

Marcin Krupski*, Agnieszka Siwocha*, Andrzej Cader*

Interpolacja fraktalna w grafice komputerowej – wykorzystanie nowej metody

1. Wstęp

Wiele kształtów występujących w przyrodzie czy też przebiegów dynamicznych odpowiadających rzeczywistym procesom ma charakter bardzo nieregularny, a odpowiadające im charakterystyki wykazują chropowatość i złożoną strukturę na wielu poziomach zagłębienia. Krzywe tego typu, zaliczane są do fraktali lub fraktali stochastycznych – multifraktali. W praktyce przybliżanie ich za pomocą klasycznych metod interpolacji jest niemożliwe, gdyż wielomiany lub inne funkcje do odtwarzania konturów dobrze wypełniają swą rolę jedynie w przypadku krzywych gładkich i w miarę regularnych. W sytuacji gdy mamy do czynienia z krzywymi o dużej złożoności i chropowatości, niezbędne jest wykorzystanie do interpolacji metod fraktalnych. Obecnie dominującą grupą metod tego typu są metody oparte na fraktalnych funkcjach interpolacyjnych (*fractal interpolation function*) FIF zaproponowanych przez Barnsleya [1]. Metody te obarczone są wieloma niedostatkami utrudniającymi powszechne ich wykorzystanie w zastosowaniach praktycznych. W pracy zaprezentowano inną alternatywną metodę wykorzystania krzywych fraktalnych do przybliżania złożonych krzywych, która jest bardziej odpowiednia niż metoda FIF do interpolacji struktur multifraktalnych. Metoda ta została po raz pierwszy przedstawiona w pracy [2]. Opiera się ona na wykorzystaniu charakterystyk nielokalnych do opisu węzłów interpolacji. Jako zbiór bazowych elementów przybliżenia zastosowano jednoparametrową rodzinę krzywych fraktalnych – odpowiednik splajnów bazowych (*base splines*). W artykule szczegółowo przedstawiono algorytm podstawowy oraz metody go wspomagające.

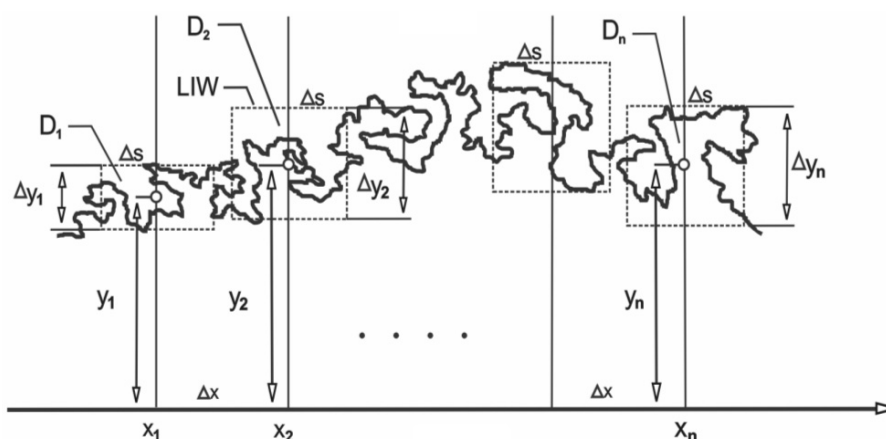
2. Idea proponowanej metody

Jednowymiarowa interpolacja polega na rekonstrukcji zadanej krzywej lub zależności funkcyjnej za pomocą zbioru punktów (węzłów interpolacyjnych), które reprezentują tę krzywą bądź zależność. W większości metod do tego celu wykorzystuje się wielomiany.

* Instytut Kształcenia na Odległość, Wydział Informatyki, Zarządzania i Transportu, Wyższa Szkoła Humanistyczno-Ekonomiczna w Łodzi

Jakość odtworzenia zależy od częstości próbkowania – kroku interpolacji oraz właściwego doboru funkcji interpolujących. Złożonych zależności nie da się interpolować za pomocą wyidealizowanych związków funkcyjnych. Nawet, jeśli stosuje się do ich przybliżania metody klasyczne, to otrzymywane wyniki daleko odbiegają od rzeczywistości. Można przyjąć, że struktury fraktalne mogą być odtwarzane jedynie za pomocą innych struktur fraktalnych lub metod generujących fraktale.

Proponowana metoda wykorzystuje, podobnie jak metody klasyczne, zasadę próbkowania ze stałym krokiem interpolacyjnym. Pojęcie węzła interpolacyjnego zostaje uogólnione, zastąpione ideą lokalnego okna interpolacyjnego LIW (*Local Interpolation Window*) [2]. Jest to obszar obejmujący fragment krzywej interpolowanej w zakresie wokół punktu interpolacyjnego x_i , gdzie Δx – szerokość okna (rys. 1). Charakterystyki z tego okna tworzą zbiór danych przechowywanych do celu odtworzenia krzywej. Szerokość okna jest stała, jeśli krok interpolacyjny jest stały. Wysokość zaś dostosowana jest do rozpiętości krzywej w przedziale ograniczonym szerokością okna. Należy zauważyć, że okno interpolacyjne jest polem widzenia otoczenia węzła interpolacyjnego, z którego chcemy pozyskać jak największą ilość informacji, by później z maksymalną dokładnością odtworzyć krzywą. Żadna z dotychczas stosowanych metod nie wykorzystuje takiego podejścia.



