

Czesław Smutnicki*, Adam Tyński*

Modelowanie przepływu zadań w elastycznym systemie produkcyjnym z wózkami AGV**

1. Wprowadzenie

Coraz wyższe wymagania stawiane przed elastycznymi systemami produkcyjnymi implikują coraz dokładniejsze (także bardziej złożone) modele i doskonalsze algorytmy rozwiązywania. W planowaniu pracy i sterowaniu coraz częściej uwzględnia się elementy takie jak transport i obsługa materiałowa już na etapie harmonogramowania zadań produkcyjnych. Wśród urządzeń transportowych, w systemach elastycznych, rosnącą popularnością cieszą się wózki AGV. Przegląd osiągnięć dotyczących projektowania systemów i harmonogramowania zadań w systemach wykorzystujących AGV znajduje się w pracy [1]. Ze względu na tematykę, na wyszczególnienie zasługują również pozycje [2–4].

Systemy wykorzystujące wózki AGV można różnicować ze względu na trzy podstawowe rozmieszczenia maszyn:

- 1) linia (*single line*),
- 2) siatka (*complex network*),
- 3) pętla (*single loop*).

System cechowany liniowym układem maszyn charakteryzuje się prostotą, jednakże nie wykorzystuje w pełni potencjału niesionego przez wózki. W przypadku układów typu siatka podstawowym problemem wydaje się powstawanie zatorów (lub innych konfliktów transportowych). Jednym ze sposobów zapobiegania zatorom jest dekompozycja takiego układu w zbiór niezależnych systemów typu pętla. Rozwiązanie tego typu rozważane jest w pracach [5–6]. Z reguły, w każdym systemie typu pętla znajduje się stacja załadownicza, stacja wyładownicza, jeden wózek AGV lub większa ich liczba i określona liczba maszyn. Aby uniknąć konfliktów transportowych, narzuca się jednokierunkowy przepływ zadań, zaś na wózkach wymusza się cykliczny, jednokierunkowy tryb pracy. W poszczególnych cyklach pracy każdego wózka do systemu wprowadzane jest jedno zadanie przez stację załadowniczą, jedno zadanie opuszcza system przez stację wyładowniczą, zaś wszystkie pozostałe zadania są transportowane w kierunku kolejnych maszyn w pętli.

* Instytut Informatyki Automatyki i Robotyki, Politechniki Wrocławskiej

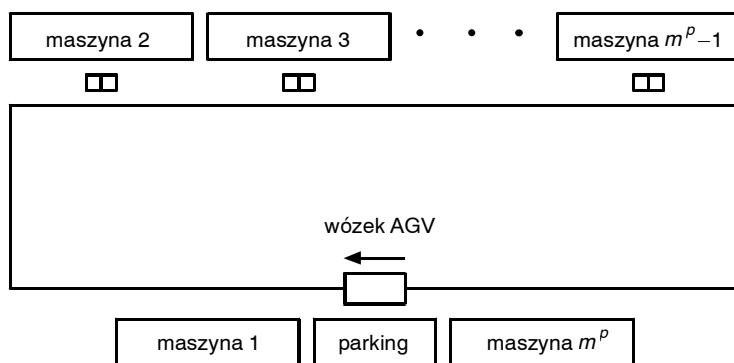
** Praca była finansowana ze środków KBN w latach 2003–2006, jako projekt badawczy nr T11A01624

Systemy typu pętla łączą w sobie prostotę układów typu linia i zalety układów typu siatka, co czyni je szczególnie interesującymi dla badaczy. Pomimo tego faktu, oraz w związku z *NP*-trudnością odpowiedniego problemu optymalizacji, w literaturze wciąż obserwuje się brak odpowiednich metod i narzędzi pozwalających na efektywne rozwiązanie przybliżone poszczególnych instancji problemu.

W niniejszej pracy rozważa się elastyczny system produkcyjny o strukturze przepływowej (permutacyjnej), w którym maszyny zorganizowane są w układ typu pętla. Do transportu zadań pomiędzy maszynami wykorzystuje się pojedynczy jednokierunkowy wózek AGV. Problem polega na określeniu takiej kolejności wykonywania zadań przez maszyny, by moment zakończenia wykonywania procesu technologicznego przyjął wartość minimalną. Dla problemu konstruuje się model matematyczny oraz wprowadza się model permutacyjny – grafowy. Dzięki wprowadzonym reprezentacjom możliwa jest łatwa modyfikacja niektórych algorytmów dedykowanych klasycznemu problemowi przepływowemu, w szczególności algorytmu poszukiwania z zabranieniami z pracy [7], i zastosowanie do przybliżonego rozwiązania poszczególnych instancji rozważanego problemu.

