{av} = 92 \text{ }\mu\text{m}$; b) $\dot{\epsilon} = 0,1 \text{ s}^{-1}$, $\epsilon = 0,18$, $d_{av} = 75 \text{ }\mu\text{m}$

8.5.2 Kinetyka rekrytalizacji statycznej

Rozpatrywanie kinetyki rekrytalizacji w podrozdziale 8.3 opierało się na podstawowym założeniu teorii KJMA i potwierdziło zgodność wyników modelowania z teorią. Spośród wszystkich zasymulowanych wariantów rekrytalizacji statycznej najlepiej powinna odpowiadać sytuacja z zarodkowaniem przed rozrostem, która w warunkach stałej prędkości rozrostu pozwoli uzyskać wykładnik Avramiego (8.23) równy $n = 3$. Wówczas do opisu kinetyki można zastosować następujący wzór:

$$\chi = 1 - \exp\left(-\frac{\pi}{3} N v^3 t^3\right) \quad (8.28)$$

gdzie:

N – liczba zarodków (w jednostce objętości);

v – prędkość rozrostu ziaren.

Jednak wszystkie dane (doświadczalne i literaturowe) wskazują, że w materiałach nigdy nie zachodzi rekrytalizacja statyczna, której kinetykę można byłoby opisać równaniem Avramiego z wykładnikiem $n = 3$. W rzeczywistych warunkach rekrytalizacji wykładnik Avramiego osiąga wartość mniejszą od dwóch ($n < 2$). W większości modeli empirycznych nie zwracano na ten fakt szczególnej uwagi i wykorzystywano taki wykładnik n o wartości ustalonej za pomocą obliczeń numerycznych lub odczytanej z wykresu Avramiego.

Problem powstaje w modelach, w których, jak w automatach komórkowych, kinetyka opiera się na zasadach geometrycznych. To, co w podrozdziale 8.3 zostało uznane za osiągnięcie, a mianowicie bardzo wysoka zgodność z wartościami teoretycznymi, teraz jest głównym, wymagającym rozwiązania problemem. Problem ten, niestety, uszedł uwadze większości badaczy specjalizujących się w automatach komórkowych. W tej chwili istnieje jedynie kilka nieuzasadnionych podejść zarówno do samego wykładnika, jak i do sposobów rozwiązania tego problemu.

W jednej z pierwszych prac na ten temat Goetz i Seetharaman (1997) zbadali wpływ warunków zarodkowania podczas rekrytalizacji statycznej na uzyskiwaną wartość wykładnika n . Podejście to jest ciekawe, jednak uzyskanie odpowiedniej kinetyki w ten sposób prowadzi do stworzenia sztucznej, nierzeczywistej struktury, która zupełnie nie jest podobna do uzyskiwanej w realnych warunkach.

Kugler i Turk (2004) spróbowali uwzględnić zdrowienie statyczne podczas rekrytalizacji metadynamicznej, ale nie ustalili jego wpływu na wykładnik n , a jedynie na czas połowy rekrytalizacji. Natomiast w innych dotyczących tego tematu publikacjach wykładnikowi n zostało poświęcone niewiele uwagi. Można to wytłumaczyć w następujący sposób: Sellars (1990) oraz Senuma *et al.* (1984) podają wartość wykładnika dla rekrytalizacji statycznej $n = 2$, inni autorzy – wartości zbliżone. Przy modelowaniu rekrytalizacji statycznej dwuwymiarowymi automatami komórkowymi wykładnik ten również wynosi 2. Nawet jeżeli w świetle badań ów wykładnik ma nieco

mniejszą wartość, na przykład $n = 1,7$, to różnice te są pomijane. Natomiast głównemu problemowi nie poświęcano uwagi. Przy zastosowaniu trójwymiarowych automatów pomijać ten problem można tylko przez jakiś czas, w którymś momencie niewątpliwie da on o sobie znać. W takim razie w dwuwymiarowych automatach komórkowych nawet niewielkie różnice wartości wykładnika n powodują, że uzyskane modele nie odzwierciedlają rzeczywistego procesu, błędnie symulują proces mięknięcia i w tym sensie są modelami nieadekwatnymi.

Wróćmy jednak do głównego problemu. Równanie (8.28) pozostaje niepodważalne, jednak główną jego wadą jest fakt, że rozpatruje ono tylko ograniczony, teoretyczny przypadek przemiany. Obydwa jego parametry są stałe. Jeżeli liczba zarodków N podczas procesu się nie zmienia, to prędkość rozrostu ziaren v może się zmieniać, zatem jest to jedyny parametr, który wpływa na kinetykę procesu. W zasadzie zamiast równania (8.28) należałoby zastosować następujący zmodyfikowany wzór Kołmogorowa:

$$\chi = 1 - \exp \left[-\frac{\pi}{3} N \left(\int_{\tau=0}^t v d\tau \right)^3 \right] \quad (8.29)$$

Wiadomo też, że wykładnik Avramiego n (8.23) może mieć wartość różną od teoretycznej wartości 3, jeżeli prędkość v jest funkcją czasu t . Czyli doświadczalne i podawane przez literaturę przedmiotu dane wskazują na to, że rekrytalizacja statyczna zachodzi ze zmienną prędkością rozrostu, a ze względu na wartość wykładnika $n < 3$ prędkość ta jest malejącą funkcją czasu.

Zgodnie z tym, co napisano w podrozdziale 8.2.4, prędkość przemieszczania się granicy ziarna podczas rekrytalizacji można przedstawić w uproszczonej postaci jako iloczyn dwóch czynników: mobilności granic m i siły napędowej rekrytalizacji ΔP (8.16). W stałej temperaturze, kiedy mobilność granic pozostaje niezmienna (8.21), tylko zmagazynowana energia odkształcenia, która zależy od gęstości dyslokacji ρ (8.16), może wpływać na zmianę prędkości v . Tak więc gęstość dyslokacji ρ niezrekrytalizowanych ziaren powinna się zmieniać. Z kolei spadek gęstości dyslokacji ρ może nastąpić tylko w procesie zdrowienia statycznego.

Warto tu przypomnieć, że zdrowienie statyczne, ze względu na jego znikomy wpływ, zwykle nie jest rozpatrywane w modelach rozwoju dyslokacji podczas odkształcenia i w takich przypadkach stosuje się wzór (8.1). Równie często zdrowienie statyczne jest pomijane w modelach mięknięcia po odkształceniu, kiedy rozpatruje się rekrytalizację jako jedyny lub jedyny warty uwagi proces wpływający na strukturę materiału. W takich modelach kinetyka mięknięcia jest opisywana równaniem Avramiego (8.23) z empirycznym wykładnikiem n , który przyjmuje dowolną wartość, natomiast ułamek mięknięcia x jest często błędnie traktowany jako ułamek rekrytalizacji χ bądź też podaje się ułamek χ bez ułamka x . Ze względu na możliwość pomyłki konieczna jest szczególna uwaga w stosunku do ułamków χ i x .

Jak wynika z analizy dotychczasowej literatury naukowej, podczas modelowania rekryształizacji statycznej za pomocą automatów komórkowych zmiany gęstości dyslokacji nie są uwzględniane, czyli nadal znajduje zastosowanie wzór (8.1), który w żaden sposób nie określa zmiany gęstości dyslokacji przy braku odkształcenia.

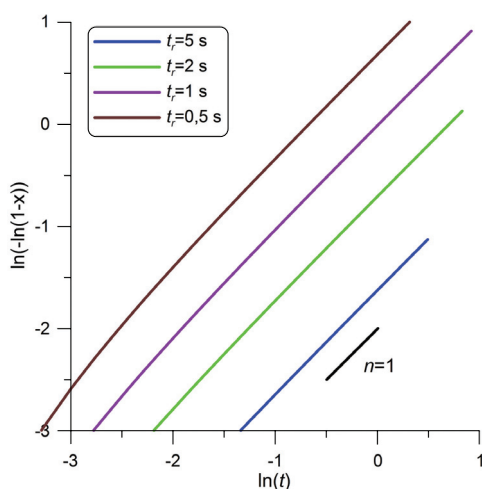
To właśnie w celu uwzględnienia zdrowienia wprowadzono dodatkowy człon $R(T)p$ do wzoru (8.3), który w przypadku braku odkształcenia pozostaje jedynym parametrem mogącym wpłynąć na gęstość dyslokacji niezrekryształizowanych ziaren i zmniejszyć siłę napędową rekryształizacji statycznej, a tym samym prędkość rozrostu ziaren i wartość wykładnika n .

W rzeczywistości zdrowienie statyczne wpływa na mięknięcie materiału na dwa sposoby: przez zmniejszenie gęstości dyslokacji niezrekryształizowanych ziaren oraz przez zmniejszenie prędkości rozrostu ziarna, to znaczy wpływając na ułamek rekryształizacji χ .

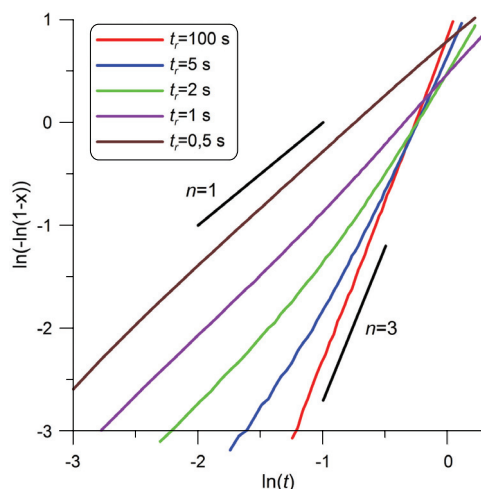
Poniżej przedstawiono wyniki symulacji procesu mięknięcia z rekryształizacją statyczną i zdrowieniem statycznym za pomocą automatów komórkowych. Wybrano proces, w którym czas połowy rekryształizacji bez uwzględnienia zdrowienia statycznego wynosi 0,687 s. Dalej do modelu gęstości dyslokacji (8.3) wprowadzono człon R , który odzwierciedla zdrowienie statyczne. Zadano pięć wartości R : 0,01, 0,2, 0,5, 1,0 i 2,0. Wartości te odpowiadają stałym czasowym t_r równym odpowiednio 100, 5, 2, 1 i 0,5 s.

Na rysunkach 8.9–8.11 przedstawiono kinetykę zdrowienia statycznego, mięknięcia i rekryształizacji statycznej.

Kinetyka zdrowienia statycznego (rys. 8.9) dotyczy ziaren niezrekryształizowanych i pozostaje niezależna od rekryształizacji. Zgodnie z założeniami jest to proces, który można opisać równaniem Avramiego z wykładnikiem $n = 1$.



Rys. 8.9. Kinetyka zdrowienia statycznego



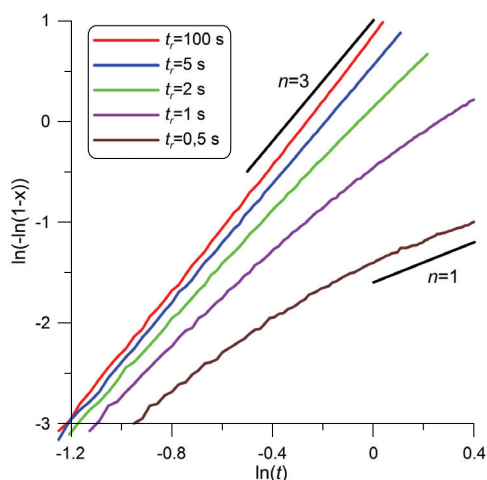
Rys. 8.10. Kinetyka mięknięcia

Kinetyka mięknięcia (rys. 8.10) jest mocno uzależniona od procesów zdrowienia i rekrytalizacji. Dopóki proces zdrowienia jest znacznie wolniejszy od rekrytalizacji ($t_r \gg t_{0,5}$), można go nie uwzględniać i wtedy wykładnik Avramiego jest bliski 3. Odchylenie od przebiegu teoretycznego z wykładnikiem $n = 3$ można zauważyć, kiedy proces zdrowienia jest szybszy ($t_r < 10t_{0,5}$). Najpierw jest to odchylenie widoczne tylko na początku mięknięcia, potem, w miarę zwiększania się prędkości zdrowienia, odchylenie trwa coraz dłużej i już przy $t_r \approx 2t_{0,5}$ wykładnik Avramiego w procesie mięknięcia zbliża się do 1. W miarę dalszego przyspieszenia procesu zdrowienia statycznego staje się on dominujący w procesie mięknięcia.

Wpływ zdrowienia statycznego nie ogranicza się tylko do procesu mięknięcia, ale jest również widoczny w procesie rekrytalizacji statycznej, o czym pisano wyżej. Kiedy proces zdrowienia jest bardzo wolny ($t_r > 10t_{0,5}$), przebieg rekrytalizacji statycznej (rys. 8.11) również w niewielkim stopniu odbiega od przebiegu teoretycznego. W wyniku zbliżenia się stałej czasowej procesu zdrowienia statycznego t_r do czasu połowy rekrytalizacji ($2t_{0,5}$) rekrytalizacja jest w znaczącym stopniu hamowana i można mówić, że nie następuje pełna, lecz częściowa rekrytalizacja statyczna. Również nieprawidłowe formalnie jest w tym przypadku stosowanie pojęcia czasu połowy rekrytalizacji $t_{0,5}$.

Wiadomo, że rekrytalizacja w materiałach o małej energii błędu ułożenia łatwiej zachodzi w wyższych temperaturach i przy większej zmagazynowanej energii, a zdrowienie na odwrót. Jeżeli przyjąć, że zdrowienie statyczne jest procesem aktywowanym do dyfuzji linii dyslokacyjnych i zależy od temperatury, ale nie od odkształcenia i gęstości dyslokacji, łatwiej zrozumieć, dlaczego mniejsze odkształcenia mogą prowadzić do niepełnej rekrytalizacji. Prędkość zdrowienia w tej samej temperaturze faktycznie pozostaje bez zmian, a rekrytalizacja w istotnym stopniu zależy od warunków odkształcenia

(odkształcenia i gęstości dyslokacji), dlatego przy większych odkształceniach rekrytalizacja zachodzi szybciej od zdrowienia, mniej od niego zależy, natomiast przy odkształceniach mniejszych przewagę może zyskać proces zdrowienia statycznego, z uwagi na fakt, że siły napędowe zarówno zarodkowania, jak i rozrostu ziaren mogą być zbyt małe.



Rys. 8.11. Kinetyka rekrytalizacji statycznej

Proces zdrowienia składa się jednak z kilku kilka mechanizmów. Może przybrać formę kontrolowanego dyfuzyjnie poślizgu dyslokacji krawędziowych, poślizgu poprzecznego dyslokacji śrubowych lub wspinania dyslokacji kontrolowanego dyfuzją wakansów, atomów lub rozpuszczonych pierwiastków. Prędkość zdrowienia dyslokacyjnego opartego na poślizgu lub wspinaniu można wyrazić wzorem:

$$\sqrt{\frac{\rho}{\rho_0}} = \left[1 + \frac{t}{\tau} \right]^{-m} \quad (8.30)$$

gdzie:

- ρ, ρ_0 – odpowiednio: bieżąca i początkowa gęstość dyslokacji;
- τ – charakterystyczny czas relaksacji;
- m – wykładnik zależny od mechanizmu zdrowienia.

We wzorze (8.30) w przypadku zdrowienia przez poślizg dyslokacji krawędziowych wykładnik $m = 1$, w przypadku wspinania kontrolowanego przez wakanse $m = 1/2$, wspinania kontrolowanego atomami – $m = 1/4$ i poślizgu poprzecznego – $m = -1$ (Nes 1995).

Kinetykę zdrowienia można alternatywnie wyrazić równaniem różniczkowym:

$$\frac{d\rho}{dt} = -k\rho^n \quad (8.31)$$

Gdy $n \neq 1$, wzór ten ma następującą postać:

$$\frac{1}{\rho^{n-1}} - \frac{1}{\rho_0^{n-1}} = (n-1)kt \quad (8.32)$$

Gdy $n = 1$:

$$\rho = \rho_0 \exp(-kt) \quad (8.33)$$

Parametr k można opisać równaniem Arrheniusa:

$$k = k_0 \exp\left(\frac{-Q}{RT}\right) \quad (8.34)$$

gdzie:

k_0 – stała materiałowa;

Q – energia aktywacji zdrowienia dyslokacyjnego;

R – stała gazowa;

T – temperatura.

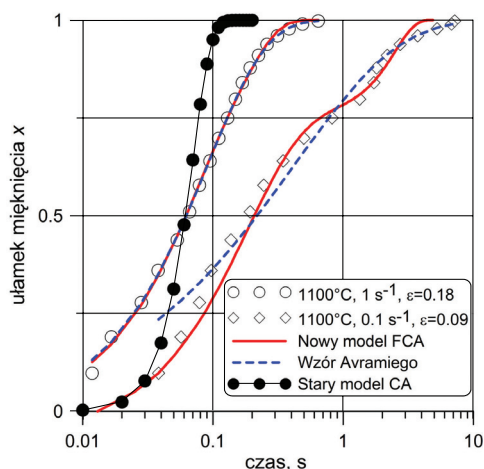
Przedstawione wyżej wyniki (rys. 8.9–8.11) dotyczą oczywiście tego ostatniego przypadku. W praktyce jednak należy wyznaczyć doświadczalnie, który mechanizm jest dominujący i zastosować go do opisu i obliczenia zdrowienia statycznego.

Dodatkowym, trudnym do uwzględnienia czynnikiem jest poziom mięknięcia wywołany zdrowieniem statycznym, czyli stopień zaawansowania tego procesu lub ułamek zdrowienia. Wiadomo, że zdrowienie obniża gęstość dyslokacji wewnątrz struktur dyslokacyjnych, czyli komórek dyslokacyjnych, podziaren lub małych ziaren. Dyslokacje dyfundują do granic (lub ścianek) komórek, podziaren lub ziaren. Tworzy się stabilna struktura, którą można rozpatrywać jako strukturę wewnętrzną ziaren i która może być usunięta tylko w efekcie rekrytalizacji. Wówczas opisując proces zdrowienia statycznego, na przykład wzorem (8.33), należy wprowadzić czynnik f odzwierciedlający stopień możliwego zaawansowania procesu zdrowienia. Wtedy wzór (8.33) przyjmie następującą postać:

$$\rho = [1 - f + f \exp(-kt)]\rho_0 \quad (8.35)$$

Na rysunku 8.12 pokazano wyniki symulacji rekrytalizacji statycznej z uwzględnieniem zdrowienia zgodnie ze wzorem (8.35), porównując je z danymi doświadczalnymi i wynikami uzyskanymi przy użyciu innych modeli. Dane doświadczalne dotyczą warunków odkształcenia stali 45, jakie zostały opisane w odniesieniu do rysunków 8.7 i 8.8, które przedstawiają uzyskaną mikrostrukturę po rekrytalizacji. Warto zwrócić uwagę, że poprzedni model, który nie uwzględniał zdrowienia statycznego, nie umożliwiał dopasowania kinetyki do rzeczywistej, ponieważ wykładnik Avramiego zawsze jest

równy 3. Uwzględnienie zdrowienia lub zastosowanie wzoru Avramiego (8.23) pozwala poprawnie opisać procesy, w których proces zdrowienia jest wolniejszy lub zachodzi w całej objętości materiału. Natomiast w warunkach ograniczenia procesu zdrowienia i jego większej prędkości w porównaniu z procesem rekrytalizacji statycznej uwzględnienie stopnia jego zaawansowania pozwala uzyskać poprawny wynik. Szybki spadek gęstości dyslokacji w procesie zdrowienia istotnie hamuje proces rekrytalizacji, ale nie na tyle, żeby w pełni go zatrzymać. Przy mniejszych odkształceniach rekrytalizacja statyczna nie nastąpi i zajdzie tylko częściowe mięknięcie materiału w efekcie zdrowienia statycznego.



Rys. 8.12. Kinetyka mięknięcia z uwzględnieniem stopnia zdrowienia statycznego

8.6. Rekrytalizacja dynamiczna

Rekrytalizacja dynamiczna zachodzi podczas odkształcenia i tym różni się od rekrytalizacji statycznej. Jej skutki są odczuwalne podczas odkształcenia, nie ograniczają się do przebudowy mikrostruktury, ale rzutują także na naprężenie uplastyczniające.

Podobnie jak przy rozpatrywaniu rekrytalizacji statycznej, aby uzyskać kompleksowe modelowanie procesu rekrytalizacji dynamicznej, należy:

- uzyskać mikrostrukturę, która będzie odpowiadała warunkom odkształcenia;
- uzyskać kinetykę rekrytalizacji dynamicznej i odpowiednie naprężenie uplastyczniające;
- uzyskać kinetykę i mikrostrukturę po rekrytalizacji metadynamicznej.

Dodatkowym elementem, który nie występuje przy rekrytalizacji statycznej, jest konieczność uwzględnienia w modelu geometrii odkształcenia. Odkształcenie było

uwzględnione również poprzednio, jednak modelowanie rekrytalizacji statycznej można prowadzić na niezmienniej siatce, pomijając proces odkształcenia i biorąc pod uwagę tylko jego skutki. Podczas modelowania rekrytalizacji dynamicznej odkształcenie wpływa na kształt zarówno istniejących od samego początku, jak i nowych ziaren. Niestety, obecnie dostępna literatura przedmiotu nie podaje przykładów uwzględnienia geometrii odkształcenia podczas modelowania rekrytalizacji dynamicznej, co istotnie obniża wiarygodność uzyskiwanych wyników.

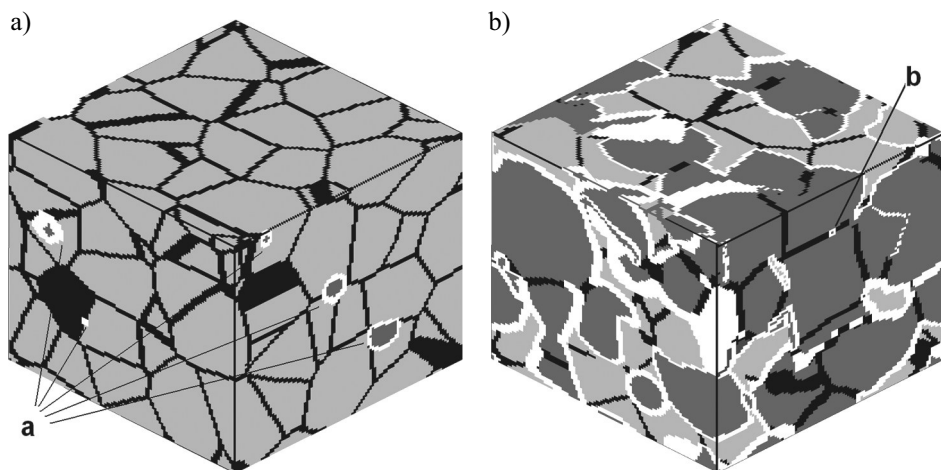
8.6.1. Zarodkowanie i rozrost ziaren

Modele zarodkowania i rozrostu ziaren zostały opisane w podrozdziale 8.2. Zawierają one wskazówki dotyczące prędkości zarodkowania, jego prawdopodobieństwa i liczby powstałych zarodków. Wiadomo też, że podczas rekrytalizacji ma miejsce wyłącznie zarodkowanie heterogeniczne. Jeżeli w odniesieniu do przemiany fazowej austenit-ferryt udowodniono teoretycznie i potwierdzono doświadczalnie różne prawdopodobieństwa zarodkowania na różnych elementach mikrostruktury, to w przypadku rekrytalizacji niezaprzeczalnych dowodów na to nie ma. Podczas przemiany fazowej największe prawdopodobieństwo zarodkowania istnieje w kątach ziaren, potem na krawędziach i na koniec na powierzchni granic. Związane to jest z mechanizmem zarodkowania, kiedy krytyczny rozmiar zarodka zależy od miejsca jego powstania, i faktycznie, w zależności od warunków najpierw następuje wysycenie pierwszych miejsc zarodkowania, potem drugich itd.

Zależność między podstawowym mechanizmem zarodkowania a rozmiarami zarodka jest w procesie rekrytalizacji inna. Pierwotne komórki dyslokacyjne lub podziarna mają podobne rozmiary i, w odróżnieniu od przemiany fazowej, szanse, by stać się zarodkami, mają większe komórki lub podziarna. Zasadnicze znaczenie ma przy tym większa powierzchnia granic wysokokątowych. Komórki lub podziarna przy kątach i krawędziach z jednej strony są nieco mniejsze, ale z drugiej mają większą procentowo powierzchnię granic wysokokątowych. Tak więc prawdopodobieństwo zarodkowania w tych miejscach jest nieco większe, ale różnica ta jest znacznie mniejsza niż w przypadku zarodkowania podczas przemian fazowych.

Kolejna różnica dotycząca przebiegu rekrytalizacji związana jest z rozrostem. Podczas przemiany austenit-ferryt ziarna ferrytu wrastają wyłącznie do swojego macierzystego ziarna austenitu, natomiast zrekrystalizowane ziarno wrasta do wszystkich innych ziaren, tylko nie do macierzystego. Dlatego w kącie, w miejscu styku kilku ziaren, w procesie przemiany fazowej zarodki mogą powstać w każdym z ziaren, natomiast w procesie rekrytalizacji może się rozrosnąć tylko jeden zarodek.

Przykład wyników symulacji przedstawiono na rysunku 8.13. Nie dotyczy on konkretnego materiału, demonstuje tylko zasady działania algorytmu. Przykład ten przedstawia proces zarodkowania podczas rekrytalizacji dynamicznej. Stare ziarna są na tym rysunku zaznaczone kolorem jasnoszarym, nowe – ciemnoszarym. Nieruchome granice ziaren oznaczono kolorem czarnym, natomiast granice nowych rosnących ziaren są białe.



Rys. 8.13. Zarodkowanie podczas rekrytalizacji dynamicznej: a) na początku procesu, pierwotne; b) wtórne, na nowych granicach ziaren

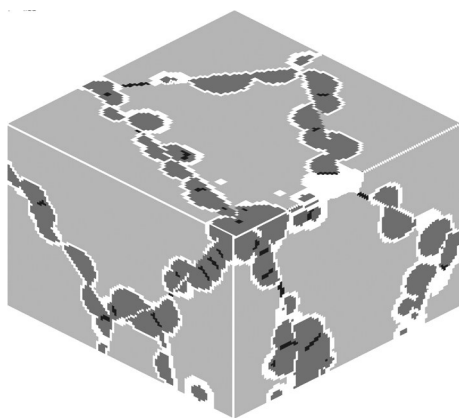
Na początku procesu zarodki nowych ziaren powstają na granicach „starych” ziaren (zaznaczono je literą „a” na rys. 8.13a). Wydaje się, że nie wszystkie ziarna powstały na granicach, jednak to wrażenie jest mylące, ponieważ w strukturze trójwymiarowej nie wszystkie granice są widoczne. Jest to zarodkowanie pierwotne. Po nim następuje rozrost. W czasie trwania procesu odkształcenia na granicach ciągle powstają nowe ziarna, dlatego też należy zwrócić uwagę na fakt, że wielkość ziaren jest różna. Oznacza to, że przy podobnej prędkości rozrostu powstały one w różnych momentach. Na rysunku 8.13a tylko w przypadku niektórych ziaren można zobaczyć, że wrastają one nie we wszystkie otaczające je ziarna, co było wyjaśnione poprzednio.

Przykład ten dotyczy również sytuacji, kiedy wielkość pierwotnych ziaren jest dość mała, a zarodkowanie odbywa się z niewielką prędkością. Dlatego podczas tego procesu praktycznie nie następuje wysycenie miejsc zarodkowania, co można zobaczyć na rysunku 8.13b, na którym obok dość dużych, wyrosniętych nowych ziaren, znajdują się ziarna drobniejsze, które powstały jako ostatnie. Należy również zwrócić uwagę na fakt, że zarodki powstają tylko na nieruchomych granicach, co jest związane z faktem, że do podtrzymania ruchu granic nowych ziaren prawie zawsze jest potrzebna mniejsza siła napędowa niż do inicjowania tego ruchu w przypadku zarodka, a poza tym do utworzenia komórki dyslokacyjnej lub podziarna, które może być miejscem powstania zarodka, potrzebny jest pewien okres inkubacyjny. Dopóki zachodzą te procesy, granica rosnącego ziarna przesuwa się dalej, komórka pojawia się wewnątrz ziarna, warunki zarodkowania zmieniają się z heterogenicznych na homogeniczne i ze względu na obecność wysokokątowych granic możliwość zarodkowania zanika. A zatem w modelu miejsc zarodkowania, w tym wtórnego, wykorzystuje się tylko nieruchome granice. Mobilne granice nowych rosnących ziaren zmieniają się w granice nieruchome w dwóch przypadkach: kiedy zderzają się z mobilnymi granicami innych ziaren

i kiedy ziarna osiągają duże rozmiary i znajdując się pod wpływem odkształcenia, gromadzą dyslokacje, siła napędowa istotnie maleje, ruch wyhamowuje i wówczas ziarna przestają rosnąć, a mobilne granice zmieniają się w granice nieruchome. Na rysunku 8.13b zilustrowano pierwszy wariant powstania nowych nieruchomych granic. Literą „b” zaznaczono na nim powstanie nowego zarodka na nowej nieruchomej granicy ziaren. Takie zarodkowanie można nazwać wtórnym. Tak więc w sytuacji przedstawionej na rysunku 8.13b jednocześnie zachodzi zarodkowanie pierwotne i wtórne, czyli mamy do czynienia z zarodkowaniem ciągłym.

Ten przykład, jak i kolejny, dotyczy „nieodkształconej” przestrzeni. Ułatwia to porównanie struktur i demonstrację zasad zarodkowania.

Drugi przykład zarodkowania pokazano na rysunku 8.14. Przedstawia on sytuację, w której struktura początkowa jest gruboziarnista, a początkowa prędkość zarodkowania – duża. Wówczas dość szybko następuje wysycenie miejsc zarodkowania i nawet, do momentu, w którym nie powstaną nowe nieruchome granice, a gęstość dyslokacji nie osiągnie wartości krytycznej dla nowego zarodkowania, chwilowe całkowite jego zatrzymanie. Na rysunku tym również dobrze widać, że zarodkowanie następuje na granicach ziaren, a rozrost ziarna zachodzi wewnątrz tylko jednego ziarna.



Rys. 8.14. Mikrostruktura pod koniec pierwszego cyklu zarodkowania.
Rozwój struktury naszyjnikowej (*necklace structure*)

8.6.2. Uwzględnienie geometrii odkształcenia

W podrozdziale 2.4.3 przedstawiono zasady uwzględniania nieregularnych lub odkształconych komórek w celu uzyskania izotropii przestrzeni, jednak nie zostały tam omówione sposoby wykorzystania parametrów odkształcenia w automatach komórkowych. Aspekt ten został również pominięty przy okazji rozpatrywania rekrytalizacji statycznej.

W zależności od modelowanego procesu przy uwzględnieniu odkształcenia w automatach komórkowych stosuje się dwa podejścia: pierwsze, w którym siatka jest nieregularna lub odkształcenie jest niejednorodne, i drugie – z regularną siatką i jednorodnym

odkształceniem. W pierwszym podejściu oblicza się współrzędne każdej komórki oddzielnie, w drugim takie obliczenia są równe dla wszystkich komórek i dla przestrzeni traktowane tak samo. W podejściu pierwszym zastosowano algorytm podobny do algorytmów stosowanych w metodzie elementów skończonych z wykorzystaniem funkcji kształtu. W drugim podejściu zastosowano prostsze obliczenia, które zostaną przedstawione niżej.

Ponieważ przy jednorodnym odkształceniu regularnej siatki wszystkie elementy będą miały taki sam kształt, aby określić współrzędne środków lub wierzchołków każdej komórki, wystarczy obliczyć zmiany położenia trzech krawędzi jednej komórki. Wówczas wykorzystując do obliczeń tensor odkształcenia, można uzyskać nowe wymiary komórki:

$$\mathbf{a}_1 = (\mathbf{1} + \mathbf{E})\mathbf{a}_0 \quad (8.36)$$

gdzie:

$\mathbf{a}_0, \mathbf{a}_1$ – wektor wymiarów komórki odpowiednio przed odkształceniem i po odkształceniu, $\mathbf{a} = \{a_x, a_y, a_z\}^T$;

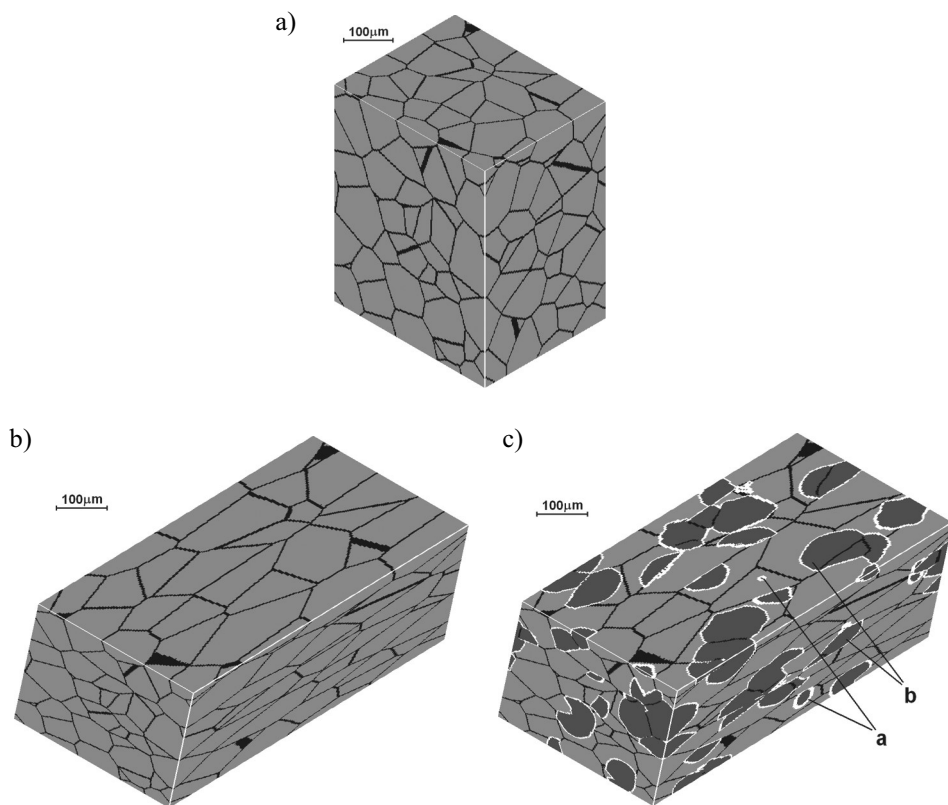
$$\mathbf{E} = \begin{vmatrix} \epsilon_x & \frac{\gamma_{xy}}{2} & \frac{\gamma_{xz}}{2} \\ \frac{\gamma_{yx}}{2} & \epsilon_y & \frac{\gamma_{yz}}{2} \\ \frac{\gamma_{zx}}{2} & \frac{\gamma_{zy}}{2} & \epsilon_z \end{vmatrix} \quad \text{– macierz (tensor) odkształcenia;}$$

$$\mathbf{1} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} \quad \text{– macierz jednostkowa.}$$

Wyniki obliczeń uwzględniające geometrię odkształcenia przedstawiono na rysunku 8.15. Ukazano na nim początkową mikrostrukturę przed odkształceniem (rys. 8.15a), po odkształceniu bez rekrytalizacji (rys. 8.15b) oraz z częściową rekrytalizacją dynamiczną (rys. 8.15c). Zasyulowano odkształcenie stali IF za pomocą automatów komórkowych o rozmiarach $200 \times 200 \times 200$ komórek i $280 \times 350 \times 437,5 \mu\text{m}$. Temperatura odkształcenia $T = 1000^\circ\text{C}$, czas odkształcenia $t = 0,5 \text{ s}$, niezerowe składowe tensora prędkości odkształcenia: $\dot{\epsilon}_x = 1,0$, $\dot{\epsilon}_z = -1,0$, $\dot{\epsilon}_{xz} = 0,07$. Warto zwrócić uwagę na fakt, że kształty przestrzeni i każdej komórki odbiegają od sześciennego, lecz są prostopadłościennie. Zadanie takich wymiarów przestrzeni służyło temu, aby proporcje między krawędziami podczas całego procesu odkształcenia pozostawały zbliżone do 1:1. Na rysunku 8.15b uwidoczniono zmiany kształtu przestrzeni i mikrostruktury podczas jednorodnego odkształcenia, czyli w sytuacji, gdy wzór (8.36) stosuje się względem każdej komórki.

Na rysunku 8.15c przedstawiono mikrostrukturę z zachodzącym procesem rekrytalizacji dynamicznej po odkształceniu w ciągu 0,5 s. Tak jak w poprzednich przykładach, tu również wymiary nowych ziaren zależą głównie od chwili ich zarodkowania.

Mimo że prędkość rozrostu nie jest stała, a siła napędowa wzrostu każdego ziarna – różna i zmienia się z czasem, to jednak wielkość ziarna i chwila zarodkowania pozostają od siebie zależne. Dostrzegalny jest również wpływ kąta dezorientacji ziaren, powodujący, że nowe ziarna rozrastają się przeważnie w kształcie półkul (półelips). Większe nowe ziarna powstałe na początku procesu rekrytalizacji uległy odkształceniu podczas swego rozrostu, co spowodowało, że ich kształt jest wydłużony pod koniec procesu, zgodnie z odkształceniem przestrzeni. Takie ziarna na rysunku 8.15c oznaczono literą „b”. Małe nowe ziarna, które jeszcze nie uległy dużemu odkształceniu, mają kształt zbliżony do kuli (półkuli) i zostały oznaczone literą „a”.



Rys. 8.15. Mikrostruktura przed odkształceniem (a) oraz po odkształceniu bez rekrytalizacji (b) i z rekrytalizacją dynamiczną (c)

8.6.3. Naprężenie uplastyczniające

Naprężenie uplastyczniające jest tym parametrem makroskalowym, który pozwala oszacować procesy zachodzące w mikroskali. To właśnie za jego pomocą wyznaczana jest kinetyka rekrytalizacji dynamicznej. Jednym ze standardowych podejść do modelowania naprężenia uplastyczniającego jest obliczenie średniej gęstości dyslokacji ρ_{av} , a na jej podstawie, z zastosowaniem wzoru (8.5) – naprężenia uplastyczniającego σ .

Średnia gęstość dyslokacji jest wówczas liczona według następujących wzorów (w zależności od przyjętych założeń):

$$\rho_{av} = \frac{1}{N_c} \sum_{i=1}^{N_c} \rho_i \quad (8.37a)$$

$$\rho_{av} = \frac{1}{N_c} \sum_{j=1}^{N_g} \rho_j n_j \quad (8.37b)$$

gdzie:

- i, j – odpowiednio: numer komórki i numer ziarna;
- ρ_i, ρ_j – odpowiednio: gęstość dyslokacji i -tej komórki i j -tego ziarna;
- n_j – liczba komórek w j -tym ziarnie;
- N_c, N_g – liczba komórek i liczba ziaren w modelowej przestrzeni.

Wzór (8.37a) stosuje się, kiedy gęstość dyslokacji jest indywidualna dla każdej komórki i liczona dla niej niezależnie od pozostałych komórek. Wówczas do wzorów rozwoju dyslokacji wprowadzony zostaje czynnik losowy i każda komórka w tym samym ziarnie ma unikalną gęstość dyslokacji. Dzięki temu prędkość rozrostu w każdym punkcie zależy od lokalnej gęstości dyslokacji. Można również uwzględnić różnicę gęstości dyslokacji podczas zarodkowania nowych ziaren.

Wzór (8.37b) stosuje się, kiedy gęstość dyslokacji jest określona jako własność całego ziarna, a nie oddzielnej komórki. Nie oznacza to jednak, że nie można uwzględnić niejednorodności gęstości dyslokacji dla każdej komórki. Czynnik losowy może być wprowadzony lokalnie i tylko przy obliczeniu lokalnej prędkości rozrostu, co pozwala znacznie skrócić czas obliczeń gęstości dyslokacji w każdym kroku obliczeniowym. Natomiast miejsce zarodkowania również wybiera się losowo, na podstawie prawdopodobieństwa zarodkowania, a nie w oparciu o różnicę gęstości dyslokacji, ponieważ skutkiem nierównomierności rozkładu gęstości dyslokacji powinno być takie samo prawdopodobieństwo zarodkowania. Wydaje się więc, że uwzględnienie gęstości dyslokacji dla każdej komórki jest zabiegiem sztucznym, a argumentem przeciwko niemu jest nie tylko fakt, że w takim przypadku niezbędne byłyby większe nakłady obliczeniowe, ale również to, że sposoby wprowadzenia tego czynnika losowego nie są uzasadnione. Nie wiadomo również, jaki rozkład gęstości prawdopodobieństwa dla gęstości dyslokacji należałoby ze względu na to zastosować i w jaki sposób uwzględniać strukturę komórkową i podziaren, kiedy wewnątrz komórek dyslokacyjnych i na ich ścianach istnieją zupełnie inne gęstości dyslokacji. Jeżeli na poziomie modelowania struktur dyslokacyjnych (komórek i podziaren) jest to podejście uzasadnione, to uważam, że na poziomie modelowania ziaren taka strategia jest sztuczna i mało skuteczna. Dlatego w modelu gęstość dyslokacji została przypisana ziarnom i do obliczeń gęstości dyslokacji każdego ziarna zastosowano w nim wzory (8.2)–(8.4), a do obliczeń średniej gęstości dyslokacji – wzór (8.37b).

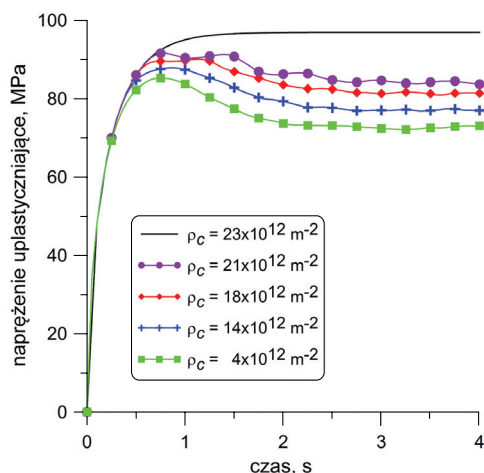
Jest to, jak wspomniano, jedno z najczęściej stosowanych podejść do obliczenia naprężenia uplastyczniającego. Jest ono szybkie, nie wymaga zastosowania skomplikowanych metod i uważa się je za wystarczająco dokładne. Jednak bardziej dokładne, ale nieporównywalnie bardziej skomplikowane i zajmujące o wiele więcej czasu jest badanie polegające na połączeniu automatów komórkowych z innymi metodami, na przykład metodą elementów skończonych. Żeby zorientować się w skali możliwych niedokładności, warto zwrócić uwagę na mikrostrukturę przedstawioną na rysunku 8.14, w której stare, grube, umocnione ziarna są całkowicie otoczone ziarnami nowymi, drobnymi i miękkimi i jakby pływają w nich, a tylko na bardzo małych obszarach stykają się z innymi grubymi ziarnami, co powoduje koncentrację naprężeń, ich wysoką niejednorodność i niejednorodność odkształceń, w szczególności jeżeli wziąć pod uwagę krystaliczną, anizotropową strukturę ziaren. W takiej sytuacji trudno być pewnym, że średnia gęstość dyslokacji, liczona w wyżej opisany sposób, zawsze będzie dobrym odzwierciedleniem tego, co dzieje się w materiale. Jednak jest to podejście zupełnie inne od wyżej opisanego i niektóre jego aspekty zostaną omówione w rozdziale 10 dotyczącym modelowania rozdrobnienia ziaren i uzyskiwania drobnoziarnistej struktury przy odkształceniu na zimno.

Wracając do modelowania rekrytalizacji dynamicznej, jej kinetyki i naprężenia uplastyczniającego, można określić trzy podstawowe parametry, które wpływają na kinetykę rekrytalizacji i naprężenie uplastyczniające. Są to: wartość krytyczna dla zajścia rekrytalizacji dynamicznej gęstości dyslokacji, prędkość zarodkowania (lub prawdopodobieństwo powstania nowych ziaren) i prędkość rozrostu ziaren.

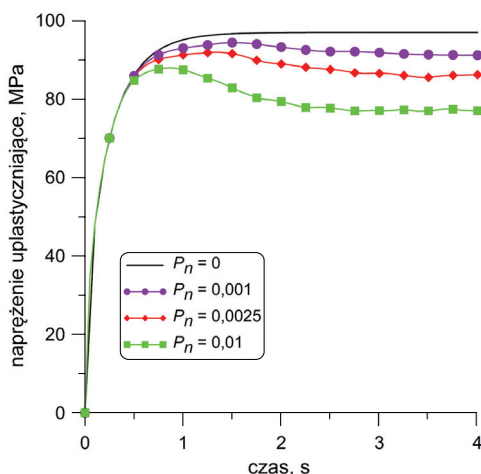
Obecnie przeprowadzono jedynie badania wstępne, które nie są związane z konkretnym materiałem ani nawet z konkretnymi parametrami procesowymi. Ich wyniki stanowią jedynie podstawę do dalszych badań.

Na rysunku 8.16 przedstawiono wpływ wartości krytycznej gęstości dyslokacji na naprężenie uplastyczniające. Obniżenie wartości krytycznej przy stałej prędkości zarodkowania i prędkości rozrostu powoduje wcześniejsze zarodkowanie, niewielkie zwiększenie liczby zarodków i większe osłabienie materiału. Zwiększenie wartości krytycznej prowadzi do okresowości w zarodkowaniu, o czym świadczy mniejsza gładkość krzywej, a w końcu do warunków, w których rekrytalizacja dynamiczna nie zachodzi. Obniżenie wartości krytycznej o 10% w porównaniu z wartością, przy której rekrytalizacja już nie zachodzi, pociąga za sobą takie samo obniżenie naprężenia uplastyczniającego jak kolejne obniżenie tej wartości do zera. Naprężenie jest bardzo czułe na wartość krytyczną zbliżoną do maksymalnej i prawie nieczułe na małe jej wartości.

Na rysunku 8.17 przedstawiono wyniki symulacji naprężenia uplastyczniającego przy różnych wartościach prędkości zarodkowania. Prędkość zarodkowania jest zadana jako prawdopodobieństwo powstania zarodka na jednostkę powierzchni granic ziaren w jednostce czasu. Wybrano wartość krytyczną dla zajścia rekrytalizacji dynamicznej równą zero, która powoduje największy spadek krzywej umocnienia. Zwiększenie prędkości zarodkowania zupełnie naturalnie prowadzi do przyspieszenia rekrytalizacji dynamicznej oraz obniżenia naprężenia uplastyczniającego. Również maksymalne (pikowe) naprężenie występuje wcześniej.

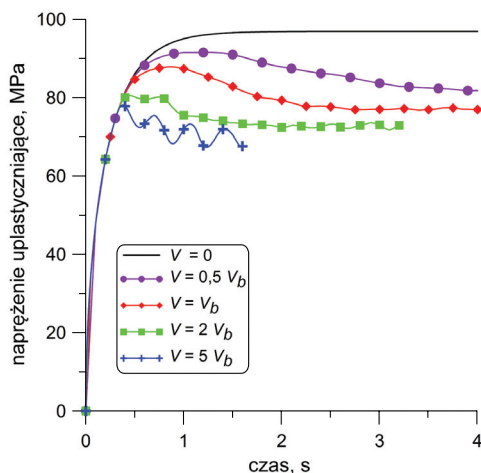


Rys. 8.16. Napężenie uplastyczniające przy różnych wartościach krytycznej gęstości dyslokacji



Rys. 8.17. Napężenie uplastyczniające w zależności od prędkości zarodkowania

Na rysunku 8.18 ukazano zmiany napężenia uplastyczniającego przy różnych prędkościach rozrostu zrekrystalizowanych ziaren. Zwiększenie prędkości zarodkowania, jak i rozrostu, powoduje przyspieszenie rekrytalizacji dynamicznej, obniżenie napężenia uplastyczniającego i wcześniejsze występowanie napężenia pikowego. Zbyt duża prędkość rozrostu prowadzi do okresowości i powoduje wystąpienie napężenia uplastyczniającego, co jest związane z szybkim wysyceniem miejsc zarodkowania i szybkim wypełnieniem przestrzeni nowymi ziarnami.