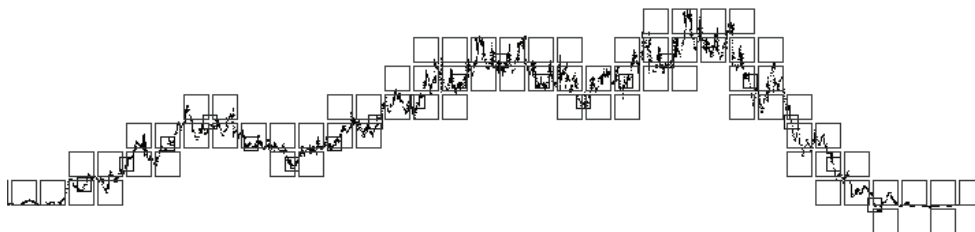
Rys. 1. LIW – lokalne okno interpolacyjne, Δs – szerokość, Δy_i – wysokość; (x_i, y_i) – węzły interpolacyjne, D_i – wymiary fraktalne [2]

W proponowanej metodzie zakładamy, że:

- opis węzła interpolacyjnego rozszerzono o dodatkowe informacje uwzględniające charakter odtwarzanej krzywej, np. wymiar fraktalny;
- dodatkowe wielkości mają charakter nielokalny – odzwierciedlają przebieg krzywej w otoczeniu węzła;
- otoczenie węzła interpolacyjnego określa apriorycznie zadane okno LIW dostosowane do kroku interpolacyjnego;
- w zakresie funkcji dopasowujących prototypem proponowanej metody jest metoda splajnów bazowych.

Warto zauważyć, że w przedstawionej koncepcji szerokość LIW nie musi być równa krokowi interpolacji. Przy szerokości mniejszej, strefa „martwa” pomiędzy oknami zostaje odtworzona za pomocą krzywych bazowych z sąsiadujących okien. Jednak niepotrzebnie tracona jest informacja z tej strefy, która mogłaby zwiększyć dokładność interpolacji. Natomiast w przypadku szerokości okna większej od kroku interpolacyjnego, w strefie przykrywania się okien dochodzi do dodatkowego uzgodnienia kształtu i wymiaru, co stanowi niewątpliwie dodatkowy atut przy interpolacji, a szersze LIW zapewnia więcej jakościowo lepszej informacji. Okno jednak nie może być zbyt szerokie, gdyż traci wtedy swój lokalny charakter, co może ujemnie wpłynąć na dokładność odtworzenia.

W praktyce odtwarzane krzywe posiadają strukturę dyskretną z określoną rozdzielczością, np. w algorytmach komputerowych uwarunkowaną wielkością piksela. Pierwotna krzywa może być odtwarzana lokalnie za pomocą zadanych krzywych fraktalnych o odpowiednim wymiarze.



Rys. 2. Ilustracja wymiaru pudełkowego

W praktyce ze względu na ułatwienie obliczeń, wymiar może być obliczany dla każdego okna LIW, jako wymiar pojemnościowy (pudełkowy, rys. 2), definiowany zależnością

$$D = \lim_{\epsilon \rightarrow 0} \frac{\log N(\epsilon)}{\log(1/\epsilon)} \quad (1)$$

gdzie N jest minimalną liczbą kostek o boku ϵ potrzebnych do nakrycia badanego zbioru fraktalnego.

3. Krzywe bazowe interpolacji i zasady interpolacji

Zaproponowana metoda wymaga określenia zbioru krzywych fraktalnych, które pełnią rolę elementów bazowych przy odtwarzaniu. Krzywe te muszą spełniać kilka warunków. Niech $S, T \in \mathbb{R}$, $S = [0, 1]$, $T = [1, 2]$ i niech K zbiór ciągłych w $U = S \times R$ krzywych fraktalnych $K_D(x, y)$, $x \in S$, $y \in R$ o wymiarze D takich, że $\forall D \in T \exists K_D(x, y)$ oraz $s K_D(x, y) = K_{D'}(sx, sy) \rightarrow D = D'$, gdzie $s \in \mathbb{R}_+$ to współczynnik skalowania. Ostatnia zależność oznacza, że operacja skalowania nie zmienia wymiaru fraktalnego.

Spośród wielu możliwych do zdefiniowania zbiorów krzywych fraktalnych spełniających powyższe warunki wybrano rodzinę krzywych opartych na dobrze znanej krzywej Kocha. Wymiar fraktalny uogólnionych krzywych Kocha zmienia się w sposób ciągły w przedziale od 1 do 2, co pozwala dobrać odpowiedni splajn do każdej sytuacji [2].