2. Model matematyczny problemu

Problem można sformułować następująco. Dany jest elastyczny system produkcyjny, w którym znajduje się zbiór maszyn produkcyjnych $M^p = \{1, 2, \dots, m^p\}$, gdzie $m^p > 2$ jest liczbą wszystkich maszyn. Każda z maszyn $2, \dots, m^p - 1$ posiada jeden bufor wejściowy i jeden wyjściowy, zaś maszynę 1 i maszynę m^p utożsamia się odpowiednio ze stacją załadowniczą i wyładowniczą o nielimitowanych pojemnościach buforów. Maszyny ze zbioru M^p zorganizowane są w układ typu pętla, tj. rozmieszczone są wzdłuż zamkniętej drogi w ten sposób, że maszyna l , $1 < l \leq m^p$, poprzedzona jest maszyną $l - 1$. Dodatkowo stacja załadownicza (maszyna 1) znajduje się w bezpośredniej bliskości stacji wyładownczej (maszyny m^p), a pomiędzy nimi znajduje się parking dla wózków AGV. Układ tego typu przedstawiony jest na rysunku 1.



Rys. 1. Układ maszyn typu pętla

Z założenia wózki znajdujące się w systemie mogą poruszać się tylko w jednym kierunku, zgodnym z kierunkiem przepływu zadań. Zbiór wózków AGV (utożsamianych z maszynami transportowymi) określa się przez $M^t = \{m^p+1, m^p+2, \dots, m^p + m^t\}$, gdzie m^t jest liczbą wózków. W pracy analizuje się szczególny przypadek, w którym zbiór wózków jest jednoelementowy, tj. $m^t = 1$. Czas przejazdu wózka $v \in M^t$ pomiędzy maszyną l i maszyną $l+1$, $1 \leq l < m^p$, będziemy oznaczać przez $t(l, l+1) > 0$. Podobnie, przez $t(m^p, 1) > 0$ oznaczmy czas przejazdu wózka pomiędzy stacją wyładowniczą i załadowniczą. Jednokierunkowy tryb pracy wózka i sposób rozmieszczenia maszyn jednoznacznie określa czasy przejazdu pomiędzy dowolną parą maszyn produkcyjnych $l, k \in M^p$ i dany jest równaniem:

$$t(l, k) = \begin{cases} \sum_{i=l}^{k-1} t(i, i+1), & l < k \\ dc - \sum_{i=k}^{l-1} t(i, i+1), & l > k \\ 0, & l = k \end{cases} \quad (1)$$

gdzie

$$dc = \sum_{i=1}^{m^p-1} t(i, i+1) + t(m^p, 1) \quad (2)$$

W systemie należy wykonać $r, r > 2$, zadań ze zbioru $J = \{1, 2, \dots, r\}$, polegających na zrealizowaniu naprzemiennej sekwencji operacji produkcyjnych i transportowych. Detal wykonywany w ramach zadania $j \in J$ trafia do systemu poprzez stację załadowniczą, gdzie przytwierdzany jest do palety. Następnie paleta z detalem transportowana jest przez wózek v w kierunku maszyn $2, \dots, m^p$. W stacji wyładowniczej detal demontowany jest z palety i opuszcza system. Bardziej precyzyjnie, każde zadanie $j \in J$ podzielone jest na m^p operacji produkcyjnych i $m^p - 1$ operacji transportowych. Operację produkcyjną l zadania j , notowaną jako para (l, j) , należy wykonać na maszynie $l \in M^p$ w czasie $p(l, j) > 0$; operację, dla której $p(l, j) = 0$ traktujemy jak operację z $p(l, j) = \epsilon > 0$ dla nieskończenie małej wartości ϵ .

Operacja ta polega na:

- (i) pobraniu palety z detalem z bufora wejściowego maszyny,
- (ii) przetworzeniu detalu na maszynie,
- (iii) wyładunku palety do bufora wyjściowego maszyny.

Zbiór wszystkich operacji produkcyjnych oznaczmy przez $O^p = \{(l, j) : l \in M^p, j \in J\}$. Nietrudno zauważyć, że liczność zbioru O^p wynosi $n^p = |O^p| = m^p \cdot r$.