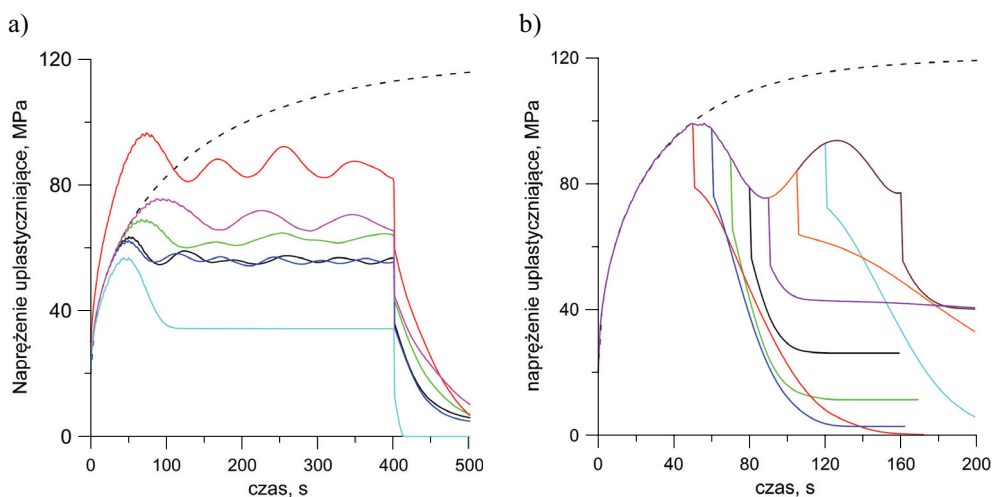
Rys. 8.18. Napężenie uplastyczniające w zależności od prędkości rozrostu zrekrystalizowanych ziaren

Należy zauważyć, że wszystkie trzy parametry w podobny sposób wpływają na charakter zmian napężenia uplastyczniającego i przy różnych kombinacjach tych parametrów można otrzymać taką samą krzywą, co oznacza, że badanie kinetyki rekrytalizacji na podstawie tylko krzywych napężenia uplastyczniającego jest niewystarczające. Zagadnienie ma więcej stopni swobody, niż jest to konieczne do poprawnego wyznaczenia parametrów zarodkowania i rozrostu ziaren. Krzywa umocnienia wprowadza więc tylko jedno ograniczenie stopni swobody. Drugie ograniczenie stopni swobody może wprowadzić badanie mikrostruktury, które pozwoli określić liczbę ziaren i pośrednio prędkość zarodkowania, ale i wówczas trudno jednoznacznie odpowiedzieć na pytanie, czy to wystarczy do określenia wszystkich trzech parametrów. Należy przy tym podkreślić trudność, jaką sprawiają badania mikroskopowe w warunkach rekrytalizacji dynamicznej, kiedy następują bardzo szybkie zmiany mikrostruktury w wysokich temperaturach i nie ma możliwości „zamrożenia” aktualnej mikrostruktury.

8.6.4. Rekrytalizacja metadynamiczna

Rekrytalizacja metadynamiczna jest kontynuacją rekrytalizacji dynamicznej po przerwaniu odkształcenia. Według jednej z koncepcji zarodkowanie wówczas dalej się nie odbywa, następuje tylko rozrost istniejących ziaren, natomiast według innej koncepcji zarodkowanie może zachodzić tak jak podczas rekrytalizacji statycznej. Ten drugi wariant czasami nazywany jest rekrytalizacją metadynamiczną połączoną z rekrytalizacją statyczną. W opracowanym modelu założono, że zarodkowanie następuje tylko podczas odkształcenia i przeprowadzono wyłącznie wstępne symulacje kinetyki rekrytalizacji.

Na rysunku 8.19 pokazano przykłady zmian naprężenia podczas odkształcenia i ich relaksacji po odkształceniu (bez uwzględnienia zdrowienia statycznego). Na rysunku 8.19a ukazano przebieg relaksacji dla kilku różnych krzywych naprężenia uplastyczniającego, na rysunku 8.19b – dla tej samej krzywej z okresowymi zmianami naprężenia uplastyczniającego, które jest związane z szybkim rozrostem ziaren. Można na tym rysunku zobaczyć, że kinetyka rekrytalizacji metadynamicznej w ostatnim przypadku jest silnie uzależniona od fazy, w której następuje przerywanie odkształcenia. Od tej fazy zależy również i stopień osłabienia materiału, co świadczy o tym, że w efekcie rozrostu zrekrystalizowanych ziaren nie następuje pełne usunięcie skutków zgniotu i w ziarnach pozostają dyslokacje, które mogą być częściowo usunięte tylko w wyniku zdrowienia statycznego. Spostrzeżenia te są zgodne z teorią i z wynikami doświadczeń. Jednak poza zbieżnością jakościową trudno mówić o jakiegokolwiek ocenie ilościowej. Do takiej oceny jest zbyt mało danych.



Rys. 8.19. Kinetyka rekrytalizacji metadynamicznej

8.7. Wielkość ziarna

Wielkość ziarna to jeden z podstawowych parametrów mikrostruktury, który jest uwzględniany zarówno w modelach empirycznych, jak i w modelowaniu numerycznym. Dotyczy to również i modelowania za pomocą automatów komórkowych, które dają znacznie większe możliwości opisu wielkości ziarna. Jedną z takich możliwości jest przedstawienie wielkości ziaren w postaci rozkładu, co zostało rozpatrzone w rozdziale 6. Jednak średnia wielkość ziarna nadal pozostaje parametrem, który jest wykorzystywany do opisu zmian zachodzących w mikrostrukturze, m.in. podczas rekrytalizacji. Jest to trudne zadanie, a jego rozwiązanie zależy od przyjętych założeń.

Najprostszym rozwiązaniem jest obliczenie średniej wielkości ziaren d_{av} na podstawie objętości modelowej przestrzeni V i liczby ziaren n_g :

$$d_{av} = \sqrt[3]{\frac{6V}{\pi n_g}} \quad (8.38)$$

Współczynnik $6/\pi$ oznacza, że wielkość ziarna jest liczona tak jak średnica kuli o tej samej co ziarno objętości. Wzór ten jest skuteczny, kiedy mamy do czynienia z ustaloną strukturą, w której nie zachodzą zmiany, a kształt ziaren jest zbliżony do równoosiowego.

W niektórych wzorach empirycznych zamiast wielkości ziarna stosuje się wskaźnik, który oprócz wymiarów ziaren uwzględnia również ich kształt lub odkształcenie. W takim przypadku określa się stosunek objętości ziaren V do powierzchni ich granic S , który w przypadku ziaren równoosiowych jest taki sam:

$$d_{av} = \frac{3V}{S} \quad (8.39)$$

Gdy według wzoru (8.38) jest modelowana na przykład rekryształizacja statyczna, wielkość ziarna podczas prawie całego procesu pozostaje bez zmian, a zmiany następują w krótkim czasie prawie skokowo, taki parametr nie niesie ze sobą żadnej istotnej informacji. Natomiast w warunkach określonych wzorem (8.39) następują ciągłe zmiany wielkości ziarna, które są opisywane wzorami zbliżonymi do wzorów wykorzystujących ułamek rekryształizacji w przypadku niepełnej rekryształizacji. Dlatego w analizowanym modelu zastosowano głównie obliczenia oparte na wzorze (8.39).

Objętość modelowa V podczas symulacji prawie zawsze pozostaje bez zmian, natomiast powierzchnia granic się zmienia. We frontalnych automatach komórkowych stosuje się dwa rodzaje granic: stałe, nieruchome (komórki w stanie q_2) i mobilne granice nowych rosnących ziaren (komórki w stanie q_1). Wówczas sumaryczną powierzchnię wszystkich ziaren można obliczyć według następującego wzoru:

$$S = (k_1 n_1 + k_2 n_2) S_c \quad (8.40)$$

gdzie:

n_1, n_2 – liczba komórek, odpowiednio, w stanie q_1 i q_2 ;

S_c – średnia powierzchnia ściany komórki;

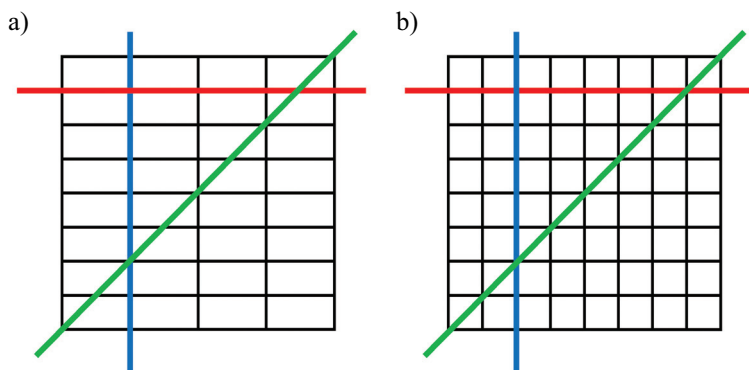
k_1, k_2 – współczynniki.

Wzór (8.40) oraz sposób wyznaczenia komórek w odpowiednim stanie wprowadzają do obliczeń powierzchni kilka uproszczeń, dlatego nie dają one dokładnego wyniku, lecz przybliżone, natomiast współczynniki k_1, k_2 służą do uwzględnienia niektórych zależności, które będą omówione niżej. Do wyjaśnienia natury niektórych uproszczeń wykorzystywane są schematy przedstawione na rysunku 8.20. Widać na nim

linie i dwuwymiarową siatkę, co upraszcza wyjaśnienia i może zostać łatwo, za pomocą zastąpienia długości powierzchnią, odniesione do przypadku trójwymiarowego.

Pierwsze uproszczenie jest związane z dyskretyzacją przestrzeni automatów komórkowych. W regularnej siatce o tej samej długości boków (rys. 8.20b) rzeczywista długość granicy (zaznaczona za pomocą kolorowych linii) będzie się różnić od długości dyskretnej. Granica dyskretna powinna przebiegać po bokach komórek, więc jej długość będzie zależała od pochylenia granicy rzeczywistej i będzie dłuższa od długości rzeczywistej granicy. Wówczas zielona linia (poprowadzona po przekątnej) będzie krótsza (1,41 razy) od jej dyskretnej reprezentacji, a czerwona (pozioma) i niebieska (pionowa) będą równe.

Jednak stwierdzenie, że linia czerwona i niebieska są równe, byłoby słuszne wtedy, gdyby liczyć długość boków komórek leżących na granicach ziaren. Natomiast zgodnie ze wzorem (8.40) liczono nie długość boków komórek (co jest zadaniem bardziej skomplikowanym), lecz same komórki. Wówczas mamy do czynienia z błędem związanym ze sposobem obliczeń i teraz już, jeśli weźmie się pod uwagę liczbę komórek, zielona linia okaże się dłuższa od jej reprezentacji. Dlatego też błąd ten będzie tym mniejszy, im mniejsze będą różnice długości boków. W przypadku komórek sześciennych czynnik korygujący ($k = 1,23$) jest wprowadzony w celu uwzględnienia różnicy między rzeczywistą powierzchnią granicy i jej komórkową reprezentacją, czyli rzeczywista powierzchnia jest większa od komórkowej reprezentacji 1,23 razy. W przypadku odkształconych komórek (lub komórek o różnej długości boków) czynnik k wzrasta i zależy od odkształcenia.



Rys. 8.20. Schemat do określenia błędów obliczania długości (powierzchni) granic ziaren

Kolejny błąd związany z dyskretyzacją przestrzeni jest spowodowany różną powierzchnią ścian komórki (różnymi długościami boków), co można zobaczyć na rysunku 8.20a. Różnice te mogą powstać podczas symulacji odkształcenia. Wówczas długość granicy w reprezentacji komórkowej dodatkowo zależy od wymiarów komórki i w takiej reprezentacji czerwona linia będzie dwukrotnie krótsza od niebieskiej. Średnia długość (powierzchnia) granic podczas odkształcenia również się zmienia, a ich długość (powierzchnia) sumaryczna ulega zmianie przy zastosowaniu innej siatki, na

przykład podczas reorganizacji przestrzeni, co na wykresie spowoduje skokową zmianę wielkości ziarna, której tak naprawdę nie będzie.

Różne sposoby symulowania za pomocą FCA komórek na granicy ziarna, czyli komórek w stanach q_1 i q_2 , prowadzą do powstania jeszcze jednej różnicy: komórki na nieruchomej granicy ziarna (w stanie q_2) określa się za pomocą otoczenia von Neumanna i każdy elementarny odcinek granicy jest reprezentowany dwukrotnie, za pomocą dwóch komórek leżących po obu stronach tej granicy, natomiast komórki na granicach mobilnych (w stanie frontalnym q_1) określone są za pomocą otoczenia Moore'a i reprezentują granicę jednokrotnie, tylko po jednej stronie.

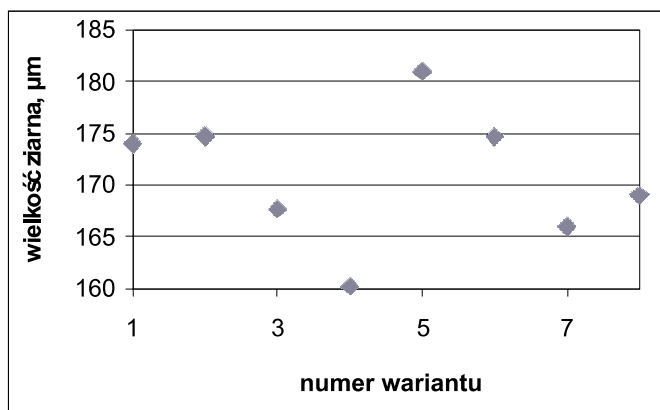
Aby uwzględnić wszystkie te różnice i zminimalizować błędy systematyczne, wprowadzono współczynniki k_1 i k_2 .

Współczynnik k_2 można określić dla początkowej lub końcowej struktury, w której nie zachodzą zmiany, czyli brak jest ruchomych granic. Wówczas należy obliczyć średnią wielkość ziarna za pomocą wzoru (8.38) i dobrać współczynnik k_2 tak, żeby otrzymać taki sam wynik za pomocą wzorów (8.39) i (8.40).

Współczynnik k_1 jest dobierany przez przerwanie symulacji w obecności ruchomych granic i zastąpienie ich granicami nieruchomymi z uwzględnieniem zmiany otoczenia i dwukrotnej reprezentacji. Wówczas obydwa warianty obliczeń powinny dostarczyć podobnych wyników.

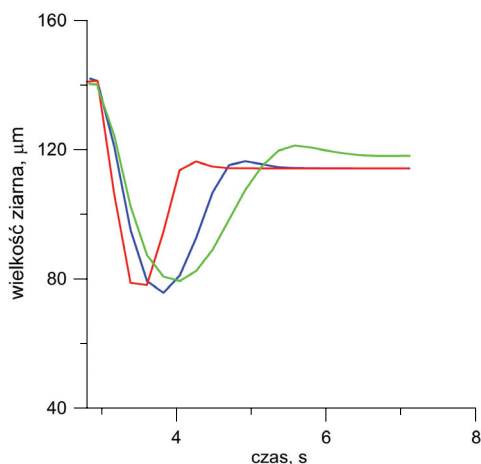
Rysunek 8.21 demonstruje sposób rozpatrywania tej samej struktury otrzymywanej w przestrzeni o różnych rozmiarach i różnym kształcie komórek. Średnia wielkość ziarna otrzymana przy zastosowaniu wzoru (8.38) wynosiła 169,5 μm , natomiast obliczona według wzorów (8.39) i (8.40) zmieniała się w zakresie od 160,2 μm do 181,0 μm .

Odształcenie prowadzi również do zmiany granic ziaren (ale w obliczeniach zmiana ta jest mniejsza niż w rzeczywistości, odształcenie 0,6 zmniejsza wielkość ziarna o 20%), a podczas reorganizacji przestrzeni powierzchnia ta wzrasta, co na wykresie widoczne jest w postaci skokowej zmiany wielkości ziarna.



Rys. 8.21. Wyniki obliczenia wielkości ziaren

Podsumowując, można stwierdzić, że zastosowana tu metoda obliczenia wielkości ziarna może się charakteryzować błędem rzędu $10\div 20\%$, ale powinna dobrze odzwierciedlać zachodzące zmiany jakościowe. Na rysunku 8.22 pokazano przykład zmian wielkości ziarna podczas rekrytalizacji statycznej. Więcej takich wykresów znajduje się w rozdziale 11, dotyczącym modelowania procesów technologicznych z wieloetapowym odkształceniem i przerwami między poszczególnymi etapami. Zmniejszenie wielkości ziarna pod koniec procesu jest związane z niedokładnością współczynnika k_1 . Minimalna wielkość ziarna jest osiągana, kiedy wielkości ziaren niezrekrytalizowanych i zrekrystalizowanych są do siebie zbliżone.



Rys. 8.22. Zmiana wielkości ziarna podczas rekrytalizacji statycznej

8.8. Kierunki dalszych badań rekrytalizacji

Przewiduje się dalszą rozbudowę modelu opartego na automatach komórkowych. Główne zasady działania owego modelu można uznać za opracowane, jednak aby mógł on poprawnie modelować rzeczywiste zjawiska i procesy, konieczne jest dopracowanie wielu szczegółów. Dlatego należy przeprowadzić dodatkowe doświadczalne badania plastometryczne przy stałych i zmiennych warunkach odkształcenia. Badania te dostarczą informacji potrzebnych nie tylko do modelowania rozwoju dyslokacji, ale również modelowania rekrytalizacji dotyczących kinetyki podczas odkształcenia. Na ich podstawie można będzie ustalić warunki zarodkowania i rozrostu ziaren. Wspomniane badania powinny być również uzupełnione badaniami mikroskopowymi, które dostarczą dodatkowych informacji o mikrostrukturze, a więc o zarodkowaniu i rozroście ziaren.

Test relaksacji naprężeń jest kolejną metodą doświadczalną, która w połączeniu z badaniami mikroskopowymi pozwoli zbadać proces rekrytalizacji (statycznej, meta-dynamicznej) po odkształceniu.

Podczas modelowania należy tak sterować pracą automatów komórkowych, żeby dokładnie odtworzyć proces nazywany rekrytalizacją dynamiczną, kiedy to następuje ciągle zarodkowanie i rozrost ziaren w czasie odkształcenia. Zabieg ten jest wielokryterialną optymalizacją, w której należy dopasować do zadanej zarówno kinetykę procesu, zmiany naprężenia uplastyczniającego, jak i końcową mikrostrukturę. Dalej należy dopasować model do zmiennych warunków odkształcenia i rekrytalizacji statycznych z uwzględnieniem zdrowienia statycznego. Dodatkowym ograniczeniem jest niezmiennosc wyników rekrytalizacji dynamicznych.

W efekcie model pozwoli na bardziej dokładne i szczegółowe symulacje przemian zachodzących w materiałach, z uwzględnieniem różnorodnych wpływów różnych parametrów procesów i zjawisk. Wprowadzenie do automatów komórkowych i powiązanie ze sobą różnych modeli, które obecnie często oddzielnie odzwierciedlają różne cechy tego samego zjawiska, pozwoli stworzyć bardziej realistyczny obraz zachodzących w materiale zmian, lepiej zrozumieć zależności między zjawiskami i uzyskać bardziej adekwatny model rozwoju dyslokacji i mikrostruktury.

Oprogramowanie może być wykorzystane do analizy rzeczywistych procesów, może też służyć za narzędzie do uzyskania mikrostruktury przeznaczonej do badań modelowych własności mechanicznych oraz jako składnik systemów komputerowego wspomaganie projektowania procesów technologicznych, w których występują zadania optymalizacji pod względem na parametrów mikrostruktury i (lub) własności mechanicznych. Oprogramowanie może być również wykorzystane w dydaktyce lub jako wirtualne laboratorium.

9. Przemiany fazowe w stanie stałym

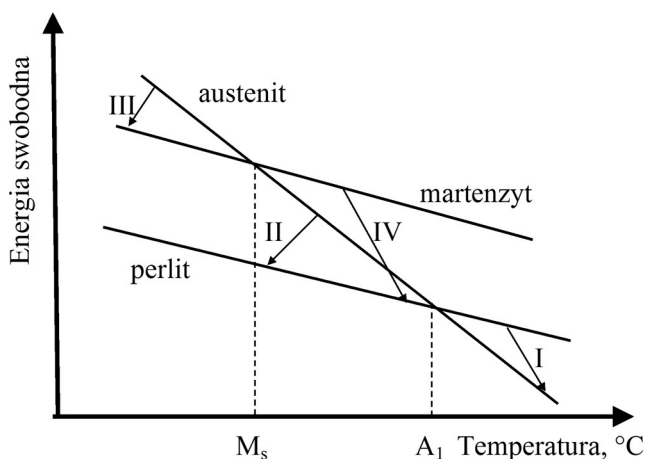
Przemiany fazowe są kolejnym zjawiskiem poddawanych symulacji w układach do modelowania rozwoju mikrostruktury. Przejście od przeróbki plastycznej na gorąco do przeróbki na zimno lub wykorzystanie materiału w temperaturze pokojowej wymaga wprowadzenia do układu modelowego bloku odpowiedzialnego za przemiany fazowe w stanie stałym. W niniejszym rozdziale zostały przedstawione podstawowe wiadomości na temat przemian fazowych w stali (podrozdz. 9.1) oraz obecny stan modelowania przemian fazowych (podrozdz. 9.2). Modelowanie zjawisk przemian fazowych dopiero zaczyna być powszechnie stosowane i ze względu na różnorodność oraz stopień skomplikowania tego typu procesów rozpatruje się tylko bardzo ograniczone odmiany przemian i warunki ich zajścia. W podrozdziale 9.3 przedstawiono model dyfuzyjnych przemian austenitu w ferryt i perlit przy małych prędkościach chłodzenia, natomiast w podrozdziale 9.4 omówiono wyniki symulacji takiej przemiany.

9.1. Podstawowe wiadomości o przemianach fazowych w stali

W stali rozpatrywano cztery podstawowe przemiany fazowe, które sklasyfikowano w zależności od fazy początkowej, kierunku i szybkości zmiany temperatury. Siłą napędową każdej przemiany jest różnica między energią swobodną składników strukturalnych. Na rysunku 9.1 schematycznie przedstawiono zależność energii swobodnej od temperatury austenitu, perlitu i martenzytu oraz wywoływane tą różnicą cztery podstawowe przemiany fazowe.

Czterema podstawowymi przemianami są:

- 1) przemiana perlitu w austenit,
- 2) dyfuzyjna przemiana austenitu,
- 3) bezdyfuzyjna przemiana austenitu w martenzyt,
- 4) rozkład martenzytu na ferryt i węgliki.

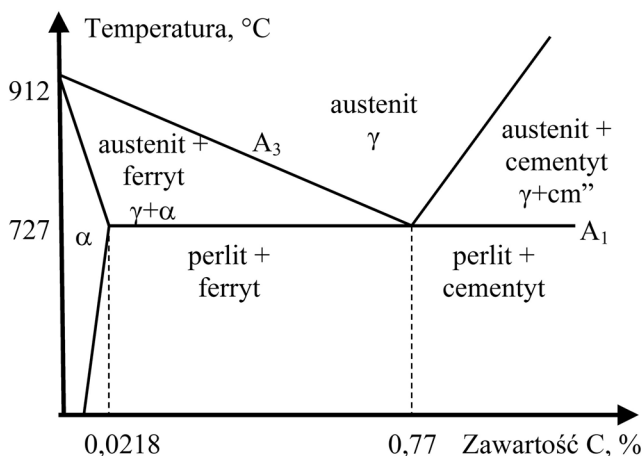


Rys. 9.1. Zależność energii swobodnej od temperatury dla austenitu, perlitu i martenzytu oraz przemiany fazowe

Pierwsza przemiana następuje przy nagrzaniu stali powyżej temperatury A_1 . Siłą napędową jest tu różnica w energii swobodnej, która wzrasta w miarę zwiększania stopnia przegrzania. Przemiana ta składa się z trzech etapów: utworzenie austenitu niejednorodnego, utworzenie austenitu jednorodnego i rozrost ziaren. Austenit niejednorodny powstaje w wyniku zarodkowania na granicy ferrytu i cementytu, a następnie rozrostu nowych ziaren austenitu wewnątrz ferrytu aż do pełnej przemiany i jednocześnie, ale z mniejszą szybkością, wewnątrz płytek cementytu. Kiedy cały cementyt zostanie przemieniony w austenit (węgiel zostanie pochłonięty przez austenit), kończy się pierwszy etap i powstaje austenit niejednorodny. Dalej następuje wyrównanie zawartości węgla w całej objętości austenitu, co oznacza utworzenie austenitu jednorodnego. Później (lub jednocześnie z ostatnimi fazami wyrównania zawartości węgla) następuje rozrost ziarna, ponieważ ze względu na energię swoich granic większe ziarna są bardziej korzystne.

Dyfuzyjna przemiana austenitu może przebiegać w różny sposób, w zależności od zawartości węgla i prędkości chłodzenia. Przy niewielkich stopniach przechłodzenia w stali zachodzi przemiana perlityczna, natomiast kiedy spadek temperatury jest znaczny i obniża się ona poniżej wartości B_s (*bainite start*), zachodzi przemiana bainityczna. Jeżeli zawartość węgla $[C]$ nie odpowiada składowi chemicznemu stali eutektoidalnej ($[C] \neq 0,77\%$) (rys. 9.2), a prędkość chłodzenia nie jest duża, to przemianę perlityczną poprzedza wydzielenie ferrytu (dla stali podeutektoidalnej $[C] < 0,77\%$, $T < A_3$) lub cementytu (dla stali nadeutektoidalnej $[C] > 0,77\%$) na granicach ziaren austenitu. Przy małych zawartościach węgla ($[C] < 0,0218\%$) cały austenit może przemienić się w ferryt.

Jeżeli szybkość chłodzenia jest niewielka, w stali podeutektoidalnej poniżej temperatury A_3 z austenitu na granicach ziaren wydzieli się ferryt z maksymalną dopuszczalną dla tej temperatury zawartością węgla (0,0218% przy $T = 727^\circ\text{C}$), przy czym austenit wzbogaca się w węgiel (do 0,77% w temperaturze $T = 727^\circ\text{C}$).



Rys. 9.2. Fragment układu równowagi fazowej Fe-Fe₃C wykorzystanego do analizy kinetyki przemiany austenitu za pomocą jednowymiarowych automatów komórkowych

W stali nadeutektoidalnej na granicach ziaren powstaje siatka drobnych cząstek cementytu ($C = 6,67\%$), natomiast zawartość węgla w austenicie zmniejsza się i również osiąga wartości $0,77\%$ w temperaturze $T = 727^\circ\text{C}$. Po spadku temperatury poniżej A_1 następuje przemiana perlityczna.

Gdy prędkości chłodzenia są większe, przed rozpoczęciem przemiany perlitycznej ani stężenie węgla w austenicie nie osiąga równoważnej wartości, ani ferryt (lub cementyt) nie wydzieli się w całości na granicach ziaren. Zatem powstający perlit będzie albo uboższy, albo bogatszy w węgiel.

Perlit jest kolonią na przemian ułożonych płytek ferrytu i cementytu. Zarodkami mogą być cząstki cementytu lub drobne wydzielenia ferrytu na granicach ziaren. Z tych zarodków rozpoczyna się rozrost płytki cementytu lub ferrytu. Rosnąca płytka cementytu pochłania węgiel z austenitu i, kiedy jego stężenie spadnie do poziomu odpowiadającego jego zawartości w ferrycie, zaczyna rosnać płytka ferrytu, która odpycha węgiel, po czym z kolei powstaje nowa płytka cementytu, a następnie cykl się powtarza aż do wyczerpania całego austenitu.

W wyniku szybkiego obniżenia temperatury po przemianie perlitycznej (lub zamiast niej) może zajść przemiana bainityczna (która jest również nazywana przemianą pośrednią). Przemiana ta rozpoczyna się od powstania zarodków ferrytu na granicach ziaren. Z tego ferrytu później wydzielają się cząstki węglików. W zależności od temperatury, w jakiej zachodzi przemiana bainityczna, rozróżnia się charakteryzujące się różną strukturą bainit górny i bainit dolny.

Trzecią przemianą fazową również jest przemiana austenitu, ale zachodzi ona przy bardzo dużych prędkościach chłodzenia i ma charakter bezdyfuzyjny. W jej wyniku powstaje martenzyt, przesycony roztwór węgla w żelazie alfa. Przemiana ta przebiega

jednokierunkowo. Jej prędkość jest bardzo duża, zbliżona do prędkości dźwięku w stali. Ułamek przemiany zależy od temperatury i składu chemicznego stali.

Ostatnią przemianą zachodzącą podczas nagrzewania zahartowanej stali jest rozkład martenzytu na ferryt i węgliki.

9.2. Modelowanie przemian fazowych

Głównymi narzędziami do modelowania przemian fazowych są automaty komórkowe (CA), często wykorzystywane w połączeniu z metodą różnic skończonych (MRS lub CA+FDM) i metodą pól fazowych (*phase-field* – PF lub *multi-phase-field* – MPF).

Prawdopodobnie jedną z pierwszych prób modelowania przemian fazowych była praca Iba *et al.* (1986). Pod koniec XX wieku powstały modele, które lepiej uwzględniały rzeczywiste warunki przemian. Za przełomową należy uznać publikację Kumara *et al.* (1998), w której uwzględniono zarodkowanie ferrytu na granicach ziaren austenitu i wykorzystano do symulacji zarodkowania model opracowany przez Thevoza *et al.* (1989) na potrzeby symulacji krzepnięcia. Prędkość zarodkowania I wyznaczono przez funkcję Gaussa, dobierając maksymalną prędkość zarodkowania N_{\max} , temperaturę przy maksymalnej prędkości zarodkowania T_{\max} i rozrzut temperatury zarodkowania ΔT_G :

$$I = \frac{N_{\max}}{\Delta T_G \sqrt{2\pi}} \exp \left[-\frac{(T_{\max} - T)^2}{2\Delta T_G^2} \right] \quad (9.1)$$

Autorzy wspomnianej publikacji uzależnili prędkość rozrostu ziaren od współczynnika dyfuzji węgla z wykorzystaniem bilansu rozpuszczalności opisanego przez Kurtza i Fischera (1986):

$$v = D_\gamma \frac{dC_\gamma}{dn} \frac{1}{C_\gamma^* (1 - k)} \quad (9.2)$$

gdzie:

dC_γ/dn – gradient stężenia węgla w austenicie w kierunku prostopadłym do granicy rosnącego ziarna;

C_γ^* – stężenie węgla w austenicie zgodnie z wykresem równowagi fazowej w aktualnej temperaturze;

k – współczynnik podziału węgla w austenicie i ferrycie;

D_γ – współczynnik dyfuzji węgla w austenicie.

Kurtz i Fisher zbadali również kinetykę przemiany przy różnych prędkościach chłodzenia. Varma *et al.* (2001) rozwinęli ten model, rozdzielając zarodkowanie oraz rozrost ferrytu i perlitu, jednak realizacja tego modelu za pomocą automatów komórkowych okazała się zbyt schematyczna.

Inny bardzo ciekawy model zaproponowali Zhang *et al.* (2003a, 2003b). Wykorzystali oni wykres równowagi fazowej układu żelazo-węgiel (por. rys. 9.2) w celu zobrazowania zarodkowania ferrytu. Do opisu rozrostu wykorzystali równanie dyfuzji, które wraz z wyznaczeniem prędkości rozrostu rozwiązywano poza automatami komórkowymi, natomiast do rozrostu ziaren zastosowano stochastyczne reguły przejścia. Zgodnie z modelem Umemoto *et al.* (1989) podczas ciągłego chłodzenia od temperatury A_{e3} do zadanej temperatury T , liczbę nowych ziaren ferrytu można wyznaczyć za pomocą równania:

$$n(T) = \int_T^{A_{e3}} \frac{I(T')(1-f)}{Q} dT' \quad (9.3)$$

gdzie:

- $I(T')$ – prędkość zarodkowania ferrytu w jednostce powierzchni granicy ziarna w temperaturze T' ;
- f – ułamek objętościowy ferrytu;
- Q – prędkość chłodzenia.

Prędkość zarodkowania wyrażono następującym wzorem:

$$I_s = \frac{K_1}{\sqrt{kT'}} D_\gamma \exp \left[-\frac{K_2}{kT'(\Delta G)^2} \right] \quad (9.4)$$

gdzie:

- K_1 – stała odpowiadająca za gęstość zarodkowania;
- K_2 – energia granic międzyfazowych austenit-ferryt;
- k – stała Boltzmanna;
- D_γ – współczynnik dyfuzji węgla w austenicie;
- ΔG – różnica energii swobodnej Gibbisa – siła napędowa przemiany austenit-ferryt.

Do obliczenia prędkości rozrostu ziarna ferrytu wspomniani autorzy zastosowali następujące równanie:

$$D_\gamma \frac{\partial C_\gamma}{\partial \mathbf{n}} - D_\alpha \frac{\partial C_\alpha}{\partial \mathbf{n}} = v(C'_\alpha - C'_\gamma) \quad (9.5)$$

gdzie:

- C_γ C_α – stężenie węgla odpowiednio w austenicie i ferrycie;
- C'_γ C'_α – stężenie węgla odpowiednio w austenicie i ferrycie w pobliżu granicy ziarna;
- D_γ D_α – współczynnik dyfuzji węgla w austenicie i ferrycie;
- v – prędkość rozrostu.

Umemoto *et al.* (1987) uwzględnili również wydzielanie cementytu, choć w przypadku przedstawionej w ich opracowaniu stali podeutektoidalnej wydaje się to zabiegiem zbędnym.

Lan *et al.* (2004) najpierw wprowadzili do modelu zbliżonego do modelu opracowanego przez Zhanga *et al.* deterministyczne reguły rozrostu ziaren ferrytu w niskowęglowej stali, a nieco później Lan wraz współautorami (Lan *et al.* 2005) uwzględnili odkształcenie poprzedzające przemianę. W tym przypadku prędkość rozrostu obliczano według wzoru:

$$v = mF \quad (9.6)$$

gdzie:

m – mobilność granicy;

F – siła napędowa, która składa się ze składowej chemicznej (F_{chem}) i składowej odkształceniowej (F_{def}).

Następnie Zheng *et al.* (2008, 2009) modelowali dynamiczną przemianę austenit-ferryt w stali niskowęglowej indukowanej odkształceniowo.

Li *et al.* (2007), zgodnie z klasyczną teorią Langa *et al.* (1988), wprowadzili okres inkubacyjny:

$$I = I_s \exp\left(-\frac{\tau}{t}\right) \quad (9.7)$$

gdzie:

I_s – prędkość zarodkowania zgodnie z (9.4);

τ – okres inkubacyjny;

t – czas.

Okres inkubacyjny wyznaczano według następującego wzoru (Lang *et al.* 1988):

$$\tau = \frac{12kTa^4\sigma_{\alpha\gamma}}{D_\gamma x_\gamma v_\alpha^2 (\Delta G_v)^2} \quad (9.8)$$

gdzie:

a – średni parametr sieci dla fazy austenitu i ferrytu;

x_γ – ułamek molowy węgla w austenicie;

$\sigma_{\alpha\gamma}$ – energia międzyfazowa nieuporządkowanego ferrytu;

v_α – objętość atomów żelaza w ferrycie.

Wartości tych parametrów również można znaleźć w publikacji Langa *et al.* (1988).

Bos *et al.* (2010) oraz Mecozzi *et al.* (2011) opracowali trójwymiarowy model przemian stali dwufazowej (ferryt + perlit) podczas procesu wyżarzania, który przedstawia rekrytalizację ferrytu oraz przemiany ferryt-austenit, perlit-austenit, austenit-ferryt i austenit-martenzyt.

Na uwagę zasługuje również podjęta przez Oparę *et al.* (2012) próba kompleksowego modelowania przemiany austenitu w ferryt, bainit i martenzyt.

9.3. Model przemian fazowych

Każda z wyżej przedstawionych czterech podstawowych przemian jest bardzo skomplikowanym zjawiskiem i wymaga szczególnej uwagi. W niniejszej pracy zwrócono uwagę tylko na dyfuzyjną przemianę austenitu, przy czym rozpatrywano jedynie przemiany zachodzące przy niewielkich prędkościach chłodzenia, czyli przemianę austenitu w perlit i poprzedzającą ją wydzielanie ferrytu w stali podeutektoidalnej. W przypadku stali o większej zawartości węgla (nadeutektoidalnej) rozpatrywano jedynie przemianę perlityczną, pomijając dokładną analizę przebiegu wydzielania cementytu. Założono jedynie, że zachodzi ono na granicach ziaren, tworząc siatkę z drobnych cząstek, nie zmieniając początkowego kształtu ziaren austenitu i ograniczając rozrost ziaren perlitu. Również wydzielania ferrytu na granicach ziaren austenitu utrudniają ten rozrost, ale prześwity między ziarnami ferrytu stwarzają możliwość wrastania ziaren perlitu w kilka ziaren, a nie w tylko jedno ziarno austenitu.

Podstawą modelu automatu komórkowego nadal jest automat frontalny, którego schemat został przedstawiony w rozdziale 4 na rysunku 4.9. Ponieważ dyfuzyjne przemiany można opisać w ten sam sposób, model oparty jest na zarodkowaniu i rozroście ziaren innej fazy.

Zarodkowanie podczas przemian fazowych ma podobne cechy jak inne, opisane w poprzednich rozdziałach procesy, ale posiada również cechy szczególne. Zarodkowanie zachodzące podczas dyfuzyjnej przemiany austenitu, podobnie jak podczas krzepnięcia i rekrytalizacji, ma wyraźnie heterogeniczny charakter. Podczas krzepnięcia zarodkowanie rozpoczyna się na przechłodzonej powierzchni krystalizatora i jego warunki można rozpatrywać jako jednakowe dla całej powierzchni, ponieważ nie trzeba uwzględniać krzywizn i zaokrągleń krystalizatora, które są znacznie większe w porównaniu z wielkością zarodków, a chropowatość powierzchni także jest wyrównana. Podczas rekrytalizacji również następuje heterogeniczne zarodkowanie na powierzchniach granicznych ziaren, ale nie stwierdzono preferowanego zarodkowania na ich krawędziach lub narożach. Natomiast podczas dyfuzyjnej przemiany austenitu prawdopodobieństwo zarodkowania jest wysoce uzależnione od lokalizacji jego miejsc. Największe prawdopodobieństwo zarodkowania istnieje w narożach, nieco mniejsze na krawędziach i wielokrotnie mniejsze na powierzchniach granicznych ziaren. Prawdopodobieństwo zarodkowania w narożach i na krawędziach również nie jest takie samo i zależy od kątów między ścianami, które tworzą daną krawędź lub naroże.

Istotna różnica prawdopodobieństwa zarodkowania na różnych elementach ziarna pociąga za sobą konieczność zastosowania innego algorytmu zarodkowania i odpowiedniej modyfikacji automatu komórkowego (rys. 4.9). Inaczej niż w przypadku

rekrytalizacji, kiedy prawdopodobieństwo powstania zarodka na dowolnym elemencie ziarna było jednakowe, teraz uwzględnienie w modelu jednego stanu na granicy ziarna q_2 nie wystarczy do prawidłowego określenia miejsc zarodkowania.

Należy więc jeszcze raz wrócić do algorytmu zarodkowania podczas rekrytalizacji, żeby lepiej uwidocznić wprowadzone zmiany. Zgodnie z założeniami można określić prawdopodobieństwo zarodkowania p_g , które jest funkcją parametrów procesowych (temperatury, gęstości dyslokacji, przechłodzenia i innych) oraz wielkością określającą liczbę zarodków, które powstają na jednostce powierzchni (długości krawędzi lub w narożu) w jednostce czasu. W takim przypadku zarodkowanie można modelować na dwa sposoby: lokalnie i globalnie. W większości modeli automatów komórkowych realizuje się metodę lokalną, która polega na skanowaniu wszystkich możliwych miejsc zarodkowania (każdej komórki), losowaniu liczby p_i z rozkładem równomiernym w zakresie od 0 do 1 i porównaniu tej liczby z prawdopodobieństwem zarodkowania. Jeżeli $p_i \leq p_g$, w danej komórce następuje zarodkowanie, jeżeli nie – pozostaje ona w poprzednim stanie. Jak już wspomniano w części pierwszej, zastosowanie tego algorytmu jest nieracjonalne z uwagi na wymagane nakłady obliczeniowe i we frontalnych automatach komórkowych wykorzystuje się inny, szybszy.

Liczbę zarodków N_n powstających w jednostce czasu można obliczyć na podstawie prawdopodobieństwa zarodkowania p_g i powierzchni zarodkowania S : $N_n = p_g S$ lub na podstawie średniej powierzchni ściany komórki S_{av} i liczby komórek na granicach ziaren n_b : $N_n = p_g S_{av} n_b$ jak w rozpatrywanym modelu. Faktycznie zastosowanie pierwszego podejścia da liczbę ziaren o rozkładzie normalnym z wartością oczekiwaną równą wartości obliczonej N_n . Wówczas jeżeli zamiast dokładnej liczby N_n zastosować rozkład normalny, wyniki w obydwu podejściach będą identyczne. Jednak drugie podejście nie wymaga na każdym kroku obliczeniowym analizy stanu wszystkich komórek, losowania liczb pseudolosowych i porównywania ich z prawdopodobieństwem zarodkowania. Wystarczy jedynie obliczyć liczbę zarodków i losowo przypisać je do miejsc możliwego zarodkowania.

Warunki zarodkowania podczas przemiany wymagają zmiany algorytmu. Możliwe są dwa rozwiązania. Można zastosować sposób, którego używa się często przy symulacji zarodkowania w modelach niezwiązanych z automatami komórkowymi. Wówczas warunki zarodkowania zależą od miejsca zarodkowania (powierzchnie graniczne, krawędzie, naroża) i nie uwzględnia się kątów wewnętrznych między ścianami ziaren. W drugim rozwiązaniu można uwzględnić nie tylko miejsce zarodkowania, ale i kąty wewnętrzne.

W obydwu wariantach najpierw można uwzględnić podział według miejsc zarodkowania, wprowadzając dodatkowe stany i wyznaczając, które komórki znajdują się na krawędziach i w narożach. Do zróżnicowania miejsc zarodkowania lub stanów mogą być wykorzystywane komórki w stanie q_2 (na granicy ziarna), jeżeli obliczenia są kontynuacją poprzednich symulacji, bądź też, jeżeli początkowa struktura jest wczytywana z pliku, a więc nie zawiera informacji o stanie komórek, dodatkowe stany będą wyznaczone wraz z wyznaczeniem stanów q_2 i q_3 . Nazwijmy dwa dodatkowe stany: q_6 – „komórka na krawędzi ziarna” i q_7 – „komórka w narożu ziarna”.

Wówczas powstają trzy rodzaje komórek, w których mogą się pojawić zarodki, i dla każdego z nich jest zadawane inne prawdopodobieństwo. Wystarczy rozszerzyć algorytm zarodkowania, stosowany poprzednio względem jednego rodzaju komórek, uwzględniając w nim pozostałe dwa. W każdym kroku na podstawie prawdopodobieństwa i liczby odpowiednich komórek jest wyznaczana liczba zarodków dla każdego rodzaju komórek i spośród wszystkich komórek losowo wybierane są takie, w których zajdzie zarodkowanie. Prawdopodobieństwa te różnią się nie tylko wartością, ale również i jednostkami – są to prawdopodobieństwa względne, odniesione do jednostki czasu oraz do powierzchni styku dwóch ziaren (p_f), do długości krawędzi (p_e) lub liczby naroży (p_n). Dlatego też prawdopodobieństwa te są w zasadzie nieporównywalne bezpośrednio, ale można oszacować ich stosunek w odniesieniu do całej objętości modelowej z uwzględnieniem takich parametrów mikrostruktury, jak stosunek powierzchni ziaren, długości ich krawędzi i liczby naroży. Parametry te zależą głównie od średniej wielkości ziaren. Zastosowanie wspomnianego algorytmu sprawia, że żmudne obliczanie prawdopodobieństw dla każdej komórki w każdym kroku obliczeniowym nie jest już konieczne. Wariant ten jest stosowany jako podstawowy. Liczba zarodków N_n będzie wynosić:

$$N_n = p_f n_f S_{av} + p_e n_e l_{av} + p_n n_n \quad (9.9)$$

gdzie:

- p_f, p_e, p_n – prawdopodobieństwo zarodkowania, odpowiednio, na powierzchni, na krawędzi i w narożu ziarna.
- n_f, n_e, n_n – liczba komórek, odpowiednio, na powierzchniach, krawędziach i w narożach ziaren;
- S_{av}, l_{av} – średnia powierzchnia ścian i długość krawędzi komórki.

Opracowano również i drugi wariant zarodkowania, który uwzględnia kąt. Wskaźnikiem związanym z kątem bryłowym w każdym punkcie na granicy ziarna jest w tym wypadku liczba komórek należących do tego samego ziarna w rozszerzonym otoczeniu. Zwiększenie promienia otoczenia istotnie wpływa na długość obliczeń, dlatego ograniczono otoczenie, przyjmując jego promień o długości dwóch komórek lub średnicy wynoszącej pięć komórek (kula wpisana w sześćcian o długości krawędzi wynoszącej 5 komórek). Takie otoczenie składa się z 81 komórek (wliczając w to komórkę rozpatrywaną). 51 komórek w tym otoczeniu należących do tego samego ziarna odpowiada płaskiej powierzchni granicy, 32 komórki – kątowi prostemu na krawędzi, a 20 – kątowi prostemu w narożu, 35 komórek odpowiada trzem równym kątom płaskim (120°) na krawędzi, a 23 – czterem równym kątom bryłowym w narożu.

W celu obliczenia prawdopodobieństwa zarodkowania w komórce p_i wprowadza się zależność aproksymującą, dla której punktami odniesienia są liczby komórek w rozszerzonym otoczeniu należących do tego samego ziarna równe 23, 35 i 51, którym są przypisane wartości prawdopodobieństwa $p_n, p_e l_{av}$ i $p_f S_{av}$.

W algorytmie tym można pozostawić dodatkowy podział na komórki w stanie q_6 i q_7 , ale nie jest to konieczne.

Jeżeli poprzednio podstawą do obliczeń była liczba komórek i ich geometryczne parametry, to teraz liczbę zarodków N_n można obliczyć, sumując prawdopodobieństwa p_i zarodkowania w każdej komórce na granicach ziaren według wzoru:

$$N_n = \sum_{i=1}^{n_g} p_i \quad (9.10)$$

Jednak ten algorytm umożliwia tylko badanie zarodkowania w każdej komórce. W zmiennych warunkach zarodkowania stałe pozostaje otoczenie komórek, które jest badane jednokrotnie.

Do wyznaczenia prędkości rozrostu zastosowano wzór (9.2).

9.4. Wyniki modelowania wstępnego

Symulację rozwoju mikrostruktury podczas przemian fazowych przeprowadzono na tej samej strukturze początkowej, która została utworzona w przestrzeni komórkowej o $260 \times 260 \times 260$ komórkach i wymiarach $500 \times 500 \times 500 \mu\text{m}^3$. Struktura początkowa zawierała 100 ziaren i została utworzona z periodycznymi warunkami brzegowymi przy swobodnym rozroście ziaren z losową lokalizacją zarodków (por. rozdz. 6).