Odpowiedni wymiar fraktalny dla uogólnionej krzywej Kocha GKC (*Generalized von Koch Curve*) można obliczyć za pomocą wyrażenia

$$D = \frac{\ln 4}{\ln [2(1 + \cos \varphi)]} \quad (2)$$

W proponowanej metodzie zbiór splajów bazowych, *B-splajny*, który w klasycznej metodzie interpolacji tworzą wielomiany trzeciego stopnia, zastąpiony zostaje zdefiniowanym powyżej zbiorem krzywych fraktalnych $\mathbf{K}$ w postaci GKC – $K_i(x, y)$. Zbiór ten będziemy nazywać FB-splajnami (*Fractal Base Splines*).

Niech będzie dany zbiór $N + 1$ węzłów interpolacyjnych: (x_0, y_0, D_0, W_0^p) , (x_1, y_1, D_1, W_1^p) , ..., (x_N, y_N, D_N, W_N^p) odpowiadających poszczególnym oknom interpolacyjnym LIW. Zakładamy, że okna rozmieszczone są z równym krokiem $x_{i+1} = x_i + \Delta x$, y_i określa położenie środka okna, D_i – wymiar fraktalny interpolowanej krzywej L w i -tym oknie, W_i^p oznacza wektor dodatkowych parametrów ($p = 1, \dots, Z$) charakteryzujących strukturę krzywej w i -tym oknie – na tym etapie rozważań przyjęto $W_i^p = 0$. Ostatecznie

$$L \approx \sum_{i=0}^N s_i K_i(x - i\Delta x, y, D_i, W_i^p) \quad (3)$$

przy czym krzywe fraktalne $K_i(x, y)$ umieszczone są centralnie w punktach interpolacji.

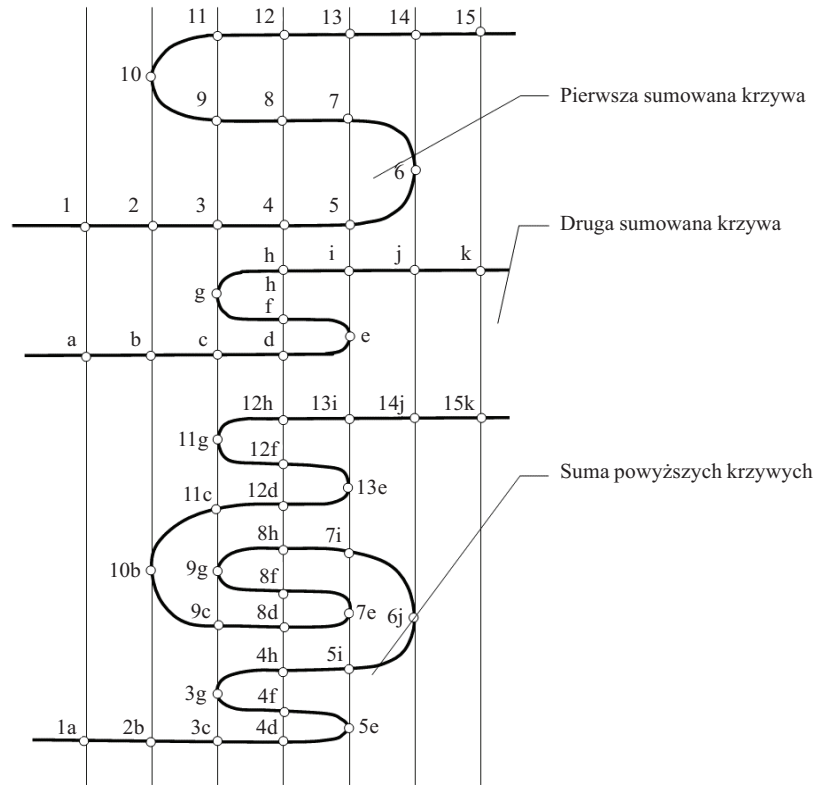
W wyrażeniu (3) należy zwrócić uwagę na to, że w odróżnieniu od klasycznej metody *B-splines*, w której skalowanie następuje tylko względem osi y , tutaj krzywe skalowane są we wszystkich kierunkach. Sumowanie określone zostało symbolicznie jako sumowanie krzywych fraktalnych, natomiast sposób tego sumowania musi zostać dokładnie zdefiniowany. Rozpatrywanie problemu interpolacyjnego w układzie współrzędnych (x, y) narzuca określone zasady – sumowanie musi dotyczyć współrzędnej y przy jednoczesnym traktowaniu współrzędnej x jako zmiennej niezależnej. Krzywa będąca wynikiem sumowania musi zachowywać własności topologiczne krzywych składowych, w szczególności ciągłość. Algorytm sumowania pokazano na rysunku 3. Opiera się on na zasadzie przemieszczania się po krzywej sumowanej, w kolejnym kroku sumowania, z punktu wyjściowego do następnego tylko do najbliższego sąsiada po lewej lub prawej stronie połączonego topologicznie z punktem wyjściowym.