Pomiędzy każdą parą operacji produkcyjnych $(l, j), (l+1, j)$, $1 \leq l < m^p, j \in J$, wykonywana jest operacja transportowa, notowana jako (m^p+l, j) , polegająca na:

- (i) pobraniu przez wózek v palety z detalem z bufora wyjściowego maszyny l ,
- (ii) przewiezieniu jej pomiędzy maszynami $l, l+1 \in M^p$ w czasie $p(m^p+l, j) = t(l, l+1)$,
- (iii) pozostawieniu jej w buforze wejściowym maszyny $l+1$.

W pracy zakładamy, że czas pobrania i pozostawienia palety przez wózek jest pomijalnie mały; założenie to nie jest ograniczające, bowiem jeśli nie jest ono spełnione dla rozpatrywanego systemu, odpowiednie czasy pobrania/pozostawienia mogą być dodane do czasu wykonania operacji transportowej i nie wpływają na prawdziwość dalszych rozważań. Zbiór wszystkich operacji transportowych, wykonywanych przez wózek v , oznaczmy przez $O^t = \{(m^p + l, j) : 1 \leq l < m^p, j \in J\}$. Liczność zbioru O^t wynosi $n^t = |O^t| = (m^p - 1) \cdot r$. Dla kompletności oznaczeń, definiujemy zbiór wszystkich operacji jako $O = O^p \cup O^t$, gdzie $n = |O| = n^p + n^t$.

Przyjęta struktura systemu oraz technika przepływu zadań wymuszają jednakową kolejność przejścia zadań przez każde ze stanowisk, tzw. problem przepływowy permutacyjny. Dalej, niech permutacja $\pi = (\pi(1), \pi(2), \dots, \pi(r))$ oznacza kolejność wykonywania zadań ze zbioru J przez każdą z maszyn produkcyjnych, zaś niech Π będzie zbiorem wszystkich permutacji. Dla permutacji $\pi \in \Pi$ przyjmuje się arbitralną kolejność wykonywania operacji transportowych przez wózek, generowaną na bazie π w sposób opisany dalej (patrz także szczegółowa dyskusja w rozdziale następnym). Wózek v wykonuje $lc = m^p + r - 2$ cykli, gdzie cykl i pracy wózka, $1 \leq i \leq lc$, przebiega następująco. Jeśli $i \leq r$, to wózek czeka przy maszynie 1 na zakończenie się operacji załadowniczej $(1, \pi(i))$, po czym wykonuje operację transportową $(m^p + 1, \pi(i))$ w czasie $p(m^p + 1, \pi(i)) = t(1, 2)$. Jeżeli ostatnie z zadań do wykonania zostało już wprowadzone do systemu, tzn. jeśli $i > r$, wózek wykonuje przejazd pusty (przejazd bez załadunku) w kierunku maszyny 2 (w czasie $t(1, 2)$). Jeśli $1 \leq i - 1 \leq r$, to wózek oczekuje na zakończenie się operacji $(2, \pi(i - 1))$, po czym wykonuje operację transportową $(m^p + 2, \pi(i - 1))$. W przeciwnym razie wózek wykonuje przejazd pusty w kierunku maszyny 3. Proces powtarza się aż do momentu przyjazdu wózka do maszyny m^p . Wykonanie przejazdu pustego pomiędzy stacją wyładowniczą i załadowniczą w czasie $t(m^p, 1)$ kończy cykl.

Dane są operacje transportowe $a = (l, i)$, $b = (k, j)$, $a, b \in O^t$. Przejazdy puste mogą być utożsamiane z przebrojeniami maszyny transportowej wykonywanymi pomiędzy operacjami a, b , w czasie $s(a, b) = t(l - m^p + 1, k - m^p)$.

Na system narzuca się dodatkowe ograniczenia polegające na tym, że:

- (i) w każdej chwili czasu może być wykonywana co najwyżej jedna operacja każdego z zadań,
- (ii) każda maszyna produkcyjna w każdej chwili może wykonywać co najwyżej jedną operację produkcyjną,
- (iii) wózek w każdej chwili może wykonywać co najwyżej jedną operację transportową,
- (iv) rozpoczęta operacja nie może być przerwana.