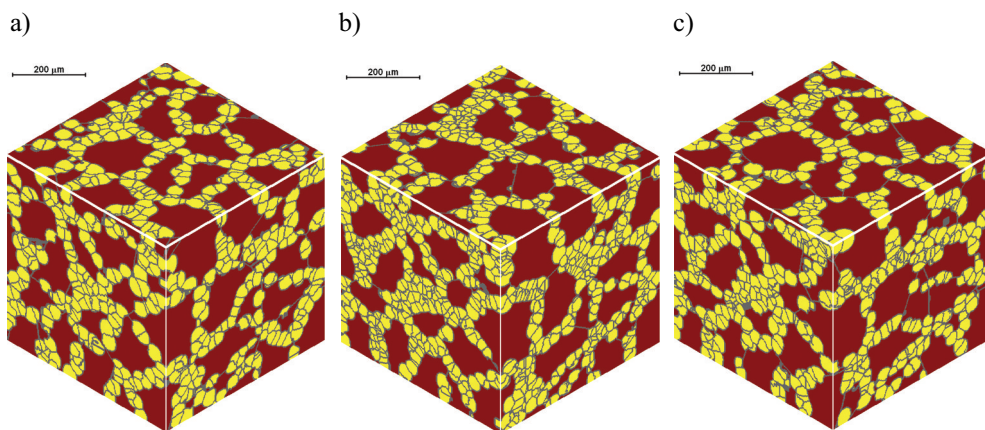
Stal o różnej zawartości węgla była chłodzona od temperatury 900°C do 730°C w ciągu 10 sekund, następnie wytrzymywana do pełnego wydzielenia ferrytu (lub cementytu) i dalej w ciągu 100 sekund temperaturę obniżano o 100°C i trzymana w tej temperaturze do zajścia pełnej przemiany austenitu w perlit.

Ponieważ model nie został zweryfikowany na podstawie danych doświadczalnych, wyniki jego zastosowania można rozpatrywać tylko jako poglądowe – pozwalają one zaobserwować tylko niektóre jego aspekty.

Pierwsza seria wyników miała dostarczyć odpowiedzi na pytanie o prawdopodobieństwo zarodkowania w różnych miejscach ziarna. Na to pytanie nie ma jednoznacznej odpowiedzi – różne źródła wskazują na obfite zarodkowanie na powierzchniach granicznych ziaren, potem na krawędziach i ostatecznie w narożach, inne z kolei podają odwrotną kolejność. Jeżeli prawdopodobieństwo zarodkowania w narożach jest wyższe, to tych miejsc jest zdecydowanie mniej niż na powierzchniach granicznych ziaren. W wyniku tego liczba zarodków na powierzchniach granicznych może być znacznie większa.

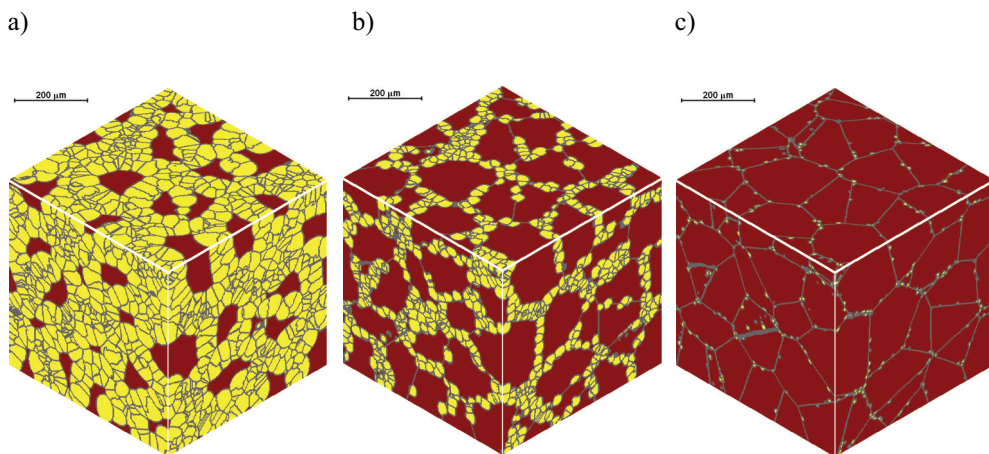
Zbadano trzy skrajne przypadki zarodkowania: na powierzchni granicy, na krawędzi i w narożach. Warunki dobrano tak, że w każdym przypadku powstawała zbliżona liczba zarodków (około 5000). Symulacji poddano stal o zawartości węgla 0,3%. Wyniki przedstawiono na rysunku 9.3. Na wszystkich kolejnych rysunkach czerwień żelazowa odpowiada austenitowi, kolor kanarkowy – ferrytowi, a kobaltowy – perlitowi. Granice zaznaczono kolorem szarym. Mało istotne różnice w mikrostrukturze pozwalają

zrozumieć, dlaczego problem zarodkowania jest tak skomplikowany. Przeprowadzenie dokładnych badań w warunkach obecności wszystkich trzech wariantów zarodkowania, z uwzględnieniem faktu, że granice między ziarnami ferrytu na zdjęciach mikroskopowych są często słabo widoczne, jest bardzo skomplikowanym zadaniem i dlatego pytanie o przewagę tego czy innego zarodkowania pozostaje otwarte. Do dalszych symulacji wybrano zarodkowanie zrównoważone, w którym liczba zarodków na powierzchni, krawędziach i w narożach jest prawie taka sama. Podobna zasada została zastosowana również w przypadku zarodkowania kolonii perlitu.

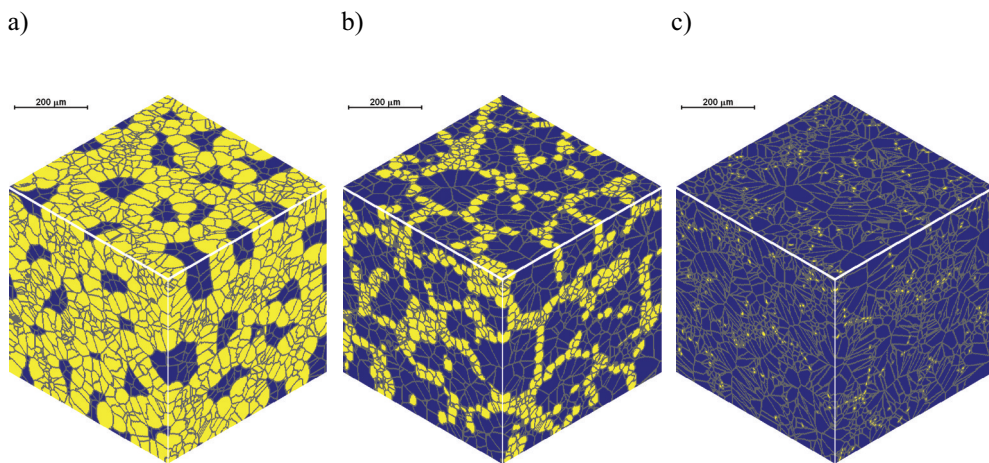


Rys. 9.3. Wpływ miejsca zarodkowania na mikrostrukturę: zarodkowanie ferrytu na powierzchni (a), na krawędziach (b) i w narożach (c) ziaren austenitu

Kolejna seria symulacji przemiany dotyczy wpływu zawartości węgla w stali na przemianę fazową. Wybrano trzy zawartości węgla: 0,1%, 0,4% i 0,73%. Wyniki przemiany austenit-ferryt zachodzącej w pierwszym etapie chłodzenia przedstawiono na rysunku 9.4, a przemiany austenit-perlit w drugim etapie chłodzenia – na rysunku 9.5. W stali niskowęglowej o zawartości 0,1% C prawie 87% austenitu przemienia się w ferryt, natomiast w stali o składzie zbliżonym do eutektoidalnego około 95% austenitu pozostaje w swojej postaci. W czasie rozrostu ziaren ferrytu granice ziaren austenitu prawie nie stanowią bariery i dlatego prędkość rozrostu w większości przypadków słabo zależy od tego, w które ziarno wstąpi ferryt. Natomiast w modelu dyfuzja węgla przez granice ziaren austenitu jest nieco zahamowana i to może powodować niewielkie różnice w prędkości rozrostu. Rozrostowi towarzyszy ciągle zmniejszanie się siły napędowej, której wartość zależy od stężenia węgla w austenicie. Po osiągnięciu stężenia eutektoidalnego siła napędowa spada do zera. W programie końcem przemiany jest obniżenie prędkości rozrostu do wartości 2% prędkości maksymalnej w aktualnej temperaturze. Wyniki przemiany na tym etapie można uznać za zadowalające, odpowiadające jakościowo warunkom rzeczywistym.



Rys. 9.4. Powstanie ferrytu podczas pierwszego etapu chłodzenia w stali o zawartości węgla 0,1% (a), 0,4% (b) i 0,73% (c)



Rys. 9.5. Mikrostruktura po przemianie perlitycznej stali o zawartości węgla 0,1% (a), 0,4% (b) i 0,73% (c)

Wyniki przemiany austenit-perlit w drugim etapie chłodzenia również jakościowo odpowiadają zjawisku rzeczywistemu. Ferryt powstały na granicach ziaren austenitu staje się naturalną barierą i przy małych zawartościach węgla ogranicza rozrost kolonii perlitu tylko do wewnątrz jednego ziarna austenitu. W miarę zwiększania się zawartości węgla coraz więcej granic między ziarnami austenitu się otwiera i wówczas niektóre kolonie perlitu z sukcesem wrastają w kilka sąsiednich ziaren. Gdy skład stali upodabnia się do stali eutektoidalnej, granice ziaren austenitu w strukturze perlitycznej stają

się bardzo słabo zaznaczone. Dalszy wzrost zawartości węgla na granicach ziaren austenitu (na rysunkach tego nie pokazano) powoduje powstanie drobnej siatki cementytu, która również stanowi barierę utrudniającą wrastanie perlitu w kilka ziaren austenitu, następuje więc ono tylko wewnątrz jednego ziarna. W modelu nie rozpatrywano powstania cementytu, jednak w odniesieniu do stali z zawartością ponad 0,8% C założono rozrost perlitu tylko do jednego ziarna austenitu, do tego, na którego granicy powstaje zarodek, czyli do tego, do którego należała komórka przed zarodkowaniem.

10. Rozdrobnienie mikrostruktury

Modelowanie rozwoju mikrostruktury w procesach przeróbki plastycznej, a szczególnie rozdrobnienia mikrostruktury w procesach z dużą akumulacją odkształceń, to stosunkowo nowy kierunek w inżynierii materiałowej. W podrozdziale 10.1 zostały opisane właściwości materiałów drobnoziarnistych, warunki ich uzyskiwania i metody modelowania. W podrozdziale 10.2 przedstawiono z kolei schemat wieloskalowego modelu do badań i symulacji materiałów drobnoziarnistych oraz omówiono metody zastosowane w tym modelu. W podrozdziale 10.3 opisano trzy oparte na automatach komórkowych modele do symulacji rozdrobnienia ziaren oraz sposób wyznaczenia parametrów tych modeli. W podrozdziale 10.4 przedstawiono z kolei wyniki symulacji rozwoju struktury drobnoziarnistej w procesie walcowania pakietowego i w technologii MAXStrain®.

10.1. Materiały drobnoziarniste

Dla uwzględnienia złożonych zjawisk zachodzących w stali podczas przeróbki plastycznej i ciepło-plastycznej konieczne jest stworzenie modelu, który pozwoli na śledzenie rozwoju mikrostruktury ze szczególnym uwzględnieniem struktury dyslokacyjnej. Zastosowanie istniejących związków konstytutywnych, wykorzystujących jako zmienną wewnętrzną jedynie gęstość dyslokacji, jest niewystarczające do dokonania kompletnego opisu zmian zachodzących w mikrostrukturze. Skonstruowany w należyty sposób model powinien uwzględniać wszystkie możliwe etapy rozwoju struktur dyslokacyjnych, poczynając od generowania i anihilacji dyslokacji, przez samoorganizację ich w podstruktury, powstawanie granic niskokątowych między podziarnami, po przekształcenie granic wąskokątowych w szerokokątowe. Ponadto właściwie działający model powinien uwzględniać zmiany kształtu ziaren i podziaren oraz dużą nierównomierność naprężeń i odkształceń, które powstają jako efekt silnej anizotropii odkształcenia poszczególnych ziaren.

Rozwiązanie wszystkich powyższych problemów przy użyciu jednego prostego modelu analitycznego nie jest możliwe. Można to natomiast osiągnąć dzięki połączeniu kilku znanych metod modelowania numerycznego. Podstawowymi takimi metodami, które należy tutaj wymienić, są metoda automatów komórkowych oraz metoda elementów skończonych. Do przewidywania przestrzennego rozwoju mikrostruktury najczęściej stosowane są modele oparte na metodzie automatów komórkowych lub na

metodzie Monte Carlo. Metody te są pod wieloma względami do siebie podobne, chociaż występują też i zasadnicze między nimi różnice.

Jednak automaty komórkowe nie pozwalają na dokładne określenie własności uzyskanych struktur. Uzyskana przy ich zastosowaniu struktura, która z założenia jest prezentacją dyskretną, powinna być przekształcona do postaci ciągłej, która będzie przystosowana do obliczeń numerycznych z zastosowaniem metody elementów skończonych. W modelu bazującym na elementach skończonych powinny wykorzystać teorię dyslokacji oraz plastyczności mono- i polikryształów, w której głównym mechanizmem odkształcenia jest poślizg z uwzględnieniem typu struktury krystalicznej oraz podstawowych (łatwych) płaszczyzn i kierunków poślizgu. W tym celu przy zastosowaniu trójwymiarowych automatów komórkowych oraz metody elementów skończonych z uwzględnieniem anizotropii materiałów polikrystalicznych, opracowano teoretyczne podstawy modelu rozwoju mikrostruktury i własności mechanicznych materiałów silnie rozdrobnionych oraz stworzono narzędzia do symulacji zmian zachodzących w mikrostrukturze i parametrach mechanicznych. Pozwoliło to na modelowanie procesów z bardzo dużą akumulacją odkształcania, które prowadzone są w celu uzyskiwania struktur ultradrobnoziarnistych w stali oraz nanonstruktur.

Zastosowanie automatów komórkowych do modelowania zjawisk zachodzących podczas odkształcenia w niższych temperaturach nie jest zbyt szeroko rozpowszechnione. Krokiem poczynionym w tym kierunku jest zastosowanie automatów komórkowych do modelowania propagacji pęknięć (Makarov *et al.* 2001) oraz rozwoju mikropasm i pasm ścinania. Madej *et al.* (2004) zastosowali klasyczne automaty, natomiast Lasko *et al.* (2004, 2006) opracowali nową metodę elementów relaksacyjnych (*Relaxation Element Method*), która w połączeniu z modelowaniem automatami komórkowymi pozwala na dokładne przewidywanie naprężenia uplastyczniającego. Głównym mechanizmem odkształcenia w tym modelu jest poślizg i lokalizacja odkształceń. Metoda owa opracowana jest w wersji dwuwymiarowej z zastosowaniem heksagonalnych komórek. Główną jej zaletą są małe nakłady obliczeniowe i, co za tym idzie, krótki czas obliczeń. Jednak poza dość dokładną symulacją efektu powstania mikropasm ścinania i naprężenia uplastyczniającego podejście owo nie pozwala ani na przewidywanie zmian mikrostruktury, ani na symulowanie struktur dyslokacyjnych, które odgrywają szczególnie istotną rolę przy dużych odkształceniach i powstawaniu struktury drobnoziarnistej. W takich automatach mamy do czynienia z problemami związanymi z kształtem komórki i wybranego otoczenia. Poza tym stosowanie dwuwymiarowych automatów komórkowych jest bardzo ograniczone (Svyetlichnyy 2007) – można je stosować wyłącznie w szczególnych przypadkach odkształcenia cienkich folii w niskich temperaturach.

Dyslokacje, gęstość dyslokacji i struktury dyslokacyjne są istotnymi parametrami uwzględnianymi podczas modelowania rozwoju mikrostruktury za pomocą automatów komórkowych. Jednak w większości dotyczących tego tematu prac zakłada się zmianę gęstości dyslokacji jako z góry ustaloną procedurę, opartą na teorii dyslokacji Taylora lub Orowana. Wykorzystuje się w niej wartości gęstości dyslokacji, które służą do obliczenia naprężenia uplastyczniającego lub inicjacji rekrytalizacji w wyższych temperaturach.

Obecnie bezpośrednie modelowanie za pomocą automatów komórkowych zmian wartości naprężenia uplastyczniającego jest ograniczone procesem odkształcenia z rekrytalizacją dynamiczną (Svyetlichnyy i Matachowski 2005). Natomiast powszechnie stosowane podejście do obliczania naprężenia uplastyczniającego na podstawie średniej wartości gęstości dyslokacji w całej objętości automatów komórkowych jest wielce dyskusyjne, o czym świadczą liczne, podejmowane na całym świecie próby mające na celu uzyskanie realistycznych krzywych płynięcia w procesie odkształcenia z rekrytalizacją dynamiczną (Ding *et al.* 2001, Qian i Guo 2004, Goetz 2005, por. też rozdz. 8). Niewiele jest prac poświęconych rozwojowi struktur dyslokacyjnych. Można tu wymienić studia Brossa i Stecka (1998) oraz Brossa *et al.* (2003). W modelu samouporządkowania struktury dyslokacyjnej rozpatrywano tylko jedną płaszczyznę poślizgu, co pozwoliło zastosować dwuwymiarowe automaty komórkowe, jednak w rzeczywistym materiale nie można pominąć wpływu innych systemów poślizgu. Niemniej we wspomnianych pracach jasno pokazano wpływ warunków odkształcenia na rozwój struktur dyslokacyjnych, kiedy zmiana tych warunków powoduje powstanie różnych struktur tego rodzaju.

Najpoważniejszym problemem związanym ze stosowaniem automatów komórkowych nadal pozostaje modelowanie odkształcenia na poziomie mikrostruktury. Praktycznie we wszystkich znanych pracach poświęconych rozwojowi mikrostruktury za pomocą automatów komórkowych odkształcenie jest traktowane jako parametr zewnętrzny, który brany jest pod uwagę w odpowiednich obliczeniach, jednak nie powoduje zmian geometrii przestrzeni, co można uznać za nonsens (por. rozdz. 2). Jeżeli w modelowaniu rekrytalizacji statycznej można wykorzystać strukturę odkształconą lub uzyskaną z badań mikroskopowych (Davies i Hong 1999, Kugler i Turk 2006), to w modelowaniu rekrytalizacji dynamicznej i rozdrobnienia struktury żadnych zmian nie przewidywano.

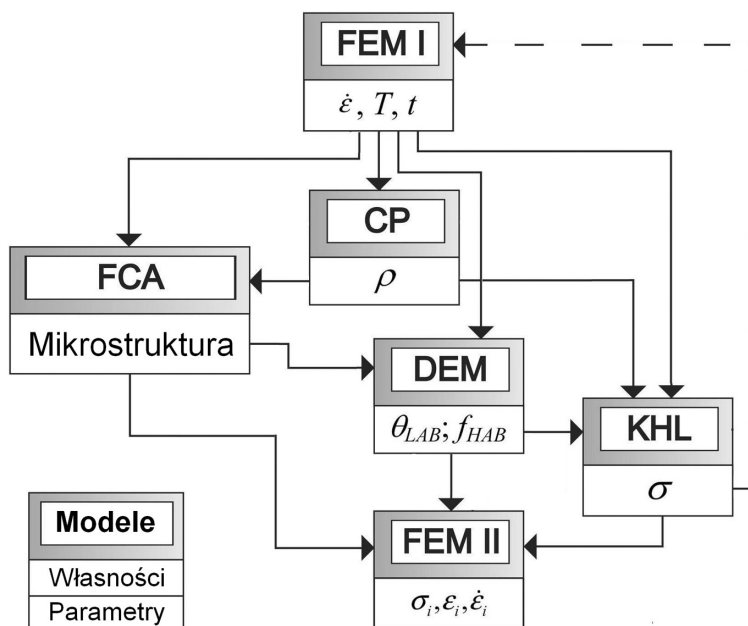
Z drugiej strony modelowanie odkształcenia za pomocą elementów skończonych jest zabiegiem bezdyskusyjnym i stosuje się je zarówno w skali makro, jak i mikro. Podczas modelowania na poziomie mikrostruktury stosowane są zasady teorii plastyczności krysztalów. Badania modelu plastyczności krysztalu z uwzględnieniem zależności umocnienia na podstawie informacji o dyslokacjach są obecnie prowadzone na całym świecie (na przykład Yamaki *et al.* 2006, Oishi *et al.* 2006). W pracy Yamakiego *et al.* (2006) zaproponowano zastosowanie równania dyfuzyjno-reakcyjnego do trzech etapów umocnienia materiału, a mianowicie do obszaru łatwego poślizgu o małej gęstości dyslokacji, do liniowego umocnienia w obszarze o wysokim stężeniu dyslokacji i wysokim ich wspólnym oddziaływaniu w pierwotnych i wtórnych płaszczyznach poślizgu, w tym przy samoorganizacji w struktury komórkowe, oraz do dynamicznego zdrowienia, gdy struktury komórkowe przekształcają się w subziarna i ziarna. Do rozwiązania równania dyfuzyjno-reakcyjnego zastosowano metodę różnic skończonych. Nieco inne podejście zaproponowano w pracy Estrina *et al.* (2006). Rozwój struktury dyslokacyjnej i jej przekształcenie w strukturę podziaren i ziaren związane w niej ze zmianą kąta dezorientacji, która z kolei uzależniona jest od rozwoju gęstości dyslokacji. Natomiast w pracy Miodownika *et al.* (1999) zaproponowano formalne podejście

z zastosowaniem metody Monte Carlo do uzyskania odpowiedniego rozkładu kąta dezorientacji granic ziaren (por. również rozdz. 6).

Przedstawione w tym rozdziale doświadczenia przeprowadzono głównie w ramach finansowanego przez Ministerstwo Nauki i Szkolnictwa Wyższego projektu badawczego własnego nr N508 3812 33 *Opracowanie modelu oraz symulacja komputerowa rozwoju struktury w procesach uzyskiwania materiałów drobnoziarnistych*. Wiele jego szczegółów przedstawiono w sprawozdaniu z tych badań (Svyetlichnyy *et al.* 2011) i związanych z nimi publikacjach (Majta i Doniec 2010, Majta *et al.* 2009, Muszka *et al.* 2009, Svyetlichnyy 2008a–2008d, 2009a–2009d, 2010a–2010b, Svyetlichnyy i Majta 2009, Svyetlichnyy i Milenin 2009, Svyetlichnyy *et al.* 2008, 2010, Wielgus *et al.* 2009).

10.2. Model rozwoju mikrostruktury i zmian własności mechanicznych materiałów silnie rozdrobnionych

Jedną z proponowanych modyfikacji schematu blokowego budowy i działania zintegrowanego modelu rozwoju mikrostruktury i zmian własności mechanicznych materiałów silnie rozdrobnionych, opracowaną z wykorzystaniem przyjętych w projekcie metod i technik, przedstawiono na rysunku 10.1 (Svyetlichnyy *et al.* 2011).



Rys. 10.1. Schemat blokowy procesu obliczeniowego z zastosowaniem proponowanych modeli oraz metod obliczeniowych (Svyetlichnyy *et al.* 2011)

Na rysunku 10.1 zastosowano następujące skróty i oznaczenia: CP – plastyczność kryształów, DEM – metoda elementów dyskretnych, FCA – frontalne automaty komórkowe, FEM I (MES I) – metoda elementów skończonych w zastosowaniu do skali makro, FEM II (MES II) – metoda elementów skończonych w zastosowaniu do skali mikro, KHL – zmodyfikowany model naprężenia uplastyczniającego (Khan–Huan–Liang), $\dot{\epsilon}$ – tensor prędkości odkształcenia, σ – tensor naprężeń normalnych, σ_p – naprężenie uplastyczniające, T – temperatura, t – czas, D – średnia wielkość ziarna, f_{HAB} – ułamek objętości ziaren z granicami wysokokątowymi, θ – kąt dezorientacji granic ziaren (niskokątowych), ρ – gęstość dyslokacji.

10.2.1. Modele oparte na metodzie elementów skończonych: MES I i MES II

Jako model MES I do modelowania w skali makro można wykorzystać dowolny program, zarówno komercyjny, jak i własny. Modelowanie powinno być prowadzone z zastosowaniem zarówno konwencjonalnego, jak i nowego modelu naprężenia uplastyczniającego (Svyetlichnyy *et al.* 2011). Na początku należy się posłużyć prostym modelem konwencjonalnym i obliczyć wstępne parametry odkształcenia. Później, po uzyskaniu danych MES II, obliczenia powinny być powtórzone z uwzględnieniem ustalonych zależności i sprecyzowanych parametrów. Wybór programu MES I może zależeć od modelowanego procesu.

Do modelowania płynięcia metali w skali mikro opracowano model MES II. Pozwala on uwzględnić nierównomierność odkształcenia na poziomie ziaren, w szczególności wewnątrz nich.

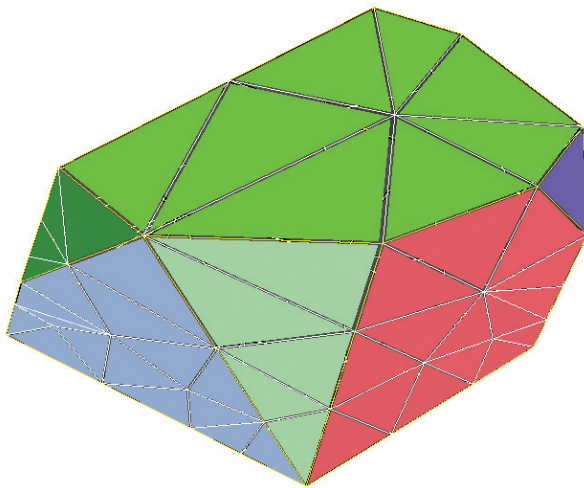
10.2.2. Automaty komórkowe do modelowania powstania nowych granic ziaren

W opracowanym schemacie automaty komórkowe odgrywają podwójną rolę. Po pierwsze, służą do stworzenia początkowej mikrostruktury (por. rozdz. 6), a po drugie – są odpowiedzialne za wszystkie zmiany mikrostruktury. Omawiany model składa się z dwóch części, symulujących odkształcenie i rozwój mikrostruktury.

Część mikrostrukturalna CA oblicza rozdrobnienie ziarna i rozwój struktur dyslokacyjnych na podstawie informacji z MES II o rozwoju tych struktur.

Ze względu na wykorzystanie frontalnych automatów komórkowych (co pozwala istotnie skrócić czas obliczeń) udało się zamodelować struktury trójwymiarowe. Ponieważ większość funkcji, które są często wykonywane przez CA (na przykład obliczenie naprężenia uplastyczniającego), w przyjętym schemacie wykonywana jest za pomocą innych modeli, automaty komórkowe wykorzystuje się głównie do symulacji procesów mikrostrukturalnych. MES II wykorzystywano do wyznaczenia własności materiału w skali mikro. W celu uwzględnienia wpływu mikrostruktury na własności materiału w CA tworzona jest jego reprezentatywna mała objętość. Pozwala to na uwzględnienie anizotropii ziaren i wpływu wąsko- i szerokokątowych granic na umocnienie struktur dyslokacyjnych.

Przestrzeń przedstawianą w CA jako dyskretyzowaną należy przekształcić w prezentację ciągłą lub dopasowaną do MES. Trójwymiarowy model CA składa się z kilkudziesięciu milionów komórek i dlatego nie może być wprost zastąpiony przez odpowiadającą mu siatkę MES. Opracowano algorytm przekształcania przestrzeni komórek w siatkę MES (por. rozdz. 5). Przykład takiego przekształcenia w odniesieniu do wybranego ziarna pokazano na rysunku 10.2.



Rys. 10.2. Trójwymiarowa siatka MES dla jednego ziarna

10.2.3. Teoria plastyczności kryształów

Badanie rzeczywistego procesu odkształcania wymaga kompleksowej analizy, kiedy wyniki uzyskane w skali mikro za pomocą teorii plastyczności kryształów należy odnieść do skali mezo i makro, tj. plastyczności ośrodków ciągłych. Efekty obliczeń wykorzystujących teorię plastyczności kryształów mogą mieć wpływ na globalny opis procesu odkształcania.

Rozdrobnienie ziaren pod wpływem dużych odkształceń jest skutkiem rozwoju struktur dyslokacyjnych, czyli przekształcania się podziaren i komórek powodującego uzyskanie bardziej stabilnych szerokokątowych granic ziaren podczas następnej obróbki cieplnej. Tak więc kluczowym czynnikiem w analizie drobnoziarnistych materiałów jest modelowanie rozwoju gęstości dyslokacji i ich podstruktury. Na potrzeby oceny ilościowej tych procesów często przyjmowano w analizie jednakowe wymiary komórek dyslokacyjnych. Trzy podstawowe rodzaje dyslokacji kształtują podstrukturę odkształcanych w procesach SPD metali i stopów z wysoką energią błędu ułożenia: nieruchome dyslokacje w ścianach komórek, nieruchome dyslokacje wewnątrz komórek oraz dyslokacje ruchome. Zagadnienia te zostały szczegółowo opisane w publikacji Svyetlichnego *et al.* (2011).

10.2.4. Naprężenie uplastyczniające

Główna idea omawianego rozwiązania polega na wykorzystaniu teorii Taylora dotyczącej rozwoju gęstości dyslokacji w opisie umocnienia w każdym systemie poślizgu. W skali makro naprężenie uplastyczniające jest liczone za pomocą modelu lepko-plastycznego zaproponowanego przez Khana. Dokładnie przejście ze skali mikro do skali makro zostało opisane w książce J. Majty (2008).

10.2.5. Model oparty na metodzie elementów dyskretnych

Główną różnicą między omawianą opcją a poprzednio zaproponowanym schematem (Svyetlichnyy *et al.* 2008) jest zastosowanie metody elementów dyskretnych (*Discrete Element Method* – DEM), która na poziomie mikro może częściowo zastąpić metody MES i CA. W tym schemacie każda komórka dyslokacyjna od chwili jej powstania do momentu pełnego przekształcenia w odrębne drobne ziarno może być zasyumulowana oddzielnym elementem dyskretnym, który będzie występował jako element prawie nieodkształcalny. Wówczas odkształcenie modelowane jest jako przemieszczenie i obrót elementów. W elementach dyskretnych obrót elementów w naturalny sposób prowadzi do zmian w kątach dezorientacji sąsiednich ziaren i wspomaga ewolucję struktury dyslokacyjnej w strukturę podziarnową i drobnoziarnistą.

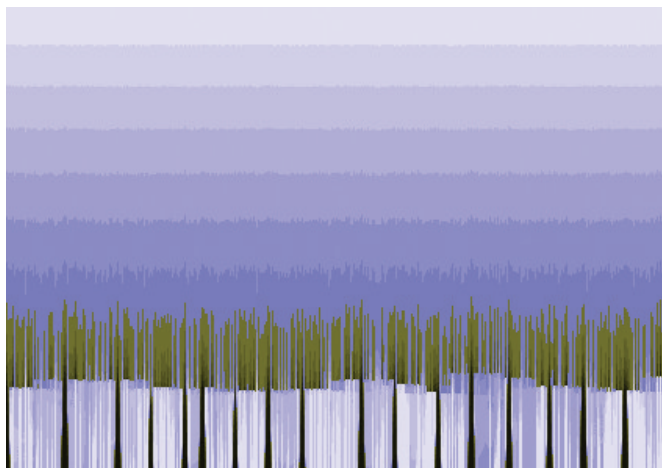
Należy również zauważyć, iż złożoność zastosowanych metod w skali mikro i bardzo duże nakłady obliczeniowe nie pozwalają na zastosowanie wszystkich przyjętych rozwiązań bezpośrednio jako podsystemów obliczeń w skali makro (MES I), nie mogą one także stanowić podstawy modelu reologicznego, niemniej są konstrukcją pozwalającą na poprawne modelowanie. Z tego powodu sprzężenie zwrotne pomiędzy skalą mikro a makro na schemacie (rys. 10.1) uwidoczniło za pomocą linii przerywanej. Realizacja tego połączenia możliwa jest dzięki zastosowaniu metamodeli (na przykład z wykorzystaniem sztucznych sieci neuronowych), niemniej może to prowadzić do jeszcze bardziej rozbudowanego i złożonego schematu obliczeniowego).

10.3. Modele automatów komórkowych do rozwoju struktur dyslokacyjnych

Jak to zostanie pokazane poniżej, krzepnięcie, kształtowanie struktury początkowej, rekrytalizacja i przemiany fazowe, mogą być w omawianym modelu brane pod uwagę tylko w niewielkim stopniu. Dlatego też opracowano trzy modele automatów komórkowych do modelowania rozwoju mikrostruktury drobnoziarnistej. Zrezygnowano z symulacji każdej odrębnej dyslokacji. Każda komórka reprezentuje wówczas małą objętość, która cechuje się określoną gęstością dyslokacji. Liczbę dyslokacji lub jej gęstość traktowano jako jeden z parametrów charakterystycznych komórki.

10.3.1. Model I

W tym wypadku model został opracowany w wersji jedno-, dwu- i trójwymiarowej. Gęstość dyslokacji zmienia się w każdej komórce wraz z odkształceniem, ale nie zmienia się ona równomiernie, lecz według rozkładu normalnego z małą wariancją. Gdy gęstość dyslokacji w pewnych komórkach osiąga wartość krytyczną, stan komórki zmienia się, w efekcie staje się ona miejscem gromadzenia dyslokacji i inicjacji powstawania granic niskokątowych (*Low-Angle Boundaries* – LAB). Komórki przechodzą w stan LAB i wpływają na swoje otoczenie. W czasie odkształcenia rozpatrywanego jako przemieszczanie się dyslokacji nowo powstałe granice niskokątowe stanowią dla takiego przemieszczania się barierę. Wówczas wszystkie dyslokacje poruszają się w kierunkach do lub od ścian komórek. Tak więc gęstość dyslokacji we wszystkich sąsiednich komórkach gwałtownie spada, natomiast w komórkach w stanie LAB – wzrasta. Średnica sąsiedztwa zależy od geometrycznych parametrów komórki (jej rozmiarów) oraz od parametrów odkształcenia. Przykład symulacji rozwoju struktury dyslokacyjnej za pomocą automatu jednowymiarowego przedstawiono na rysunku 10.3. Zmiany zachodzące w jednowymiarowych automatach komórkowych ukazano na nim w układzie dwuwymiarowym we współrzędnych przestrzenno-czasowych. Domyślna oś czasu na rysunku 10.3 skierowana jest w dół.

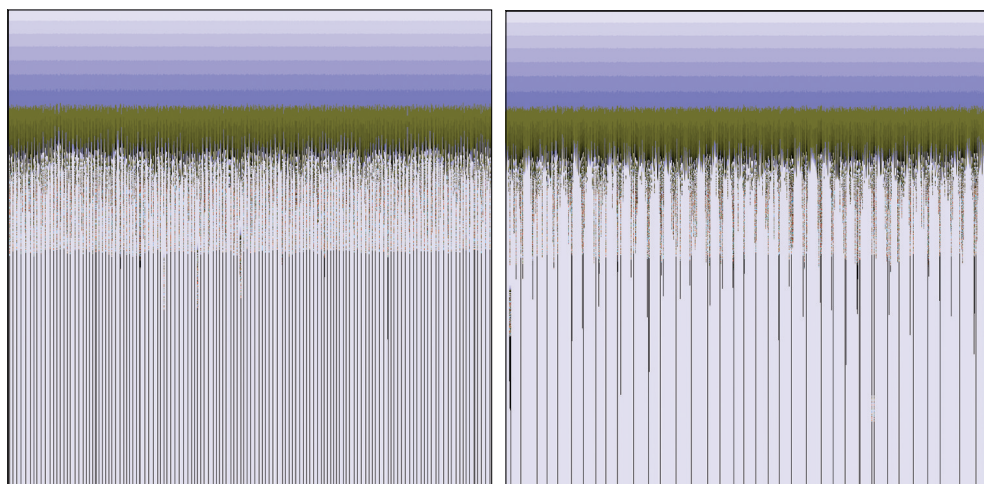


Rys. 10.3. Rozwój struktury dyslokacyjnej zamodelowanej jednowymiarowym automatem komórkowym. Oś czasu jest skierowana w dół

Kolorami oznaczono różne poziomy gęstości dyslokacji, zaczynając od jasnoniebieskiego, przez ciemnoniebieski i dalej zielony, który oznacza osiągnięcie krytycznej gęstości dyslokacji, kończąc na czarnym, odzwierciedlającym gęstość nieruchomych dyslokacji w ściankach komórek dyslokacyjnych, czyli powstanie granic niskokątowych.

towych. Zróznicowanie gęstości dyslokacji na początku procesu odkształcania nie jest znaczące, ale wprowadzenie dodatkowego dodatniego sprzężenia zwrotnego we współdziałaniu sąsiednich komórek w jego trakcie prowadzi do wzrostu zróznicowania i po osiągnięciu odkształcenia krytycznego (krytycznej gęstości dyslokacji) następuje podział komórek na te, które posiadają znaczącą gęstość dyslokacji (w ścianach nowo powstałych komórek dyslokacyjnych), i na pozostałe (wewnątrz komórek dyslokacyjnych), w których gęstość dyslokacji spada prawie do poziomu początkowego.

Na parametry automatów komórkowych wpływa zarówno średnia odległość między ścianami powstających komórek, jak i moment powstania struktury komórkowej lub wartość odkształcenia (gęstość dyslokacji). Na rysunku 10.4 pokazano, w jaki sposób zmiana średnicy sąsiedztwa w automatach komórkowych wpływa na rozmiary komórek dyslokacyjnych. Zadając parametr średnicy, można sterować wymiarami komórek dyslokacyjnych, a w konsekwencji i wymiarem ziaren oraz podziaren.



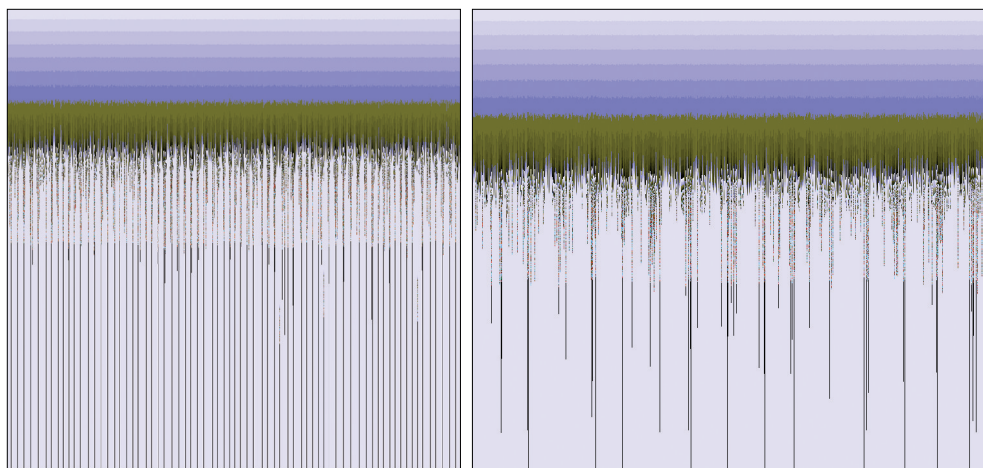
Rys. 10.4. Rozwój struktury dyslokacyjnej o różnej średnicy sąsiedztwa

Na rysunku 10.5 pokazano wynik symulacji, w których oprócz średnicy zmieniano również prędkość gromadzenia dyslokacji oraz czas osiągnięcia jej krytycznej gęstości.

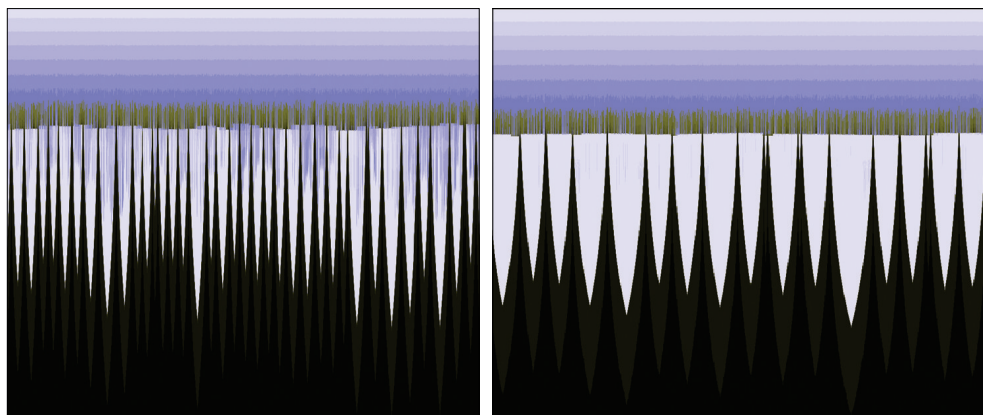
Opracowano również wariant modelu z kształtowaniem ciągłej granicy komórki dyslokacyjnej, czyli wariant półtorawymiarowy, w którym granice przebiegają właśnie wzdłuż jednowymiarowego automatu komórkowego. Warianty symulacji pokazano na rysunku 10.6. Początek symulacji jest podobny do opisanego poprzednio, natomiast zwiększenie liczby komórek w stanie LAB oznacza w pełni ukształtowaną granicę (czarny kolor wypełnia całą przestrzeń).

Dwuwymiarowe automaty komórkowe zostały opracowane w dwóch wersjach. Pierwsza jest oparta na tych samych założeniach co wersja jednowymiarowa, tylko

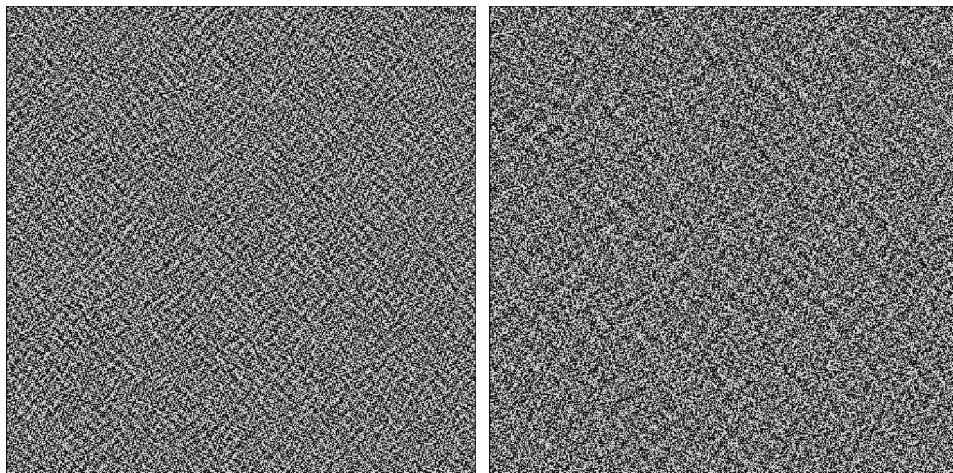
z rozszerzeniem otoczenia do dwóch wymiarów. Sąsiednie komórki rozpatrywano wzdłuż dwóch kierunków poślizgu na płaszczyźnie poślizgu – takie podejście pozwoliło uzyskać strukturę dyslokacyjną pokazaną na rysunkach 10.7 i 10.8. Przedstawiony na rysunku 10.7 rozkład gęstości dyslokacji wskazuje na kształtowanie tekstury, która mogłaby się stać matrycą do powstania struktury komórkowej, jednak jest ona jeszcze stosunkowo słabo rozwinięta i mało spójna. Jednak inaczej, niż to miało miejsce w przypadku automatu jednowymiarowego, po osiągnięciu przez gęstość dyslokacji wartości krytycznej przy zastosowaniu dodatniego sprzężenia zwrotnego nie udało się uzyskać spójności granic, co można zobaczyć na rysunku 10.8.



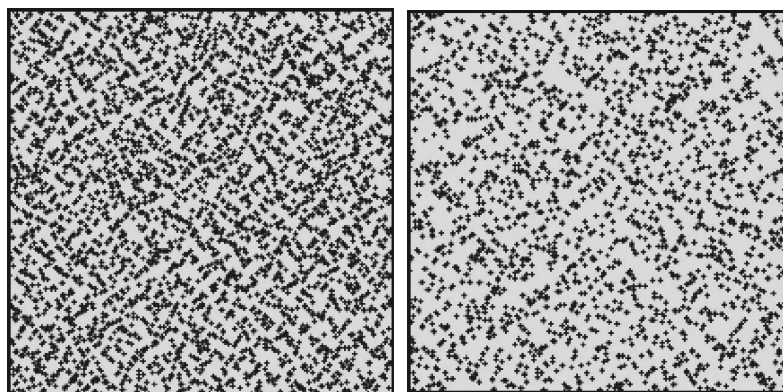
Rys. 10.5. Rozwój struktury dyslokacyjnej o różnej średnicy sąsiedztwa i prędkości gromadzenia dyslokacji



Rys. 10.6. Kształtowanie granic



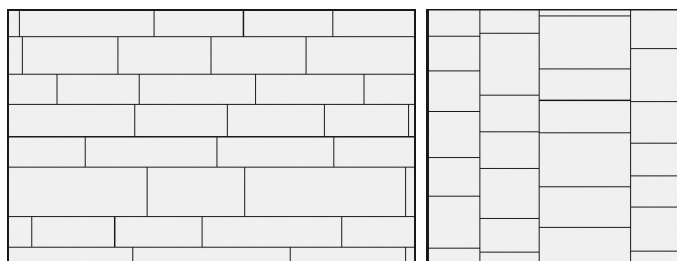
Rys. 10.7. Rozkład gęstości dyslokacji przed kształtowaniem granic



Rys. 10.8. Rozkład gęstości dyslokacji podczas kształtowania granic

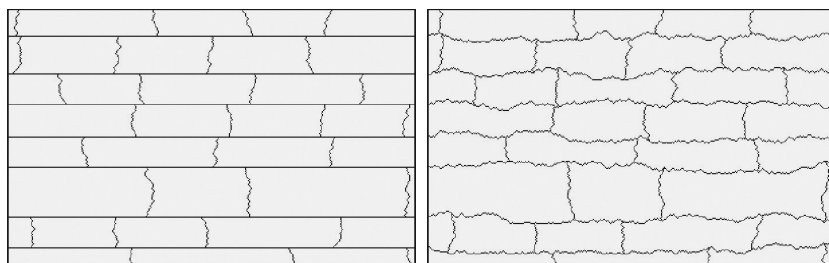
W drugim podejściu obliczenia prowadzono w dwóch kierunkach poślizgu niezależnie. Na początku każdy kierunek poślizgu rozpatrywano jako odrębny, jednowymiarowy automat komórkowy. Algorytm działania automatów komórkowych jest taki sam jak opisany powyżej. Ponieważ prędkość poślizgu jest różna dla różnych jego kierunków, struktury dyslokacyjne rozwijają się w nich w różnym tempie. Komórki z granicami LAB pojawiają się najpierw w jednym kierunku; oznacza to pojawienie się pierwszych geometrycznie niezbędnych granic. W przypadku drugiego kierunku poślizgu jednowymiarowy automat komórkowy rozpada się na wiele jednowymiarowych automatów. Ich liczba jest równa liczbie części, na które została podzielona przestrzeń automatów komórkowych ścianami komórek w pierwszym kierunku. Z wykorzystaniem tych automatów obliczenia powtarzano od początku.

Na rysunku 10.9 pokazano wyniki pierwszych obliczeń – granice ziaren zostały uproszczone i przedstawione jako linie proste. W jednym przypadku najpierw powstają granice poziome, w drugim – pionowe.

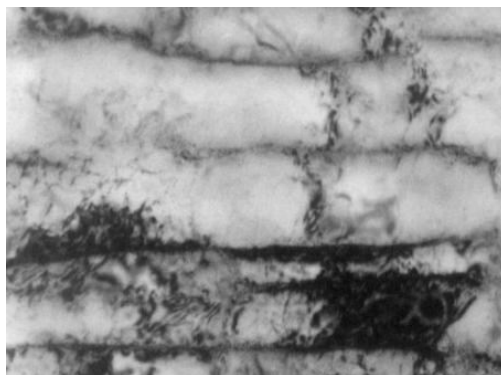


Rys. 10.9. Końcowa struktura komórkowa

Wprowadzenie dodatkowych czynników losowych pozwala zmienić prostoliniowość granic w jednym lub obydwu kierunkach (rys. 10.10). Na rysunku 10.11 przedstawiono dla porównania zdjęcie rzeczywistej struktury. Opisane podejście zostało również rozszerzone do wersji trójwymiarowej.