Współczynniki skali s_j^n w n -tym kroku wyliczono iteracyjnie w oparciu o równanie

$$y_i = \sum_{j=0}^N s_j^{n+1} y_j \left(\frac{(i-j)\Delta x}{s_j^n}, D_j \right)^* \quad (4)$$

gdzie:

- s_j^0, y_i – zadane wartości węzłów interpolacyjnych,
- $y_j(x, D_j)$ – zbiór wartości y krzywej $K_j(x, y, D_j)$ o numerze j dla zadanej wartości x , przy czym $K_j(x, y, D_j)$ – startowe FB-splajny ulokowane są centralnie w każdym punkcie interpolacyjnym.



Rys. 3. Algorytm sumowania ciągłych krzywych nie stanowiących zależności funkcyjnych [2]

Do obliczeń przyjmowano rozpiętość krzywych bazowych równą $6\Delta x$. W wyrażeniu (4) uwzględniono możliwość wystąpienia wielu wartości y FB-splajnów przy ustalonym x . Stąd

$$y_j^n(x, D_j)^* = \frac{1}{2} \left[\sup y_j^n(x, D_j) + \inf y_j^n(x, D_j) \right] \quad (5)$$

Miarą dokładności obliczeń jest średni błąd kwadratowy iteracji

$$\delta^n = \sum_{i=0}^N \left(y_i - \sum_{j=0}^N y_j \left((i-j)\Delta x, D_j \right)^* \right)^2 \quad (6)$$

Przy interpolacji należy uwzględnić dobrze znany problem braku danych na końcach interpolowanej krzywej. Wartości te należy zdefiniować dodatkowo.

4. Opis algorytmu FBS

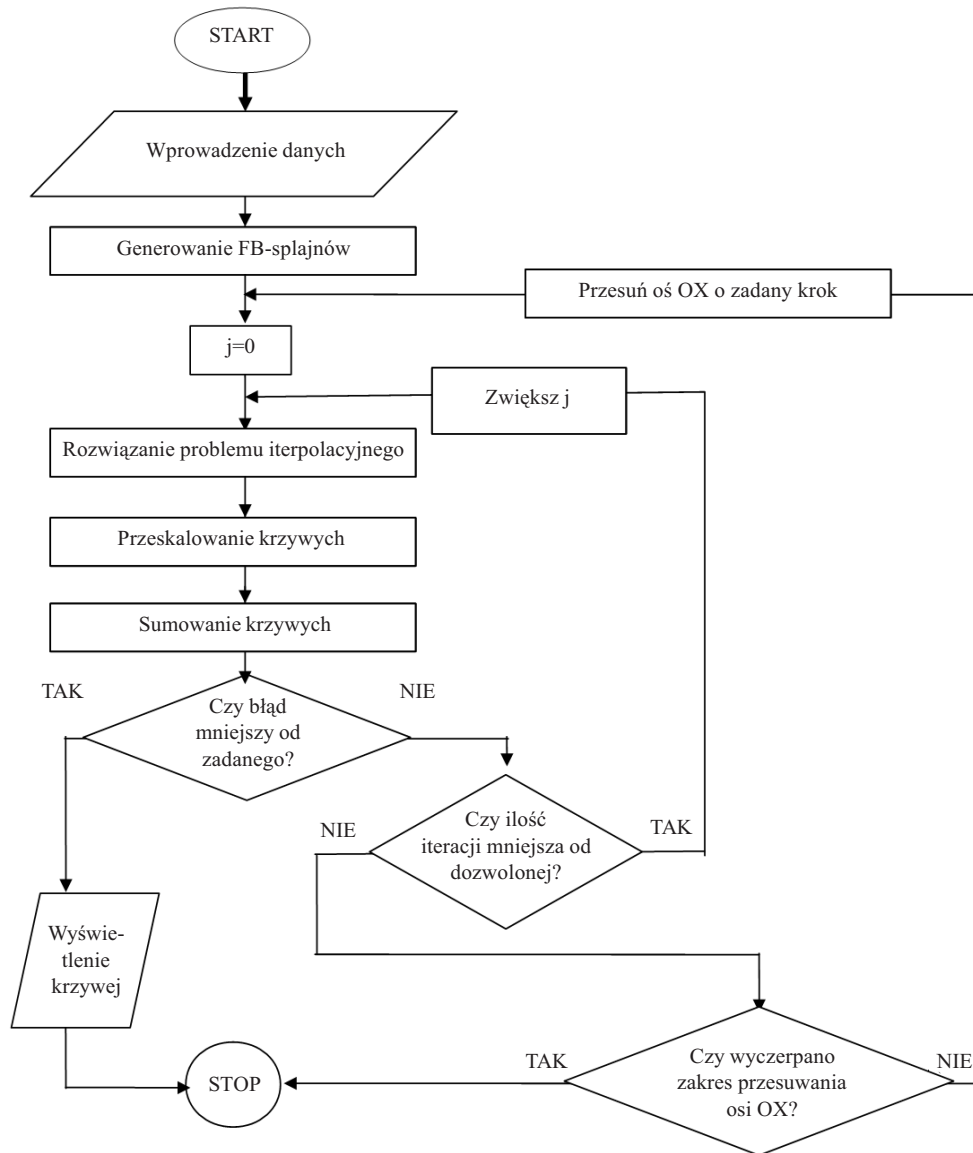
Komputerowy algorytm implementujący metodę fraktalnych splajnów bazowych FBS (*FB-splines*) można podzielić na trzy zasadnicze części. Pierwsza z nich analizuje strukturę interpolowaną ze szczególnym uwzględnieniem charakterystyk nielokalnych. Druga część generuje bazowe elementy przybliżające w procesie iteracyjnym, dobierając taką kombinację liniową elementów bazowych, która z jak największą dokładnością odtworzy zadany kształt krzywej. Wreszcie w ostatnim etapie aplikacja w opisany powyżej sposób sumuje przeskalowane krzywe.