Uszeregowanie określa się jako zbiór wartości $S(i) \geq 0$, $i \in O$, gdzie $S(i)$ oznacza moment rozpoczęcia wykonywania operacji i . Uszeregowanie jest dopuszczalne, gdy spełnia wszystkie ograniczenia technologiczne narzucone na system. Problem polega na odnalezieniu takiego uszeregowania dopuszczalnego, by moment zakończenia wykonywania procesu technologicznego, równy $S(m^p, \pi(r)) + p(m^p, \pi(r))$, przyjął wartość minimalną.

3. Polityka pracy wózka

Mimo tak prostych założeń dotyczących systemu, permutacja $\pi \in \Pi$ nie określa jednoznacznie kolejności wykonania operacji transportowych przez wózek. Przyjęta w rozdziale 2 arbitralna polityka kolejności realizacji operacji transportowych jest wynikiem sposobu programowania komputera pokładowego wózka AGV i sposobem koordynacji realizacji harmonogramu przez system produkcyjny.

W przyjętym modelu, program w komputerze pokładowym wózka jest bardzo prosty:

1. jedź i zatrzymaj się przy buforze wejściowym maszyny;
2. czekaj na zdarzenie (opróżnienie się bufora wejściowego maszyny);
3. jeśli nie stacja załadowcza, to wyładuj paletę do bufora wejściowego;
4. jeśli nie stacja wyładowcza, to pobierz paletę z bufora wyjściowego;
5. idź do kroku 1.

Jeśli maszyny również rozpoczynają serię automatycznych czynności po wypełnieniu się ich bufora wejściowego, to realizacja zaplanowanego harmonogramu rozpoczyna się automatycznie po dostarczeniu zadań do stacji załadowczej w ustalonej kolejności. W takim systemie ingerencja centralnego komputera, koordynującego na bieżąco pracę maszyn i wózka jest w zasadzie nie potrzebna.

Alternatywnym rozwiązaniem jest użycie kolejności wykonania operacji transportowych jako dodatkowej zmiennej sterującej. Wymaga zastosowania komputera na bieżąco komunikującego się i „informującego” wózek o dalszych czynnościach, które należy wykonać. Takie rozwiązanie jest droższe, bardziej skomplikowane i raczej charakterystyczne dla systemów modelowanych za pomocą gniazdowych problemów szeregowania.

Różnice w efektach zastosowania odmiennych polityk ruchu wózka można prześledzić na krótkim przykładzie. Rozważmy zbiór $r = 2$ zadań oraz system zawierający $m^P = 3$ stanowiska (załadowcze, produkcyjne, wyładowcze) oraz jeden wózek AGV. Czasy wykonywania zadania pierwszego na stanowiskach 1..3 wynoszą odpowiednio (1;1,4). Czasy wykonywania zadania drugiego na stanowiskach 1..3 wynoszą odpowiednio (1;1,1). Zakładamy, że czasy przejazdu wózka pomiędzy każdą parą sąsiednich maszyn są równe jednostki. Przyjmijmy kolejność wprowadzania zadań do systemu $\pi = (1,2)$.

Rozważmy dwie skrajnie odmienne strategie ruchu wózka AGV:

- (a) wózek realizuje wszystkie operacje transportowe zadania pierwszego czekając każdorazowo przy stanowisku na zakończenie przetwarzania zadania, po czym obsługuje w identyczny sposób transport zadania drugiego;
- (b) wózek realizuje operacje transportowe w sposób przeplatany, raz zadania pierwszego, raz drugiego, zgodnie z koncepcją podaną w rozdziale 2.

Termin zakończenia przetwarzania obu zadań według intuicyjnie irracjonalnej strategii (a) wynosi 9, podczas gdy według strategii (b) wynosi 11. Można podać inny przykład, w którym relacja ta jest odwrotna. Stąd wniosek, że kolejność obsługi operacji transportowych przez wózek zależy od danych przykładu i może być optymalizowana podobnie jak

sekwencja wejściowa zadań do systemu. Jak wspomniano, pociąga to za sobą zarówno skomplikowanie modelu, algorytmu optymalizacji i realizacji sterowania wózkiem. Dlatego problem tego typu nie będzie tutaj rozważany.