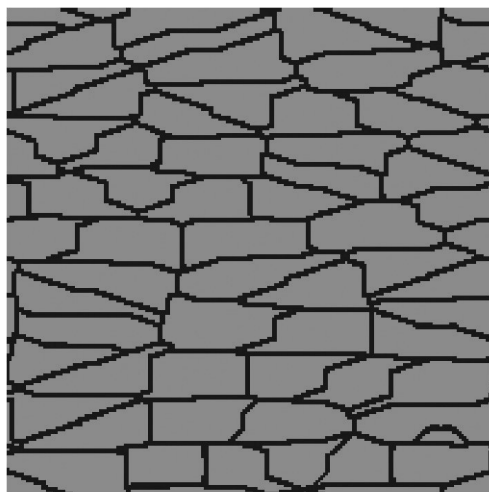
Rys. 10.10. Końcowa struktura komórkowa z pofalowanymi granicami



Rys. 10.11. Rzeczywista struktura stali z mikrododatkami przy odkształceniu 0,6

10.3.2. Model II

Jeżeli w modelu I jako główny mechanizm rozwoju struktury dyslokacyjnej zastosowano proces przemieszczania się dyslokacji do ścian komórek, które mają większą gęstość dyslokacji (przyciąganie), to w modelu II przyjęto proces przeciwny, tzn. odpychanie dyslokacji od stref z mniejszą gęstością, czyli samooczyszczanie tych stref. Wówczas w dwu- i trójwymiarowych wersjach modelu I (pierwsza wersja) doprowadzi to do gromadzenia się dyslokacji w różnych punktach (rys. 10.8), natomiast w modelu II dyslokacje będą oddalać się od wybranych komórek, a ich kumulacja w innych miejscach utworzy granice wąskokątowe (LAB). Przykład struktury uzyskanej dwuwymiarowym modelem II przedstawiono na rysunku 10.12.



Rys. 10.12. Struktura dyslokacyjna uzyskana dwuwymiarowym automatem komórkowym (model II)

10.3.3. Model III

W praktyce działanie modelu II oparte jest na zasadach podobnych do tych stosowanych w modelowaniu innych zjawisk związanych z rozwojem mikrostruktury (rekrytalizacja, przemiany fazowe). Między głównym mechanizmem modelu II a procesami zarodkowania i wzrostu da się zauważyć duże podobieństwo. W tym wypadku wystarczające jest wytrzymanie oraz dobranie odpowiednich warunków zarodkowania i rozrostu. Wynik takiej symulacji przedstawiono na rysunku 10.13. Nie zadano tutaj anizotropii wynikającej z ukierunkowanego płynięcia plastycznego, charakterystycznego dla każdego z analizowanych procesów odkształcania. Efekt uprzywilejowanej orientacji granic ziaren zostanie omówiony w dalszej części książki (podrozdz. 10.4).



Rys. 10.13. Struktura dyslokacyjna uzyskana trójwymiarowym automatem komórkowym bez uwzględnienia anizotropii plastycznego płynięcia (model III)

10.4. Weryfikacja parametrów automatów komórkowych

Model oparty na automatach komórkowych składa się z dwóch podstawowych części symulujących: zarodkowanie nowych granic ziaren w warunkach nierównomiernego rozkładu gęstości dyslokacji i kształtowanie samej struktury, czyli wyostrenie granic i ich transformację w granice szerokokątowe. Rozwój mikrostruktury zależy głównie od prędkości tych dwóch procesów. Dodatkowym czynnikiem jest krytyczna gęstość dyslokacji, od której wartości zależy przejście od pierwszego do drugiego etapu.

Dlatego też na początku powinniśmy przedstawić założenia przyjęte w automatach komórkowych. Główne założenia modelu są następujące:

- początkowa gęstość dyslokacji wszystkich ziaren jest taka sama;
- gęstość dyslokacji wzrasta zgodnie z równaniami teorii dyslokacji, które uwzględniają umocnienie i zdrowienie;
- gęstość dyslokacji w komórkach CA jest rozpatrywana dla każdego systemu poślizgu z uwzględnieniem rzeczywistych warunków odkształcania;
- gdy gęstość dyslokacji w jednej z komórek CA w jednym z systemów poślizgu osiągnie wartość krytyczną r_c , rozpoczyna się w komórce CA kształtowanie granicy komórki dyslokacyjnej;
- rozwój granic ziaren polega na obrocie względem siebie sąsiednich komórek dyslokacyjnych.

Rozpatrzmy najpierw trzy podstawowe wielkości, które mogą zostać sprawdzone. Pierwsza z nich to zależność zarodkowania pierwotnej struktury komórkowej od odkształcenia w różnych materiałach. Odkształcenie takie wynosi od 0,3 do 0,6, a nawet więcej. Wymiary elementarnej komórki dyslokacyjnej i prędkość zmiany kąta dezorientacji ziaren również zostały uwzględnione.

Opisane wyżej parametry dla każdego z materiałów wyznaczano na podstawie badań z wykorzystaniem transmisyjnej mikroskopii elektronowej oraz skaningowej (EBSD), z zastosowaniem metod statystyki matematycznej. Badania te zostały opisane w jednej z publikacji Svyetlichnego *et al.* (2011).

Model gęstości dyslokacji wyznacza naprężenie uplastyczniające na każdym etapie odkształcenia. Współczynniki równań dyslokacyjnych (por. rozdz. 8) są wyznaczone dość dokładnie; ponieważ pozostałe parametry nie mają na nie istotnego wpływu, współczynniki równań dyslokacyjnych mogą być wprost przeniesione do automatów komórkowych.

Główne wnioski z przeprowadzonej analizy są następujące:

- Gdy rozważanym parametrem jest wielkość ziarna, końcowa mikrostruktura otrzymana po rozdrobnieniu może się odznaczać dużą dokładnością.
- Kinetyka rozwoju granic może być różna w zależności od wybranego materiału, natomiast nie zbadano dokładnie wpływu różnych pierwiastków na prędkość zmian kąta dezorientacji. Opierając się na danych doświadczalnych lub zawartych w literaturze przedmiotu, można uzyskać kinetykę zbliżoną do uzyskiwanej za pomocą automatów komórkowych.
- Naprężenie uplastyczniające symulowane za pomocą automatów komórkowych wykazuje pewne błędy, dlatego jego wartość należy traktować jako przybliżoną i doprecyzować z wykorzystaniem MES.

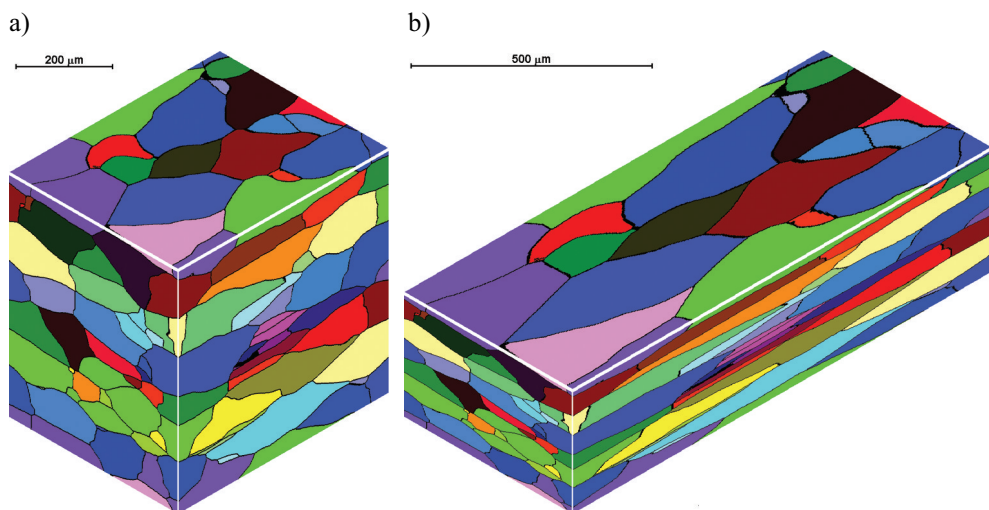
10.5. Zastosowanie opracowanego modelu do przeprowadzenia testowych obliczeń

10.5.1. Walcowanie pakietowe

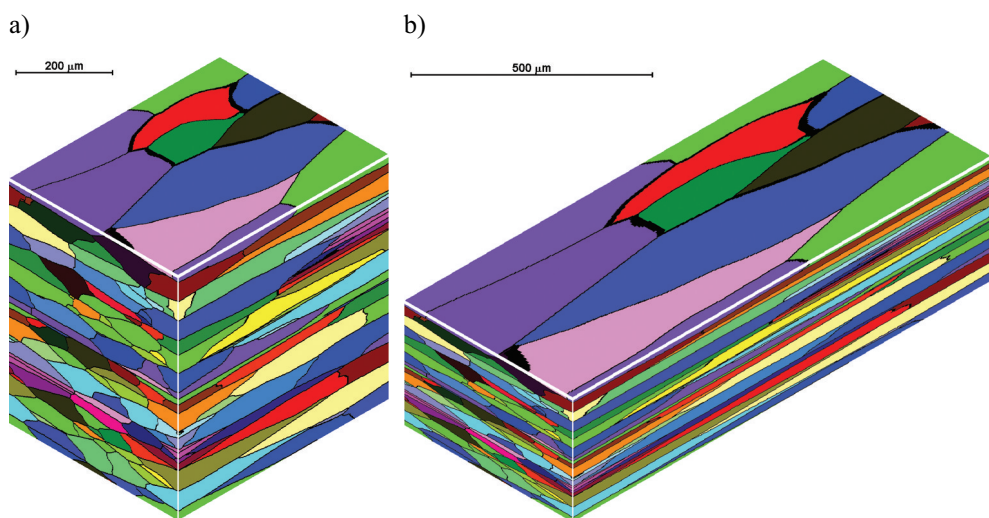
W pierwszej kolejności przeprowadzono symulacje procesu walcowania pakietowego. Punktem startowym przeprowadzonych symulacji był proces modelowania bez uwzględnienia rozdrobnienia ziarna. Symulacje przeprowadzono w dwóch wariantach: w pierwszym po walcowaniu przestrzeń automatów komórkowych rozcinano i składało tak samo jak w rzeczywistym procesie walcowania. W drugim wariantcie wycięto część przestrzeni. Tak więc w wariantcie pierwszym wymiary przestrzeni pozostały bez zmian, natomiast w drugim każdy jej wymiar zmniejszył się dwukrotnie.

Na rysunkach 10.14–10.17 pokazano zmiany makrostruktury po pięciu przepustach. Możliwości modelowania są ograniczone z jednej strony rozdzielczością przestrzeni automatów komórkowych, a z drugiej – dużym zwiększeniem liczby ziaren, co

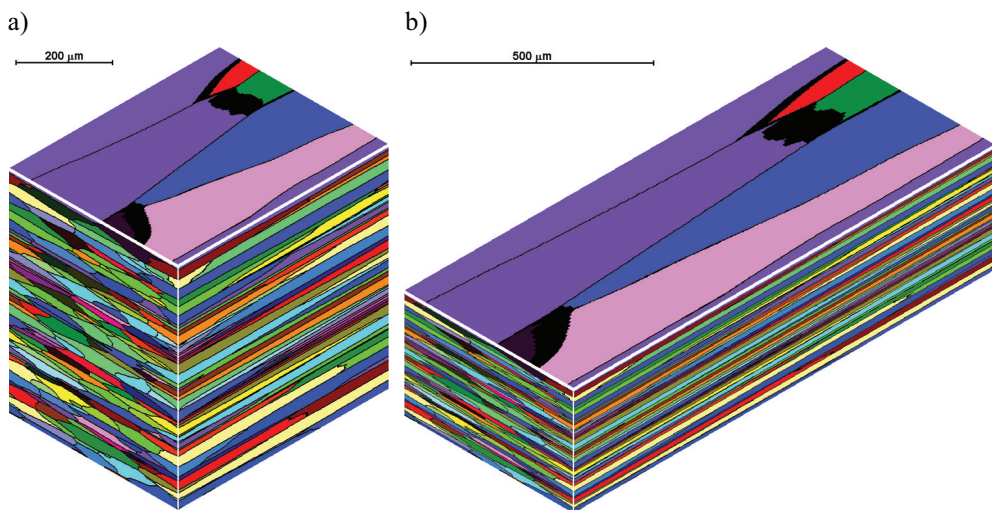
w istotnym stopniu hamuje proces obliczeń. Po czterech złożeniach (po pierwszym przepuszczeniu złożenia nie było) powstaje 16 warstw i jeżeli w początkowej strukturze na pionowym przekroju warstwy znajdowało się ponad 10 ziaren, to w strukturze przedstawionej na rysunku 9.5 na grubości modelowanej przestrzeni będzie ponad 160 ziaren. Przy założeniu, że na pionowym przekroju modelowanej przestrzeni znajduje się 500 komórek, łatwo ustalić, że grubość ziarna wynosi około trzech komórek. Przy czym niektóre ziarna osiągają grubość jednej komórki i dalsze modelowanie staje się niemożliwe.



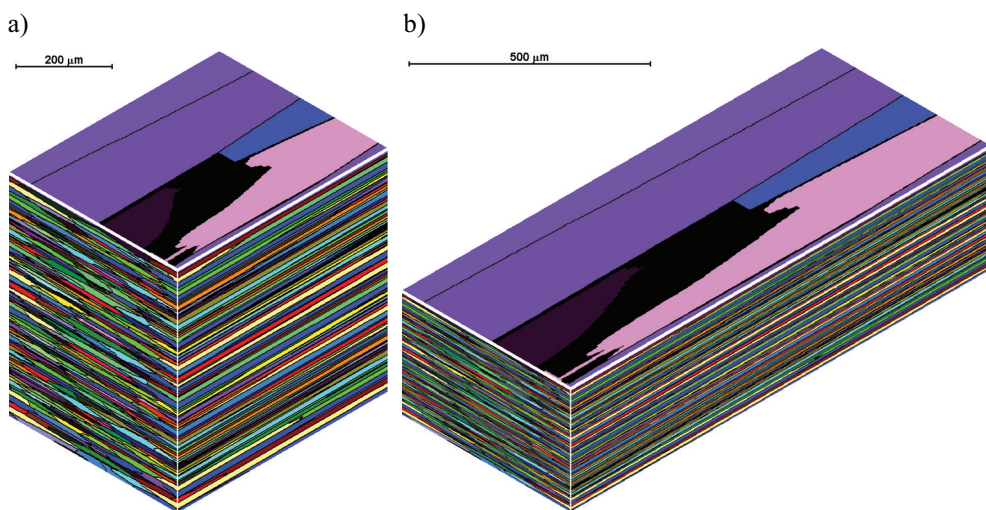
Rys. 10.14. Mikrostruktura po pierwszym (a) i drugim (b) przepuszczeniu



Rys. 10.15. Mikrostruktura przed trzecim przepustem (a) i po nim (b)



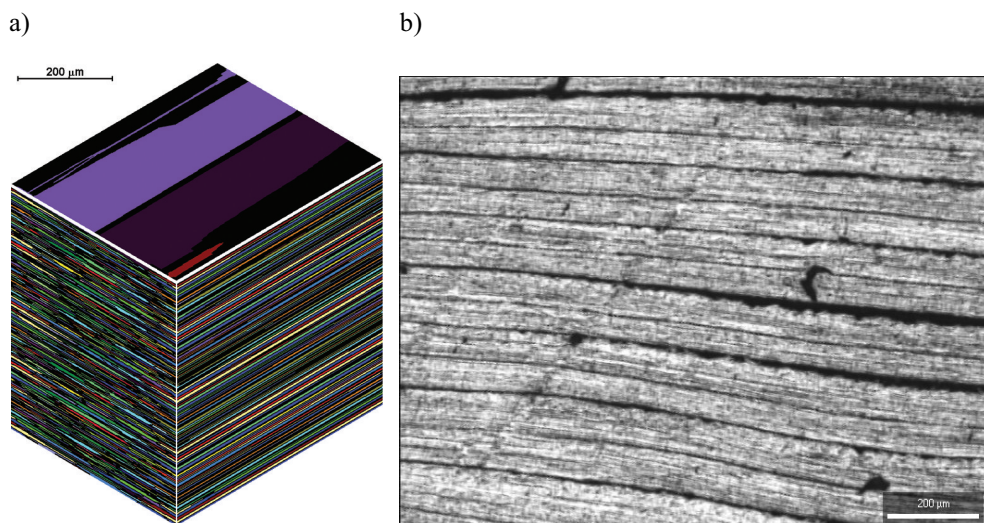
Rys. 10.16. Mikrostruktura przed czwartym przepustem (a) i po nim (b)



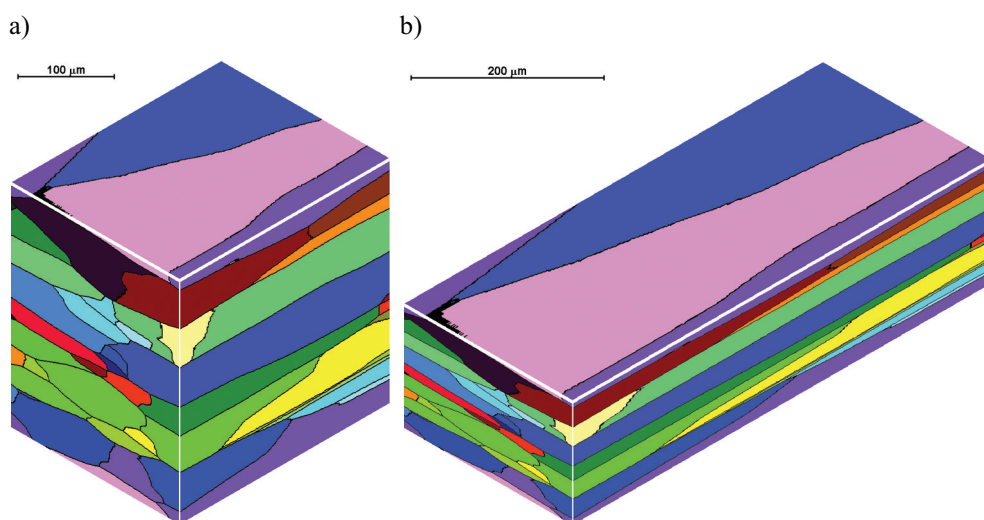
Rys. 10.17. Mikrostruktura przed piątym przepustem (a) i po nim (b)

Na rysunku 10.18 porównano wynik symulacji z rzeczywistą strukturą po pięciu przepustach (czterech złożeniach). Na obydwu rysunkach można zauważyć strukturę pasmową. Na rysunkach 10.19 i 10.20 pokazano zmiany mikrostruktury w trzecim i czwartym przepuszczeniu. W tym przypadku po przepuszczeniu jest wycinana część pasma. Wycinano czwartą część jego długości oraz połowę szerokości i w ten sposób uzyskiwano kawałek, który proporcjonalnie jest dwukrotnie mniejszy od początkowego. Liczbę komórek po wycinaniu zwiększano do wartości początkowej, czyli skala

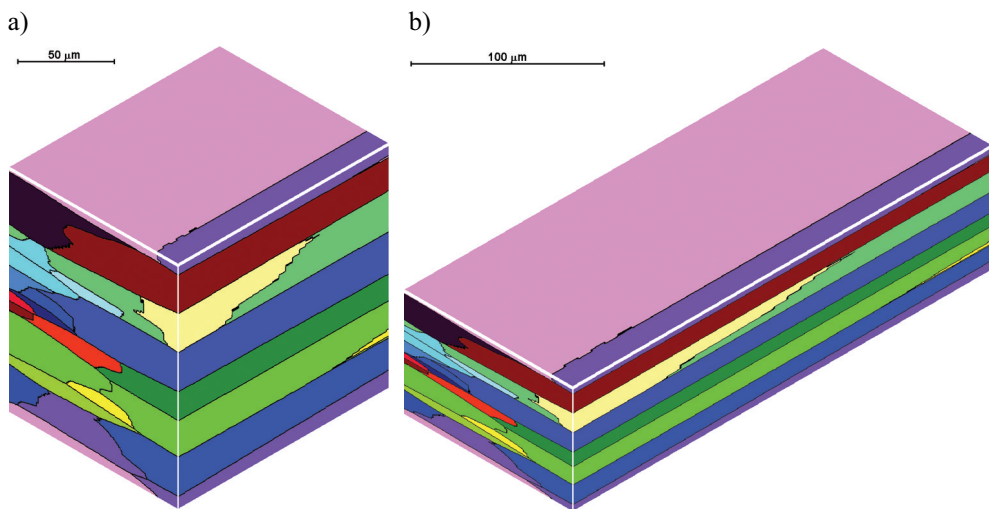
przestrzeni proporcjonalnie dwukrotnie się zmniejsza, zwiększając owej przestrzeni rozdzielczość. W takim podejściu faktycznie jest badana jedna warstwa, w której pozostaje taka sama liczba ziaren. Obydwie te metody mają swoje zalety i wady. W dalszych badaniach wykorzystywano je naprzemiennie.



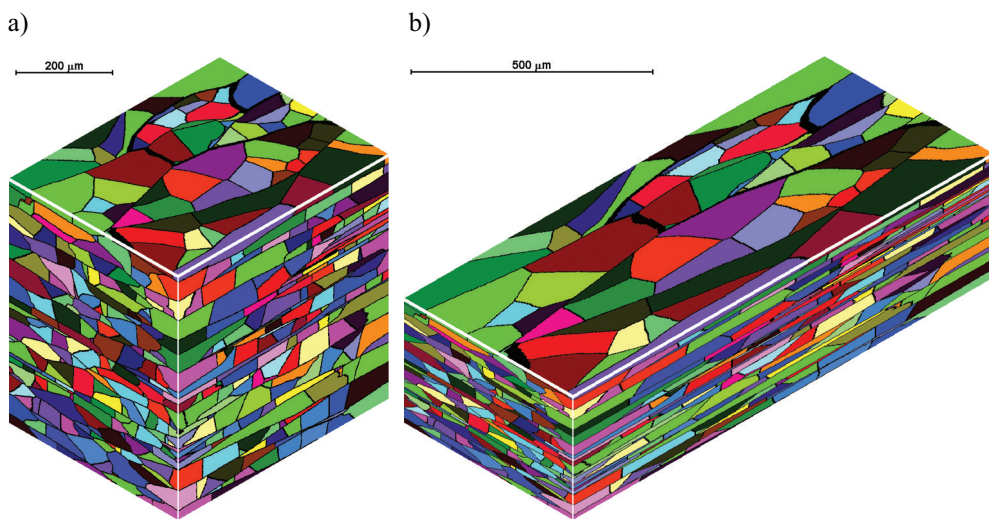
Rys. 10.18. Mikrostruktura po złożeniu po piątym przepuszczeniu (a) oraz zdjęcie rzeczywistej mikrostruktury aluminium po pięciu przepustach (b)



Rys. 10.19. Mikrostruktura przed trzecim przepustem (a) i po nim (b) (wycięcie części pasma)



Rys. 10.20. Mikrostruktura przed czwartym przepustem (a) i po nim (b) (wycięcie części pasma)

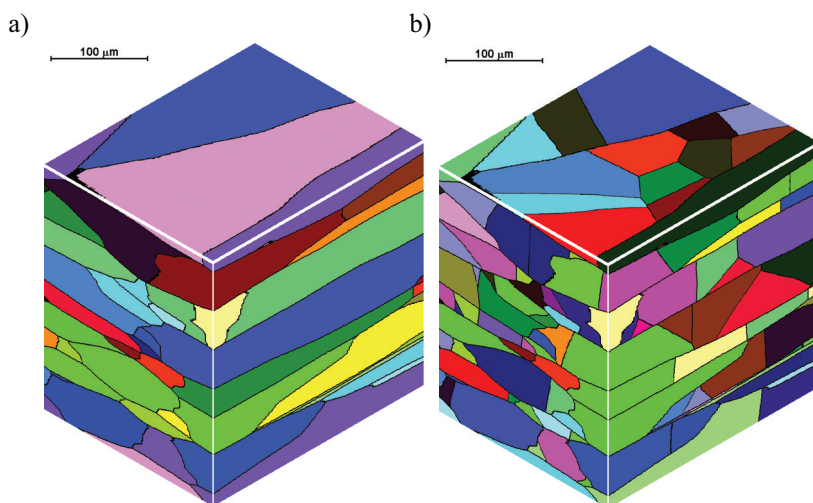


Rys. 10.21. Mikrostruktura przed trzecim przepustem (a) z uwzględnieniem rozdrobnienia w drugim przepuszczeniu (b)

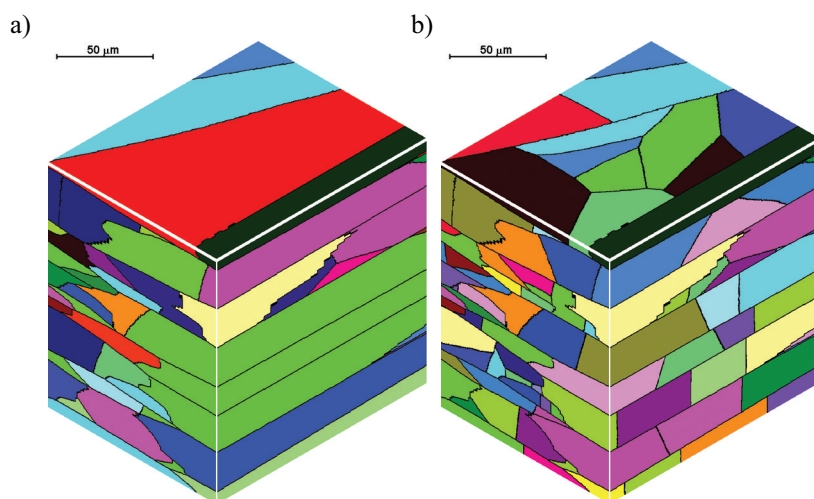
Dalsze symulacje prowadzono z uwzględnieniem rozdrobnienia ziaren. Zastosowanie rozcinania i składania pasma dość szybko doprowadziło do osiągnięcia ograniczenia przestrzeni automatów komórkowych i po pierwszym złożeniu dalsze rozdrobnienie struktury było niemożliwe. Wyniki przedstawione na rysunku 10.21 można porównać z wynikami na rysunku 10.15. Rozdrobnienie ziaren nastąpiło w drugim

przepuszcie, dlatego przed przepustem trzecim uzyskano bardziej drobnoziarnistą strukturę. Niestety, modelowanie walcowania w trzecim przepuszcie można było przeprowadzić tylko z pominięciem symulacji dalszego rozdrobnienia ziaren.

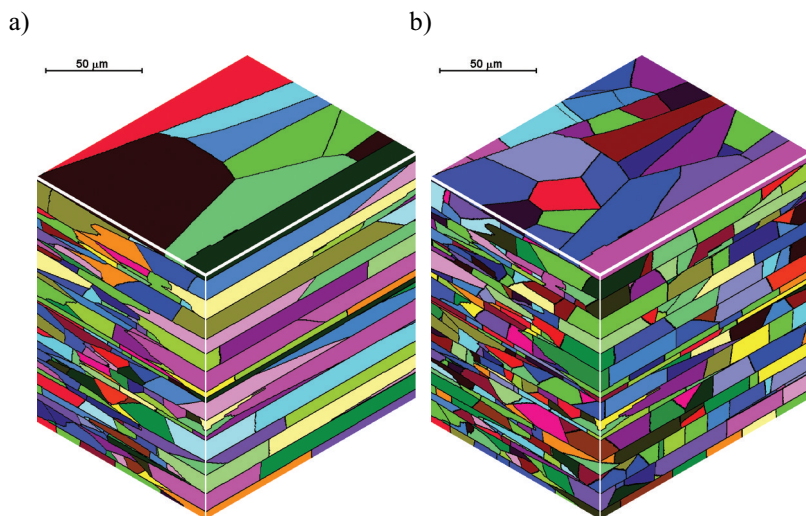
Modelowanie rozdrobnienia struktury można przeprowadzić bez istotnych ograniczeń, w sytuacji, kiedy po każdym przepuszcie jest wycinany kawałek do dalszego walcowania. Wyniki symulacji przedstawiono na rysunkach 10.22–10.24, a rzeczywistą mikrostrukturę na rysunku 10.25.



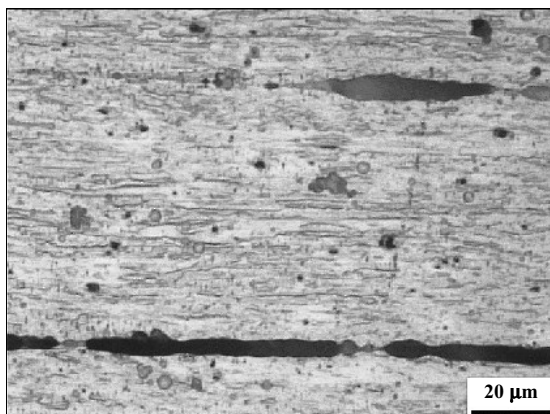
Rys. 10.22. Mikrostruktura po drugim przepuszcie bez uwzględnienia (a) i z uwzględnieniem (b) rozdrobnienia ziaren



Rys. 10.23. Mikrostruktura po trzecim przepuszcie bez uwzględnienia (a) i z uwzględnieniem (b) rozdrobnienia ziaren



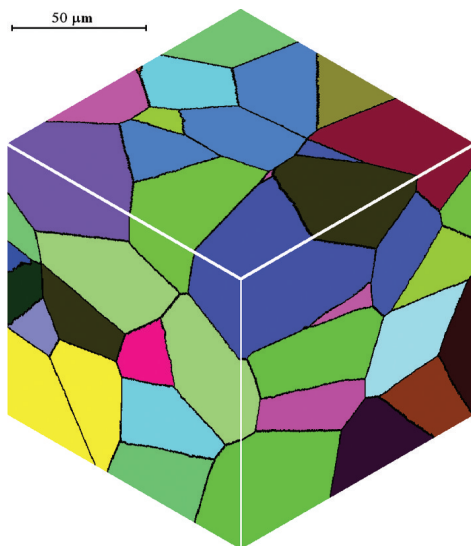
Rys. 10.24. Mikrostruktura po czwartym przepuście bez uwzględnienia (a) i z uwzględnieniem (b) rozdrobnienia ziaren



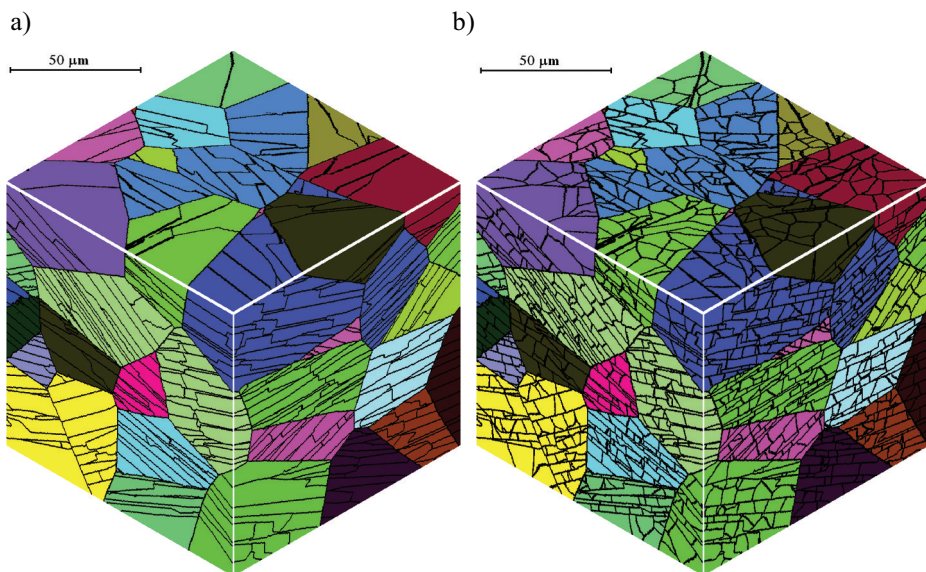
Rys. 10.25. Mikrostruktura aluminium po czwartym przepuście.

10.5.2. Symulacja procesu MAXStrain[®]

Proces MAXStrain[®] polega na wieloetapowym odkształcaniu materiału. Każdy etap prowadzono w warunkach płaskiego stanu odkształceń, kiedy jeden wymiar pozostaje bez zmian, natomiast pozostałe zmieniają się dwukrotnie. Po każdym etapie próbkę obracano tak, żeby po kolejnym odkształceniu uzyskać początkowe wymiary próbki. Proces został zasymulowany za pomocą modelu opisanego poprzednio. Początkową mikrostrukturę pokazano na rysunku 10.26, a przykładowe wyniki symulacji jej rozwoju po dwóch cyklach odkształcenia – na rysunku 10.27.



Rys. 10.26. Początkowa mikrostruktura



Rys. 10.27. Zmiana mikrostruktury w procesie MAXStrain[®] po pierwszym (a) i drugim (b) cyklu odkształcenia

Rysunek 10.27 odzwierciedla proces rozwoju struktury dyslokacyjnej w trakcie odkształcenia z silną akumulacją jego efektów. Na przykładzie zaprezentowanych mikrostruktur można zauważyć, jak w miarę zwiększania się wartości całkowitego

odkształcenia zakumulowanego dochodzi do tworzenia się struktury komórkowej, która następnie ulega dalszemu przekształcaniu w strukturę ultradrobnoziarnistą. Prezentowany proces tworzenia drobnoziarnistej struktury stanowi ilustrację zmian jakościowych. Do oceny ilościowej konieczna jest znajomość parametrów wyznaczanych doświadczalnie oraz wiedza na temat przebiegu procesu obróbki cieplnej, co wykracza poza zakres niniejszej pracy.

11. Przykłady zastosowania frontalnych automatów komórkowych do modelowania mikrostruktury w procesie walcowania

Przedstawiona w rozdziale 1 struktura modelu rozwoju mikrostruktury w procesach technologicznych (rys. 1.1) jest układem kompleksowym z elementami hierarchii. Podstawą tego modelu są frontalne automaty komórkowe, które zostały opisane w części pierwszej niniejszego opracowania. Na nich opierają się modele odpowiadające za symulację podstawowych zjawisk mikrostrukturalnych oraz działań przygotowawczych i pomocniczych (por. rozdz. 6–10). Kolejnym poziomem układu są modele procesów technologicznych. Opierając się na modelach zjawisk mikrostrukturalnych i układając je w odpowiedniej kolejności, pozwalają one zasymulować zmiany mikrostruktury w procesach technologicznych.

W tym rozdziale przedstawiono dwa przykłady symulacji rozwoju mikrostruktury w procesach technologicznych. Pierwszy dotyczy modelowania procesu wieloetapowego odkształcenia w połączeniu z wynikami modelowania metodą elementów skończonych. W ten sposób można zasymulować rozwój mikrostruktury praktycznie w dowolnym procesie technologicznym. Drugi przykład dotyczy bardziej szczegółowego przypadku walcowania wyrobów płaskich, kiedy podstawowe parametry procesu mogą być obliczone metodami analitycznymi bez zastosowania MES i wówczas cały wieloprzepustowy proces walcowania jest modelowany wyłącznie za pomocą automatów komórkowych.

W przykładach tych zastosowano dwa warianty połączenia modeli FCA i MES. Pierwszy zapewnia wykorzystanie w obliczeniach zarówno automatów komórkowych, jak i metody elementów skończonych na poziomie skali mezo. W przypadku spęczania trójwymiarowego obliczenia automatami komórkowymi przebiegają równolegle z modelowaniem metodą elementów skończonych. Wówczas rozpatrywany jest sześcián reprezentujący małą objętość materiału z narzuconymi warunkami odkształcenia. Faktycznie jest to próba zbadania nierównomierności odkształcenia struktury podczas rekrytalizacji dynamicznej.

Łącząc działanie automatów komórkowych z programem bazującym na metodzie elementów skończonych, przyjęto zasadę sztywnego połączenia siatki elementów skończonych z przestrzenią automatów komórkowych. Zrealizowanie tej zasady jest możliwe tylko w przypadku, gdy kształty elementów skończonych i komórek są kompatybilne. W idealnym przypadku komórki mają ten sam kształt i te same wymiary.

Mogą to być sześciiany lub czworościany. Tu zastosowano w elementach skończonych i w automatach komórkowych kształt sześcienny. Niestety, opracowany program oparty na metodzie elementów skończonych nie pozwala na uwzględnienie liczby elementów porównywalnej z liczbą komórek. Kostka modelowa jest dzielona jednocześnie na sześciennie elementy skończone i takie same, ale mniejszych rozmiarów komórki. Innymi słowy, każdy element skończony składa się z kilkunastu lub kilkadziesiątu komórek o tym samym kształcie.

W drugim z przedstawionych wariantów mamy do czynienia z mniej ścisłym połączeniem metody automatów komórkowych i MES. W tym podejściu obliczenia MES są niezależne od wyników symulacji automatami komórkowymi, natomiast te ostatnie otrzymują od MES niezbędne informacje o parametrach procesu. Stwarza to możliwość prowadzenia obliczeń automatami komórkowymi nie tylko równoległe z obliczeniami MES, ale również i po zakończeniu ich pracy. W przypadku walcowania trójwymiarowego w metodzie elementów skończonych dane zostają zapisane do plików, które są następnie czytane przez model automatów komórkowych.

11.1. Procesy wieloetapowego odkształcenia (walcowanie w wykrojach)

Matematyczny model i oprogramowanie (SortRoll) do modelowania trójwymiarowego płynięcia metalu podczas odkształcenia plastycznego, którego wyniki obliczeń zastosowano w niniejszej publikacji, zostały opracowane i ogłoszone w pracach Milenina (1998, 2002). Do rozwiązania przestrzennego zagadnienia teorii plastyczności zastosowano metodę elementów skończonych oraz warunek stacjonarności funkcjonału Markowa. Warunki brzegowe procesu walcowania zostały uwzględnione przy wykorzystaniu metody zaproponowanej w pracy Milenina *et al.* (2004). Metoda ta wymaga zastosowania funkcji kary do uwzględnienia warunków współdziałania metalu z narzędziem w złożonej konfiguracji przestrzennej. Model zawiera również algorytm przenoszenia pola temperatury między kolejnymi przepustami.

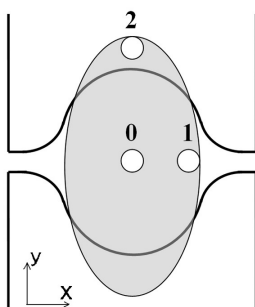
Program SortRoll pozwala na wyznaczenie wielu parametrów termomechanicznych procesu walcowania, między innymi: jego składowych prędkości, tensorów odkształceń, prędkości odkształcania, naprężeń, temperatury, pól powierzchni przekrojów poprzecznych walcowanego materiału oraz nacisku metalu na walce.

Rozbudowa programu do współpracy z modelem automatów komórkowych dotyczyła przenoszenia pól temperatury oraz intensywności odkształceń między przepustami, a także przedstawienia wyników obliczeń w formie zmiany rozpatrywanych parametrów w określonych punktach przekroju (Malinowski *et al.* 2009). Wybrano kilka punktów, których ogólne położenie przedstawiono na rysunku 11.1. W początkowym etapie symulacji zostały znalezione te węzły trójwymiarowej siatki metody elementów skończonych, które znajdowały się najbliżej rozpatrywanych punktów.

Dla wybranych warunków walcowania w wykrojach przeprowadzono metodą elementów skończonych (MES) obliczenia rozkładu odkształceń, ich prędkości i temperatury w wybranych punktach przekroju. Otrzymane dane wykorzystano do obliczenia zmiany mikrostruktury i naprężenia uplastyczniającego za pomocą opracowanego modelu automatów komórkowych.

Wyniki obliczeń dla każdego wybranego punktu są zapisywane w odpowiednich, odrębnych plikach zgodnie z następującym schematem: czas, aktualne współrzędne punktu (węzła), temperatura, prędkość odkształcenia, naprężenie uplastyczniające, intensywność odkształceń w punkcie oraz składowe tensora prędkości odkształcenia.

Faktycznie w obliczeniach automatami komórkowymi uwzględniono tylko czas, temperaturę i składowe tensora prędkości odkształcenia.



Rys. 11.1. Płożenie analizowanych punktów w przekroju wsadu
(Svyetlichnyy 2012a)

11.1.1. Warunki symulacji

Możliwości opracowanego układu omówiono na przykładzie technologii walcowania 38-milimetrowych prętów okrągłych gładkich. Warunki symulacji przyjęto na podstawie danych z literatury przedmiotu (Laber i Milenin 2007). Do analizy wykorzystano trzy ostatnie przepusty (klatki 10, 11 i 12). Temperatura pasma przed dziesiątym przepustem została zadana równomiernie na całym przekroju i miała wartości 900°C i 1000°C. Podstawowe parametry procesu walcowania podano w tabeli 11.1. W symulacjach komputerowych zadano następujące dane wejściowe: temperatura otoczenia oraz temperatura wody 20°C, temperatura walców 60°C, współczynnik tarcia $\mu = 0,8$, współczynnik wymiany ciepła pasma z otoczeniem 120 W/(m²·K), współczynnik wymiany ciepła pasma z walcami 6000 W/(m²·K).

Tabela 11.1

Schemat walcowania

Kształt wykrojów	Wymiary, mm	Prędkość walcowania, m/s	Czas przerwy, s
okrąg	45,6 × 45,6	2,94	1,43
owal	50,4 × 35,0	3,46	1,21
okrąg	38,5 × 38,5	3,99	1,5

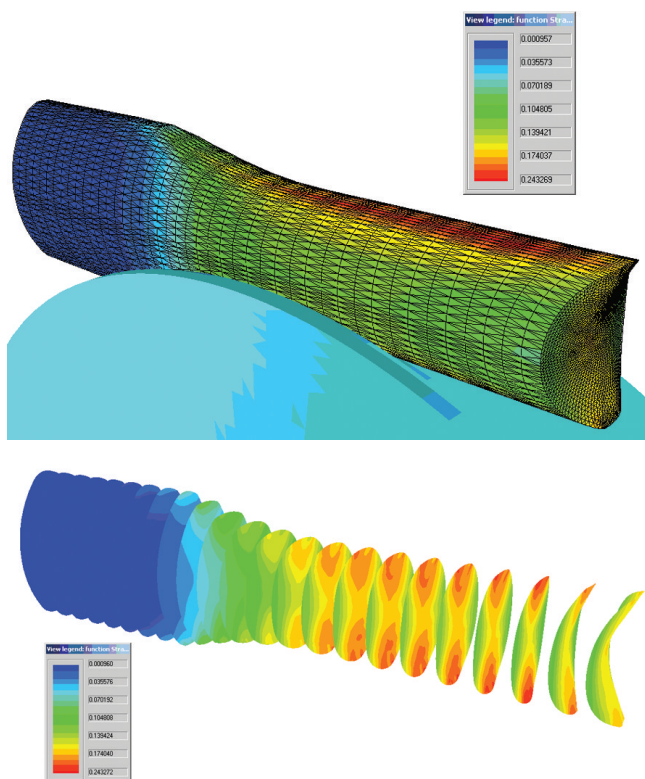
11.1.2. Charakterystyka materiału

Materiałem wykorzystanym do obliczeń była stal niestopowa konstrukcyjna S355J2G3. Jest to stal jakościowa powszechnie stosowana w konstrukcjach spawanych, zgrzewanych, nitowanych oraz łączonych śrubami. Produkowana jest w postaci blach grubych, taśm, prętów oraz kształtowników.

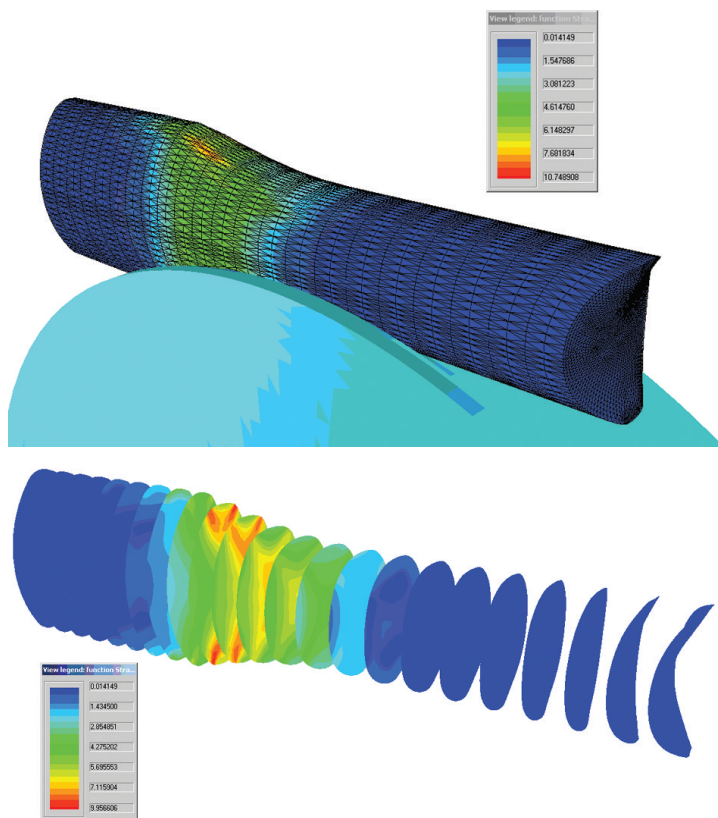
W celu wprowadzenia zależności od temperatury własności termofizycznych materiału do programu MES SortRoll aproksymowano je równaniem liniowym.

11.1.3. Wyniki modelowania MES

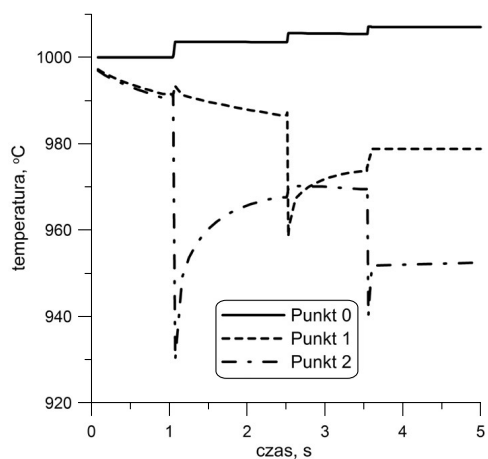
Wybrane wyniki symulacji programem MES wykonane przez A. Milenina (Malinowski *et al.* 2009) przedstawiono na rysunkach 11.2 i 11.3 w postaci trójwymiarowych rozkładów temperatury, intensywności odkształceń oraz prędkości odkształcenia. Zmiany temperatury w wybranych punktach uwidocznił na rysunku 11.4. Więcej rysunków i wykresów można znaleźć w sprawozdaniu (Malinowski *et al.* 2009).



Rys. 11.2. Rozkład intensywności odkształceń w dziesiątym przepuście (Malinowski *et al.* 2009, oprac. Milenin)



Rys. 11.3. Pole intensywności prędkości odkształcenia w dziesiątym przepływie (Malinowski *et al.* 2009, oprac. Milenin)

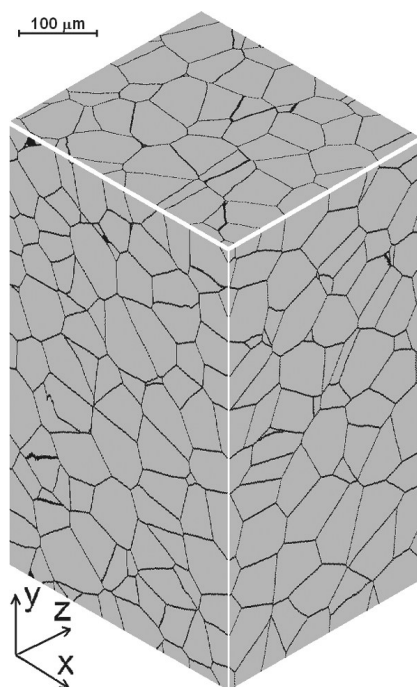


Rys. 11.4. Temperatura w wybranych punktach

11.1.4. Modelowanie rozwoju mikrostruktury

Na podstawie obliczeń MES przeprowadzono symulację zmian mikrostruktury w trzech punktach (rys. 11.1). Pliki z wynikami MES wykorzystano do utworzenia nowych plików, które zawierały tylko informacje konieczne do obliczeń automatami komórkowymi, a mianowicie: czas, temperaturę i tensor prędkości odkształcenia. Poza tym uwzględniono kantowanie w przerwie między przepustami, które polegało na łączeniu kilku plików i zmianie składowych tensora.