Moduł aplikacji odpowiedzialny za badanie interpolowanej struktury otrzymuje zadane współrzędne węzłów interpolacyjnych. Następnie buduje w ich otoczeniu lokalne okna interpolacyjne i z tych okien pozyskuje charakterystyki nielocalne opisujące krzywą w otoczeniu węzła. Program stosuje tu algorytm wynikający bezpośrednio z podanej przez Kołmogorowa definicji wymiaru pojemnościowego. Obszar lokalnego okna interpolacyjnego dzielony jest na kwadraty o zadanym boku (4, 2 piksele lub 1 piksel). Do obliczania wymiaru brana jest pod uwagę liczba tych obszarów, w których znajdzie się choćby najmniejszy fragment interpolowanej krzywej. Wartością wyjściową jest iloraz logarytmu z liczby bloków i odwrotności rozmiaru bloku (1). Te dane interpretowane są za pomocą metody najmniejszych kwadratów, dając w efekcie wymiar fraktalny oddzielnie w każdym oknie interpolacyjnym.

Na podstawie tak wyznaczonego wymiaru fraktalnego budowane są bazowe elementy przybliżające, czyli jednoparametrowa rodzina krzywych fraktalnych – uogólnione krzywe Kocha. Zgodnie z zależnością (2) istnieje bezpośredni związek pomiędzy wymiarem fraktalnym krzywej a parametrem ϕ uogólnionej krzywej Kocha. W praktyce dla sprawniejszego działania algorytmu, budowana jest tablica wartości wymiaru fraktalnego z zakresu $[1, 2]$ z dokładnością do $1/1000$ w zależności od parametru ϕ . Przeszukanie tej tablicy jest szybsze niż wyznaczenie wartości parametru z podanego wzoru. Na tej podstawie konstruowane są uogólnione krzywe Kocha rozmieszczone centralnie w kolejnych węzłach interpolacyjnych. Początkowa długość FB-splajnu może zostać narzucona przez użytkownika.

Ponieważ wpływ przeskalowanego FB-splajnu na krzywą sumaryczną w innym punkcie niż ekstremum splajnu jest złożoną zależnością nieliniową, to do bezpośredniego wyliczenia algorytm stosuje rozwiązanie iteracyjne – wewnętrzna pętla na schemacie (rys. 4). Każda iteracja polega na zastosowaniu „próbek” wartości splajnów pobranych z poprzedniego kroku iteracyjnego i przeskalowanych o obliczone współczynniki tak, by osiągnąć zadane wartości w węzłach interpolacyjnych. Rozwiązany jest przy tym liniowy układ równań w oparciu o klasyczną metodą eliminacji Gaussa. W ten sposób uzyskiwane są kolejne wartości współczynników interpolacyjnych. Błąd iteracji obliczany jest na podstawie zależności (6).

W przeprowadzonych badaniach stwierdzono, że kształt krzywej interpolującej pomiędzy węzłami silnie zależy od położenia linii bazowej – osi x , względem której FB-splajny są generowane. Właściwy wybór tego położenia zapewnia zbieżność procesu iteracyjnego i optymalne dopasowanie kształtu. Właściwość powyższą można wykorzystać, tworząc dodatkową pętlę iteracyjną w algorytmie dopasowującym – pętlą zewnętrzną na schemacie (rys. 4).