4. Model permutacyjno-grafowy

Poniżej przedstawimy model permutacyjno-grafowy problemu, wygodny do analizowania własności, jak i konstrukcji metody rozwiązywania. Niech $G(\pi) = (O, E)$ będzie skierowanym grafem reprezentującym dowolną permutację $\pi \in \Pi$ ze zbiorem wierzchołków O i zbiorem łuków $E = E^T \cup E^K \cup E^V \cup E^S$, gdzie:

$$E^T = \bigcup_{j \in J} \bigcup_{l=1}^{m^p-1} \{((l, j), (m^p + l, j)), ((m^p + l, j), (l+1, j))\} \quad (3)$$

$$E^K = \bigcup_{l \in M^p} \bigcup_{j=1}^{r-1} \{((l, j), (l, j+1))\} \quad (4)$$

$$E^V = \bigcup_{j=2}^r \bigcup_{l=1}^{m^p-2} \{((m^p + l, j), (m^p + l+1, j-1))\} \quad (5)$$

Zbiór E^S stanowi sumę zbiorów $E^S = E^{S1} \cup E^{S2} \cup E^{S3}$, określonych następująco:

$$E^{S1} = \bigcup_{z=1}^{k-1} \{((m^p + z, 1), (m^p + 1, z+1))\} \quad (6)$$

$$E^{S2} = \bigcup_{z=1}^{k-2} \{((2 \cdot m^p - 1, r - z), (2 \cdot m^p - z, r))\} \quad (7)$$

$$E^{S3} = \begin{cases} \{((m^p + z, 1), (m^p + z - k + 2, r)) : k \leq z < m^p\}, & r < m^p \\ \{((2 \cdot m^p - 1, z), (m^p + 1, z + k - 1)) : 1 < z \leq r - k + 1\}, & r > m^p \\ \emptyset, & r = m^p \end{cases} \quad (8)$$

gdzie $k = \min\{m^p, r\}$.

Każdy wierzchołek $(l, j) \in O$ ma obciążenie $p(l, \pi(j))$. Łuki ze zbiorów E^T, E^K, E^V mają zerowe obciążenie. Reprezentują odpowiednio kolejność wykonania operacji w zadaniu, kolejność wykonania operacji produkcyjnych przez maszyny produkcyjne i kolejność wykonania operacji transportowych przez wózek. Każdy łuk $(i, j) \in E^S$ ma obciążenie $s(i, j)$.

Łuki te reprezentują przejazdy wózka bez załadunku. Definicja zbioru łuków E gwarantuje acykliczność grafu dla każdej permutacji $\pi \in \Pi$. Oznaczmy przez $r^\pi(i)$, $q^\pi(i)$ najdłuższą ścieżkę odpowiednio dochodzącą i wychodzącą z wierzchołka $i \in O$ bez jego obciążenia $p(i)$ w grafie $G(\pi)$. Nie trudno zauważyć, że wartość $r^\pi(i)$ jest równa najwcześniejszemu możliwemu momentowi rozpoczęcia operacji i , tj. $r^\pi(i) = S(i)$. Zatem rozważany w pracy problem sprowadza się do odnalezienia takiej permutacji $\pi \in \Pi$, że najdłuższa ścieżka (zwana ścieżką krytyczną) w grafie $G(\pi)$, równa $C_{\max}(\pi) = r^\pi(m^p, r) + p(m^p, \pi(r))$, przyjmuje wartość minimalną.

Dzięki zaproponowanemu modelowi permutacyjno-grafowemu można w łatwy sposób zmodyfikować niektóre istniejące algorytmy lokalnego poszukiwania dedykowane klasycznemu problemowi przepływowemu i zastosować do przybliżonego rozwiązania instancji rozpatrywanego problemu. Jednym z przedstawicieli tej grupy algorytmów jest algorytm poszukiwania z zabronieniami, prezentowany w pracy [7].