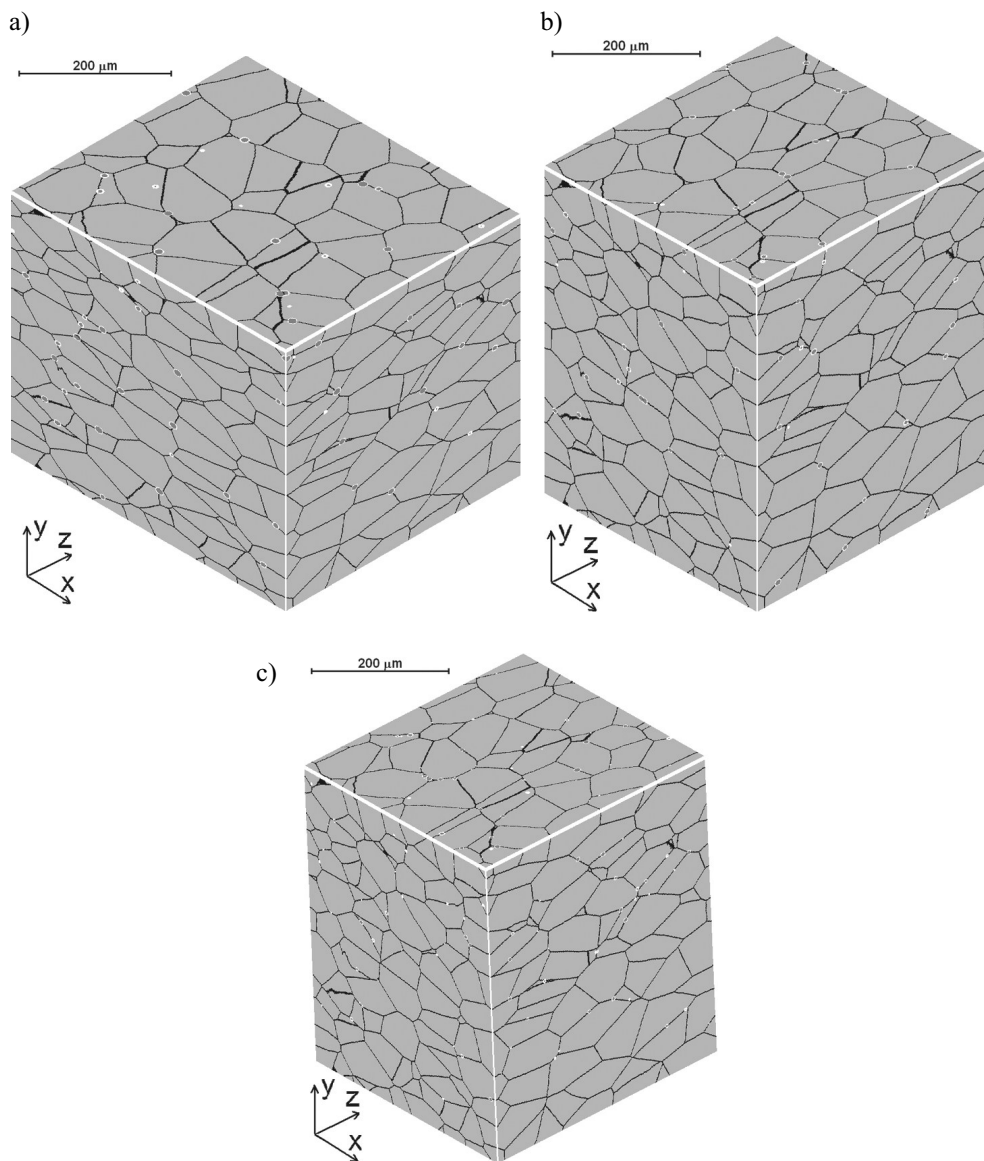
Reprezentującą mikrostrukturę objętość modelową przedstawia przestrzeń automatów komórkowych o wymiarach początkowych $280 \times 320 \times 560 \mu\text{m}^3$, która zawiera 125 mln komórek ($500 \times 500 \times 500$). Początkową mikrostrukturę, taką samą dla wszystkich trzech punktów, wygenerowano za pomocą automatów komórkowych (por. rozdz. 6) z periodycznymi warunkami brzegowymi i liczbą ziaren równą 300, co daje średnią początkową wielkość ziarna wynoszącą około $52 \mu\text{m}$ (Svyetlichnyy 2012a). Mikrostrukturę tę przedstawiono na rysunku 11.5.



Rys. 11.5. Początkowa mikrostruktura w punktach 0, 1 i 2
(Svyetlichnyy 2012a)

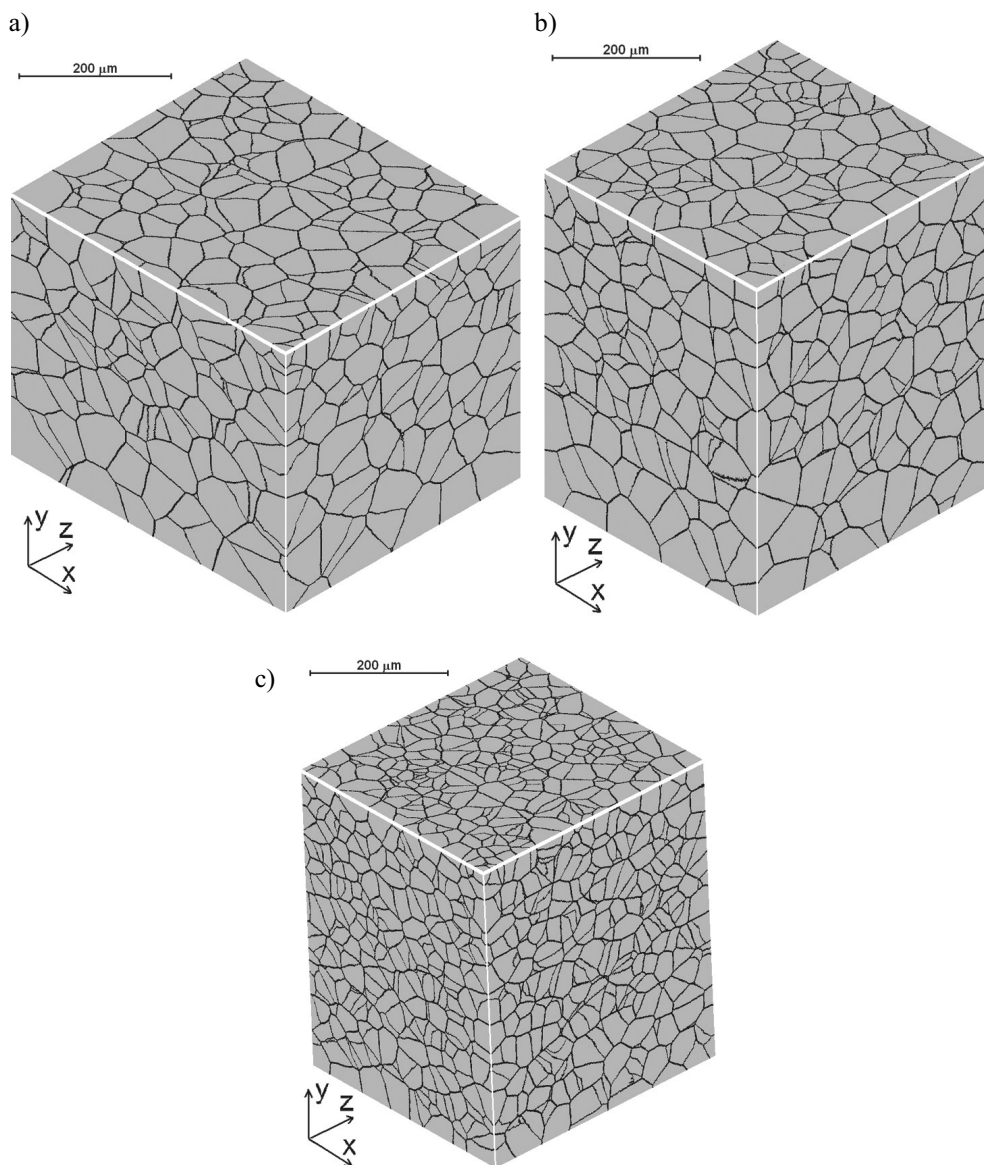
Wybrane wyniki symulacji (rys. 11.6–11.12) przedstawiono jako obrazy mikrostruktury na bokach sześcianów (rys. 11.6–11.8, 11.11, 11.12) i wykresy zmian wielkości ziaren (rys. 11.9) oraz naprężenia uplastyczniającego (rys. 11.10). Litera „a” na

rysunkach 11.6–11.8 odpowiada mikrostrukturze w punkcie 0 (rys. 11.1), „b” – w punkcie 1, a „c” – w punkcie 2. W wariacie symulacji z początkową temperaturą 900°C wybrano trzy czasy, aby pokazać mikrostrukturę w trzech momentach: po pierwszym przepięciu (rys. 11.6), przed drugim przepięciem (rys. 11.7) i po ostatnim (trzecim) przepięciu oraz chłodzeniu w ciągu 1,5 s (rys. 11.8).



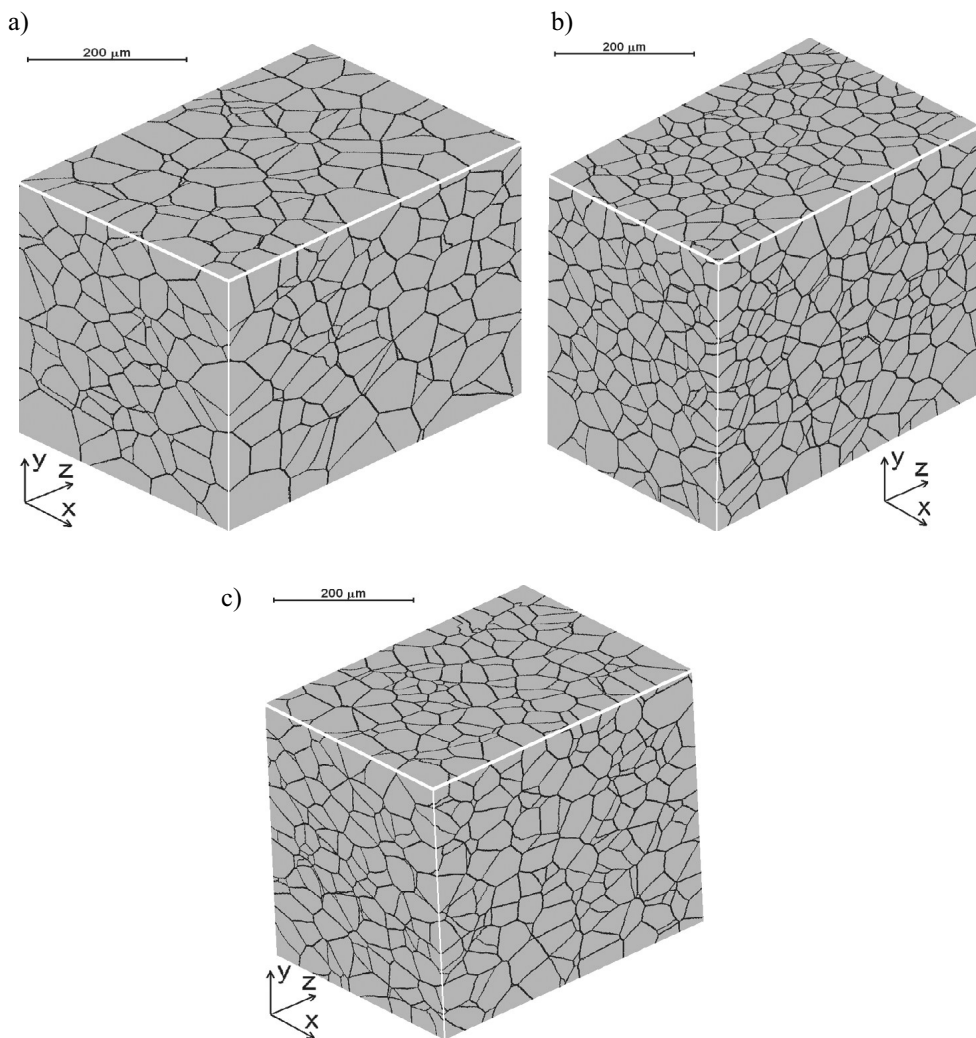
Rys. 11.6. Mikrostruktura po pierwszym przepięciu w punktach 0 (a), 1 (b) i 2 (c)
(Svyetlichnyy 2012a)

Warunki odkształcenia w tych trzech punktach są różne i wpływa to nie tylko na kształt komórek i całej przestrzeni, ale również na kształt i wielkość ziaren po odkształceniu i rekrytalizacji. Ponieważ zastosowano drugi algorytm uwzględniający równomierne odkształcenie (podrozdz. 2.4.3, rys. 2.17), komórki mają kształt prostopadło- lub równoległoscianów, odwzorowując kształt przestrzeni w mniejszej skali.



Rys. 11.7. Mikrostruktura przed drugim przepustem w punktach 0 (a), 1 (b) i 2 (c) (Svyetlichnyy 2012a)

W centralnym punkcie (punkt 0), który leży na osi symetrii przekroju, składowe ścinające tensora prędkości odkształcenia są równe zero. W dwóch innych punktach leżących na liniach symetrii przekroju składowe ścinające są dość małe. Niewielkie odchylenia od kształtu prostopadłościanu można zauważyć na rysunkach 11.6c, 11.7c i 11.8c w punkcie 2.



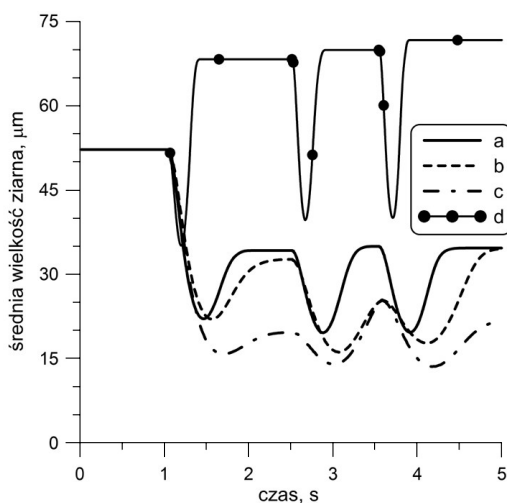
Rys. 11.8. Mikrostruktura po trzecim przepuście i chłodzeniu przed przemianą fazową w punktach 0 (a), 1 (b) i 2 (c) (Svyetlichnyy 2012a)

Odształcenie wpływa głównie na wymiary równoległościanów. Jego warunki nie odpowiadają inicjowaniu rekrytalizacji dynamicznej – w przerwach między odształceniami zachodzi tylko rekrytalizacja statyczna. Materiał stykający się z walcami

podlega szybszemu chłodzeniu, co rzutuje na kształtowanie bardziej drobnej mikrostruktury (rys. 11.7c, 11.8b, c). Związane jest to z różnymi parametrami odkształcenia – jego temperaturą i prędkością – w wybranych punktach. Parametry te wpływają na liczbę nowych ziaren i prędkość ich rozrostu. Różnice wielkości ziaren widoczne są od pierwszego przepustu.

Rekrystalizacja rozpoczyna się podczas odkształcenia (drobne ziarna, których granice zaznaczono białym kolorem na rysunku 11.6), przebiega w czasie przerwy między przepustami i kończy się przed kolejnym przepustem (rys. 11.7). Następuje pełna rekrystalizacja statyczna. Różna wielkość ziaren po rekrystalizacji w trzech punktach jest widoczna na rysunku 11.7. Ten sam wynik otrzymamy również po ostatnim przepuszczeniu. Przykłady takiej mikrostruktury po ostatnim przepuszczeniu i chłodzeniu przedstawiono na rysunku 11.8.

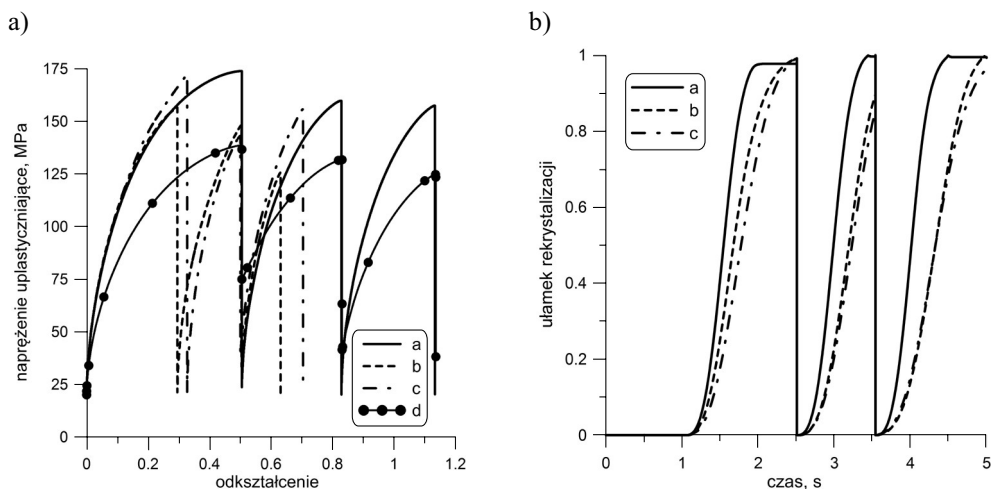
Zmiany wielkości ziarna obliczono w każdym punkcie w ciągu całego procesu walcowania i pokazano je na rysunku 11.9. Wielkość ziarna liczono jako stosunek jego reprezentatywnej objętości (która jest stała) do powierzchni wszystkich ziaren. Sposób ten został szczegółowo opisany w podrozdziale 8.6. Na wspomnianym rysunku przedstawiono cztery wykresy. Trzy odpowiadają modelowaniu z początkową temperaturą 900°C (linie oznaczone literami „a”, „b” i „c”), natomiast czwarty ilustruje zmianę wielkości ziarna w centralnym punkcie 0 przy początkowej temperaturze 1000°C (krzywa oznaczona literą „d”). Ponieważ temperatura w punkcie 0 wzrasta (por. rys. 11.3), średnia wielkość ziaren po rekrystalizacji również się zwiększa oraz staje się większa od początkowej. Największy wpływ na wielkość ziarna w omawianym modelu ma temperatura. Końcowa wielkość ziaren po walcowaniu z początkową temperaturą 900°C w punktach 0, 1 i 2 wynosi odpowiednio 34,6 μm , 34,5 μm i 21,6 μm .



Rys. 11.9. Zmiana wielkości ziarna podczas walcowania
(Svyetlichnyy 2012a)

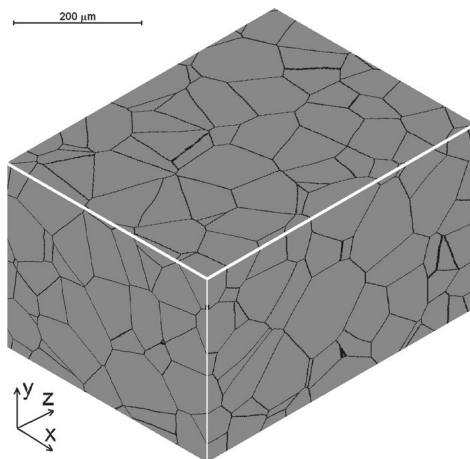
Naprężenie uplastyczniające obliczono na podstawie gęstości dyslokacji metodą przedstawioną w podrozdziale 8.2. Zmiany tego naprężenia pokazano na rysunku 11.10: zmiany względem odkształcenia (a) oraz ułamek rekrytalizacji względem czasu (b). Na rysunku 11.10a można zauważyć, że wszystkie trzy krzywe naprężenia uplastyczniającego podczas odkształcenia prawie się pokrywają, mimo różnej temperatury punktach 0, 1 i 2. Można to tłumaczyć faktem, że w niższej temperaturze w punktach 1 i 2 w porównaniu z punktem 0 odkształcenie zachodzi z mniejszą prędkością, co w pewnym stopniu kompensuje różnice. Na wspomnianym rysunku widać również istotne różnice w odkształceniu w różnych punktach. Ponieważ w modelach MES i FCA przyjęto ten sam schemat rozwoju gęstości dyslokacji przy braku odkształcenia rekrytalizacji dynamicznej, naprężenia uplastyczniające uzyskane za pomocą obu metod są identyczne.

Przedstawienie ułamka rekrytalizacji w funkcji czasu (rys. 11.10b) pozwala przeanalizować kinetykę tego procesu. Wówczas widać, że po pierwszym przepuszczeniu we wszystkich punktach następuje pełna rekrytalizacja. Podobnie wygląda sytuacja dla wszystkich przepustów w punkcie 0. Związane to jest z wysoką temperaturą i wystarczająco długim czasem przerwy. Natomiast w punktach przy powierzchni, gdzie mamy do czynienia z większym spadkiem temperatury, długość przerwy jest już niewystarczająca do zajścia pełnej rekrytalizacji statycznej.



Rys. 11.10. Zmiana naprężenia uplastyczniającego podczas walcowania jako funkcja odkształcenia (a) i czasu (b) (Svyetlichnyy 2012a)

Na rysunku 11.11 pokazano jeszcze jeden obraz mikrostruktury w punkcie 0 po ostatnim przepuszczeniu i chłodzeniu w ciągu 1,5 s od początkowej temperatury 1000°C. Tu mamy ziarna wyraźnie grubsze w porównaniu z wariantem symulacji w niższej temperaturze w tym samym punkcie (rys. 11.8a). W dalszej symulacji przemiany fazowej podczas chłodzenia walcowanego pręta mikrostruktura ta występowała jako początkowa.



Rys. 11.11. Zmiana naprężenia uplastyczniającego podczas walcowania jako funkcja odkształcenia (a) i czasu (b) (Svyetlichnyy 2012a)

Po walcowaniu pręt chłodzono z różną prędkością. Mikrostruktury po przemianie fazowej przy zastosowaniu trzech prędkości chłodzenia (0,5 K/s, 1,0 K/s i 3,0 K/s) przedstawiono na rysunku 11.12.

Do obliczeń liczby zarodków N_α (więc i ziaren ferrytu) często stosowano empiryczne wzory uwzględniające prędkość chłodzenia Cr i powierzchnię ziaren S . Zamiast powierzchni ziaren uwzględniono średnią wielkość ziarna austenitu D_γ i zakumulowane odkształcenie resztkowe ϵ_{ret} :

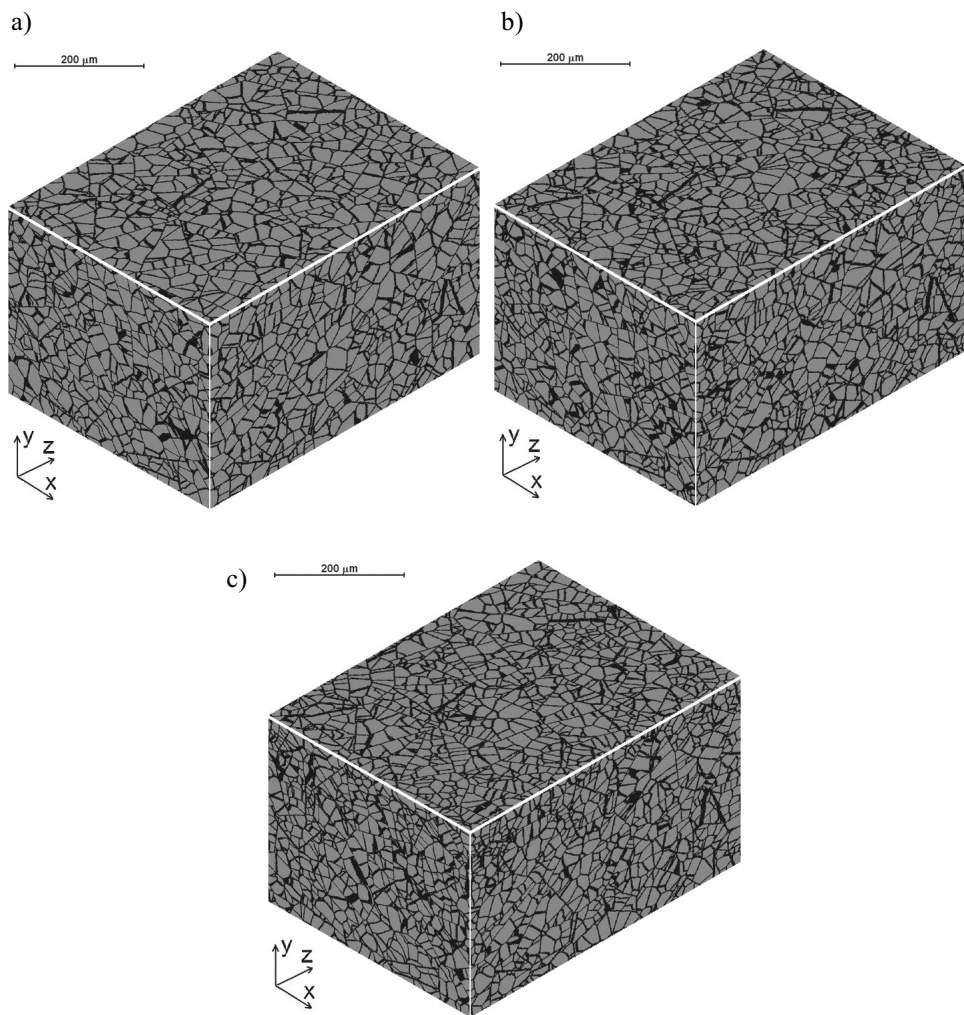
$$N_\alpha = a_f Cr^{1/2} S = \frac{1}{\left\{1,4 + 5Cr^{-1/2} + 22\left[1 - \exp(-0,015D_\gamma)\right]\right\}\left(1 - 0,45\epsilon_{ret}^{0,5}\right)} \quad (11.1)$$

Średnią wielkość ziaren ferrytu D_α liczono na podstawie objętości modelowej przestrzeni V i liczby ziaren:

$$D_\alpha = \sqrt[3]{\frac{6}{\pi} \frac{V}{N_\alpha}} \quad (11.2)$$

Średnia wielkość ziarna austenitu po walcowaniu z początkową temperaturą 1000°C w punkcie 0 wyniosła 72 μm. Liczba ziaren ferrytu po przemianie fazowej z prędkościami 0,5 K/s, 1,0 K/s i 3,0 K/s wynosiła odpowiednio 7876, 10446 i 14386. Daje to średnią wielkość ziarna 23,0 μm, 20,9 μm i 18,8 μm. Średnia liczba ziaren ferrytu przypadająca na jedno ziarno austenitu wynosi 30,7, 40,7 i 56,0.

W wariacie o początkowej temperaturze 900°C średnia wielkość ziarna austenitu wynosiła 35 μm, a liczba ziaren ferrytu po przemianie – 18022, 26324 i 41006. Średnia wielkość ziaren ferrytu wynosiła 17,5 μm, 15,4 μm i 13,3 μm. Średnia liczba ziaren ferrytu przypadająca na ziarno austenitu wynosiła 8,0, 11,8 i 18,3.



Rys. 11.12. Mikrostruktura po trzecim przepuście i chłodzeniu przed przemianą fazową w punktach 0 (a), 1 (b) i 2 (c) (Svyetlichnyy 2012a)

Za pomocą omówionego algorytmu można zamodelować dowolny proces technologiczny, nie tylko walcowanie, a więc automaty komórkowe mogą być wykorzystane do symulacji dowolnego procesu przeróbki plastycznej na gorąco.

11.2. Walcowanie wyrobów płaskich

W pewnych procesach technologicznych, do których zalicza się walcowanie wyrobów płaskich, w którym może nie być wymagany wysoki poziom dyskretyzacji

w całym przekroju, da się zastosować inne podejście: zamiast wykonywania skomplikowanych obliczeń parametrów procesu można wykorzystać ich średnie wartości, obliczone z wystarczającą dokładnością na podstawie rozwiązań analitycznych. Ponieważ w przypadku automatów komórkowych wystarczy ograniczona charakterystyka procesu, symulację takich procesów można przeprowadzać bez stosowania metody elementów skończonych.

Model rozwoju mikrostruktury oparty na automatach komórkowych potrzebuje do obliczeń czasu trwania przeróbki plastycznej informacji o następujących parametrach: czasie przerw między kolejnymi etapami odkształcenia oraz temperaturze i prędkości odkształcenia. Parametry te można uzyskać zarówno z rzeczywistego procesu, jak i w wyniku przeprowadzenia obliczeń analitycznych na podstawie planu gniotów.

11.2.1. Dane wejściowe

Przedstawione w tym rozdziale symulacje przeprowadzono na podstawie planów gniotów opracowanych w monografii autora (Svyetlichnyy 2004), z których jeden przedstawiono w tabeli 11.2. Obliczenia przeprowadzono dla walcarki wykańczającej nawrotnej z walcami o średnicy 1000 mm. Zasymulowano walcowanie blachy o grubości 40 mm i długości 10 m. Prędkość walcowania we wszystkich przepustach wyniosła 5 m/s, czas między wyjściem blachy z walców a wejściem w walce w kolejnym przepuszczeniu – 2,3 s. Między przepustami założono chłodzenie w powietrzu. Symulowano mikrostrukturę metalu w osi blachy. Modelowanym materiałem była stal 45.

Wejściowy plan gniotów zawiera jedynie informację o temperaturze oraz grubości wejściowej i wyjściowej, czyli pierwsze 4 kolumny tabeli 11.2. Na podstawie rozwiązań analitycznych obliczono dodatkowe parametry procesu walcowania, część z nich również zamieszczono w tabeli 11.2. Są to: gniot (kolumna 5), średnie odkształcenie, prędkość odkształcenia (kolumna 6), długość wyjściowa, czas walcowania i czas przerwy (kolumna 7) z uwzględnieniem czasu od wyjścia blachy z poprzedniego przepustu i wejścia w kolejny oraz czasu walcowania. Założono, że temperatura podczas odkształcenia pozostaje bez zmian, tak jak i prędkość odkształcenia.

Tabela 11.2

Plan gniotów (Svyetlichnyy 2004)

Numer przepustu	Grubość wejściowa h_0 , mm	Grubość wyjściowa h_1 , mm	Temperatura T , °C	Gniot r , %	Prędkość odkształcenia $\dot{\epsilon}$, s ⁻¹	Czas przerwy τ , s
1	2	3	4	5	6	7
1	40,00	27,18	1098	32	90,6	5,24
2	27,18	18,55	1087	31,7	90,1	6,60
3	18,55	12,75	1061	31,3	89,4	8,70
4	12,75	8,95	1010	29,8	87,3	11,20
5	8,95	6,49	928	27,5	83,6	14,60

Symulowano tylko rekryształizację statyczną, ponieważ nawet w temperaturze 1100°C i przy prędkości odkształcenia 10 s^{-1} odkształcenie pikowe wynosi ponad 0,35. Natomiast zgodnie z warunkami modelowania (wszystkie temperatury poniżej 1100°C, odkształcenie około 0,3 i prędkość około 90 s^{-1}) rekryształizacja dynamiczna w takim przypadku nie zachodzi.

Po odkształceniu obliczano wymiary komórki w celu określenia kroku czasowego i liczby iteracji koniecznych do symulacji walcowania. Pod koniec walcowania powstają zarodki nowych ziaren. Szczegółowo model ten został opisany w rozdziale 8. W przerwie między przepustami nowe zarodki nie powstają, symulowano więc rozrost ziaren bez zarodkowania. Zmiany temperatury między przepustami aproksymowano liniową zależnością od czasu. Po ostatnim przepuście materiał chłodzono z prędkością 10 K/s w ciągu 5 s.

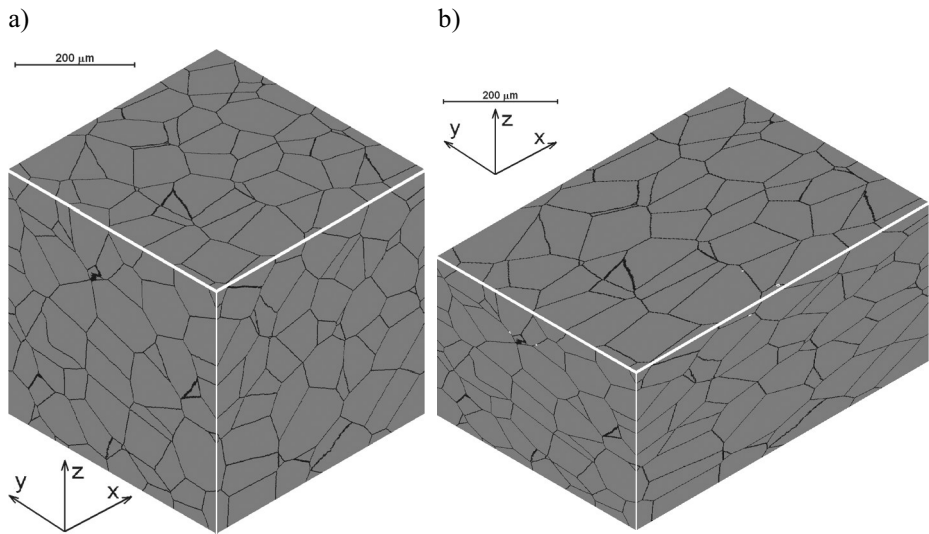
11.2.2. Wyniki symulacji

Za pomocą FCA utworzono początkową mikrostrukturę (rys. 11.13) o następujących parametrach: przestrzeń FCA zawiera $n_x \times n_y \times n_z = 500 \times 500 \times 500$ komórek z reprezentatywną objętością $V_r = a_x \times a_y \times a_z = 0,4 \times 0,4 \times 0,4 = 0,064 \text{ mm}^3$. Początkowa mikrostruktura zawiera 200 ziaren, co odpowiada średniej wielkości ziarna $70 \text{ }\mu\text{m}$. Oś x jest kierunkiem walcowania, y – szerokością blachy, z – grubością. Wymiar blachy w kierunku y pozostaje bez zmian, x wydłuża się, z się skraca. Możliwe są dwa warianty wizualizacji wyników obliczeń. W pierwszym wykorzystano paletę 45 kolorów, które odzwierciedlają różną orientację krystalograficzną. Jednak w niniejszym rozdziale w wizualizacjach posłużono się skalą szarości, która prezentuje stan ziaren. Stare ziarna zaznaczono kolorem ciemnoszarym, nowe są jasnoszare. Granice ziaren są czarne, natomiast ruchome granice rosnących ziaren zaznaczono kolorem białym.

Wyniki symulacji przedstawiono w postaci rysunków izometrycznych z mikrostrukturą w wybranych chwilach czasu pokazaną na powierzchni prostopadłościannów ograniczających przestrzeń automatów (rys. 11.13–11.18) oraz wykresów demonstrujących zmiany wielkości ziarna (rys. 11.19) i naprężenia uplastyczniającego (rys. 11.20) podczas całego procesu walcowania. Mikrostruktura została pokazana w chwili czasu związanej z początkiem lub końcem odkształcenia. W związku z istotnym sumarycznym wydłużeniem walcowanego materiału zastosowano reorganizację przestrzeni automatów opisaną w rozdziale 3 i automatyczną przebudowę przed kolejnym przepustem zgodnie z warunkami zapisanymi we wzorach (3.1). Wartości przyjęte we wzorach (3.1) są następujące: minimalna dopuszczalna liczba ziaren w całej przestrzeni $n_{g\min} = 500$, maksymalna $n_{g\max} = 4000$.

Kolejność obliczeń i wybrane wyniki zebrano w tabeli 11.3. Do przedstawienia reorganizacji przestrzeni wykorzystano takie same oznaczenia jak w rozdziale 3: BXZ – przecinanie przestrzeni w poprzek długości ($a_x/2$) i sklejanie jej po powierzchni poziomej ($2a_z$); HX i HY – dzielenie w poprzek długości ($a_x/2$) i w poprzek szerokości ($a_y/2$). Przed drugim przepustem przestrzeń nie jest zmieniana ze względu na małą liczbę ziaren ($n_g = 453 < n_{g\min} = 500$), co uniemożliwia dzielenie, natomiast stosunek

krawędzi jest niewystarczający do zastosowania przecinania i sklejania przestrzeni ($a_x = 0,588 < 3a_z = 3 \cdot 0,272 = 0,716$). Przed trzecim przepustem następuje przecinanie i sklejanie, a przed czwartym i piątym – jeden i dwa podziały (por. przedostatnia kolumna tabeli 11.3).



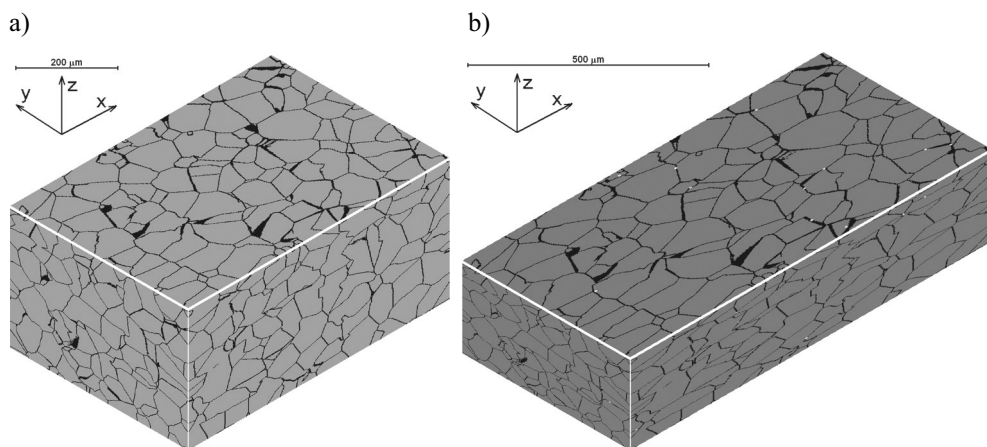
Rys. 11.13. Mikrostruktura przed pierwszym przepustem (a) i po nim (b).
 Reprezentatywna objętość $V_r = 0,064 \text{ mm}^3$ (Svyetlichnyy 2012b)

Tabela 11.3
 Wyniki obliczeń (Svyetlichnyy 2012b)

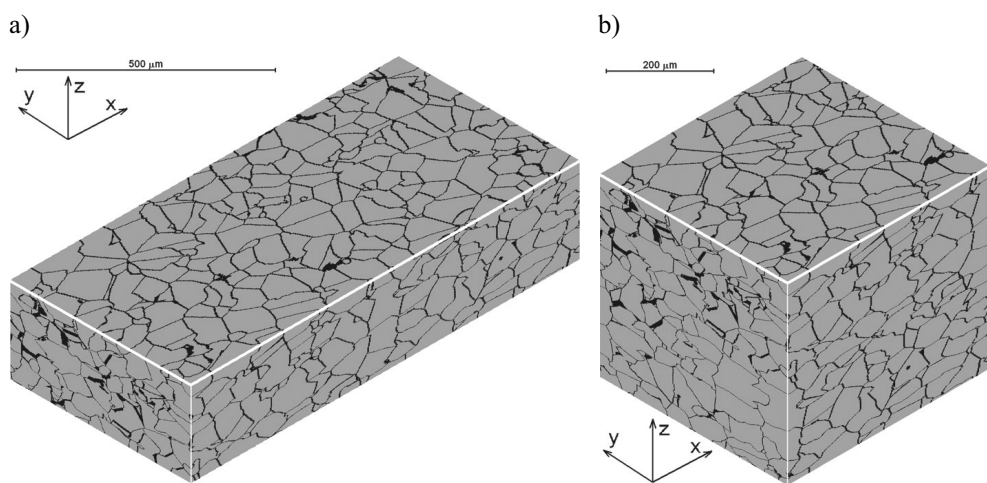
Numer przepustu	Długość × szerokość × grubość $a_x \times a_y \times a_z$, mm	Objętość V , mm^3	Liczba ziaren przed przepustem i po nim	Przekształcenie przestrzeni	Rysunek
1	$0,4 \times 0,4 \times 0,4$	0,064	200/453	–	11.13
2	$0,588 \times 0,4 \times 0,272$	0,064	453/576	<i>BXZ</i>	11.14 i 11.15
3	$0,431 \times 0,4 \times 0,371$	0,064	576/994	<i>HX</i>	–
4	$0,314 \times 0,4 \times 0,255$	0,032	674/1464	<i>HX + HY</i>	11.16 i 11.17
5	$0,224 \times 0,2 \times 0,179$	0,008	625/4850	–	11.18

Na rysunkach 11.13b, 14b, 16b, 18b można zaobserwować efekt walcowania w postaci małych białych punktów. Są to zarodki nowych ziaren, które ze względu na brak rekrytalizacji dynamicznej nie mają warunków do rozrostu podczas odkształcenia. Następuje pełna rekrytalizacja statyczna, z wyjątkiem dwóch ostatnich przepustów.

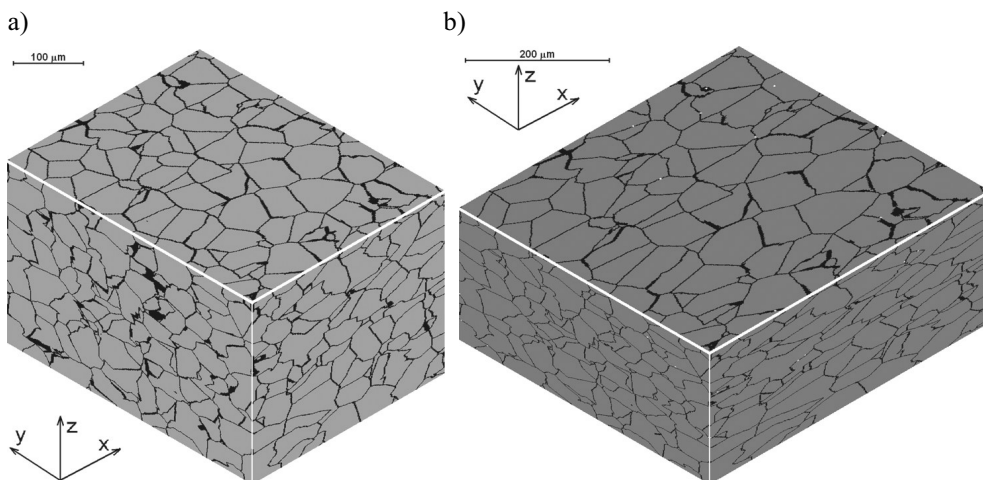
Podczas pierwszych przepustów panuje dostatecznie wysoka temperatura i jest wystarczająco dużo czasu, aby zaszła pełna rekryształizacja. Na rysunku 11.17b ilustrującym sytuację przed ostatnim przepustem widać kilka białych granic rosnących ziaren, co oznacza, że rekryształizacja nie dobiegła końca. Po ostatnim, piątym, przepuszczeniu i chłodzeniu rekryształizacja na skutek spadku temperatury jest zatrzymywana (mała siła napędowa rozrostu ziaren i dość krótki czas) i dlatego na rysunku 11.18b można zobaczyć niewiele małych nowych ziaren.



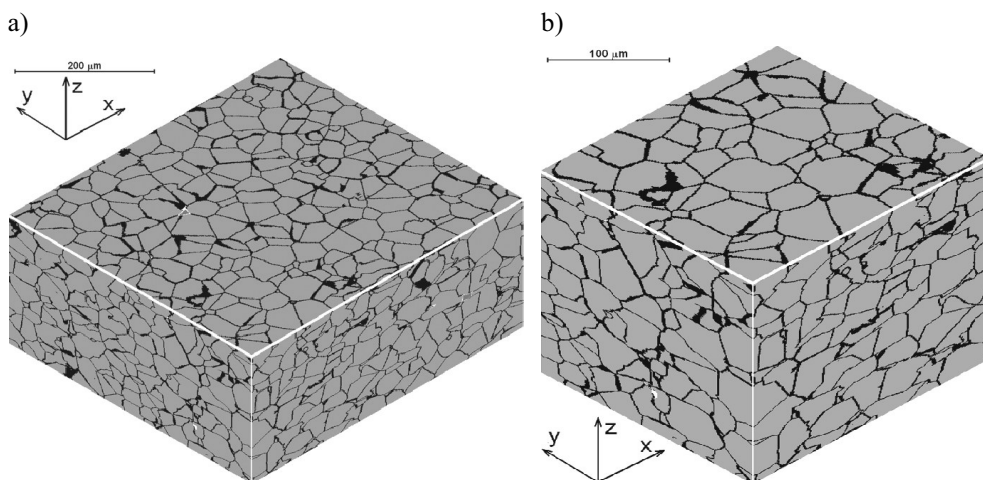
Rys. 11.14. Mikrostruktura przed drugim przepustem (a) i po nim (b).
Reprezentatywna objętość $V_r = 0,064 \text{ mm}^3$ (Svyetlichnyy 2012b)



Rys. 11.15. Mikrostruktura po rekryształizacji po drugim przepuszczeniu przed przecinaniem i sklejeniu przestrzeni automatów komórkowych (a) i po tym zabiegu (b).
Reprezentatywna objętość $V_r = 0,064 \text{ mm}^3$ (Svyetlichnyy 2012b)



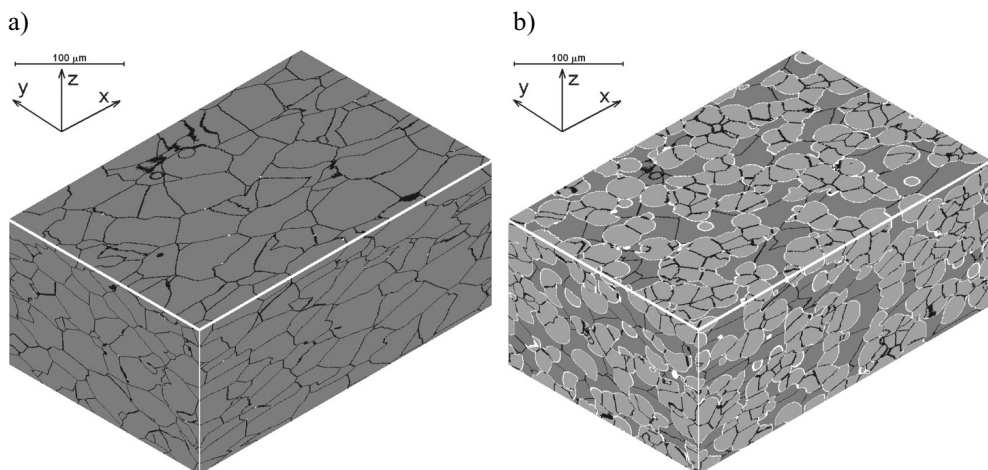
Rys. 11.16. Mikrostruktura przed czwartym przepustem (a) i po nim (b).
Reprezentatywna objętość $V_r = 0,032 \text{ mm}^3$ (Svyetlichnyy 2012b)



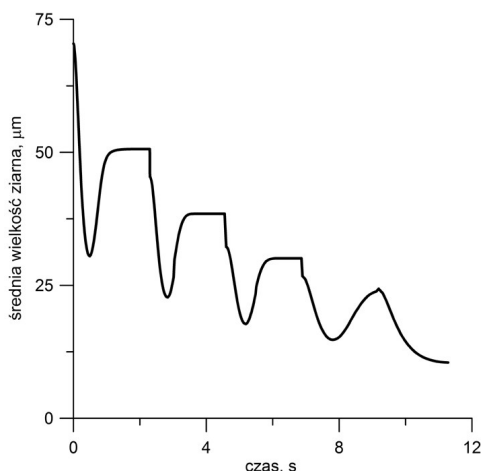
Rys. 11.17. Mikrostruktura po rekrytalizacji po czwartym przepuszczeniu przed podziałem
w dwóch kierunkach (a) i po tym zabiegu (b). Reprezentatywna objętość V_r
zmniejsza się z $0,032 \text{ mm}^3$ do $0,008 \text{ mm}^3$ (Svyetlichnyy 2012b)

Procedura reorganizacji przestrzeni z przecinaniem i sklejanym jest dobrze widoczna na rysunku 11.15. Po przecięciu przestrzeni w poprzek długości obie jej części zajmują pozycję jedna nad drugą. Ten wariant jest dokładnie opisany w rozdziale 3, gdzie omówiono również inny sposób przeprowadzenia takiej przebudowy (rys. 3.15). Ponieważ struktura w miejscu połączenia (sklejania) jest identyczna nie powstają żadne nieciągłości. Dlatego też na rysunku 11.19b nie można zobaczyć linii ani powierzchni

łączenia dwóch części przestrzeni automatów komórkowych. Podwójne dzielenie po czwartym przepuszczeniu (na długości i szerokości) pokazano na rysunku 11.17. Do dalszych obliczeń wykorzystano tylko ćwiartkę reprezentatywnej objętości V_r . Objętość V_r zmniejsza się z $0,032 \text{ mm}^3$ do $0,008 \text{ mm}^3$.