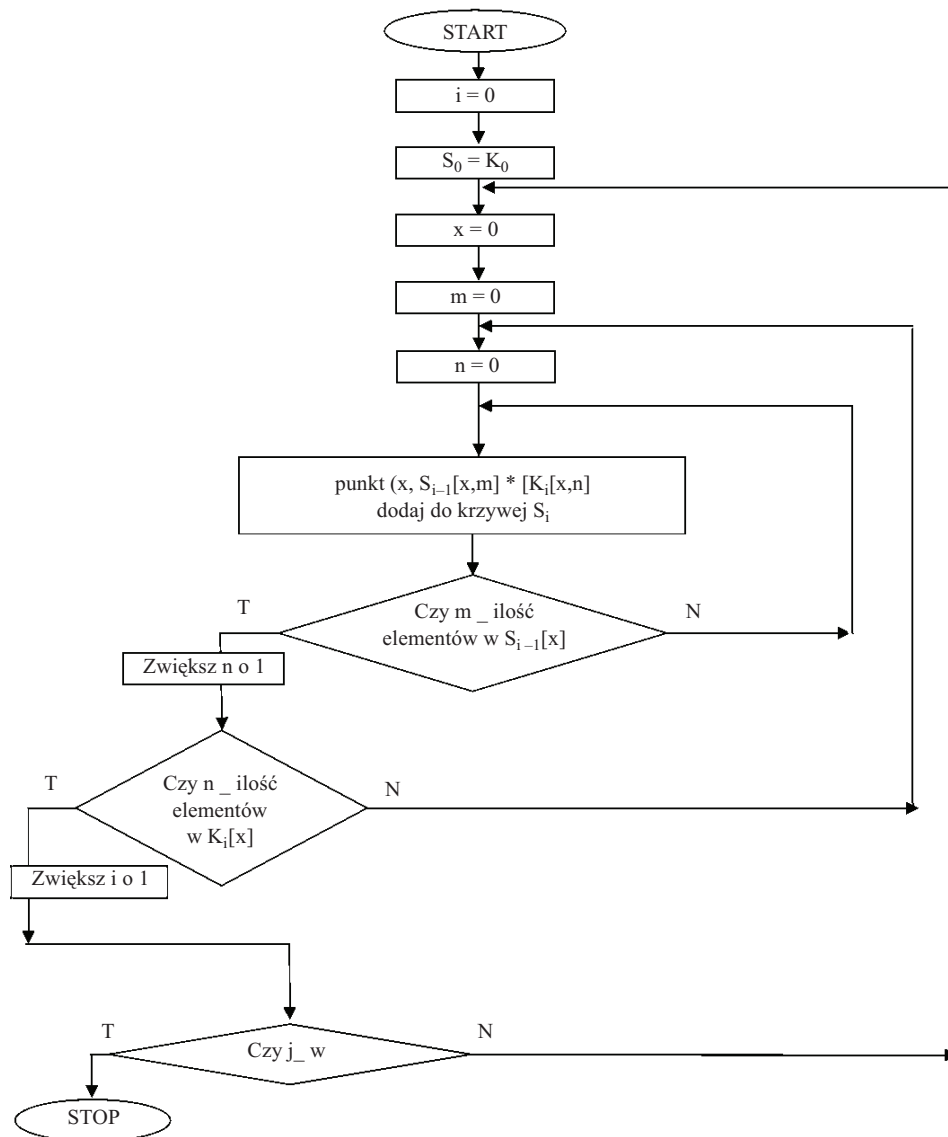
Rys. 4. Schemat blokowy prezentowanego algorytmu interpolacyjnego

Wybór odpowiedniego położenia osi x wynika z minimalizacji średniego błędu kwadratowego liczonego dla wszystkich punktów (pikseli) krzywej

$$\delta(z_i) = \sum_m \left(y_m^{z_i} - y'^{z_i}_m \right)^2 \quad (7)$$

gdzie:

- $y_m^z, y_m'^z$ – wartości y odpowiednio krzywej interpolowanej i krzywej interpolującej, przy czym sumowanie przebiega dla wszystkich wartości (pikseli) x ;
 z_i – określa różne przemieszczenie, w kolejnym kroku iteracyjnym, osi x względem położenia początkowego.



Rys. 5. Algorytm sumowania krzywych fraktalnych

FB-splajny przesuwane są wzdłuż osi y w jedną stronę. Jeśli globalny błąd ulega zmniejszeniu, w kolejnym kroku następuje dalsze przesunięcie osi x . Jeśli natomiast błąd ulega zwiększeniu, algorytm „wycofuje się” z ostatniej modyfikacji przesunięcia i zmniejsza krok dziesięciokrotnie. Iteracje kończą się z chwilą uzyskania satysfakcjonującego dopasowania.

Algorytm sumowania krzywych fraktalnych przedstawiony został na rysunku 5. Sumowane są kolejno pary krzywych, przy czym kolejność sumowania nie jest istotna, gdyż zdefiniowana operacja sumowania jest przemieniana. Sumowanie pary krzywych odbywa się w pętli, której licznikiem jest kolejny numer piksela rzędnej w najmniejszym przedziale zawierającym obie krzywe. Oddzielnie dla każdej wartości rzędnej, w kolejnej pętli obsługiwany jest każdy piksel pierwszej krzywej, którego współrzędna x równa się licznikowi zewnętrznej pętli.

W trzeciej wewnętrznej pętli uwzględnia się kolejne wartości x -pikseli drugiej sumowanej krzywej. Wreszcie dla każdego elementu iloczynu kartezjańskiego podzbiorów pikseli obu krzywych spełniających warunek zgodności wartości rzędnej z wartością ustaloną w zewnętrznej pętli tworzony jest oddzielny piksel o tej samej rzędnej i odciętej będącej sumą odciętych tworzących go pikseli. Tak wygenerowana krzywa interpolująca jest końcowym wynikiem opisywanej metody FB-splajnow.