Rys. 11.18. Mikrostruktura po piątym przepuszczeniu przed rekrytalizacją (a) i po niej (b). Reprezentatywna objętość $V_r = 0,008 \text{ mm}^3$ (Svyetlichnyy 2012b)



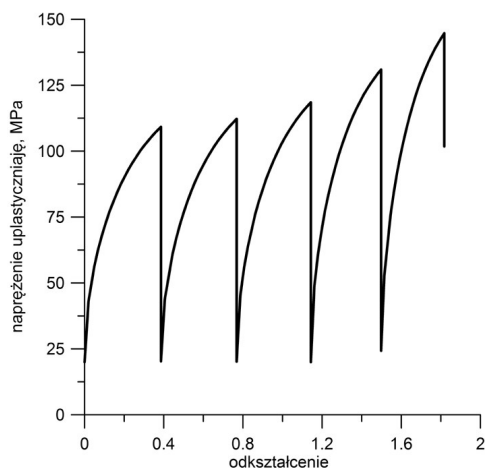
Rys. 11.19. Zmiana wielkości ziarna podczas walcowania (Svyetlichnyy 2012b)

Na rysunku 11.19 przedstawiono zmiany wielkości ziarna podczas symulacji. Metoda ich obliczenia została opisana w podrozdziale 8.6, a przykład dotyczący walcowania w wykrojach pokazano na rysunku 11.9. Rekrytalizacja statyczna po czwartym

przepuszczenie nie jest pełne, co można zaobserwować na rysunku 11.19. Dla pełnej rekrytalizacji statycznej charakterystyczna jest obecność prawie poziomego odcinka, na którym dana wielkość się nie zmienia. Parametr pozostaje bez zmian, ponieważ siła napędowa termicznego rozrostu ziarna jest wielokrotnie mniejsza od siły jego rozrostu podczas rekrytalizacji związanej z gęstością dyslokacji i dlatego w modelu została pominięta.

Średnia wielkość ziarna osiąga podczas rekrytalizacji minimalną wartość wtedy, kiedy rosnące ziarna jeszcze nie wypełniły całej przestrzeni, nie wyrosły maksymalnie, a stare nie zniknęły zupełnie, choć istotnie się zmniejszyły.

Do obliczenia naprężenia uplastyczniającego zastosowano model opisany w podrozdziale 8.2, a wyniki przedstawiono na rysunku 11.20. Odzwierciedlają one zmiany naprężenia względem sumarycznego odkształcenia. Na rysunku tym można zaobserwować również wzrost naprężenia od przepustu do przepustu. Wzrost ów można wytłumaczyć spadkiem temperatury walcowania. Po walcowaniu, tak jak w przypadku walcowania w wykrojach, istnieje możliwość symulacji przemian fazowych.



Rys. 11.20. Zmiana naprężenia uplastyczniającego podczas walcowania (Svyetlichnyy 2012b)

Podsumowanie i analiza wyników badań teoretycznych

Analiza metody automatów komórkowych na obecnym etapie jej rozwoju, zasad ich działania i stosowanych w tej metodzie algorytmów pozwoliła uwidocznic jej ograniczenia oraz trudności w zakresie symulacji mikrostruktury podczas plastycznej przeróbki metali. Omówiono również zalety tej metody. Oto lista jej najistotniejszych cech:

- Klasyczny algorytm rozrostu ziarna jest najprostszy w zastosowaniu; wymaga najmniejszych nakładów prac projektowych i obliczeniowych. Cechuje go jednak wysoka przestrzenna anizotropia. Kształt ziarna związany jest z wybranym otoczeniem (sąsiedztwem), co nie pozwala sterować ani kształtem, ani prędkością

rozrostu. Wprowadzenie reguł losowych powoduje zmniejszenie prędkości obliczeń, pozwala nieznacznie poprawić izotropowość ziaren w przestrzeni i umożliwia wprowadzenie częściowej kontroli nad prędkością rozrostu. W praktyce jednak kształt ziaren i prędkość ich rozrostu można uznać za parametry niesterowalne.

- Zastosowanie stanu przejściowego pozwala na pełne sterowanie prędkością rozrostu, a w konsekwencji daje możliwość uzyskania struktury izotropowej przy użyciu ograniczonej liczby rodzajów sąsiedztwa. W przestrzeni dwuwymiarowej możliwe jest uzyskanie rosnących ziaren o kształcie czworokątnym, okrągłym oraz osiowosymetrycznym, a w przestrzeni trójwymiarowej: sześciennym i kulistym.
- Prędkość rozrostu może być uzależniona od parametrów zewnętrznych, takich jak temperatura, oraz wewnętrznych, do których można zaliczyć skład chemiczny, gęstość dyslokacji, kąt dezorientacji i inne.
- Zaleca się stosowanie periodycznych warunków brzegowych, rzadziej warunków otwartych.
- Kształt komórek i przestrzeń automatów komórkowych są niezmiennie, nie uwzględniają rzeczywistego odkształcenia materiału. Geometria komórek i przestrzeni są stałe w czasie.
- Konwencjonalne automaty komórkowe wymagają przeszukiwania całej przestrzeni automatów i otoczenia każdej komórki w każdym kroku czasowym, co prowadzi do istotnego wydłużenia czasu obliczeń, w szczególności w algorytmach trójwymiarowych.

Na podstawie analizy wyników badań opracowano następujące zasady i algorytmy dotyczące automatów komórkowych:

- Określono zasady tworzenia algorytmów i metody uzyskania przestrzeni izotropowej w dwuwymiarowym i trójwymiarowym sąsiedztwie Moore'a, dla komórek o kształcie czworokątnym i sześciennym, o kątach prostych, zdeformowanych lub o kształcie dowolnym.
- Określono reguły uzyskania praktycznie dowolnego kształtu rosnącego ziarna w przestrzeni dwu- i trójwymiarowej. Opracowane algorytmy i programy testowano na następujących kształtach: kulistym, sześciennym, ośmiościennym, równoległościennym, elipsoidalnym oraz walcowatym, przyjmując dowolne wymiary tych figur.
- Podczas modelowania założono zależność prędkości rozrostu ziaren od parametrów zewnętrznych i wewnętrznych oraz mikrostrukturalnych, związanych z kształtem ziaren, ich krystalograficzną orientacją i kierunkiem rozrostu.
- Uzupełniono warunki brzegowe periodyczne i otwarte o warunki periodyczne z przesunięciem, mieszane i kombinowane. Uwzględniono również zależności warunków brzegowych od rosnącego ziarna, a nie tylko od przestrzeni. Uzależniono warunki brzegowe od przewidywanego typu przebudowy przestrzeni.

- Opracowano reguły i algorytmy uwzględniania rzeczywistego odkształcenia materiału przez odpowiednie zmiany kształtu i wymiarów komórki oraz całej przestrzeni. Związano przestrzeń automatów z modelowanym metodą elementów skończonych płynięciem punktów materialnych.
- Opracowano zasady i algorytmy przekształcenia przestrzeni automatów komórkowych i komórek przy ich istotnym zniekształceniu. Uzależniono wybierany typ przebudowy od parametrów mikrostrukturalnych modelowanego procesu.
- Opracowano i zastosowano frontalne automaty komórkowe, które w każdej iteracji wykorzystują tylko część komórek znajdujących się w pobliżu granicy rosnących ziaren, a zatem niewielką część wszystkich komórek. Każda komórka jest poddawana symulacjom jednokrotnie w całym cyklu obliczeń, a nie, jak dotychczas, wielokrotnie w każdym kroku czasowym. Zastosowanie frontalnych automatów wydawnie skraca czas obliczeń.

Przeprowadzono badania plastometryczne i mikrostrukturalne. Badania plastometryczne wybranych gatunków stali wykonano, przyjmując określone warunki odkształcenia, w tym z przerwami w procesie i ze zmiennymi parametrami odkształcenia. Do wyznaczenia związanych z rozwojem mikrostruktury parametrów procesu odkształcenia wykorzystano fenomenologiczny model naprężenia uplastyczniającego z zastosowaniem zmiennych wewnętrznych materiału. Na podstawie wyników badań plastometrycznych i obserwacji mikroskopowych za pomocą modelu fenomenologicznego przeprowadzono identyfikację parametrów procesu związanych z rozwojem gęstości dyslokacji, prędkością zarodkowania i prędkością rozrostu nowych ziaren. Parametry te zostały sprecyzowane w wyniku symulacji odkształcenia za pomocą automatów komórkowych.

Ich weryfikacja pozwoliła sformułować następujące wnioski:

- Gdy rozważanym parametrem jest wielkość ziarna, końcową mikrostrukturę po pełnej rekryształizacji można otrzymać z dużą dokładnością.
- Kinetyka rekryształizacji jest różna w zależności od tego, czy rozrost ziarna odbywa się podczas odkształcenia, czy po nim. Opierając się na wynikach doświadczeń i danych zawartych w literaturze przedmiotu, można uzyskać zbliżoną kinetykę rekryształizacji automatami komórkowymi. W efekcie zastosowania czynnika zależnego od prędkości odkształcenia model rekryształizacji został skorygowany ze względu na jej rodzaj.
- Zastosowany model naprężenia uplastyczniającego, oparty na średniej gęstości dyslokacji w całej objętości automatów komórkowych, przy symulacji procesów z rekryształizacją dynamiczną wykazuje istotne błędy. Zatem może on być wykorzystany jedynie jako przybliżony lub dostarczyć informacji o kinetyce rekryształizacji. Model ten należy dopracować i uwzględnić w nim parametry mikrostrukturalne, takie jak: udział w mikrostrukturze nowych ziaren i ich wpływ na spadek naprężenia uplastyczniającego oraz połączenie metody automatów komórkowych i metody elementów skończonych w skali mezo.

Omówiono dwa warianty takiego opracowania danych uzyskanych za pomocą metody elementów skończonych do wykorzystania w modelach automatów komórkowych. Opracowano również schemat bezpośredniego zastosowania automatów komórkowych do symulacji procesu walcowania bez użycia metody MES.

Pierwszy wariant zapewnia wykorzystanie w obliczeniach zarówno automatów komórkowych, jak i metody elementów skończonych na poziomie skali mezo. W przypadku spęczania trójwymiarowego obliczenia automatami komórkowymi przebiegają równolegle z obliczeniami za pomocą modelu metody elementów skończonych. Wówczas rozpatrywany jest sześciian reprezentujący niewielką objętość materiału z narzucenymi warunkami odkształcenia. Łącząc metodę automatów komórkowych z programem bazującym na metodzie elementów skończonych, przyjęto zasadę sztywnego połączenia siatki elementów skończonych z przestrzenią automatów komórkowych.

Wariant drugi polega na mniej ścisłym połączeniu automatów komórkowych i metody elementów skończonych (MES). W tym podejściu obliczenia MES są niezależne od wyników symulacji automatami komórkowymi, natomiast automaty komórkowe otrzymują z modelu MES niezbędną informację o parametrach procesu. Istnieje więc możliwość dokonywania obliczeń automatami komórkowymi nie tylko równolegle z obliczeniami MES, ale także po zakończeniu tych drugich. W przypadku walcowania trójwymiarowego dane uzyskane z MES zapisywane są do plików, które są następnie czytane przez model automatów komórkowych.

Opracowano dwa schematy przygotowania danych do symulacji rozwoju mikrostruktury podczas walcowania na gorąco. Pierwszy z nich nie wymaga wykorzystania metody elementów skończonych i może być stosowany do symulacji rozwoju mikrostruktury podczas walcowania wyrobów płaskich. Schemat drugi wymaga obliczeń metodą elementów skończonych i może być zastosowany równolegle z MES lub w obliczeniach po zakończeniu działania programu opartego na metodzie elementów skończonych.

Przeprowadzono symulacje procesu walcowania na gorąco. Do analizy zmian zachodzących podczas walcowania wyrobów płaskich wykorzystano plan gniotu. Wynikiem symulacji była mikrostruktura w dowolnej chwili całego procesu walcowania w klatce wykańczającej od pierwszego przepustu aż do przemiany austenitu w ferryt, z możliwością przedstawienia struktury ferrytycznej, a także wykresy zmian wielkości ziarna i naprężenia uplastyczniającego w każdym przepuszczeniu oraz kinetyki rekryształizacji statycznej w przerwach między przepustami. Wyniki te są zgodne z założeniami.

Do symulacji procesu walcowania w wykrojach wykorzystano wyniki modelowania tego procesu za pomocą metody elementów skończonych. W tym przypadku również uzyskano mikrostrukturę w dowolnej chwili procesu oraz wykresy zmian wielkości ziarna, naprężenia uplastyczniającego i kinetyki rekryształizacji.

Część trzecia

Praktyczny kurs automatów komórkowych

Część trzecia niniejszej pracy jest praktycznym kursem opracowywania i wykorzystywania automatów komórkowych oraz zasad działania frontalnych automatów komórkowych.

Pierwsze dwa ćwiczenia mają charakter wprowadzenia do jednowymiarowych automatów, demonstrują podstawowe cechy tej metody, stwarzają możliwość zapoznania się z jej głównymi zasadami oraz sposobem prowadzenia symulacji prostych zjawisk. Jednak niniejsze opracowanie nie dotyczy głównie automatów jednowymiarowych i dlatego ci czytelnicy, których interesuje przede wszystkim wprowadzenie do automatów pozwalających symulować zjawiska mikrostrukturalne, mogą je pominąć.

Pozostałe ćwiczenia związane są z automatami dwuwymiarowymi i pozwalają lepiej zrozumieć i zrealizować w praktyce większość zasad przedstawionych w części pierwszej niniejszej pracy. Automaty dwuwymiarowe zostały wybrane dlatego, że za ich pomocą znacznie łatwiej jest otrzymać wizualizację, są szybsze oraz z tego powodu, że potrzebują mniej pamięci niż automaty trójwymiarowe. Ponadto na świecie większość badań nadal prowadzona jest z wykorzystaniem automatów dwuwymiarowych, a rozszerzenie modeli dwuwymiarowych do trójwymiarowych można wykonać samodzielnie. Będzie to możliwe dzięki wiedzy zdobytej podczas wykonywania ćwiczeń z wykorzystaniem wskazówek i uwag przedstawionych w części pierwszej.

Poniżej przedstawiono w skrócie treść ćwiczeń.

W ćwiczeniu 1 przedstawiono zasady działania i główne cechy automatów: geometrię, otoczenie, stany, reguły, stan początkowy oraz sposób wizualizacji.

Ćwiczenie 2 umożliwia zapoznanie się z różnymi sposobami aktualizacji stanów i ocenę jej wpływu na wyniki symulacji.

Ćwiczenie 3 jest wprowadzeniem do dwuwymiarowych automatów komórkowych i jego celem jest wyznaczenie głównych cech tych automatów oraz umożliwia stworzenie najprostszego automatu komórkowego, który będzie wykorzystywany we wszystkich następnych ćwiczeniach. Zostanie w nim zasymulowany rozrost pojedynczego ziarna w kształcie kwadratu. Ćwiczenie to pozwala również zasymulować najprostsze figury wykorzystane w grze „Życie”.

Ćwiczenie 4 z kolei stwarza możliwość zapoznania się z różnymi rodzajami otoczenia. Na przykładzie rozrostu pojedynczego ziarna można zbadać wpływ na kształt

ziarna otoczenia von Neumanna, Moore'a, heksagonalnego, pseudoheksagonalnego, losowych i innych.

Ćwiczenie 5 pokazuje, jak można otrzymać przestrzeń izotropową. Jego wynikiem jest idealnie okrągły kształt rosnącego ziarna.

Ćwiczenie 6 ilustruje zasady działania frontального automatu komórkowego. Wynik ćwiczenia dowodzi, że zastosowane FCA prowadzi do wielokrotnego skrócenia czasu obliczeń.

Ćwiczenie 7 demonstruje zasady sterowania kształtem oraz przestrzenną orientacją rosnących ziaren na przykładzie pochylonego w stosunku do głównych osi przestrzeni prostokąta.

W ćwiczeniu 8 zaproponowano zamodelowanie początkowej mikrostruktury. Jako punkt wyjścia rozpatrywano zarodkowanie przed rozrostem. Podano też wskazówki dotyczące organizacji ciągłego zarodkowania podczas całego procesu kształtowania początkowej mikrostruktury.

Ćwiczenie 9 jest pierwszym z dwóch dotyczących warunków brzegowych. Rozpatrywane są w nim warunki periodyczne i periodyczne z przesunięciem. Poza tym w modelu uwzględniono stan odpowiadający lokalizacji komórki na granicy ziarna.

Ćwiczenie 10 wprowadza otwarte (nie półotwarte) warunki brzegowe, a dodatkowo pogłębia wiedzę czytelników na temat kroku czasowego.

Ćwiczenie 11 jest bardzo uproszczonym przykładem symulacji przemiany fazowej. Tworzona jest w nim początkowa mikrostruktura, wyznaczone są miejsca preferowanego zarodkowania (przyjęto ich lokalizację w narożach ziaren) oraz zasymulowany rozrost nowych ziaren.

Ćwiczenie laboratoryjne 1

Automaty komórkowe. Podstawowe pojęcia. Jednowymiarowe CA. Lokalne reguły

Automaty komórkowe (*Cellular Automata* – CA) – to model dyskretny stosowany w matematyce, teorii obliczeń, fizyce, biologii teoretycznej, mikromechanice i innych dziedzinach. CA jest strukturą obejmującą siatkę komórek, z których każda może się znajdować w jednym ze skończonego zbioru stanów. Siatka może mieć dowolny wymiar. Dla każdej komórki definiowany jest zestaw wielu komórek nazywany otoczeniem. Aby CA mogły działać, wymagane jest wyznaczenie początkowego stanu wszystkich komórek oraz reguł (zasad) przejścia tych komórek z jednego stanu do innego. W każdym kroku, stosując reguły przejścia i uwzględniając stany komórek sąsiednich, wyznacza się stan każdej komórki. Reguły przejścia są zazwyczaj takie same dla wszystkich komórek i stosuje się je bezpośrednio do całej siatki.

CA charakteryzuje się czterema elementami:

- 1) geometrią użytej przestrzeni,
- 2) lokalną regułą przejścia,
- 3) stanami komórki,
- 4) stanami otoczenia (sąsiedztwa) komórek.

Prawdopodobnie najprostszymi automatami komórkowymi są automaty jednowymiarowe (1D CA) z komórkami o dwóch stanach i otoczeniem składającym się z komórki i jej dwóch sąsiadów (z lewej i z prawej). Regułami i innymi zagadnieniami związanymi z automatami zajmiemy się nieco później, przy okazji opracowywania pierwszych automatów.

Rozpocniemy od stworzenia 1D CA. Jak wynika z samej ich nazwy, przestrzeń tych automatów jest jednowymiarowa i może być przedstawiona w postaci linii. Linia ta może być podzielona na równe odcinki, które nazywamy komórkami. Każdej komórce można przypisać numer lub współrzędną. Zazwyczaj są to liczby całkowite, przypisywane komórkom po kolei, sekwencyjnie. Graficznie, zamiast w postaci linii, 1D CA jest przedstawiany jako taśma z komórkami o kwadratowym kształcie. Jednakże przy demonstrowaniu działania 1D CA istnieje możliwość wykorzystania dodatkowego, drugiego wymiaru i przedstawienia ewolucji CA na płaszczyźnie. Powszechnie

przyjęło się prezentowanie 1D CA w poziomie. Po zastosowaniu lokalnych reguł przejścia nowy stan każdej z komórek jest ukazywany bezpośrednio pod stanem poprzednim i w ten sposób kolejny rząd będzie kolejnym przedstawieniem stanu automatów w kolejnym kroku obliczeniowym. Innymi słowy, w poziomie ukazywana jest oś przestrzeni, a pionowo – oś czasu.

Wobec tego budowę automatu komórkowego należy rozpocząć od stworzenia przestrzeni, w której będzie on działać. W tym celu najpierw należy stworzyć odpowiedni projekt w *Visual Studio* (na przykład w *QuickWin Application*).

Następnie wybieramy *Source Files* → *Add* → *New Item*. Następnie w okienku wybieramy *Source*, nadajemy nazwę i naciskając *Add*, tworzymy plik *Fortran File* (.f90). Będzie to główny plik, w którym zapiszemy cały program.

Tworzenie programu rozpoczynamy od standardowych wierszy:

```
Program *****; ! Wpisać nazwę programu
USE IFQWIN
implicit none

stop
END
```

Konieczne jest podłączenie modułu *IFQWIN*, ponieważ będziemy wykorzystywać jego funkcje do wprowadzania informacji na ekran monitora. Zalecane jest polecenie „**implicit none**”, ponieważ eliminuje ono błędy związane z omyłkowymi nazwami zmiennych.

Dalej program powinien stworzyć odpowiednie okno, w którym będą działały automaty komórkowe:

```
TYPE (QWINFO) winfo
INTEGER(4) result1, bcolor
LOGICAL(4) status

winfo%TYPE = QWIN$MAX
status = SETWSIZEQQ(QWIN$FRAMEWINDOW, winfo)
bcolor = GETCOLORRGB( )
result1 = SETCOLOR(15) ! white

OPEN (UNIT= 1, FILE= 'USER', TITLE= '1d CA')
CALL SETVIEWPORT (0, 0, nx, ny) ! W tym miejscu należy zadać wymiary okna
result1 = FLOODFILLRGB (1, 1, bcolor)

result1 = SETWSIZEQQ(1, winfo)
result1 = FOCUSQQ (1)
```

Teraz należy opracować algorytmy automatów komórkowych. Zakładamy, że komórka może się znajdować w dwóch stanach. Jeden, początkowy, oznaczymy jako „0”, i przypiszemy mu kolor biały, drugi stan oznaczymy symbolem „1” i będzie on przedstawiany w kolorze czarnym. Ponieważ nie będziemy brać pod uwagę dodatkowych informacji o komórkach i jedynym parametrem będzie stan komórki, będziemy używać koloru jako oznaki stanu, czyli stan będzie przyjmował wartość, która odpowiada kodowi danego koloru. W związku z tym do przechowywania informacji o stanie komórki będziemy wykorzystywać pamięć graficzną, bez wykorzystania pamięci podręcznej. Program do swojej pracy praktycznie nie będzie potrzebował pamięci.

Dlatego dodatkowo wykorzystamy zmienne:

```
INTEGER(2) state, x, y
```

i polecenia:

```
state = SETPIXEL(x, y)
state = GETPIXEL(x, y)
```

Pierwsze polecenie zmienia kolor piksela, który ma współrzędne x i y , a drugie odczytuje ten kolor z ekranu (pamięci graficznej). W pierwszym poleceniu zmienna *state* zachowuje kod poprzedniego koloru, w drugim – obecnego. Ponieważ na początku programu całe okno zostało wypełnione kolorem białym (kod 15), to w celu zmiany koloru na czarny należy wpisać odpowiedni kod (0):

```
state = SETCOLOR(INT2(0))
```

Warunki brzegowe. Istnieje kilka możliwości zadania warunków brzegowych. W naszym programie najprostszym rozwiązaniem będzie przyjęcie stałych zerowych warunków brzegowych, co oznacza, że komórki pierwsza i ostatnia zawsze będą się znajdowały w stanie oznaczonym kodem 15 (kolor biały). Osiągnąć to można przez niezmiennianie ich stanów i stosowanie reguł względem komórek, poczynając od drugiej do przedostatniej.

Kolejność zmiany stanów. Istnieją automaty synchroniczne i asynchroniczne. Proponowana struktura automatów jest synchroniczna ze względu na algorytm, który będzie w nich zastosowany. Kolejność aktualizacji stanów jest nieistotna i lepiej przyjmując sekwencyjną, zaczynając od drugiej komórki i kończąc na przedostatniej, albo na odwrót. Wówczas cykl

```
DO x = 2, nx-1
```

uwzględnia kolejność zmiany stanów i warunki brzegowe (nx – liczba komórek).

Warunki początkowe. Bardzo ważnym elementem wpływającym na ewolucję automatów komórkowych są warunki początkowe, czyli stan, w którym znajdują się wszystkie komórki przed pierwszą iteracją. Ze względów metodycznych początkowy stan komórki lepiej zdefiniować oddzielnym podprogramem. Wykorzystamy tu dwa

warianty warunków początkowych: z jedną czarną komórką w środku przestrzeni oraz z komórkami z losowo ustalonym stanem początkowym.

Otoczenie (sąsiedztwo). Otoczeniem komórki o numerze i będą trzy komórki z indeksami $i - 1$, i oraz $i + 1$. Jeżeli w automatach jednowymiarowych można zastosować niewiele różnych otoczeń, to już w automatach dwuwymiarowych istnieje ich kilka. Również do obsługi otoczenia można stworzyć oddzielny podprogram, ale w tej chwili nie jest to konieczne.

Lokalna reguła. Do lokalnych reguł też byłoby sensownym stworzenie podprogramu i rozdzielenie ich przez ten podprogram lub pozostawienie razem w jednym podprogramie.

Zastosujemy notację zaproponowaną przez Stephena Wolframa. Przy założeniu, że są trzy komórki, które mogą się znajdować w dwóch stanach, istnieje osiem możliwych kombinacji ich stanów, którym można przypisać wartości od 0 do 7, jeżeli przyjąć, że w trzycyfrowym dwójkowym zapisie stanów trzech komórek cyfra o wyższej wartości oznacza lewego sąsiada, a cyfra o wartości niższej – prawego. Wszystkie możliwe kombinacje stanów trzech komórek mogą być zatem uporządkowane według ich zapisu w systemie dwójkowo-dziesiątkowym i oznaczone cyframi od 7 do 0. Każdej takiej kombinacji odpowiada nowy stan środkowej komórki. Nowy stan komórki może być więc zapisany cyfrą 0 lub 1. Zapisując nowe stany w kolejności narzuconej przez stany wejściowe, otrzymujemy ciąg ośmiu cyfr w systemie dwójkowym, które można również przedstawić w systemie dziesiętnym.

W ten sposób otrzymamy lokalną regułę. Przykład reguły 30 pokazano w tabeli 1.

Tabela 1

Aktualny stan komórki	111	110	101	100	011	010	001	000
Nowy stan środkowej komórki	0	1	0	1	1	0	1	0

Uzyskana liczba (drugi wiersz tabeli) w systemie dwójkowym 01011010 jest równa 30. Stąd pochodzi nazwa reguły.

Zadanie 1

Stworzyć automat komórkowy z regułą 30 i zasymulować ewolucję tego automatu z jedną centralną komórką w stanie 1. Kolor biały odpowiada stanowi 0, czarny – stanowi 1.

Powtórzyć symulację z regułami 90 i 110.

Zadanie 2

Zasymulować ewolucję tego automatu z losowym rozkładem początkowego stanu komórek, wybierając 25%, 50% i 75% komórek w stanie 1 z regułą 184. Do uzyskania liczby losowej zastosować polecenie `call Random(z)`, przy czym z przyjmuje wartość od 0 do 1.

Ćwiczenie laboratoryjne 2

Jednowymiarowe CA.

CA synchroniczne i asynchroniczne

Kolejność zmiany stanów

Istnieją automaty synchroniczne i asynchroniczne.

Automaty komórkowe zazwyczaj traktują czas jako dyskretny i stany aktualizacji jako występujące synchronicznie. Stan wszystkich komórek jest aktualizowany jednocześnie, zanim którykolwiek z nowych stanów wpłynie na stan innych komórek. W przeciwieństwie do nich asynchroniczne automaty komórkowe aktualizują stany poszczególnych komórek niezależnie, w taki sposób, że nowy stan komórki od razu wpływa na obliczenie stanów w sąsiednich komórkach.

Synchronizacja zwykle odbywa się w dwóch fazach. W pierwszej oblicza się nowy stan każdej komórki na podstawie sąsiedztwa i lokalnej reguły. Nowa wartość stanu jest przechowywana w pamięci tymczasowej. W drugim etapie wartości stanu są aktualizowane przez implementowanie do komórek informacji o nowych stanach.

Natomiast do aktualizacji asynchronicznej w najprostszym przypadku (w pełni aktualizacja asynchroniczna) nie jest konieczne rozdzielenie tych dwóch faz, zmiany stanu są realizowane natychmiastowo.

O ile w automatach synchronicznych sposób realizacji tych dwóch faz nie wpływa na wyniki, w automatach asynchronicznych bardzo istotna jest również metoda wyboru komórki do aktualizacji. Tak więc schematy aktualizacji stanu komórek można sklasyfikować w następujący sposób:

- 1) Schemat synchroniczny – stany wszystkich komórek są aktualizowane równolegle w każdym kroku czasowym. Jest to model tradycyjny.
- 2) Asynchroniczny schemat uporządkowany – w każdym kroku czasowym wszystkie komórki są aktualizowane w zadanej kolejności.
- 3) Schemat cykliczny – w każdym kroku czasowym komórki są wybierane według ustalonej kolejności aktualizacji, która została wyznaczona losowo podczas inicjacji automatu.
- 4) Losowy schemat uporządkowany – wszystkie komórki w każdym kroku czasowym są aktualizowane, ale w kolejności losowej.

- 5) Schemat losowy niezależny – w każdym kroku czasowym losowo wybierana jest komórka do aktualizacji.
- 6) Schemat samotaktowany – każda komórka posiada niezależny zegar z ustawianymi losowo okresem i fazą. Po upływie okresu komórka jest aktualizowana, a zegar ustawiany ponownie. Aktualizacja jest autonomiczna i przebiega w różnym tempie w różnych komórkach.
- 7) Schemat samosynchronizujący się – podobny do schematu taktowanego, ale faza jest w nim objęta sprzężeniem z zegarami sąsiadów, a więc może być osiągnięta lokalna synchronizacja.

W przedstawionych poniżej ćwiczeniach zostaną stworzone automaty pozwalające na aktualizację stanu komórki czterema pierwszymi sposobami.

Automaty komórkowe stworzone w ćwiczeniu 1 są zaliczane do synchronicznych, ponieważ reguła przejścia wykorzystywała wyłącznie stany w poprzednim kroku. Dopóki wypełniała się kolejna linia, nowy stan komórek nie wpływał na komórki sąsiednie. W automatach tych trudno było oddzielić jedną fazę od drugiej. Jednak w większości przypadków komórki zawierają dodatkową informację i do jej odczytania nie wystarczy jedynie pamięć graficzna.

Głównym zadaniem będzie stworzenie automatów z zachowaniem informacji o komórkach w pamięci podręcznej i zbadanie ich zachowania w zależności od przyjętego sposobu aktualizacji.

Rozpoczynamy od utworzenia struktury komórki. Pewne informacje będą wykorzystywane we wszystkich algorytmach, inne tylko w niektórych. Tworzymy typ zmiennej, którą można nazwać *cell* (komórka) i która będzie zawierała pola: *state* (aktualny stan komórki) oraz *oldstate* (poprzedni stan komórki).

Teraz można stworzyć składające się z komórek automaty, wprowadzając na przykład zmienną *cells(n)*, gdzie *n* będzie liczbą komórek.

Spróbujmy powtórzyć działanie poprzedniego automatu komórkowego (nr 1), przyjmując pierwszy sposób aktualizacji stanu komórek. Należy oddzielić wyznaczenie nowego stanu od aktualizacji tego stanu.

Algorytm działania będzie następujący:

1. Zadać początkowy stan komórek, przyjmując taką samą wartość nowego stanu *state* i poprzedniego *oldstate* dla wszystkich komórek. Zmienić kolor pierwszej linii ekranu zgodnie ze stanem komórek.
2. Stosując odpowiednią regułę, wyznaczyć na podstawie stanu *oldstate* po kolei dla wszystkich komórek ich nowy stan *state*.
3. Skopiować nowy stan *state* do stanu poprzedniego *oldstate* dla wszystkich komórek. Zmienić kolor kolejnej linii ekranu zgodnie z nowym stanem komórek.
4. Wrócić do etapu 2.

Nowy stan można wyświetlać na etapie 2. lub 3.

Zadanie 1

Zasymulować działanie synchronicznych automatów komórkowych zgodnie z poprzednimi regułami (30, 90, 110, 184).

Tworzenie asynchronicznego automatu komórkowego nr 2 z uporządkowaną aktualizacją odbywa się następująco:

Modyfikujemy działanie algorytmu w następujący sposób:

1. Zadać początkowy stan komórek (zmienna *state*). Wyświetlić stan komórek na ekranie w pierwszej linii.
2. Stosując odpowiednią regułę na podstawie stanu *state* po kolei do wszystkich komórek, wyznaczyć i zapisać ich nowy stan *state*.
3. Wrócić do etapu 2.

Jest to asynchroniczny, uporządkowany sposób aktualizacji, który w zasadzie nigdy nie jest stosowany w praktyce.

Zadanie 2

Zasymulować działanie asynchronicznych automatów komórkowych (nr 2) zgodnie z regułami 30 i 90.

Tworzenie asynchronicznego automatu komórkowego nr 3 z cykliczną aktualizacją odbywa się następująco:

Cykliczną aktualizację można zrealizować na dwa sposoby: po pierwsze, za pomocą indeksowania komórek (listy indeksów), a po drugie – przez stworzenie listy komórek. W pierwszym przypadku tworzy się statyczną listę, która jest wypełniana losowo numerami komórek:

```
INTEGER CellNumber(n).
```

Wówczas organizowany jest cykl:

```
do i=1,n.
```

Komórki są wybierane zgodnie z kolejnością wyznaczoną przez przypisane im numery:

```
cells(CellNumber(i))
```

Drugi sposób różni się od wyżej omówionego jedynie tym, że tworzona jest w nim lista dynamiczna.

Przykład podprogramu do uzyskania losowej sekwencji jest podany na końcu ćwiczenia.

Zadanie 3

Zasymulować działanie asynchronicznych automatów komórkowych (nr 3) zgodnie z regułą 90.

Tworzenie asynchronicznego automatu komórkowego nr 4 z losowo uporządkowaną aktualizacją na każdym kroku odbywa się następująco:

Różnica w stosunku do poprzedniego automatu polega na tym, że w automacie nr 3 w każdym kroku czasowym stosowana jest ta sama losowa kolejność, natomiast w automacie nr 4 kolejność losowa jest inna w każdym kroku czasowym.

Wówczas podprogram do wyznaczenia kolejności będzie wywoływany nie jednokrotnie przed całym cyklem obliczeń, a wielokrotnie, na początku każdego kroku czasowego.

Zadanie 4

Zasymulować działanie asynchronicznych automatów komórkowych (nr 4) zgodnie z regułą 90.

Podprogram do uzyskania losowej sekwencji. Podprogram tworzy losową sekwencję w postaci listy n elementów, z których każdy zawiera jeden z n numerów i może wskazywać na jeden z k elementów innej uporządkowanej listy. Przy czym każdy numer n jest podawany jednokrotnie. Warunkiem prawidłowej pracy automatu jest $n \leq k$. W niniejszych ćwiczeniach przyjęto $k = n$.

```
Subroutine RandomSequence(ind1,n,k)
implicit none
Real ran
Integer(4) i, j, k, l, m, n, ind1(n)
Integer(4) ind2(k), ind3(n), ind4(k), ind5(n), ind6(k)

1 do i=1,n
    call Random(ran); ind1(i)=ran*k+1.; if (ind1(i)>k)
        ind1(i)=k;
enddo
do i=1,k
    ind2(i)=0;
enddo
2 do i=1,n
    if (ind2(ind1(i))==0) then
        ind2(ind1(i))=1;
    else
        ind1(i)=0;
    endif
enddo
3 j=0
do i=1,n
    if (ind1(i)==0) then
        j=j+1; ind3(j)=i;
    endif
```

```

enddo
if (j==0) return
m=j; j=0;
4 do i=1,k
    if (ind2(i)==0) then
        j=j+1; ind4(j)=i;
    endif
    ind6(i)=0;
enddo
l=j
5 do i=1,m
    call Random(ran); ind5(i)=ran*1+1.;
    if (ind5(i)>1) ind5(i)=1;
enddo
6 do i=1,m
    if (ind6(ind5(i))==0) then
        ind6(ind5(i))=1; ind1(ind3(i))=ind4(ind5(i));
ind2(ind4(ind5(i)))=1;
    endif
enddo
goto 3
end subroutine RandomSequence;

```


Ćwiczenie laboratoryjne 3

Dwuwymiarowe CA

Dwuwymiarowe automaty komórkowe różnią się od jednowymiarowych nie tylko liczbą wymiarów, ale również istotnym wzrostem skali złożoności i nowymi możliwościami stwarzanymi przez bogatsze otoczenie, reguły lokalne, warunki brzegowe itp. Z drugiej strony nie ma w nich możliwości graficznego przedstawienia ewolucji w czasie lub znacznie się ta kwestia komplikuje. Tego rodzaju zmiany w automatach dwuwymiarowych można przedstawić tylko jako zestaw obrazków lub za pomocą animacji.

Powtórzmy teraz główne etapy tworzenia jednowymiarowych automatów komórkowych, a także automatów dwuwymiarowych:

- tworzenie przestrzeni modelowej z wizualizacją,
- utworzenie komórek z zestawem zmiennych wewnętrznych i zewnętrznych (w tym stanu komórki),
- określenie warunków brzegowych,
- zadanie warunków początkowych,
- ustalenie kolejności zmiany stanów, aktualizacja,
- warunek zatrzymania,
- otoczenie,
- lokalna reguła.

Kolejność wprowadzenia elementów (a zatem kolejność powyższych etapów) nie jest sztywna i można ją zmieniać według uznania, ale żaden z nich nie powinien zostać pominięty. Spróbujmy stworzyć automat komórkowy, przyjmując zaprezentowaną powyżej kolejność poszczególnych etapów.

Przestrzeń modelowa z wizualizacją

Możemy wykorzystać te same polecenia co w dwóch poprzednich ćwiczeniach:

```
USE IFQWIN
implicit none
TYPE (QWINFO) winfo
```

```

INTEGER(4) result1, bcolor
LOGICAL(4) status
INTEGER(2) state

winfo%TYPE = QWIN$MAX
status =      SETWSIZEQQ(QWIN$FRAMEWINDOW, winfo)
bcolor = GETCOLORRGB( )
result1 = SETCOLOR(15) ! white

OPEN (UNIT= 1, FILE= 'USER', TITLE= '1d CA')
CALL SETVIEWPORT (0, 0, nx, ny)
result1 = FLOODFILLRGB (1, 1, bcolor)

result1      = SETWSIZEQQ(1, winfo)
result1 = FOCUSQQ (1)

```

Wizualizacja będzie zrealizowana przez bezpośredni zapis do pamięci graficznej:

```

result1 = SETCOLOR(c) ! c - wyznacza kolor, c=0 - czarny, c=15 - biały
state = SETPIXEL(x, y)

```

Komórki

Zestaw zmiennych zależy od tego, jakie zjawisko będzie modelowane, od lokalnych reguł, sposobu aktualizacji stanów i innych czynników. Jednak w tym zestawie zawsze znajduje się stan komórki. Na początku wprowadzamy dwie zmienne, *state* i *oldstate*, które przyjmują dwie wartości – 0 i 1. W ten sposób utworzymy, jak poprzednio, typ zmiennej, który odpowiada komórce, oraz wypełnimy przestrzeń komórkami:

```

TYPE cell
    byte state, oldstate
END TYPE cell;

TYPE (cell), dimension (:,:), pointer :: cells
INTEGER nx, ny ! rozmiary przestrzeni komórkowej
INTEGER i, j ! indeksy wskazujące na położenie komórki w przestrzeni

allocate (cells(nx,ny))

```

Warunki brzegowe

Na początku będziemy wykorzystywać otwarte warunki brzegowe (stałe zerowe). Oznacza to, iż automaty komórkowe są otoczone komórkami w stanie zerowym. Program, kiedy potrzebna jest informacja o stanie komórek spoza obszaru ($i < 1$, $i > nx$, $j < 1$ lub $j > ny$), przyjmuje stan tych komórek równy 0 lub pomija te komórki. Do warunków brzegowych można wrócić po ustaleniu otoczenia i reguły lokalnej.

Warunki początkowe

Zamodelujemy rozrost pojedynczego ziarna. Dlatego wszystkie komórki, poza jedną, centralną, będą w stanie zerowym, a komórka centralna – w stanie 1:

```
state = SETCOLOR(INT2(0)) ! black
do i=1,nx
  do j=1,ny
    cells(i,j).oldstate=0; cells(i,j).state=0;
  enddo
enddo
cells(nx/2,ny/2).oldstate=1; state = SETPIXEL(nx/2,ny/2);
```

Aktualizacja stanów

Jako pierwszy, wstępny wariant aktualizacji wybieramy sposób synchroniczny. Dlatego typ zmiennej *cell*, jak poprzednio, zawiera dwa pola: *state* i *oldstate*. W pierwszej fazie będzie wyznaczany stan, a w drugiej – prowadzona jego aktualizacja.

Następnie należy zaprogramować dwa cykle, w których będą rozpatrywane wszystkie komórki po kolei. W pierwszym zostanie zastosowana reguła lokalna, która na podstawie informacji o stanie komórki (*oldstate*) będzie wyznaczała nowy stan komórki (*state*). W drugim cyklu następuje aktualizacja (*oldstate = state*) i wyświetlanie wyników, na przykład tak, jak pokazano poniżej:

```
do i=1,nx
  do j=1,ny
    ! Reguła lokalna
  enddo
enddo

do i=1,nx
  do j=1,ny
    cells(i,j).oldstate=cells(i,j).state;
    state = SETPIXEL(nx/2,ny/2);
  enddo
enddo
```

Warunek zatrzymania

Omówione w poprzednim punkcie obliczenia automatami komórkowymi nazywamy iteracją lub krokiem czasowym, który należy powtórzyć wielokrotnie. Wprowadzając długość kroku czasowego i czas modelowania, można zaprogramować cykl, w którym będzie stosowana reguła lokalna i aktualizacja stanów:

```
dTime=1; FinalTime=500;
do Time=0,FinalTime, dTime
  ! ← Reguła lokalna+Aktualizacja
enddo
```

W tym przykładzie warunkiem zakończenia obliczeń będzie wykonanie ustalonej liczby kroków (501).

Otoczenie

Na początku będziemy wykorzystywać dwa rodzaje otoczenia: ortogonalne, które nosi nazwę otoczenia von Neumanna, i mające postać kwadratu jednostkowego (sąsiedztwo Moore'a). Na rysunku 1 ukazano komórkę i jej otoczenie Moore'a (osiem komórek lub dziewięć, jeśli uwzględnić komórkę rozpatrywaną). Otoczenie von Neumanna zawiera cztery (lub pięć) komórek oznaczonych numerami 2, 4, 5 i 7.

6	7	8
4		5
1	2	3

Rys. 1. Komórka i jej otoczenie

Zaprogramować badanie otoczenia można na kilka sposobów, na przykład wyznaczając je wprost (otoczenie von Neumanna):

```
Subroutine NeiNeumann(i,j, ni1,nj1)
ni1(1)=i-1; nj1(1)=j;
ni1(2)=i; nj1(2)=j-1; itd
```

programując cykl (otoczenie Moore'a):

```
Subroutine NeiMoore(i,j,ni2,nj2)
k=1
do j1=-1,1
do i1=-1,1
if ((i1.ne.0).and(j1.ne.0)) then
ni2(k)=i+i1; nj2(k)=j+j1; k=k+1
endif
enddo
enddo
```

lub przygotowując tablicę i stosując operacje tablicowe (otoczenie Moore'a):

```
Integer*4 ndx1(8),ndxn(8)
Integer*4 ndy1(8),ndyn(8)
```

```

ndx1=0; ndx1(1)=-1;ndx1(4)=-1;ndx1(6)=-1;
ndxn=0; ndxn(3)=1; ndxn(5)=1; ndxn(8)=1
ndy1=0; ndy1(1)=-1;ndy1(2)=-1;ndy1(3)=-1;
ndyn=0; ndyn(6)=1; ndyn(7)=1; ndyn(8)=1;

```

a potem wyznaczając otoczenie:

```

ni2=ndx1+ndxn+i
nj2=ndy1+ndyn+j

```

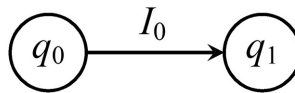
Każdy z tych sposobów może być zastosowany również w odniesieniu do automatów trójwymiarowych.

Lokalna reguła

Zastosujemy najprostszy algorytm (rys. 2).

Regułę przejścia można określić jednym zdaniem: komórka zmienia swój stan (przechodzi ze stanu 0 w stan 1) wtedy i tylko wtedy, gdy w jej otoczeniu znajduje się przynajmniej jedna komórka w stanie 1. W otoczeniu Moore'a regułę tę można zapisać w następujący sposób:

$$I_0 = c_1 \vee c_2 \vee c_3 \vee c_4 \vee c_5 \vee c_6 \vee c_7 \vee c_8 \quad (1)$$



Rys. 2. Automat komórkowy dla algorytmu rozrostu ziaren

Realizacja tej reguły może wyglądać następująco:

```

call NeiMoore(i,j,ni2,nj2,ndx1,ndxn,ndy1,ndyn) ! Moore
neighborhood
do n=1,8
  if
    (ni2(n).gt.0.and.ni2(n).le.nx.and.nj2(n).gt.0.and.nj2(n).le.ny)
  then
    if (cells(ni2(n),nj2(n)).oldstate==1) then !local rule
      cells(i,j).state=1; state
      = SETPIXEL(int2(i),int2(j)); goto 1
    endif
  endif
enddo
1 continue

```

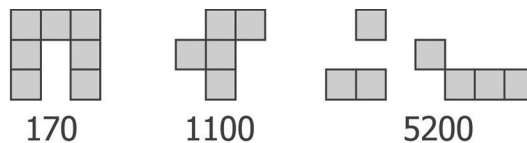
Pozostaje tylko przetestować program, usunąć możliwe błędy i uruchomić go, obserwując ziarna w kształcie kwadratu.

Opcjonalne zadanie dla zaawansowanych

Gra „Życie” może być opisana następującą regułą lokalną:

```
! Life
nz=0
do n=1,8
  if
    (ni2(n).gt.0.and.ni2(n).le.nx.and.nj2(n).gt.0.and.nj2(n).le.ny)
  then
    if (cells(ni2(n),nj2(n)).oldstate==1) then
      nz=nz+1
    endif
  endif
enddo
if (nz<2.or.nz>3) then
  if (cells(i,j).oldstate==1) then
    cells(i,j).state=0; state = SETCOLOR(INT2(15));
    state = SETPIXEL(int2(i),int2(j));
  endif
else
  if ((nz==3).and.(cells(i,j).oldstate==0)) then
    cells(i,j).state=1; state = SETCOLOR(INT2(0));
    state = SETPIXEL(int2(i),int2(j));
  else
    cells(i,j).state=cells(i,j).oldstate;
  endif
endif
endif
```

Najprostsze ciekawe początkowe konfiguracje nazywane π , r i *acorn* z liczbą iteracji koniecznych do osiągnięcia stanu ustalonego pokazano na rysunku 3.