5. Przykłady zastosowań metody

Wstępna analiza możliwości metody w oparciu o wybrane przykłady pokazuje, że metoda ta może być bardzo efektywnie stosowana do generowania i rekonstrukcji występujących w naturze rzeczywistych kształtów i konturów różnych obiektów oraz przebiegów dynamicznych.

Przykład 1

Często spotykanym przypadkiem jest konieczność generowania krzywej (konturu) o zadanym z góry profilu – kształcie i strukturze wewnętrznej. Miara takiej struktury może być lokalna wielkość, którą będziemy określać mianem chropowatości. Wielkość tę zdefiniować można jako

$$\lambda = \arccos \left(\frac{1}{2} e^{\frac{\ln 4}{D}} - 1 \right),$$

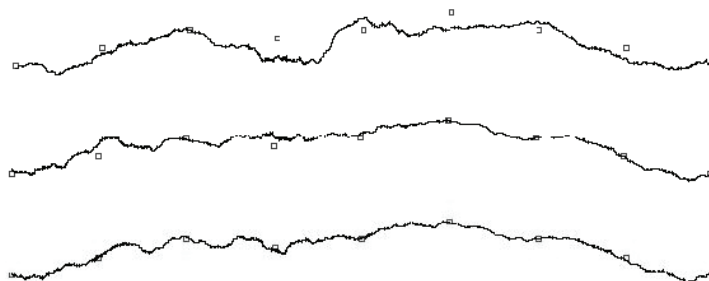
gdzie D – lokalny wymiar fraktalny krzywej.

Pojęcie to jest uogólnieniem wielkości ϕ występującej w definicji GKC. Uogólnienie to jest o tyle zasadne, że w proponowanym algorytmie krzywa o zadanej chropowatości konstruowana jest w oparciu o sumę krzywych GKC.

Poniższy przykład ilustruje zadany problem. W tabeli 1 przedstawiono wymagane parametry krzywej, a jej kształt konstruowany jest w kolejnych iteracjach (rys. 6).

Tabela 1
Wymagane parametry krzywej

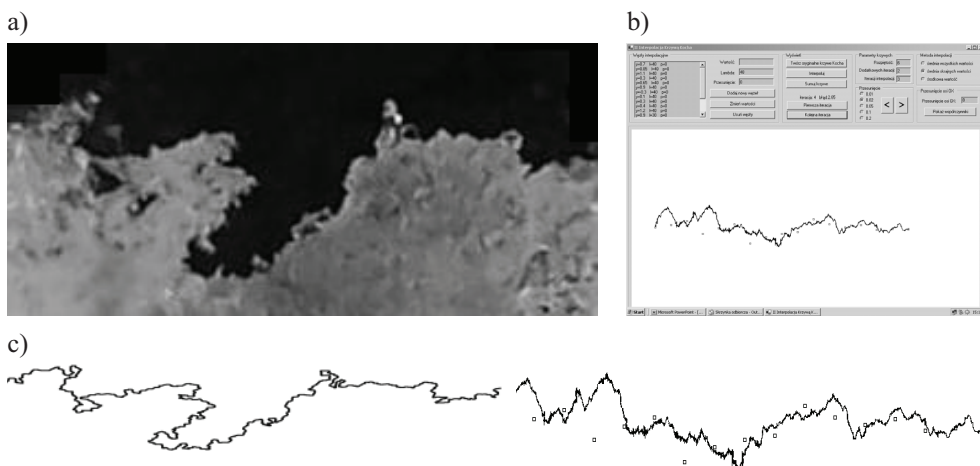
X	0	1	2	3	4	5	6	7	8
Y	0.11	0,3	0,5	0,4	0,5	0,7	0,5	0,3	0,1
Chropowatość	15	15	18	25	18	14	11	14	18