Rys. 3. Początkowe konfiguracje gry „Życie”

Ćwiczenie laboratoryjne 4

Otoczenie (sąsiedztwo)

Wynikiem ćwiczenia laboratoryjnego 3 były automaty komórkowe rozrostu pojedynczego ziarna w kształcie kwadratu.

W tym ćwiczeniu będziemy badali różne warianty otoczenia w odniesieniu do rozrostu pojedynczego ziarna. Przed rozpoczęciem pracy zmodyfikujemy nasz projekt.

Po pierwsze, ponieważ w Fortranie podprogramy działają ze zmiennymi lokalnymi, wprowadzimy zmienne globalne przez podłączenie odpowiedniego modułu ze zmiennymi i inicjalizującym podprogramem. Tworzymy nowy plik i przenosimy do niego część zmiennych oraz część kodu.

Plik rozpoczyna się od deklaracji:

```
MODULE {Nazwa modulu};
```

Następnie wprowadza się do niego typy zmiennych i zmienne. W tej chwili do takiego modułu możemy przenieść z programu opracowanego w poprzednich ćwiczeniach typ *cell* oraz zmienne *state*, *cells*, *i*, *j*, *nx*, *ny*, *ni(8)*, *nj(8)*, *Time* i *FinalTime*, *dTime*.

Dalej wprowadzamy do modułu sekcję, która będzie zawierała podprogram, i zamykamy go:

```
CONTAINS
!*****
subroutine InitCA;

end subroutine InitCA;
!*****
end MODULE Global;
```

W podprogramie zakodowano inicjalizację okna graficznego (s. 302–303) oraz wprowadzenie zerowych warunków początkowych dla wszystkich komórek (s. 304). Nie można zapomnieć o wywołaniu tego podprogramu w odpowiednim miejscu programu głównego.

Teraz będziemy włączać utworzony moduł do głównego programu i podprogramów:

```
USE {Nazwa modulu};
```

Po wykonaniu wszystkich czynności uruchamiamy program i sprawdzamy, czy działa on prawidłowo. Po wprowadzeniu kolejnego rodzaju sąsiedztwa też należy uruchomić program, sprawdzić jego działanie oraz obserwować wyniki na ekranie.

W celu ułatwienia pracy z różnymi wariantami otoczenia sensowne jest wydzielenie podprogramu obsługującego wszystkie otoczenia w oddzielnym pliku. Obecnie zaprogramowano w nim tylko jeden wariant otoczenia (Moore'a).

Proponuje się utworzenie jednego podprogramu, który będzie uwzględniał wszystkie otoczenia wybierane według numerów. Wówczas należałoby przekazywać dwa parametry niezbędne do pracy podprogramu oraz pobierać wyniki do wykorzystania w innych podprogramach. Będą to numer otoczenia i liczba znajdujących się w nim komórek. Można te dwa parametry wprowadzić jako zmienne globalne lub jako parametry przekazywane przy wywołaniu:

```
Integer NeighborhoodNumber, NumberN;
```

Wybór otoczenia w podprogramie można zrealizować na przykład za pomocą operacji **goto**:

```
goto (1,2,3...) NeighborhoodNumber ! wybor otoczenia
```

Otoczenie Moore'a i von Neumanna wprowadzamy w następujący sposób:

```
1 NumberN=8; ! Moore
ni(1)=i-1; ni(2)=i; ni(3)=i+1; ni(4)=i-1;
ni(5)=i+1; ni(6)=i-1; ni(7)=i; ni(8)=i+1;
nj(1)=j-1; nj(2)=j-1; nj(3)=j-1; nj(4)=j;
nj(5)=j; nj(6)=j+1; nj(7)=j+1; nj(8)=j+1;
return
```

```
2 NumberN=4; ! von Neumann
ni(1)=i-1; ni(2)=i; ni(3)=i; ni(4)=i+1;
nj(1)=j; nj(2)=j-1; nj(3)=j+1; nj(4)=j;
return
```

Teraz wprowadzamy trzeci rodzaj otoczenia – heksagonalne, które można zaprogramować na kilka sposobów. Wykorzystamy jeden z nich, jednak należy pamiętać, że możliwe do zastosowania są w tym przypadku również inne warianty. Ten sposób wykorzystania otoczenia heksagonalnego symetrycznego polega na tym, że dla parzystych (lub nieparzystych) wierszy (lub kolumny) stosujemy otoczenie Moore'a, dla pozostałych – otoczenie von Neumanna. Tak samo symetryczne otoczenie „pentagonalne” będzie się składało z jednego otoczenia Moore'a na trzy otoczenia von Neumanna, a „heptagonalne” – z trzech otoczeń Moore'a na jedno otoczenie von Neumanna.

Otoczenie niesymetryczne można wprowadzać stopniowo, co omówiono wyżej w odniesieniu do wariantów otoczenia Moore’a i von Neumanna. W taki sam sposób można wprowadzić również otoczenia symetryczne.

Rozpatrzmy jeszcze jeden wariant otoczenia – pseudoheksagonalne. Polega ono na naprzemiennym stosowaniu otoczenia Moore’a i von Neumanna, lecz nie przestrzennie, a czasowo. Zatem w iteracjach nieparzystych wykorzystuje się jedno otoczenie, a w parzystych – drugie. W taki sam sposób można zastosować i inne symetryczne pseudo-otoczenia (pseudopentagonalne i inne). Wyglądać to może w następujący sposób:

```
7 continue; ! pseudo-heksagonalne
if (mod(Time,2*dTime)<dTime) goto 1
goto 2
```

Stosując zamiast 2 (niekoniecznie musi to być liczba całkowita) w podstawie moduła ($2*dTime$) warunku `if` oraz zamieniając miejscami `goto 1` i `goto 2`, można zmieniać proporcje między długością boków ośmiokąta.

Ostatni wariant, który tu rozpatrzymy, pozwoli uzyskać prostokątny kształt rosnącego ziarna. Jest on zbliżony do pseudootoczeń. Jeżeli będzie to na przykład „poziomy” prostokąt za stosunkiem boków 3:1, to dwaj sąsiedzi w poziomie (w kierunku osi x), będą wybierani w każdym kroku, a wszyscy pozostali (otoczenie Moore’a) raz na trzy iteracje:

```
if (mod(Time,3*dTime)<dTime)
```

Jeżeli w tym wariantcie zastosowane zostanie otoczenie von Neumanna, wynikiem będzie poziomo wydłużony sześciokąt.

Otoczenie z elementami losowymi

Zmienna losowa potrzebna do wyznaczenia otoczenia może być wprowadzona na kilka sposobów. Pierwszy, najłatwiejszy, polega na tym, że w odniesieniu do wybranej komórki badamy otoczenie lub go nie badamy w zależności od wylosowanej liczby. Drugi sprowadza się do losowego wyboru dowolnego z opracowanych otoczeń. Trzeci polega na losowym wyborze każdego z możliwych sąsiadów komórki.

Wykorzystanie pierwszego sposobu w naszym podprogramie będzie polegało na wyznaczeniu wartości zmiennej losowej `call Random(ran)` i porównaniu jej z wartością ustaloną (na przykład 0,1). Jeżeli będzie ona mniejsza, to opuszczamy podprogram i nie badamy otoczenia ($NumberN = 0$), w pozostałych przypadkach wybieramy otoczenie zgodnie z przypisanym mu numerem.

W drugim sposobie należy uwzględnić dwa aspekty. Pierwszy to zestaw otoczeń (minimum dwa, maksimum wszystkie opracowane). Jeśli wybierzemy dwa otoczenia, to niech to będą otoczenia główne (von Neumanna i Moore’a). Drugi aspekt to prawdopodobieństwo występowania każdego z otoczeń. Zmieniając ten parametr,

można przechodzić od otoczenia Moore’a do otoczenia von Neumanna przez dowolne otoczenia przejściowe. Będzie to najpierw „poziomy” kwadrat (o bokach skierowanych wzdłuż głównych osi przestrzeni) z „zaokrąglonymi” kątami, potem nieco zniekształcony ośmiokąt o różnym stosunku boków i na koniec „przekątny” kwadrat (o przekątnych skierowanych wzdłuż osi przestrzeni) z „zaokrąglonymi” kątami. Faktycznie będą to takie same kształty, jakie można uzyskać za pomocą symetrycznych pseudootoczeń, różnica polega tylko na tym, iż w pseudootoczeniach granice są proste, a kąty ostre, natomiast przy zastosowaniu zmiennych losowych granice są zniekształcone, a kąty zaokrąglone.

W trzecim wariancie każdy z sąsiadów jest wylosowany. Sens takiego wyboru polega na tym, iż można przyjąć, że stosunek prawdopodobieństw wylosowania każdego sąsiada jest proporcjonalny do prędkości rozrostu w kierunku wyznaczonym przez tego sąsiada. W ten sposób można zmieniać kształt i, w niewielkim stopniu, orientację głównych osi ziarna. Część takiego symetrycznego otoczenia pokazano poniżej:

```
10 NumberN=0;
call Random(ran)
if (ran<0.9) then ! wzdłuż osi x
    ni (NumberN+1)=i-1; ni (NumberN+2)=i+1;
    nj (NumberN+1)=j; nj (NumberN+2)=j;
    NumberN=NumberN+2
endif
```

Na przykład można zadać prawdopodobieństwo wylosowania sąsiadów ortogonalnych i po przekątnych (0,9; 0,3; 0; 0) i (0,1; 0,1; 0,9; 0) lub dowolnego innego zestawu.

Ćwiczenie laboratoryjne 5

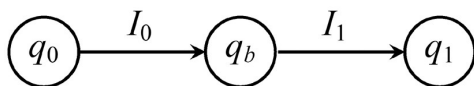
Sterowanie prędkością rozrostu. Izotropia przestrzeni

Podsumowując zastosowane w poprzednich ćwiczeniach algorytmy rozrostu ziarna, możemy stwierdzić, że mają pewne zalety, pozwalające uzyskać okrągły kształt bądź też kształt bardziej lub mniej wydłużonej lub obróconej o praktycznie dowolny kąt elipsy. Jednak algorytmy te mają poważne wady, do których można zaliczyć brak gładkiej granicy, niemożliwość uzyskania dowolnego kształtu i ograniczoną możliwość sterowania prędkością rozrostu.

Najpierw należy zauważyć, że sterowanie prędkością rozrostu rozumiemy jako możliwość wprowadzenia zależności względnej prędkości ruchu granicy od kierunku rozrostu, a nie jest to wyznaczenie odpowiedniości prędkości rozrostu symulowanego do rzeczywistej jego prędkości. Zatem chodzi tu o zadanie i zrealizowanie odpowiednich zależności, które pozwolą dokładnie sterować kształtem i orientacją ziaren.

Rozpocząć należy od modyfikacji samego automatu komórki. Ponieważ tworzone automaty zalicza się do automatów synchronicznych, zostały wprowadzone dwie zmienne odpowiadające za stan komórki. Teraz stworzymy automaty asynchroniczne, które jednak zachowują wszystkie cechy automatów synchronicznych. Pozostawiamy tylko jedną zmienną i wprowadzamy dla niej trzeci stan, przejściowy.

Wówczas każda komórka może przebywać w jednym z trzech stanów: początkowym, przejściowym lub końcowym. Schematycznie przedstawiono to na rysunku 4. W programach autora stan przejściowy oznaczony jest numerem 4 ($state = q_p = 4$).



Rys. 4. Automat komórki oparty na algorytmie rozrostu ziaren z dwoma podstawowymi stanami i stanem przejściowym q_p

Modyfikacja algorytmu i programu polega na tym, że w pierwszej fazie (wyznaczenie nowego stanu) stan przejściowy oznacza, iż w poprzednim kroku komórka znajdowała się w stanie 0. Natomiast w drugiej fazie (aktualizacji stanu) stan przejściowy automatycznie zmienia się w stan końcowy 1.

Wprowadzamy odpowiednie zmiany do programu i sprawdzamy jego działanie:

1. Przy inicjowaniu nadajemy stan początkowy zmiennej *state*, a nie *oldstate*.
2. Sprawdzamy, czy sąsiednia komórka znajduje się w stanie 1 (*state*, a nie *oldstate*) i jeżeli jest w tym stanie, zmieniamy stan rozpatrywanej komórki na przejściowy. Oczywiście nie sprawdzamy otoczenia komórek, które już są w stanie końcowym.
3. Zamiast kopiowania stanu *state* do *oldstate* zmieniamy tylko stan komórek, które są w stanie przejściowym, na stan końcowy.

Poprzedni synchroniczny schemat aktualizacji zawierał drugą fazę jako element konieczny, niezbędny do prawidłowej pracy automatów komórkowych. Nowy schemat stwarza całkiem inne możliwości. Niewskazane jest przecież wprowadzanie aktualizacji od razu po przejściu w stan przejściowy. Przejście to można opóźnić lub przeprowadzić dokładnie w wyznaczonym czasie. A jeżeli tak, to dlaczego nie uzależnić prędkość rozrostu od dowolnego parametru lub nie skonstruować automatów mniej czułych na zastosowane otoczenie?

W otoczeniu von Neumanna są cztery kierunki rozrostu, w otoczeniu Moore'a – osiem. Kierunki te ograniczają możliwość uzyskania dowolnego kształtu ziaren i dowolnej ich orientacji. Zatem nie można było otrzymać idealnie okrągłego kształtu lub na przykład kształtu kwadratu albo prostokąta obróconego o 10° lub 15° , a tylko o 45° lub nieobróconego. W proponowanym układzie to również nie będą idealne kształty, lecz ograniczenie owo sprowadza się jedynie do poziomu rozdzielczości zależnej od wymiarów komórki.

Przystąpimy teraz do opracowania programu do obliczeń rozrostu okrągłego ziarna. Algorytm będzie się składał z dwóch etapów: wyznaczenia kierunku rozrostu oraz wyznaczenia czasu opóźnienia.

Do wyznaczenia kierunku rozrostu zastosujemy najprostszy algorytm, który będzie polegał na wyznaczeniu wektora od punktu, w którym powstał zarodek ziarna, do rozpatrywanej komórki. Algorytm ten ma bardzo ograniczone zastosowanie, ale w tym przypadku można go użyć. Do obliczeń niezbędna jest znajomość współrzędnych dwóch punktów – jednym z nich jest punkt, którego współrzędne znamy, natomiast współrzędne drugiego punktu powinniśmy określić za pomocą dodatkowych zmiennych.

Ponieważ obliczenia nie zawsze będą ograniczone do jednego ziarna i współrzędne nie będą jedynym parametrem go dotyczącym, sensowne będzie stworzenie odpowiedniego typu *Grain* i zmiennej *Grains* (podobnie do *cell* i *cells*) i umieszczenie ich w module globalnych zmiennych. Pierwszymi parametrami *Grain* będą współrzędne zarodka.

Typ *cell* wzbogacamy zmienną *GrainNumber*, która będzie wskazywała na numer ziarna, do którego należy komórka.

Oczywiście podczas inicjowania programu lepiej przypisać wszystkie komórki do fikcyjnego, nieistniejącego ziarna oznaczonego numerem 0.

Jedyną niezerową komórkę należy przypisać do pierwszego utworzonego ziarna.

Po tym przygotowaniu tworzymy nowy plik, który będzie odpowiadał za opracowywany podprogram. Nazwiemy go na przykład *GrainGrowthRate*.

Pierwszy etap działania programu będzie polegał na wyznaczeniu jednostkowego wektora kierunkowego \mathbf{r} :

$$l = \sqrt{(i_c - i_g)^2 + (j_c - j_g)^2}, \quad r_x = \frac{i_c - i_g}{l}, \quad r_y = \frac{j_c - j_g}{l},$$

gdzie:

- l – odległość rozpatrywanej komórki od miejsca zarodkowania ziarna,
- r_x, r_y – składowe wektora \mathbf{r} ,
- i_c, j_c – współrzędne rozpatrywanej komórki,
- i_g, j_g – współrzędne zarodka.

Drugi etap działania programu to wyznaczenie opóźnienia i czasu przejścia komórki w stan końcowy. Należy wprowadzić do typu zmiennej *cell* dodatkowy parametr *time*, który będzie wskazywał na chwilę przejścia w stan końcowy.

Granica ziarna powinna przejść przez komórkę, więc przy założeniu wymiarów jednostkowych granica powinna pokonać odległość równą $|r_x| + |r_y|$, co przy jednostkowej prędkości rozrostu da czas opóźnienia $\tau = |r_x| + |r_y|$. Jednak opóźnienie to będzie równe dokładnie $\tau = l_c - l_n$, gdzie l_c – odległość danej komórki od zarodka, l_n – odległość od zarodka komórki, której rozrost ziarna dana komórka naśladuje (dla lepszego odwzwiedlenia okrągłego kształtu ziarna należy nieco wydłużyć czas τ , mnożąc jego wartość przez parametr nieco większy od 1, na przykład 1,1).

Pozostaje dodać to opóźnienie do chwili przejścia w stan końcowy komórki, której rozrost ziarna dana komórka naśladuje, w ten sposób wyznaczymy chwilę przejścia w stan końcowy danej komórki.

Wróćmy do programu głównego – w miejscu, gdzie zaprogramowano przejście komórki w stan przejściowy, należy dodać podprogram obliczania chwili przejścia w stan końcowy opisany wyżej, z uwzględnieniem faktu, że komórka ta powinna się znajdować w stanie początkowym, oraz należy przyłączyć daną komórkę do ziarna.

Natomiast przejście ze stanu przejściowego do stanu końcowego uzależniamy od czasu:

```
if (cells(i,j).time<=Time) cells(i,j).state=1;
```

Na koniec wprowadźmy zmiany, które będą potrzebne w kolejnym ćwiczeniu. Po pierwsze, zmieniamy warunek zakończenia obliczeń: zamiast oznaczonej zadawanej liczby kroków programujemy koniec obliczeń na moment, kiedy wszystkie komórki przejdą w stan końcowy. Tak więc zamiast cyklu *do* wykorzystamy cykl *while*. Wprowadzimy dwie zmienne *ncells* i *icells*, które będą oznaczały liczbę wszystkich komórek ($ncells = nx*ny$) i liczbę komórek w stanie końcowym. Zatem przejście każdej komórki w stan końcowy będzie zwiększało wartość licznika *icells* o 1. Również

wewnątrz cyklu `do while` należy wprowadzić zmianę czasu (który był podstawą cyklu `do`). Sprawdzamy, czy program działa poprawnie.

Po drugie, należy wprowadzić pomiar czasu działania całego programu lub jego części dotyczącej rozrostu ziarna. Wprowadzamy zmienną i przeprowadzamy odpowiednie operacje:

```
real*4 timecalc
INTEGER*4 result1

timecalc = SECNDS(0.0) ! włączenie zegarka

timecalc = SECNDS(timecalc) ! pomiar czasu
result1 = FOCUSQQ (0) ! przełączenie aktywnego okna
write(*,*)'Obliczenia trwały «,timecalc, 'sekund.'
! wyświetlenie wyniku
```

Zatem włączamy zegar przed głównym cyklem, mierzymy czas trwania cyklu i wyświetlamy wynik.

Ćwiczenie laboratoryjne 6

Frontalne automaty komórkowe

W tym ćwiczeniu zajmiemy się przyspieszeniem obliczeń automatami komórkowymi. Przejdziemy od klasycznych automatów komórkowych do frontalnych i porównamy wyniki.

Zadajemy jak największą przestrzeń (tak dużą, na ile pozwala ekran monitora albo przynajmniej 800×800 komórek (pikseli)), uruchamiamy opracowany w poprzednim ćwiczeniu program i mierzymy czas. Powtarzamy ten cykl kilkakrotnie i obliczamy z uzyskanych wyników wartość średnią.

Wyłączamy wizualizację na ekranie procesu rozrostu ziaren i powtarzamy obliczenia. Jeżeli mamy do dyspozycji kilkuprocesorowy komputer lub kilkurdzeniowy procesor, różnica prawdopodobnie będzie minimalna.

Uzyskane wyniki będą stanowiły dla nas punkt odniesienia. Ten wariant programu należy zachować, żeby można było do niego wrócić i przeprowadzić obliczenia z wykorzystaniem innych danych.

Główny cykl opracowanego poprzednio programu zawiera dwa podcykle, w których następuje „skanowanie” przestrzeni i sprawdzenie każdej komórki po kolei. W pierwszym podcyklu ma miejsce badanie otoczenia komórki, w drugim – zmiana jej stanu. Pierwszy podcykl powinien w zasadzie dotyczyć wyłącznie komórek w stanie zerowym, więc można nie badać sąsiedztwa komórek w innym stanie, ale to nie zapewnia znaczącego przyspieszenia obliczeń (do tego zagadnienia wrócimy później). Drugi podcykl dotyczy tylko komórek w stanie przejściowym, natomiast skanowana jest cała przestrzeń. Za punkt wyjścia naszego rozumowania przyjmiemy następującą kwestię: co by było, gdybyśmy zamiast skanowania po kolei wszystkich komórki mieli listę komórek w stanie przejściowym? Oczywiście w każdym cyklu można by było na tej operacji zaoszczędzić wiele czasu, dlatego też powstała idea utworzenia listy komórek w stanie przejściowym. Wówczas w pierwszym cyklu będziemy wprowadzać do tej listy elementy, natomiast w drugim będziemy ją opróżniać.

Istnieje kilka sposobów zorganizowania list. Zastosujemy wariant najprostszy (choć nie najlepszy).

Utworzymy typ struktury *CellList*, który będzie zawierał trzy zmienne wewnętrzne: *n* – liczbę elementów listy, *FirstCell* – pierwszy element listy, *LastCell* – ostatni element

listy. Wszystkie te zmienne są liczbami całkowitymi. Tworzymy zmienną typu *CellList* z nazwą, na przykład *L(0:4)* (pięć list zawierających komórki w stanie od 0 do 4).

Rozszerzamy strukturę typu *Cell* o dwie zmienne całkowite: *PrevCell* – poprzedni element listy i *NextCell* – następny element.

Teraz należy napisać podprogramy dodawania elementów na koniec listy i usunięcia elementu z dowolnego miejsca listy, nie zapominając o inicjowaniu listy na początku programu. Pusta lista będzie zawierała $n = 0$, $FirstCell = 0$, $LastCell = 0$. Wskazywać na daną komórkę można, podając jej numer (lub współrzędne). Numer komórki może być związany ze współrzędnymi poprzez wzory: $CellNumber = (j - 1) * nx + i$, $j = \text{int}((CellNumber - 1) / nx) + 1$, $i = \text{mod}(CellNumber - 1, nx) + 1$. *CellNumber* to numer komórki podawany zmiennymi *FirstCell*, *LastCell*, *PrevCell* i *NextCell*. Najpierw wszystkie komórki możemy wprowadzić do listy *L(0)*:

```
Subroutine TransferList(i1,j1)
USE Global;
implicit none
Integer*4 i1, j1
Integer*4 ii, jj
Integer*4 PrevCell, NextCell

PrevCell=Cells(i1,j1).PrevCell
NextCell=Cells(i1,j1).NextCell

if (NextCell==0) then ! ostatni element listy Lout
  L(Lout).LastCell= PrevCell;
else
  ii=mod(NextCell-1,nx)+1; jj=int((NextCell-1)/nx)+1;
  Cells(ii,jj).PrevCell= PrevCell;
endif

if (PrevCell==0) then ! pierwszy element listy Lout
  L(Lout).FirstCell= NextCell;
else
  ii=mod(PrevCell-1,nx)+1; jj=int((PrevCell-1)/nx)+1;
  Cells(ii,jj).NextCell= NextCell;
endif

L(Lout).n=L(Lout).n-1

CellNumber=(j1-1)*nx+i1
if (L(Lin).FirstCell>0) then ! nie pierwszy element listy
  ii= mod(L(Lin).LastCell-1,nx)+1; jj= int((L(Lin).LastCell-1)/
    nx)+1
  Cells(ii,jj).NextCell= CellNumber
  Cells(i1,j1).PrevCell=L(Lin).LastCell
```



```

Else ! pierwszy element listy
    Cells(i1,j1).PrevCell=0
    L(Lin).FirstCell= CellNumber
endif
Cells(i1,j1).NextCell=0;
L(Lin).n=L(Lin).n+1
L(Lin).LastCell= CellNumber
Cells(i1,j1).state=Lin

CellNumber = NextCell;

return
end subroutine TransferList

```

Po przygotowaniu podprogramu powinniśmy wywołać go w dwóch miejscach programu głównego – przy zmianie stanu z początkowego na przejściowy, dodając w pierwszym cyklu:

```
Lout=0;Lin=4;call TransferList(i,j);
```

oraz zmieniając drugi cykl:

```

CellNumber=L(4).FirstCell
do while (CellNumber.ne.0)
    i=mod(CellNumber-1,nx)+1; j=int((CellNumber-1)/nx)+1;
    if (cells(i,j).time<=Time) then
        icells=icells+1;
        Lout=4;Lin=1;call TransferList(i,j)
    else
        CellNumber=Cells(i,j).NextCell;
    endif
enddo

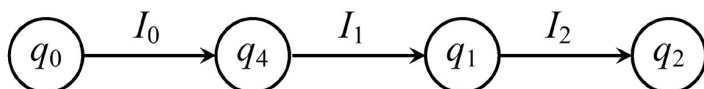
```

Należy również zwrócić uwagę na to, że pierwszej komórce, w której powstał zarodek, należałoby nadać stan przejściowy, a zmienne *Lout* i *Lin* dodać do zmiennych globalnych.

Po wprowadzeniu zmian i sprawdzeniu działania programu można porównać czas obliczeń – zauważymy wówczas, że przyspieszenie jest niewielkie. Związane jest to z faktem, że drugi cykl był kilkakrotnie krótszy od pierwszego (nie wymagał badania otoczenia). Można by zastosować podobną metodę jak w drugim cyklu, czyli nie badać całej przestrzeni, lecz tylko komórki w stanie początkowym. Powinno to doprowadzić do niemal dwukrotnego skrócenia czasu (na początku będą badane prawie wszystkie komórki i ich otoczenie, a pod koniec tylko ich niewielka część).

Istotą frontalnych automatów komórkowych jest właśnie to, że nie trzeba badać wszystkich komórek w stanie początkowym, ale tylko te przylegające do komórek, w których zachodzą zmiany, innymi słowy – usytuowane przy granicy rozrostu ziarna (froncie zmian).

Aby zastosować metodę frontalnych automatów komórkowych, zmodyfikujemy nasz automat, dodając jeszcze jeden stan, tak jak to pokazano na rysunku 5. Od tej chwili będziemy nazywać stany w następujący sposób (przy czym indeks będzie odpowiadał wartości liczbowej stanu komórki $q_i = i$): q_0 – stan początkowy, q_4 – stan przejściowy, q_1 – stan frontalny i q_2 – stan końcowy.



Rys. 5. Frontalny automat komórki

Pierwsza zmiana, którą można wprowadzić w programie, to dodanie jeszcze jednego cyklu zmieniającego stan komórki ze stanu 1 na stan 2. Będzie to cykl podobny do cyklu drugiego, ale bez żadnych warunków (żeby program działał, należy w pierwszym cyklu zmienić warunek przejścia ze sprawdzaniem stanu końcowego z 1 na 2, ale nie jest to konieczne w tej chwili).

Teraz komórki w stanie frontalnym istnieją tylko po wykonaniu drugiego cyklu, do końca cyklu trzeciego, a wszystkie inne komórki tylko w krótkiej chwili między końcem drugiego i początkiem trzeciego cyklu. Zastosujmy teraz regułę lokalną względem komórek, które będą zmieniały stan w kolejnych krokach. Tworzymy dodatkowy cykl, w którym wskaźnik bieżącej komórki (*CellNumber*) wskazuje po kolei na wszystkie elementy listy $L(1)$. Faktycznie powinniśmy nim zastąpić pierwszy cykl.

Porównajmy te dwa cykle. W pierwszym skanowana jest cała przestrzeń, w drugim, dodatkowym – tylko komórki na ruchomej granicy ziarna, których liczba jest małym ułamkiem liczby wszystkich komórek. W pierwszym cyklu badane jest otoczenie w celu sprawdzenia, czy znajduje się w nim komórka w stanie końcowym, której stan należy zmienić, czyli badane jest otoczenie wejściowe. W cyklu dodatkowym komórka znajduje się już prawie w stanie końcowym, więc bada się otoczenie wyjściowe w celu znalezienia komórek w stanie początkowym, aby przekazać im informację i zmienić ich stan na przejściowy. Ponieważ zmienia się kierunek przekazywania informacji, zmienia się otoczenie komórek z wejściowego na wyjściowe i z listy do listy należy przenosić nie samą komórkę, lecz jej sąsiada.

Ten cykl będzie wyglądał następująco:

```

CellNumber=L(1).FirstCell
do while (CellNumber.ne.0)
  i=mod(CellNumber-1,nx)+1; j=int((CellNumber-1)/nx)+1;

```

```

        NeighborhoodNumber=1; call Neighborhood
    do n=1,NumberN
        if (ni(n).gt.0.and.ni(n).le.nx.and.nj(n).gt.0.and.nj(n).le.ny)
    then
        if (cells(ni(n),nj(n)).state==0) then
            cells(ni(n),nj(n)).GrainNumber=cells(i,j).GrainNumber;
            call GrainGrowthRate;
            state = SETPIXEL(int2(ni(n)),int2(nj(n)));
            Lout=0;Lin=4;call TransferList(ni(n),nj(n))
        endif
    endif
    enddo
    CellNumber=Cells(i,j).NextCell;
enddo

```

Natomiast skorygowany podprogram obliczenia czasu będzie wyglądał tak:

```

Subroutine GrainGrowthRate
USE Global;
implicit none
Real*8 l1, l0

l0=sqrt((i-Grains(cells(i,j).GrainNumber).i)**2.+(j-Grains
(cells(i,j).GrainNumber).j)**2.)
l1=sqrt((ni(n)-Grains(cells(i,j).GrainNumber).i)**2.+(nj(n)-
Grains(cells(i,j).GrainNumber).j)**2.)
cells(ni(n),nj(n)).time=cells(i,j).time+(l1-l0)*1.1;

return
end subroutine GrainGrowthRate

```

Po wprowadzeniu zmian uruchamiamy program i mierzymy czas obliczeń. Przyspieszenie będzie prawdopodobnie kilkakrotne.

Teraz wyłączamy wyświetlanie na ekran procesu rozrostu i uruchamiamy program ponownie, by przyjrzeć się efektom wprowadzonych zmian.

Ćwiczenie laboratoryjne 7

Kształt ziaren. Prostokąt

Efekt zastosowania frontalnych automatów, w szczególności kiedy uzyskany wynik był całkowicie nieoczekiwany, powinien stanowić dla wszystkich przekonujący argument za tym, by używać właśnie automatów frontalnych, a nie klasycznych. Co prawda należy zaznaczyć, że algorytm frontalnych automatów nie może być stosowany we wszystkich przypadkach, a w niektórych trzeba go poddać pewnym modyfikacjom, jednak na potrzeby modelowania zjawiska, którym będziemy się zajmować w ćwiczeniach, może być wykorzystany bez ograniczeń.

Wdrożenie izotropii przestrzeni, której najlepszym testem był rozrost ziarna uzyskującego kształt okręgu, otwiera nieograniczone możliwości sterowania kształtem rosnących ziaren. Można zadać rozrost ziarna w kształcie prostokąta, rombu lub elipsy o dowolnym stosunku boków, przekątnych lub półosi, trójk-, sześć- i ośmiokąta, przy czym orientacja tych kształtów w przestrzeni może być dowolna.

Tak więc kolejne ćwiczenie będzie dotyczyło rozrostu prostokąta obróconego o 15° .

Rozpocniemy od obrotu. Ponieważ łatwiej jest wykonać obliczenia w układzie współrzędnych związanym z rozpatrywanym ziarnem, wprowadzamy układ lokalny, różny od układu globalnego związanego z przestrzenią komórkową. Wówczas można powiedzieć, że układ lokalny jest obracany wokół punktu zarodkowania ziarna w układzie globalnym o wyznaczony kąt Θ .

Wektor \mathbf{r} od punktu zarodkowania ziarna do dowolnego punktu w układzie lokalnym i globalnym reprezentowany jest odpowiednio przez wektory \mathbf{r}_l i \mathbf{r}_g :

$$\begin{aligned}\mathbf{r}_l &= (x_l, y_l)^T = x_l \mathbf{i}_l + y_l \mathbf{j}_l; \\ \mathbf{r}_g &= (x_g, y_g)^T = x_g \mathbf{i}_g + y_g \mathbf{j}_g\end{aligned}\tag{1}$$

Owe dwa wektory są związane następującym równaniem:

$$\mathbf{r}_l = \mathbf{R}_{g,l} \mathbf{r}_g\tag{2}$$

We wzorze (2) $\mathbf{R}_{g,l}$ oznacza macierz obrotu:

$$\mathbf{R}_{g,l} = \begin{bmatrix} \mathbf{i}_g \mathbf{i}_l & \mathbf{j}_g \mathbf{i}_l \\ \mathbf{i}_g \mathbf{j}_l & \mathbf{j}_g \mathbf{j}_l \end{bmatrix} \quad (3)$$

Zatem jeżeli lokalny układ obrócono o kąt θ wokół osi z (prostopadłej do przestrzeni automatów komórkowych), macierz $\mathbf{R}_{g,l}$ tego przekształcenia będzie wyrażone wzorem:

$$\mathbf{R}_{g,l} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (4)$$

Faktycznie, wiersze macierzy $\mathbf{R}_{g,l}$ są równoznaczne z wersorami \mathbf{i}_l i \mathbf{j}_l układu lokalnego zapisanymi w układzie globalnym. Podczas inicjacji powstawania zarodka ziarna powinniśmy zapisać tę macierz jako stały parametr tego ziarna. Oznacza to, że rozszerzamy strukturę *Grain* o macierz *Rgl(2,2)*.

Kolejną czynnością będzie utworzenie podprogramu odpowiadającego za zarodkowanie (*Nuclea*), który przygotuje wstępną informację dla ziarna i będzie bardzo pomocny w przyszłości, w czasie tworzenia wielu ziaren. Jednak jest to jedynie opcja; wrócimy do niej przy okazji wykonywania kolejnych ćwiczeń.

Formujemy macierz *Rgl(2,2)* zgodnie ze wzorem (4), uwzględniając $\theta = 15^\circ$.

Wprowadzamy dodatkową zmienną *GrainKind*, która będzie wyznaczała kształt rosnącego ziarna.

Przechodzimy do podprogramu *GrainGrowthRate*, za pomocą którego definiujemy selektor, który w zależności od wymaganego kształtu ziarna będzie wybierał odpowiednią gałąź podprogramu. W pierwszej pozostawiamy okrąg, drugą rezerwujemy dla prostokąta.

Najpierw zadajemy stosunek boków prostokąta, na przykład 2:1:

```
side1 = 2; side2 = 1;
```

Obliczamy wektor \mathbf{r} w układzie globalnym do komórki, której czas przejścia w stan końcowy będziemy obliczać:

```
rgx=ni(n)-Grains(cells(i,j).GrainNumber).i;  
rgy=nj(n)-Grains(cells(i,j).GrainNumber).j;
```

Następnie obliczamy współrzędne lokalne:

```
rlx=Rgl(1,1)*rgx+Rgl(1,2)*rgy;  
rly=Rgl(2,1)*rgx+Rgl(2,2)*rgy;
```

Wyznaczamy, na którym z boków prostokąta będzie się znajdowała komórka. Jeżeli $\left| \frac{rlx}{rly} \right| < \frac{side1}{side2}$, będzie ona należała do boku oznaczonego *side1*; kierunek rozrostu

$\mathbf{r}_r = (r_x, r_y)$ jest prostopadły do boku prostokąta, więc będzie równy wektorowi lokalnemu \mathbf{j} : $\mathbf{r}_r = \mathbf{j} = (\mathbf{R}_{g,l}(2,1), \mathbf{R}_{g,l}(2,2))$. W innym wypadku komórka będzie należała do boku oznaczonego *side2* i kierunek rozrostu będzie równy wektorowi lokalnemu \mathbf{i} : $\mathbf{r}_r = \mathbf{i} = (\mathbf{R}_{g,l}(1,1), \mathbf{R}_{g,l}(1,2))$.

W kierunku \mathbf{i} prędkość rozrostu będzie zależała wprost proporcjonalnie od długości boku *side1*, w kierunku \mathbf{j} – od długości boku *side2*. Aby nie spowodowało to, że czas opóźnienia będzie zbyt krótki, krótszy niż krok czasowy, należy znormalizować prędkość rozrostu: $v_i = \text{side1}/\max(\text{side1}, \text{side2})$ i $v_j = \text{side2}/\max(\text{side1}, \text{side2})$.

Czas opóźnienia będzie równy czasowi niezbędnemu do przejścia granicy przez komórkę. W przypadku komórek, które naśladują rozrost ziarna komórek znajdujących się na tym samym boku prostokąta, ten czas obliczano jako: $\tau = l/v$, gdzie l – rzut wektora \mathbf{r}_b od jednej komórki do drugiej (od sąsiada, którego rozrost ziarna jest naśladowany): $l = |\mathbf{r}_r \cdot \mathbf{r}_b|$. Jednak sytuacja komplikuje się, kiedy rozpatrywana i sąsiednia komórka znajdują się na różnych bokach prostokąta. Konieczna jest wówczas dodatkowa analiza, dlatego też zastosujemy łatwiejszą metodę, która jednak nie jest odpowiednia dla każdego przypadku.

Ustalając, na którym boku prostokąta znajduje się komórka, wyznaczamy, która ze współrzędnych będzie charakteryzowała poszukiwaną granicę ziarna. Współrzędna lokalna *rlx* lub *rly* odpowiadać będzie nowej granicy ziarna. Ponieważ zakładamy, że ziarno rośnie ze stałą prędkością, czas osiągnięcia komórki przez granicę ziarna będzie wyznaczany jako stosunek odpowiedniej lokalnej współrzędnej (*rlx* lub *rly*) do właściwej prędkości rozrostu (v_i lub v_j).

Wpisujemy do programu odpowiednie operacje.

Teraz program jest gotowy i można go uruchomić. Jeżeli wszystko działa poprawnie i rozrasta się ziarno w kształcie prostokąta, można spróbować zmieniać stosunek długości jego boków (w tym uwzględnić sytuację, w której będą one równe, jak w kwadracie) oraz kąt jego pochylenia. Analizę kształtu łatwiej przeprowadzić, zatrzymując pracę programu po wypełnieniu ziarnem 15–20% całej przestrzeni.

Ćwiczenie laboratoryjne 8

Początkowa mikrostruktura

Celem poniższych ćwiczeń będzie opracowanie programu, który pozwoli na utworzenie mikrostruktury. Algorytm takiego programu składa się z dwóch części: zarodkowania i rozrostu ziarna. Program w obecnej chwili pozwala na symulację rozrostu pojedynczego ziarna w kształcie okręgu lub prostokąta z dowolnym stosunkiem boków (w tym kwadratu) i o dowolnej orientacji przestrzennej. Zarodkowanie było zadawane wprost w głównym programie lub mogło być sterowane w oddzielnym podprogramie jak w poprzednim ćwiczeniu.

Wróćmy do podprogramu zarodkowania (*nuclea*) i wyznaczmy, jakie główne elementy on zawiera i jakie czynności powinien wykonać.

1. Informacją wejściową będą współrzędne i parametry ziarna (jego kształt, orientacja przestrzenna, stosunek boków prostokąta). Parametry można zadać w głównym programie lub w podprogramie. Mogą być one zadane wspólne dla wszystkich ziaren lub losowo dla każdego z osobna. Zaleca się wpisać na początku wspólne parametry dla wszystkich. Wówczas faktycznie jedyną wykorzystywaną informacją będą współrzędne zarodka.
2. Na podstawie informacji o współrzędnych należy sprawdzić, czy nie ma już w tej pozycji innego ziarna, a w przypadku stwierdzenia jego obecności należy wyjść z podprogramu.
3. Zwiększyć wskazania licznika ziaren (na przykład zmienna *NumGrains*). Wskazania te będą również numerem utworzonego ziarna.
4. Wpisujemy informacje o ziarnie: współrzędne zarodka i macierz obrotu. Jeżeli zakładamy, że ziarna będą miały różny kształt, to dodajemy odpowiednie zmienne (typ ziarna, stosunek boków), a jeżeli zakładamy, że takie same ziarna będą się znajdować w całej przestrzeni, nie należy rozszerzać struktury *Grain*.
5. W komórce o zadanych współrzędnych zapisujemy numer ziarna, zmieniamy stan na przejściowy (4) i przenosimy ją z listy 0 do listy 4 oraz wpisujemy bieżący czas aktualizacji komórki (*cells(i,j).time = Time*).
6. W podprogramie *GrainGrowthRate* sprawdzamy i zmieniamy wszystko, co dotyczy informacji o ziarnie (na przykład, *Rgl*).

Teraz można uruchomić program i sprawdzić, czy działa poprawnie.

Ponieważ w układzie będzie powstawało nie jedno ziarno, a kilka, trzeba je rozróżnić. Najłatwiej to zrobić, przypisując inny kolor każdemu ziarnu o danym numerze. Wówczas przed operacją:

```
state = SETPIXEL(int2(ii),int2(ii));
```

należy zaprogramować operację:

```
state = mod(NumGrains,15)+1; state = SETCOLOR(state)
```

Pozostaje zaprogramować podprogram, który będzie symulował powstawanie zarodków w czasie. Pierwszy wariant zarodkowania jest realizowany przed rozrostem. Wówczas program zarodkowania jest uruchomiany jednokrotnie na początku programu głównego.

Po uruchomieniu podprogramu (*nucleation*) wybieramy losowo ustaloną liczbę komórek, w których powstaną zarodki. Programujemy więc cykl, powtarzany tyle razy, ile zadano ziaren. W cyklu tym najpierw losujemy komórkę, a potem wywołujemy podprogram *nuclea*. Musimy pamiętać, żeby lista ziaren (*grains*) była wystarczająco długa.

Na początku głównego cyklu programu głównego wywołujemy podprogram zarodkowania *nucleation*. Uruchamiamy go i sprawdzamy jego działanie. W efekcie powinniśmy uzyskać mikrostrukturę.

Drugi wariant to zarodkowanie podczas rozrostu. Powtarzamy cykl z zarodkowaniem co kilka kroków czasowych (na przykład po każdym 20–30 krokach czasowych).

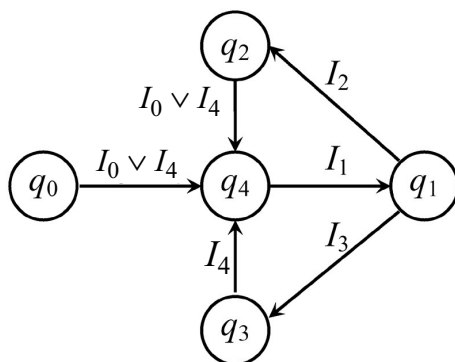
Ćwiczenie laboratoryjne 9

Granice ziaren. Warunki brzegowe

Celem poniższych ćwiczeń jest uwzględnienie w symulacjach dodatkowego stanu wskazującego na znajdowanie się komórki na granicy ziarna, a także wprowadzenie periodycznych warunków brzegowych.

Wprowadzenie dodatkowego stanu, który wskazuje na to, że komórka znajduje się na granicy ziarna, można uzasadnić przynajmniej dwoma przyczynami. Pierwsza, mniej istotna, to fakt, że pozwala to podkreślić na obrazie granice, w szczególności, kiedy sąsiadujące ze sobą ziarna mają ten sam kolor lub gdy struktura jest wizualizowana w barwach czarno-białych – wówczas ziarna są białe, a granice czarne. Druga przyczyna, ważniejsza, sprowadza się do tego, że w dalszych symulacjach można wykorzystać taką informację i przyspieszyć obliczenia. Na przykład jeżeli uzyskana struktura będzie wykorzystana do modelowania rekrytalizacji lub przemiany fazowej, to zarodkowanie w tych procesach będzie się odbywać wyłącznie na granicach ziaren. W tej chwili wykorzystamy tylko pierwszą ze wspomnianych zalet.

Modyfikujemy automat komórkowy w sposób pokazany na rysunku 6.



Rys. 6. Automat komórkowy

W poprzednio omówionym automacie struktura była liniowa, jednokierunkowa, z jednym stanem końcowym q_2 . Teraz będą istniały dwa stany końcowe: na granicy

ziarna q_2 i wewnątrz ziarna q_3 . Jeżeli przedtem przenoszenie komórek ze stanu frontального q_1 do końcowego q_2 odbywało się automatycznie, bezwarunkowo, pod koniec kroku czasowego, to teraz należy wprowadzić warunki I_2 i I_3 ($I_3 = \bar{I}_2$), w zależności od spełnienia których komórka będzie w stanie q_2 lub q_3 . Takim warunkiem będzie obecność w otoczeniu von Neumanna przynajmniej jednej komórki należącej do innego ziarna.

W ostatnim cyklu przed zmianą stanu komórki należy zbadać otoczenie von Neumanna, wybrać, jaki będzie nowy stan komórek; w przypadku komórek w stanie „na granicy ziarna” należy zmienić ich kolor na czarny. Wprowadzamy te zmiany do programu i sprawdzamy jego działanie.

Następnie należy wprowadzić periodyczne warunki brzegowe. Oznacza to, że przestrzeń komórkowa nie będzie otwarta, ze stałymi zerowymi warunkami brzegowymi, lecz zamknięta. Lewa jej granica będzie połączona z prawą, a górna z dolną. Warunki te nie były wprowadzane w oddzielnym podprogramie, a zazwyczaj zadawano je jako modyfikację otoczenia. Ponieważ operujemy głównie otoczeniem Moore’a i von Neumanna, to zmodyfikujemy je z uwzględnieniem nowych warunków brzegowych.

Na początku należy wprowadzić dodatkową zmienną globalną, która będzie wyznaczała, które warunki brzegowe zostaną zastosowane. Niech to będą zmienne logiczne lx i ly , każda z nich będzie odpowiadała za jedną współrzędną (czyli warunki mogą być mieszane: w jednym kierunku periodyczne, w drugim – otwarte). Niech „true” odpowiada warunkom periodycznym, a „false” – otwartym.

Zwykle stosuje się jedno z dwóch podejść. W pierwszym sprawdza się, czy współrzędna nie wskazuje na zakres zmiennej spoza obszaru przestrzeni (na przykład dla i : $i < 1$ lub $i = 0$, oraz $i > nx$ lub $i = nx + 1$). W razie spełnienia tego warunku współrzędna owa jest zmieniana. Zatem zamiast $i = 0$ przyjmuje się $i = nx$, a zamiast $i = nx + 1 - i = 1$. Tak samo postępuje się w odniesieniu do drugiej współrzędnej.

W drugim podejściu w ogóle nie sprawdza się warunków brzegowych, a dla każdej komórki stosuje się ten sam wzór, który już uwzględnił periodyczny warunek brzegowy. Tak więc zamiast i lub j przyjęto $\text{mod}(i - 1, nx) + 1$ lub $\text{mod}(j - 1, ny) + 1$. Wówczas pożądaný wynik jest uzyskiwany automatycznie.

Należy wybrać jedną z tych metod i zmodyfikować podprogram wyznaczający otoczenie komórki. Po wprowadzeniu zmian trzeba go uruchomić i sprawdzić jego działanie podczas symulacji powstania początkowej mikrostruktury.

Uwaga: Wprowadzenie periodycznych warunków brzegowych wymaga zmian w sposobie obliczania wektora rozrostu w podprogramie *GrainGrowthRate*. Jeżeli wektor ten jest wyznaczany wprost, jako różnica współrzędnych komórki i zarodka ziarna, będzie on obliczany niepoprawnie w przypadku, kiedy zarodek i rozpatrywana komórka leżą na różnych brzegach przestrzeni. Należy go wówczas wyznaczać przyrostowo, jak pokazano na rysunku 2.14: $\mathbf{r}_b = \mathbf{r}_p + \mathbf{r}_{bp}$, przy czym ostatni wektor trzeba wyznaczać nie w układzie globalnym, lecz przez sąsiedztwo, lokalnie. Oznacza to, że komórka znajdująca się na prawym brzegu ($i = nx$) ma prawego sąsiada na lewym brzegu ($i = 1$). W układzie globalnym składowa wektora będzie równa $dx = 1 - nx$, a przy zastosowaniu metody przyrostowej $dx = 1$.

Jeżeli wartość dx nie została w ten sposób wyznaczona w czasie wykonywania poprzednich ćwiczeń, należy określić ją teraz.

1. Trzeba się upewnić, czy struktura *cell* zawiera współrzędne wektora rozrostu rx i ry , a jeżeli nie, dodać je.
2. Należy sprawdzić, czy przy zarodkowaniu komórka ta otrzymuje współrzędne wektora rozrostu równe zeru.
3. Należy sprawdzić, czy przy określeniu otoczenia jest wyznaczany wektor od komórki do sąsiada. Jeżeli nie, należy najpierw wyznaczyć ten wektor, następnie wyznaczyć na jego podstawie współrzędne sąsiada, a później uwzględnić warunki brzegowe.
4. W podprogramie *GrainGrothRate* należy najpierw wyznaczyć wektor rozrostu i jego długość dla aktualnie rozpatrywanej komórki metodą przyrostową, a następnie wyznaczyć czas przejścia komórki w nowy stan.

Jeżeli wszystkie powyżej omówione elementy zostały uwzględnione, program powinien działać poprawnie.

Dodatkowe zadanie

Oprócz prostych periodycznych warunków brzegowych istnieją periodyczne warunki brzegowe z przesunięciem. W związku z tym przeciwległe brzegi przestrzeni nie mają dokładnie tych samych współrzędnych, lecz łączone są z przesunięciem. Na przykład znajdująca się na prawym brzegu komórka o współrzędnych (nx, j) ma za prawego sąsiada nie komórkę ze współrzędnymi $(1, j)$, jak w przypadku zadania periodycznych warunków brzegowych, a ze współrzędnymi $(1, j + dxy)$, gdzie dxy jest przesunięciem. Dodatkowym zadaniem jest wprowadzenie jednego takiego warunku ze stałym przesunięciem dla wszystkich komórek, które jest równe połowie długości przestrzeni $dxy = ny/2$.

Ćwiczenie laboratoryjne 10

Otwarte warunki brzegowe. Krok czasowy

Celem zaprezentowanych poniżej ćwiczeń jest wprowadzenie otwartych warunków brzegowych, a także uwzględnienie warunków rzeczywistych: wymiarów przestrzeni, prędkości rozrostu, czasu rzeczywistego oraz doboru optymalnego kroku czasowego.

Otwarte warunki brzegowe, które zostaną opisane poniżej, różnią się od przedstawionych w ćwiczeniu 9, ponieważ poprzednie warunki były otwarte ze stałym, zerowym warunkiem brzegowym. Przy zerowym warunku granica przestrzeni może być nazwana pochłaniającą. Również warunki periodyczne często są nazywane przenikającymi lub zamkniętymi. Przewaga warunków periodycznych nad otwartymi polega na tym, że wszystkie komórki przestrzeni są równoważne, nierozróżnialne, identyczne, co, po pierwsze, rzutuje na prawidłowe odzwierciedlenie kinetyki zachodzących procesów, a po drugie – niweluje wpływ granic, chociaż mikrostruktura z powtarzającym się wzorcem na krawędziach wygląda nieco sztucznie.

W pełni otwarte warunki brzegowe różnią się od periodycznych tym, że przeciwległe brzegi przestrzeni nie są połączone, natomiast ziarna nie tylko rosną na zewnątrz, ale wrastają do przestrzeni automatów z zewnątrz. Takie warunki tworzą zatem wycinek przestrzeni, zachowując zalety periodycznych warunków brzegowych. Kiedy ziarno dorasta do granicy, nadal będzie rosnąć, ale już nie będziemy mieć o tym procesie informacji. Aby skompensować nieuwzględnienie części objętości ziarna znajdującej się na zewnątrz przestrzeni komórkowej, w innym miejscu granicy przestrzeni wrasta do niej inne ziarno – niekoniecznie na przeciwległym brzegu i niekoniecznie w tej samej chwili.

Na tej podstawie można sformułować zasady działania odpowiedniego algorytmu:

1. Wyjście na brzeg ziarna jest zdarzeniem jednokrotnym, które powinno zostać zarejestrowane. Dla każdego ziarna należy wprowadzić po jednej zmiennej odpowiadającej za dojście do brzegu (na przykład l_{bx} , l_{by}). Zmienne te przy zwykłym zarodkowaniu ziarna przyjmują wartość „true”, a przy docieraniu do brzegu – wartość „false”.
2. Gdy pierwsza komórka rosnącego ziarna próbuje przejść przez brzeg przestrzeni, odpowiednia zmienna l_{bx} lub l_{by} przyjmując wartość „false” przełącza się i wywołuje podprogram zewnętrznego zarodkowania.

3. Zewnętrzne zarodkowanie różni się od zwykłego tym, że zarodek znajduje się poza obszarem automatów komórkowych, natomiast samo ziarno wrasta do przestrzeni z zewnątrz. Druga różnica polega na tym, że fikcyjnym zarodkiem jest komórka na brzegu przestrzeni (dowolnym brzegu wolnym od ziaren). W związku z tym należy wylosować na brzegu miejsce fikcyjnego zarodkowania.
4. Dla fikcyjnego zarodka na brzegu przestrzeni zamiast rzeczywistych współrzędnych tej komórki należy określić współrzędne „rzeczywistego” zarodka, którego miejsce można wyznaczyć na linii prostopadłej do brzegu przestrzeni w miejscu fikcyjnego zarodka i na odległości od brzegu równej (lub nieco się różniącej) odległości od brzegu zarodka ziarna, które dotarło do brzegu i spowodowało to fikcyjne zarodkowanie. Na przykład jeżeli ziarno A dorosło do brzegu $i = nx$ i znajduje się od niego w odległości l (zarodek ma współrzędną $i = nx - l$), natomiast dla wrastającego ziarna wybrano komórkę o współrzędnych (i_1, ny) , to zarodek będzie miał współrzędne $(i_1, ny + l)$. Współrzędne zarodka należy nadać zmienным dotyczącym nowego ziarna, natomiast wektor rozrostu dla komórki będzie równy ($rx = 0, ry = l$) oraz komórka przechodzi w stan przejściowy i należy obliczyć aktualny czas zmiany stanu.
5. Dotarcie granicy ziarna powstałego z zewnętrznego zarodka do brzegu przestrzeni nie powinno wywoływać kolejnego zewnętrznego zarodkowania ($lby = .false$ lub $lby = .false$).
6. Do algorytmu można dodać czynnik losowy na dwa sposoby. Pierwszy będzie oznaczał inną odległość zewnętrznego zarodka od granicy ($l \pm \Delta l$), zaś drugi inny czas zarodkowania ($\pm \Delta t$). Można również zastosować obydwa sposoby.

Krok czasowy odgrywa dość istotną rolę podczas modelowania. Poprzednio czas pozostawał wielkością względną, nie wpływając w istotny sposób na uzyskiwane wyniki. Dlatego był on równy jeden, natomiast podstawowy czas opóźnienia przejścia został wybrany tak, żeby był dłuższy od kroku czasowego. Jest to jedna z podstawowych zasad działania automatów komórkowych, w szczególności tych aktualizowanych asynchronicznie. Ponieważ nasze automaty trudno już zaliczyć do synchronicznych, ten szczegół jest również istotny. Aby zapobiec oddziaływaniu zmian poza obszarem sąsiedztwa, w jednym kroku czasowym przewidziano tylko jeden cykl automatów, w którym każdy stan jest badany jednokrotnie. Jeżeli czas opóźnienia jest krótszy od kroku czasowego, od razu wpływa to na kształt rosnącego ziarna. Niewielki wpływ niespełnienia tego warunku na kształt można zaobserwować w przypadku rosnących ziaren w kształcie prostokąta, ale najbardziej widoczne jest to przy zastosowaniu otwartych warunków brzegowych. Zewnętrzne zarodkowanie na brzegu przestrzeni doprowadzi do tego, że opóźnienie w przypadku komórek na brzegu przestrzeni będzie wielokrotnie mniejsze niż długość kroku czasowego (dla pozostałych komórek opóźnienie pozostaje dłuższym od kroku czasowego).

Wówczas można zastosować dwa rozwiązania. Pierwsze polega na skróceniu kroku czasowego, drugie – na zezwoleniu na wielokrotne wykonanie cykli automatów komórkowych podczas jednego kroku czasowego. Są to rozwiązania równoważne ze względu na uzyskiwany wynik. Rozpatrzmy obie możliwości.

W pierwszym podejściu, polegającym na wielokrotnym powtarzaniu cyklu, najpierw należy sprawdzić czas przejścia w stan końcowy każdej komórki. Wówczas na początku cyklu (kroku czasowego) nadajemy zmiennej logicznej (na przykład *lrepeat*) wartość „false” i wykonujemy wszystkie czynności po kolei. Następnie, jeżeli po obliczeniu czasu przejścia w stan końcowy dowolnej komórki okaże się, że jest on krótszy od czasu bieżącego (może się tak stać, ale niekoniecznie się stanie, jeżeli czas opóźnienia jest krótszy od kroku czasowego), nadajemy zmiennej logicznej wartość „true”. Na końcu cyklu, gdy zmienia się czas przez dodanie kroku czasowego, jeżeli zmienna logiczna pozostała w cyklu bez zmian, czyli ma wartość „false”, należy zmienić czas i przejść do kolejnego kroku czasowego. Natomiast jeżeli zmiana *lrepeat* ma wartość „true”, powtarzamy ten krok czasowy. Wprowadzenie tych zmian od razu poprawi kształt ziaren.

Zmiana długości kroku czasowego wymaga znajomości wszystkich czasów opóźnień. Wówczas podczas wykonywania cyklu porównywane są czasy opóźnienia wszystkich komórek w stanie przejściowym (pozostających w nim albo przechodzących do niego) i wyznacza się najmniejszą wartość. Krok czasowy jest dopasowywany do najmniejszej wartości opóźnienia (powinien być odrobinę mniejszy, równy 0,9–0,99 najmniejszej wartości). W ten sposób krok czasowy będzie się przystosowywał do warunków obliczeniowych. Wówczas na początku cyklu zmiennej (na przykład *dtmin*) jest nadawana bardzo duża wartość, którą następnie porównuje się z czasem opóźnienia każdej komórki w stanie przejściowym, a na końcu cyklu wartość ta jest wykorzystywana do wyznaczenia kroku czasowego. Ze względów obliczeniowych drugie podejście wydaje się mniej korzystne.

Rzeczywiste warunki symulacji

Wybór rzeczywistych wymiarów modelowej przestrzeni zależy od wstępnej wiedzy o procesie, który jest modelowany. Jeżeli jest to na przykład rekrytalizacja, ważne są wymiary ziaren przed procesem i po nim. Wymiary ziarna nie mogą być większe niż 20–30% wymiarów przestrzeni, ziarna nie powinny być też zbyt drobne (składać się z mniejszej liczby komórek niż 20–30). W ten sposób można ustalić wymiary przestrzeni i odpowiednio obliczyć wymiary każdej komórki. Na przykład jeżeli tworzymy początkową mikrostrukturę i zadamy liczbę komórek dla jednego ziarna (przykładowo 100), będziemy mogli obliczyć liczbę ziaren. Dalej, znając średnią wielkość ziarna (na przykład 30 μm), można obliczyć powierzchnię przekroju jednego ziarna i wszystkich, które tworzą przestrzeń modelową, jej wymiary i w końcu wymiary komórki. Oczywiście wymiary można zaokrąglić i sprecyzować je, podobnie jak liczbę ziaren.

Prędkość rozrostu jest zwykle zadawana jako stała lub za pomocą wzoru i nie mamy żadnego wpływu na jej wartość. Może ona być taka sama w całej przestrzeni lub zależna od różnych czynników. Jest to jednak bardzo ważna wielkość, która pomaga powiązać dwie podstawowe zmienne: wymiary i czas.

Czas jest zmienną zależną od wymiarów. Krok czasowy zależy od wymiarów komórki i prędkości rozrostu. Może on pozostawać bez zmian lub automatycznie zmieniać się w opisany wyżej sposób. Czas symulacji w ogólnym przypadku zależy od

modelowanego zjawiska, głównie od liczby komórek w przestrzeni oraz od liczby kroków czasowych. Dlatego symulacje, w których prędkości rozrostu w różnych miejscach istotnie się różnią, trwają dłużej. Uzyskanie struktury, w której rosnące ziarna mają kształt prostokąta o dużym stosunku długości boku dłuższego do krótszego trwa dłużej niż uzyskanie struktury z rosnącymi ziarnami w kształcie okręgów lub kwadratów

Należy zasymulować kilka struktur, porównując czas obliczeń.

Ćwiczenie laboratoryjne 11

Rozwój mikrostruktury. Przemiana fazowa

Celem tego ćwiczenia jest opracowanie i symulacja uproszczonego modelu przemiany dyfuzyjnej.

Opracowany w trakcie poprzednich ćwiczeń program można rozbudować w celu przystosowania go do modelowania różnych zjawisk zachodzących w materiale, symulacji całych procesów technologicznych oraz ich badania.

Wymaga to jednak dużego nakładu pracy i na ćwiczeniach spróbujemy tylko stworzyć bardzo uproszczony model. Opracowane wcześniej automaty komórkowe zostały zastosowane do tworzenia początkowej mikrostruktury. Automatami owymi można zamodelować dowolny proces, który składa się z dwóch etapów – zarodkowania i rozrostu. Mogą to być: krystalizacja (krzepnięcie), rekrytalizacja (dynamiczna, statyczna i metadynamiczna), przemiany fazowe, rozwój struktur dyslokacyjnych, rozdrobnienie ziaren i inne. Jako przykład będzie tu opracowany uproszczony model przemiany fazowej dyfuzyjnej, typu austenit-feryt.

Do modelowania tego zjawiska potrzebne nam są informacje o mikrostrukturze początkowej oraz warunkach zarodkowania i rozrostu.

Mikrostruktura początkowa została już utworzona, więc do dalszych symulacji będą nam służyć wyniki uprzednio przeprowadzonych obliczeń. Przy projektowaniu modelu nowego zjawiska lepiej każdy nowy model zachowywać jako oddzielny podprogram i wywoływać, kiedy zajdzie taka konieczność. Jednak pozostawimy to na razie bez zmian; przemiana będzie programowana w tym samym głównym programie. Po pierwszym cyklu tworzenia początkowej mikrostruktury zostanie dodany cykl odpowiadający przemianie fazowej.

Ponieważ będziemy stosować ten sam automat komórkowy, można skopiować poprzedni cykl i tylko wprowadzić do niego odpowiednie zmiany.

Teraz należy określić warunki zarodkowania. Ze względu na warunki energetyczne najlepszymi do zarodkowania miejscem będą naroża, punkty potrójne, w których stykają się trzy ziarna, trzy granice. Nieco gorszymi miejscami są granice ziaren. Zakładamy, że największe prawdopodobieństwo zarodkowania będzie istniało w punktach potrójnych, mniejsze na granicach, a wewnątrz ziaren będzie ono równe zero. W tej chwili na oddzielnej liście zgrupowane są komórki na granicy ziarna, co już ułatwia dalsze czynności. Niestety, nie mamy komórek w narożach, zatem powinniśmy je

wyznaczyć i utworzyć kolejną listę $L(5)$, do której będziemy przenosić odpowiednie komórki. Wówczas między cyklem tworzenia początkowej struktury a przemianą fazową należy uruchomić cykl, podczas którego będą badane komórki na granicy ziarna ze względu na ich przynależność do punktów potrójnych. Algorytm badania każdej komórki będzie obejmował kilka czynności. Najpierw należy zbadać otoczenie von Neumanna (4 komórki). Jeżeli sąsiednia komórka należy do innego ziarna, należy zapamiętać numer tego ziarna. Jeżeli inna sąsiednia komórka należy do innego ziarna, trzeba sprawdzić, czy nie należy ona do tego samego, zapamiętanego ziarna, a jeżeli nie należy ani do jednego, ani do drugiego, to znaczy, że ma przynajmniej dwóch sąsiadów należących do dwóch innych, różnych ziaren i powinna być przeniesiona do nowej listy. Po zbadaniu wszystkich komórek na granicy ziaren wszystkie ziarna należące do punktów potrójnych zostaną przeniesione do listy 5.

Przykładowo można założyć, że zarodkowanie następuje w 30–80% komórek należących do punktów potrójnych. W przypadku punktów na granicy ziarna prawdopodobieństwo zarodkowania będzie wynosić około 0,5–1%. Do symulacji zarodkowania można wykorzystać podprogram, który był stosowany wcześniej. Jednak należy wprowadzić do niego pewne korekty, które są związane z warunkami zarodkowania

Zakładamy, że kształt ziaren będzie okrągły. Poprzednio warunkiem rozrostu ziarna był zerowy początkowy stan komórki q_0 , w tej chwili nie ma żadnej komórki znajdującej się w tym stanie, więc nowe ziarna ferrytu będą rosły w istniejącej strukturze komórek w stanie q_2 , q_3 i q_5 , czyli na granicy ziaren, wewnątrz ziaren i w punktach potrójnych. Z jednej strony należy pozwolić wrastać ziarnom w komórki w stanach q_2 , q_3 i q_5 należące do ziaren austenitu, ale z drugiej strony nie należy pozwolić im wrastać w ziarna ferrytu. Można wprowadzić dodatkową zmienną dla każdego ziarna, która będzie określała jego typ (austenit-ferryt). Jest to dobre rozwiązanie, metodycznie uzasadnione, ale niekonieczne, co zostanie wyjaśnione niżej. Aby zbliżyć symulację do warunków rzeczywistych, wprowadzamy dodatkowe ograniczenie: ziarna ferrytu będą rosły nie we wszystkich ziarnach austenitu, lecz tylko w jednym, macierzystym, w którym powstały ich zarodki. Należy więc wprowadzić dodatkowy warunek: sprawdzanie, czy ziarno może w danym miejscu rosnąć, czy nie. Zatem podczas zarodkowania nowego ziarna ferrytu należy zapamiętać numer ziarna, które jest dla niego macierzyste. Ten warunek jest znacznie ostrzejszy niż sprawdzenie typu austenit-ferryt, zatem poprzedni warunek w tym podejściu nie jest konieczny, podobnie jak i wprowadzenie zmiennej określającej typ ziarna nie jest konieczne (ale dobre ze względów metodycznych). Podczas rozrostu pozostaje tylko sprawdzać, czy ziarno może wrastać w tę komórkę, czy nie.

Wprowadzenie opisanych zmian pozwala zasymulować najprostszy wariant zmian struktury.

Należy pamiętać, że w rzeczywistości warunki zarodkowania i rozrostu są znacznie bardziej złożone i są funkcją wielu zmiennych: temperatury, orientacji krystalograficznej, odkształcenia i stężenia pierwiastków.

Literatura

- Adrian H., 2011: *Numeryczne modelowanie procesów obróbki cieplnej*, Wyd. AGH, Kraków.
- Avrami M., 1939: *Kinetics of phase change I*, J. Chem. Phys. 7 (12), 1103–1112.
- Avrami M., 1940: *Kinetics of phase change II*, J. Chem. Phys. 8 (2), 212–224.
- Avrami M., 1941: *Kinetics of phase change III*, J. Chem. Phys. 9 (2), 177–184.
- Beck P.A., 1949: *The formation of recrystallization nuclei*, J. Appl. Phys. 20 (6), 633–634.
- Beck P.A., Sperry P.R., 1950: *Strain induced boundary migration in high purity aluminum*, J. Appl. Phys. 21 (2), 150–152.
- Bos C., Mecozzi M.G., Sietsma J., 2010: *A microstructure model for recrystallisation and phase transformation during the dual-phase steel annealing cycle*, Comp. Mater. Sci. 48, 692–699.
- Bross S., Hahner P., Steck E.A., 2003: *Mesoscopic simulations of dislocation motion in dynamic strain ageing alloys*, Comp. Mater. Sci. 26, 46–55.
- Bross S., Steck E.A., 1998: *Simulation of selforganised dislocation structures in b.c.c. single crystals*, Comp. Mater. Sci. 13, 16–22.
- Burks A.W. (ed.), 1970: *Cellular Automata*, University of Illinois Press, Urbana.
- Cahn R.W., 1949: *Recrystallization of single crystals after plastic bending*, J. Inst. Metals 76 (2), 121–143.
- Cahn R.W., 1950: *A new theory of recrystallization nuclei*, Proc. Physic. Soc. ser. AI 63 (364), 323–336.
- Cahn W., 1996: *The Time Cone Method for an Allen-Cahn/Cahn-Hilliard System in Thermodynamics and Kinetics of Phase Transformation*, 398 of Mater. Res. Soc. Symp. Proc. 425–438.
- Chapman P., 2002: <http://www.igblan.free-online.co.uk/igblan/ca/>.
- Davies C.H.J., 1995: *The effect of neighbourhood on the kinetics of a cellular automaton recrystallization model*, Scr. Metall. Mater. 33 (7), 1139–1143.
- Davies C.H.J., 1997: *Growth of nuclei in a cellular automaton simulation of recrystallization*, Scr. Mater. 36 (1), 35–40.
- Davies C.H.J., Hong L., 1999: *The cellular automaton simulation of static recrystallization in cold rolled AA1050*, Scr. Mater. 40 (10), 1145–1150.

- Derby B., Ashby M.F., 1987: *On dynamic recrystallization*, Scr. Metall. 21, 879–884.
- Dewdney A.K., 1990: *Computer recreations: The cellular automata programs that create Wireworld, Rugworld and other diversions*, Sci. Am., Jan., 136–139.
- Ding R., Guo Z.X., 2001: *Coupled quantitative simulation of microstructural evolution and plastic flow during dynamic recrystallization*, Acta Mater. 49, 3163–3175.
- Doherty R., Hughes D.A., Humphreys F., Jonas J., Jensen D.J., Kassner M., King W., McNelley T., McQueen H., Rollett A., 1997: *Current issues in recrystallization: a review*, Mater. Sci. Eng. A 238, 219–274.
- Estrin Y., Toth L.S., Brechet Y., Kim H.S., 2006: *Modelling of the Evolution of Dislocation Cell Misorientation under Severe Plastic Deformation*, Mat. Sci. Forum 503–504, 675–680.
- Ferry M., Humphreys F.J., 1996: *Discontinuous subgrain growth in deformed and annealed $\{110\}<001>$ aluminium single crystals*, Acta Mater. 44 (4), 1293–1308.
- Heidenreich R.D., 1949: *Electron microscope and diffraction study of metal crystal textures by means of thin sections*, J. Appl. Phys. 20 (10), 993–1010.
- Hurley P.J., Humphreys F.J., 2003: *Modelling the recrystallization of single-phase aluminium*, Acta Mater. 51, 3779–3793.
- Gardner M., 1970: *Mathematical games. The fantastic combinations of John Conway's new solitaire game "life"*, Sci. Am. 223, 120–123.
- Gardner M., 1972: *Математические досуги*, Moskwa, Mir (tłumaczenie: *New mathematical diversions from Scientific American*, Simon & Shuster, NY, 1966; *The unexpected hanging and other mathematical diversions*, Simon & Shuster, NY, 1969).
- Goetz R.L., 2005: *Particle simulated nucleation during dynamic recrystallization using a cellular automata model*, Scr. Mater. 52, 851–856.
- Goetz R.L., Seetharaman V., 1997: *Modeling dynamic recrystallization using cellular automata*, Scr. Mater. 38 (3), 405–413.
- Hadała B., Malinowski Z., 2009: *Accuracy of the finite element solution to steady convection-diffusion heat transport equation in continuous casting problem*, Comp. Meth. Mat. Sci. 9 (2), 302–308.
- Hesselbarth H., Gobel I., 1991: *Simulation of recrystallization by cellular automata*, Acta Metall. Mater. 39 (9), 2135–2143.
- Iba Y., Akutsu Y., Kaneko K., 1986: *Phase Transitions in 2-dimensional Stochastic Cellular Automata*, Science on Form, KTK Scientific Publ., Tokyo, 103–111.
- Janssens K.G.F., 2010: *An introductory review of cellular automata modeling of moving grain boundaries in polycrystalline materials*, Math. Comput. Simul. 80, 1361–1381.
- Janssens K.G.F., Raabe D., Kozeschnik E., Miodownik M.A., Nestler B., 2007: *Computational Materials Engineering: An Introduction to Microstructure Evolution*, Elsevier.
- Jin Z., Cui Z., 2010: *Investigation on strain dependence of dynamic recrystallization behavior using an inverse analysis method*, Mater. Sci. Eng. A 527, 3111–3119.

- Johnson W.A., Mehl R.F., 1938: *Reaction kinetics in processes of nucleation and growth*, Trans. Metallur. Soc. AIME 135, 416–441.
- Kolmogorov A.E., 1937: *On the static theory of metal crystallization*, Akad. Nauk SSSR, Izv., Ser. Mater. 1, 355–359.
- Kugler G., Turk R., 2004: *Modeling the dynamic recrystallization under multi-stage hot deformation*, Acta Mater. 52, 4659–4668.
- Kugler G., Turk R., 2006: *Study of the influence of initial mikrostructure topology on the kinetics of static recrystallization using a cellular automata model*, Comp. Mater. Sci. 37, 284–291.
- Kumar M., Sasikumar R., Kesavan Nair P., 1998: *Competition between nucleation and early growth of ferrite from austenite – studies using cellular automation simulations*, Acta Mater. 46, 6291–6303.
- Kurtz W., Fisher D.J., 1986: *Fundamentals of solidification*, Switzerland, Germany: Trans. Tech. Publications.
- Lan Y.J., Li D.Z., Li Y.Y., 2004: *Modeling austenite decomposition into ferrite at different cooling rate in low-carbon steel with cellular automaton method*, Acta Mater. 52, 1721–1729.
- Lan Y.J., Li D.Z., Li Y.Y., 2005: *Mesoscale simulation of ferrite transformation from deformed austenite during continuous cooling in a C–Mn steel using a cellular automaton metod*, Comp. Mater. Sci. 32, 147–155.
- Laber K., Milenin A., 2007: *Numeryczna analiza rozkładu temperatury oraz parametrów odkształcenia w materiale podczas regulowanego walcowania prętów okrągłych*, Mat. Konf. VIII „Nowe technologie i osiągnięcia w metalurgii i inżynierii materiałowej”, Częstochowa 25 maja 2007, 1, Seria: Konferencje nr 61, 320–325.
- Lange W.F., Enomoto M., Aaronson H.I., 1988: *The kinetics of ferrite nucleation at austenite grain-baundaries in Fe-C alloys*, Metall. Trans. A19, 427–440.
- Lasko G.V., Deryugin Y.Y., Schmauder S., 2004: *Plastic Deformation Development in Polycrystals Based on the Cellular Automata end Relaxation Element Method*, Proc. on Cellular Automata for Research and Industry, ACRI 2004, LNCS 3305, 375–384.
- Lasko G.V., Deryugin Y.Y., Schmauder S., 2006: *Simulation of the Evolution of Band Structures in Polycrystals on the Basis of Relaxation Element Method and Cellular Automata*, Proc. on Cellular Automata for Research and Industry, ACRI 2006, LNCS 4173, 367–372.
- Macioł P., Gawad J., Kuziak R., Pietrzyk M., 2010: *Internal variable and cellular automata-finite element models of heat treatment*, Int. J. Multiscale Comp. Eng. 8, 267–285.
- Madej L.K., Hodgson P.G., Gawad J., Pietrzyk M., 2004: *Modeling of Rheological behavior and microstructure evolution using cellular automation technique*, Proc. 7th Conf. ESFORM 2004, 143–146.
- Majta J., 2008: *Odształcanie i własności. Stale mikroskopowe. Wybrane zagadnienia*, Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków.

- Majta J., Doniec K., 2010: *On the utilization of plastic instability criterion in ductility assessment of ultrafine-grained microalloyed*, Mater. Sci. Forum 638–642, 1977–1982.
- Majta J., Muszka K., Doniec K., Svyetlichnyy D., 2009: *Mapping the dislocation strengthening in rheological model of BCC ultrafine-grained structures*, Computational plasticity X: fundamentals and applications – Barcelona, Spain, 2–4 September 2009, CD-ROM 1–4.
- Makarov P.V., Schmauder S., Cherepanov O.I., Smolin I.Yu., Romanova V.A., Balokhonov R.R., Saraev D.Yu., Soppa E., Kizler P., Fischer G., Hu S., Ludwig M., 2001: *Simulation of elastic-plastic deformation and fracture of materials at micro-, meso- and macrolevels*, Theor. Appl. Fract. Mech. 37, 183–244.
- Malinowski Z., 2006: *Numeryczne modele w przeróbce plastycznej i wymianie ciepła*, Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków.
- Malinowski Z., Milenin A., Kędzierski Z., Svyetlichnyy D., Ciomborowski J., Stepiński J., Dyja H., Matachowski J., 2009: Projekt własny nr N508 002 32/0158 „Opracowanie i zastosowanie modelu rozwoju mikrostruktury i własności mechanicznych stali w procesach przeróbki plastycznej z wykorzystaniem trójwymiarowych automatów komórkowych”. Sprawozdanie merytoryczne z wykonanych zadań (niepublikowane).
- Matachowski J., Svyetlichnyy D., 2005: *Badanie wpływu kształtu ziarna oraz parametrów przestrzeni automatów komórkowych na proces krystalizacji*, Nowe technologie i osiągnięcia w metalurgii i inżynierii materiałowej, Częstochowa, 264–267.
- Matachowski J., Svyetlichnyy D.S., 2006: *Trójwymiarowy automat komórkowy do modelowania rekrytalizacji*, Nowe technologie i osiągnięcia w metalurgii i inżynierii materiałowej, Częstochowa, 358–361.
- Mecozzi M. G., Bos C., Sietsma J., 2011: *3D cellular automata modelling of solid-state transformations relevant in low-alloy steel production*, Solid State Phenom. 172–174, 1140–1145.
- Milenin A.A., 1998: *Исследование численных свойств алгоритмов метода конечных элементов применительно к трехмерным задачам обработки металлов давлением*, Изв. РАН. Металлы 5, 33–37.
- Milenin A.A., 2002: *Проблемы разработки и применения трехмерных численных моделей для оптимизации процессов прокатки*, Сучасні проблеми металургії. Наукові вісті. Том. 5. Пластична деформація металів, Системні технології, 36–46.
- Milenin A., Dyja H., Mróz S., 2004: *Simulation of metal forming during multi-pass rolling of shape bars*, J. Mater. Proc. Technol. 153–154, 108–114.
- Miodownik M., Godfrey A.W., Holm E.A., Hughes D.A., 1999: *On boundary misorientation distribution functions and how to incorporate them into three-dimensional models of microstructural evolution*, Acta Mater. 47, 2661–2668.

- Moore E.F., 1962: *Machine Models of Self Reproduction*, Proc. Symp. Appl. Math. 14, 17–33.
- Mukhopadhyay P., Loeck M., Gottstein G., 2007: *A cellular operator model for the simulation of static recrystallization*, Acta Mater. 55, 551–564.
- Muszka K., Doniec K., Majta J., 2009: *Possibilities of the mechanical behaviour modelling of structures after severe plastic deformation*, XVI Conference on Computer Methods in Materials Technology: Komplastech' 2009, 11–14 January 2009, Krynica-Zdroj.
- Nes E., 1995: *Recovery revisited*, Acta Metall. Mater. 43, 2189–2207.
- Neumann J. von, 1963: *Papers of John von Neumann on computing and computer theory*, w: Aspray W., Burks A. (Eds), Reprint Series on the History of Computing, 12, MIT Press, Cambridge, MA.
- Neumann J. von, 1966: *Theory of Self-Reproducing Automata. Edited and completed by Burks A.W.*, University of Illinois Press, Champaign, IL.
- Oishi S., Aoyagi Y., Shizawa K., 2006: *Modeling and Simulation of Crystal Plasticity Based on GN Crystal Defects for Ultrafine-Grained Metals Induced by Severe Plastic Deformation*, Mat. Sci. Forum 503–504, 215–220.
- Opata J., Jabłoński G., Rudzki D., Pietrzyk M., 2012: *Modelowanie metodą automatów komórkowych cyklu przemian fazowych w stalach*, KomPlasTech 2012, dokument elektroniczny, 1–9.
- Qian M., Guo Z. X., 2004: *Cellular automata simulation of microstructural evolution during dynamic recrystallization of an HY-100 steel*, Mater. Sci. Eng. A 365, 180–185.
- Raabe D., 2002: *Cellular automata in materials science with particular reference to recrystallization simulation*, Ann. Rev. Mater. Res. 32, 53–76.
- Raabe D., 2004: *Mesoscale simulation of spherulite growth during polymer crystallization by use of a cellular automaton*, Acta Mater. 52, 2653–2664.
- Raabe D., Roters F., Barlat F., Chen L.-Q., 2004: *Continuum Scale Simulation of Engineering Materials*, Wiley-VCH.
- Rappaz M., Gandin C.-A., 1993: *Probabilistic modeling of microstructure formation in solidification processes*, Acta Metall. Mater. 41, 345–360.
- Rios P.R., Siciliano Jr F., Sandim H.R.Z., Plaut R.L., Padilha A.F., 2005: *Nucleation and Growth During Recrystallization*, Mater. Res. 8 (3), 225–238.
- Roberts W., Ahlblom B., 1978: *A nucleation criterion for dynamic recrystallization during hot working*, Acta Metall. 26, 801–813.
- Rollett A.D., 1997: *Overview of modeling and simulation of recrystallization*, Prog. Mater. Sci. 42, 79–99.
- Sellars C.M., 1990: *Modelling Microstructural Development During Hot Rolling*, Mater. Sci. Technol. 6, 1072–1081.
- Senuma T., Yada H., Matsumura Y., Futamura T., 1984: *Structure of austenite of carbon steels in high speed hot working processes*, Testu-to-Hagane 70, 2112–2119.

- Sorby H.C., 1886: *The application of very high powers to the study of the microscopical structure of steel*, J. Iron and Steel Inst. 30 (1), 140–145.
- Yang H., Wu C., Li H. W., Fan X.G., 2011: *Review on cellular automata simulations of microstructure evolution during metal forming process: Grain coarsening, recrystallization and phase transformation*, Sci. China Ser. E: Technol. Sci. 54, 2107–2118.
- Spittle J.A., Brown S.G.R., 1994: *3D cellular automaton model of coupled growth in 2-component systems*, Acta Metall. Mater. 42 (6), 1811–1815.
- Svyetlichnyy D., 2004: *Zastosowanie technik teorii sterowania i sztucznych sieci neuronowych w modelowaniu on-line walcowania wyrobów płaskich*, Politechnika Częstochowska, Częstochowa.
- Svyetlichnyy D.S., 2006: *Uwzględnienie odkształcenia podczas modelowania za pomocą automatów komórkowych*, Nowe technologie i osiągnięcia w metalurgii i inżynierii materiałowej, Częstochowa, 533–536.
- Svyetlichnyy D., 2007: *Three-dimensional cellular automata for simulation of microstructure evolution*, COMPLAS IX: COMputational PLASticity: fundamentals and applications, Pt. 2 Barcelona, 5–7 September 2007, CIMNE, 986–989.
- Svyetlichnyy D.S., 2007: *Some aspects which deny a use of 2D cellular automata for modeling of recrystallization*, Comp. Meth. Mater. Sci. 7, 1, 175–181.
- Svyetlichnyy D.S., 2008a: *Algorytmy wstępne przekształcenia dyskretnej przestrzeni automatów komórkowych w siatkę MES*, KomPlasTech 2008, Informatyka w technologii metali, materiały XV konferencji: Korbielów, 6–9 stycznia 2008, Wydawnictwo Naukowe Akapit, 207–212.
- Svyetlichnyy D.S., 2008b: *Trehmernye kletochnye avtomaty dlâ modelirovaniâ mikrostruktury vo vremâ rekristallizacii*, Udoskonalennâ procesiv ò obladnannâ obrobki tiskom v metalurgii ò mašinobuduvannî: tematiënij zbîrnik naukovih prac', Mînistertstvo osvîti ò nauki Ukraïni, Donbas'ka derzavna mašinobudivna akademîa, Kramators'k, 87–93.
- Svyetlichnyy D.S., 2008c: *Frontalnye kletochnye avtomaty*, Systemni technologii – Dnipropetrovsk 3, 56, 2 42–47.
- Svyetlichnyy D., 2008d: *Modeling of Microstructure Evolution of Steel during the Rolling by using Frontal Cellular*, Proc. of the 3rd International Conference Thermo-mechanical Processing of Steel, 10–12.09.08, Padua, Italy, CD-ROM.
- Svyetlichnyy D.S., 2012a: *Simulation of microstructure evolution during shape rolling with the use of frontal cellular automata*, ISIJ Int. 52, 559–568.
- Svyetlichnyy D., 2009a: *Modeling of Microstructure Evolution during the Rolling by using Cellular Automata*, Comp. Meth. Mater. Sci. 9, 2, 256–263.
- Svyetlichnyy D.S. 2009b: *Simulation of microstructure evolution during and after the rolling by using cellular*, CMM 2009: 18th international conference on Computer Methods in Mechanics, 18–21 May 2009, Zielona Góra, Poland, The University of Zielona Góra Press, 453–454.

- Svyetlichnyy D., 2009c: *Simulation of microstructure evolution during the shape rolling by using cellular automata*, Computational plasticity X: Fundamentals and applications, Barcelona, Spain, 2–4 September 2009, CD-ROM, 1–4.
- Svyetlichnyy D., 2009d: *Simulation of microstructure evolution during the rolling*, SEECCM 2009: The 2nd South-East European Conference on Computational Mechanics, Athens, National Technical University. Institute of Structural Analysis & Seismic Research, 172.
- Svyetlichnyy D.S., 2010a: *Modeling of microstructure evolution in process with severe plastic deformation by cellular automata*, Mater. Sci. Forum 638–642, 2772–2777.
- Svyetlichnyy D.S., 2010b: *Modelling of the microstructure: from classical cellular automata approach to the frontal one*, Comp. Mater. Sci. 50, 1, 92–97.
- Svyetlichnyy D.S., 2011: *Modeling of macrostructure formation during the solidification by using frontal cellular automata*, w: Salcido A. (Ed.), Cellular automata – Innovative modelling for science and engineering, InTech, Croatia, 179–196.
- Svyetlichnyy D.S., 2012a: *Simulation of microstructure evolution during shape rolling with the use of frontal cellular automata*, ISIJ Int. 52, 559–568.
- Svyetlichnyy D.S., 2012b: *Reorganization of cellular space during the modeling of the microstructure evolution by frontal cellular automata*, Comp. Mater. Sci. 60, 153–162.
- Svyetlichnyy D.S., www: http://pl.wikipedia.org/wiki/Frontalny_automat_komórkowy.
- Svyetlichnyy D., Hadała B., Malinowski Z., Łach Ł., 2012: *Modelowanie rozwoju struktury krzepnącego wlewka w procesie ciągłego odlewania stali*, Hutnik – Wiadomości Hutnicze 79 (4), 203–211.
- Svyetlichnyy D.S., Łach Ł., 2012: *Digital material representation of given parameters*, Steel Res. Int. (w druku).
- Svyetlichnyy D., Majta J., 2009: *Modeling of microstructure evolution in process with severe plastic deformation*, Computational methods for coupled problems in science and engineering III, Ischia, Italy, 08–10 June 2009, CIMNE, Barcelona 2009, CD-ROM, 1–4.
- Svyetlichnyy D., Majta J., Muszka K., 2008: *Modeling of microstructure evolution of BBC metals subjected to severe plastic deformation*, Steel Res. Int. 79, 452–458.
- Svyetlichnyy D., Majta J., Muszka K., Jurczyk T., 2011: Projekt badawczy własny nr N508 3812 33 „Opracowanie modelu oraz symulacja komputerowa rozwoju struktury w procesach uzyskiwania materiałów drobnodziarnistych”. Sprawozdanie merytoryczne z wykonanych zadań (niepublikowane).
- Svyetlichnyy D.S., Majta J., Muszka K., Łach Ł., 2010: *Modeling of microstructure evolution subjected to severe plastic deformation*, AMPT 2010 : Advances in Materials and Processing Technologies: international conference, Paris, France, Oct. 24–27, 2010.
- Svyetlichnyy D., Matachowski J., 2005a: *Modelowanie rozwoju mikrostruktury i naprężenia uplastyczniającego za pomocą automatów komórkowych*, Przegląd Mechaniczny LXIV, 7–8, 46–50.

- Svyetlichnyy D., Matachowski J., 2005b: *Modelowanie rozwoju mikrostruktury i naprężenia uplastyczniającego za pomocą automatów komórkowych*, Mat. Konf. FIMM 2005, Warszawa, 2005, Zeszyt Naukowy PW „Mechanika” z. 207, Warszawa, 25–30.
- Svyetlichnyy D., Matachowski J., 2007: *Three-Dimensional Cellular Automata For Simulation of Microstructure Evolution during Recrystallization*, AIP Conference Proceedings 908, 1357–1362.
- Svyetlichnyy D.S., Matachowski J., Milenin A.A., 2005: *Моделирование процесса динамической рекристаллизации с помощью клеточных автоматов*, Сучасні проблеми металургії, Наукові вісті. Том 8. Пластична деформація металів. Дніпропетровськ: Системні технології, 207–211.
- Svyetlichnyy D., Milenin A., 2005: *Modelowanie procesów rekrytalizacji za pomocą automatów komórkowych*, Mat. Konf. KomPlasTech, Ustroń 16–19 stycznia, 2005, 115–122.
- Svyetlichnyy D., Milenin A., 2009: *Modelirovanie izmeneniâ mikrostruktury vo vremâ sortovoj prokatki s pomos'û trehmernyh kletočnyh avtomatov*, Materials working by pressure, Ministerstvo obrazovaniâ i nauki Ukrainy, Donbasskaâ gosudarstvennaâ mašinstroitel'naâ akademiâ, Kramatorsk 1, 104–110.
- Thevoz Ph., Desbiolles J.L., Rappaz M., 1989: *Modeling of equiaxed microstructure formation in casting*, Metall. Trans. A20, 311–321.
- Ulam, S.M., 1952: *Random Processes and Transformations*, Proc. Int. Congress of Mathematicians, Cambridge, MA, 1950, 2, 264–275.
- Ulam S.M., 1962: *On Some Mathematical Problems Connected with Patterns of Growth of Figures*, Proc. Symp. Appl. Math. 14, 215–224 (również w: Burks A.W. (ed.), 1970: *Cellular Automata*, University of Illinois Press, Urbana, 219–231).
- Umamoto M., Guo Z.H., Tamura I., 1987: *Effect of cooling rate on grain size of ferrite in a carbon steel*, Mater. Sci. Technol. 3, 249–255.
- Varma M.R., Sasikumar R., Pillai S.G.K., Nair P., 2001: *Cellular automaton simulation of microstructure evolution during austenite decomposition under continuous cooling conditions*, Bull. Mater. Sci. 24 (3), 305–312.
- Vitanyi P., 1973: *Sexually reproducing cellular automata*, Math. Biosci. 18, 23–54.
- Voronoi G., 1908: *Nouvelles applications des paramètres continus à la théorie des formes quadratiques*. J. Reine Angew. Math. 133, 97–178.
- Wielgus M., Majta J., Łuksza J., 2009: *Nowa metoda kątownego wielostopniowego ciągnięcia*, Politechnika Warszawska. Seria: Mechanika, z. 226, 131–136.
- Wiener N., Rosenblueth A., 1946: *The mathematical formulation of the problem of conduction of impulses in a network of connected excitable elements, specifically in cardiac muscle*, Arch. Inst. Cardiol. México 16, 205–265.
- Wolfram S., 1983: *Statistical mechanics of cellular automata*, Rev. Mod. Phys. 55, 601–644.
- Wolfram S., 2002: *A New Kind of Science*, Wolfram Media, Inc.

- Xiao N., Zheng C., Li D., Li Y., 2008: *A simulation of dynamic recrystallization by coupling a cellular automaton method with a topology deformation technique*, Comp. Mater. Sci. 41, 366–374.
- Yamaki N., Yoshiteru A., Shizawa K., 2006: *Dislocation-Crystal Plasticity Simulation Based on Self-Organization for Repartition of Dislocation Cell Structures and Subgrain*, Mat. Sci. Forum 503–504, 989–994.
- Yazdipour N., Davies C.H.J., Hodgson P.D., 2008: *Microstructural modeling of dynamic recrystallization using irregular cellular automata*, Comp. Mater. Sci. 44, 566–576.
- Zhang L., Wang Y.M., Zhang C.B., Wang S.Q., Ye H.Q., 2003a: *A cellular automaton model of the transformation from austenite to ferrite in low carbon steels*, Modell. Simul. Mater. Sci. Eng. 11, 791–802.
- Zhang L., Zhang C.B., Wang Y.M., Wang S.Q., Ye H.Q., 2003b: *A cellular automaton investigation of the transformation from austenite to ferrite during continuous cooling*, Acta Mater. 51, 5519–5527.
- Zheng C.W., Li D.Z., Lu S.P., Li Y.Y., 2008a: *On the ferrite refinement during the dynamic strain-induced transformation: A cellular automaton modeling*, Scr. Mater. 58, 838–841.
- Zheng C., Xiao N., Li D., Li Y., 2008b: *Microstructure prediction of the austenite recrystallization during multi-pass steel strip hot rolling: a cellular automaton modeling*, Comp. Mater. Sci. 44, 507–514.
- Zheng C.W., Xiao N.M., Hao L.H., Li D.Z., Li Y.Y., 2009: *Numerical simulation of dynamic strain-induced austenite–ferrite transformation in a low carbon steel*, Acta Mater. 57, 2956–2968.
- Zuse K., 1969: *Rechnender Raum, Schriften zur Datenverarbeitung*, 1 Freidr. Vieweg & Sohn, Braunschweig (tłumaczenie na angielski: Zuse K., 1970: *Calculating Space*, MIT Press, Cambridge).
- Zuse K., 1982: *The computing Universe*, Int. J. Theor. Phys. 21, 6–7, 589–600.



DMYTRO SVYETLICHNYY, dr hab. inż., profesor nadzwyczajny Akademii Górniczo-Hutniczej w Krakowie, jest absolwentem Krzyworońskiego Instytutu Górnictwa Rud Żelaza (obecnie Krzyworoński Uniwersytet Techniczny, Ukraina). Doktoryzował się w Instytucie Budowy Maszyn Metalurgicznych im. Aleksandra Celikova (VNIIMETMASH, Moskwa, 1990). Stopień doktora habilitowanego inżyniera uzyskał w Politechnice Częstochowskiej (2004) oraz w Narodowej Metalurgicznej Akademii Ukrainy (NMetAU, Dniepropetrowsk, 2006). Doświadczenia w pracy naukowo-badawczej zdobywał w Rosji (VNIIMETMASH, Moskwa), na Ukrainie (Instytut Napędów Elektrycznych, NMetAU), w Uzbekistanie (huta w Bekabadzie), Polsce i Kanadzie (Uniwersytet Waterloo). W ramach działalności naukowej i badawczej zajmuje się teorią i technologią plastycznej przeróbki metali, automatyką oraz wybranymi zagadnieniami inżynierii materiałowej, m.in. własnościami metali, zjawiskami relaksacji i rekrytalizacji, rozwojem mikrostruktury oraz zastosowaniem do modelowania automatów komórkowych. Jest autorem ponad stu publikacji w czasopiśmie i materiałach konferencyjnych. Był wykonawcą lub kierował realizacją kilkudziesięciu prac naukowo-badawczych. Obecnie pracuje na Wydziale Inżynierii Metali i Informatyki Przemysłowej w Akademii Górniczo-Hutniczej.