Rys. 6. Krzywa interpolująca po pierwszej, trzeciej i piątej iteracji

Przykład 2

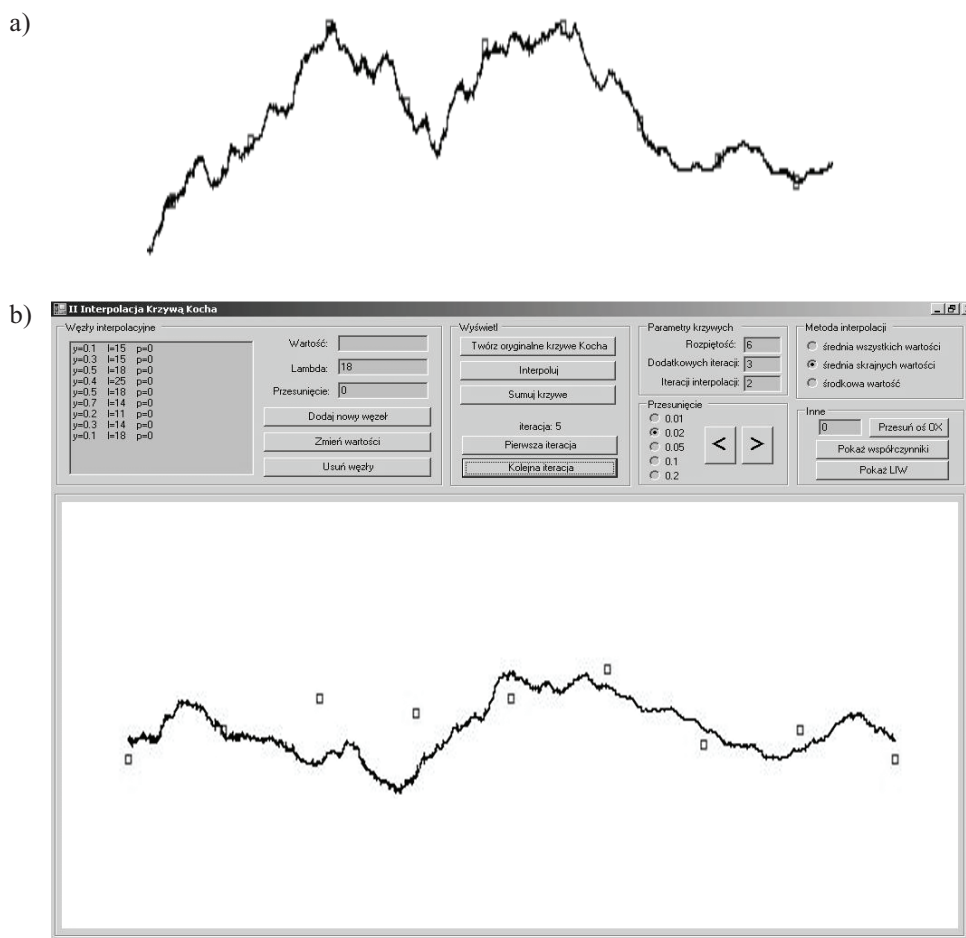
Na rysunku 7 przedstawiono fotografię struktury obrzeża krystalizującego H_2O , natomiast na rysunku 7c za pomocą algorytmu FBS zrekonstruowano wydobyty z niego kontur.



Rys. 7. Struktura obrzeża krystalizującego H_2O : a) fotografia; b) wydobyty kontur; c) zrekonstruowany kształt

Przykład 3

Większości kształtów występujących w świecie naturalnym nie da się zakwalifikować do zbioru elementów geometrii klasycznej. Mają one charakter fraktalny lub pseudofraktalny [3, 4, 5]. Często są to fraktale stochastyczne [6, 7]. Dlatego przedstawioną metodę można stosować zarówno do odtwarzania kształtów w skali mikro, jak i makro. Na rysunku 8 przedstawiono odtworzenie elewacji rzeźby górskiego terenu.



Rys. 8. Linia horyzontu: a) wydobyta z fotografii z obliczonymi wymiarami fraktalnymi; b) zrekonstruowana za pomocą metody FBS

6. Wnioski

Biorąc pod uwagę obiecujące wyniki przedstawione w przykładach, autorzy doszli do wniosku, że metoda powinna być dalej rozwijana. Istnieje wiele dodatkowych nielokalnych

charakterystyk, o które mogłaby być wzbogacana informacja dotycząca węzła interpolacyjnego. Wskazane jest też prowadzenie dalszych badań w celu ustalenia sposobu wykorzystania dodatkowych informacji z otoczenia węzła do rekonstrukcji krzywej. Dlatego prace nad metodą FBS będą kontynuowane. Ponadto już w obecnej fazie doskonalenia metody można pokusić się do zastosowania jej w niektórych rozwiązaniach. W pracy [8] autorzy zaproponowali podjęcie próby stworzenia fraktalnej koncepcji kompresji obrazu wykorzystującej metodę FBS jako podstawowego narzędzia odtwarzającego wykres jasności lub natężenia składowej koloru.

Literatura

- [1] Barnsley M.F.: *Fractal Functions and Interpolation*. Constructive Approximation. 2, 1986, 303–332
- [2] Cader A., Krupski M.: *New interpolation method with fractal curves*. Artificial Intelligence and Soft Computing. Lecture Notes in Artificial Intelligence. Springer 2006
- [3] Mandelbrot B.: *The Fractal Geometry of Nature*. Freeman, San Francisco, CA, 1982
- [4] Barnsley M.F.: *Fractals Everywhere*. 2nd ed., Academic Press, Inc., Boston, 1993
- [5] Kawaguchi Y.: *A morphological study of the form of nature*. Comput. Graph. 16, 3, 1982
- [6] Voss R.: *Random fractals: characterisation and measurement*. In Scaling Phenomena in Disordered Systems, edited by R. Pynn and A. Skjeltorp, New York, Plenum Press, 1985
- [7] Calvet L., Fisher A.: *Multifractality in Asset Returns: Theory and Evidence*. The Review of Economics and Statistics, 84, 3, 2002, 381–406
- [8] Siwocha A., Krupski M., Cader A.: *Koncepcja fraktalnej kompresji obrazu*. Przetwarzanie i analiza sygnałów w systemach wizji i sterowania, Słok k/Bełchatowa, 2006