5. Przykład ilustracyjny

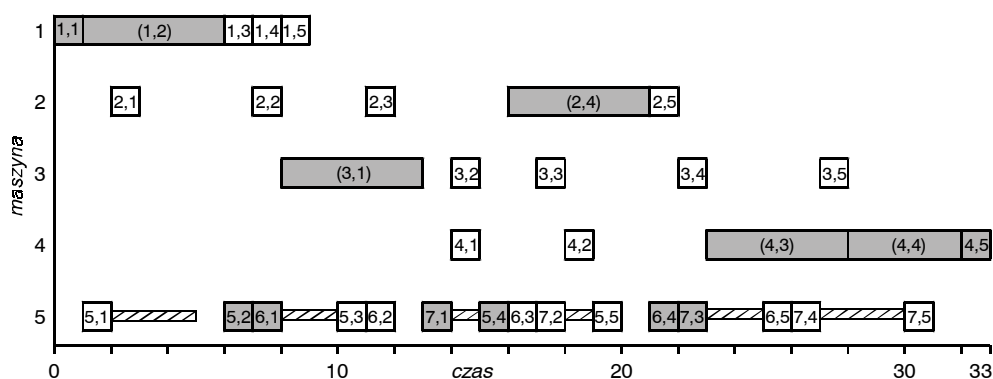
Poniżej zajmujemy się szczegółową analizą przykładowej instancji rozpatrywanego problemu. Dane są $m^p = 4$ ($M^p = \{1, 2, \dots, 4\}$) maszyny produkcyjne zorganizowane w układ typu pętla oraz $m^t = 1$ wózek AGV ($M^t = \{5\}$). Zakłada się jednostkowy czas przejazdu wózka $v \in M^t$ pomiędzy następującymi po sobie maszynami produkcyjnymi, tj. $t(l, l+1) = 1$, $1 \leq l < m^p$ oraz $t(m^p, 1) = 1$. W systemie należy wykonać $r = 5$ zadań ($J = \{1, 2, \dots, 5\}$). Każde zadanie $j \in J$ składa się z naprzemiennej sekwencji $m^p = 4$ operacji produkcyjnych i $m^p - 1 = 3$ operacji transportowych, które należy wykonać w kolejności $(1, j)$, $(5, j)$, $(2, j)$, $(6, j)$, $(3, j)$, $(7, j)$, $(4, j)$. Czasy wykonania wszystkich $n^p = 20$ operacji produkcyjnych zostały podane w tabeli 1. Czasy wykonania wszystkich $n^t = 15$ operacji transportowych wynikają z czasów przejazdu wózka v pomiędzy maszynami produkcyjnymi i wynoszą $p(j) = 1, j \in O^t$.

Tabela 1
Czasy wykonania operacji produkcyjnych

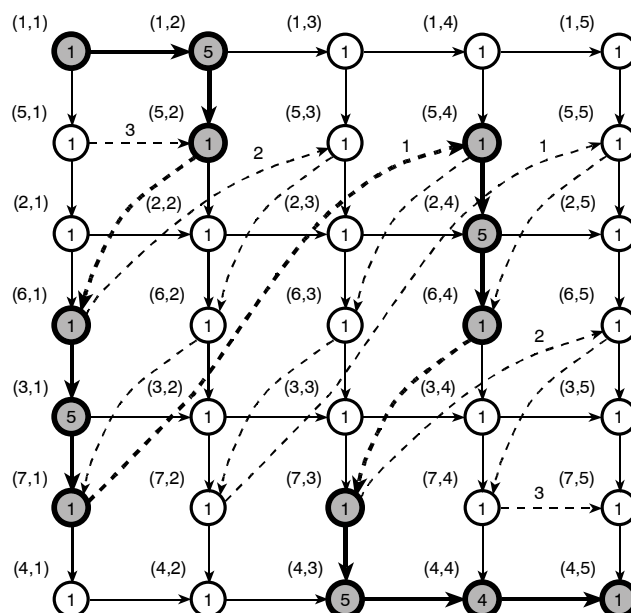
	Zadanie				
	1	2	3	4	5
$p(1, j)$	1	5	1	1	1
$p(2, j)$	1	1	1	5	1
$p(3, j)$	5	1	1	1	1
$p(4, j)$	1	1	5	4	1

Przyjmujemy, że permutacją określającą kolejność wykonania zadań ze zbioru J w systemie jest permutacja $\pi = (1, 2, 3, 4, 5)$. Uszeregowanie zgodne z permutacją π przedstawione jest na rysunku 2. Moment zakończenia wykonywania procesu technologicznego wynosi $S(4, 5) + p(4, 5) = 33$. Operacje determinujące wartość funkcji kryterialnej zostały wyróżnione. Graf $G(\pi)$ ze zbiorem obciążonych wierzchołków O i zbiorem łuków E został przedstawiony na rysunku 3.

Zbiory E^T , E^K określają odpowiednio kolejność wykonania operacji w zadaniach, kolejność wykonania operacji przez maszyny produkcyjne i nie wymagają komentarza. Łuki należące do tych zbiorów oznaczone są linią ciągłą. Zbiory łuków E^V , E^S określają kolejność wykonywania operacji transportowych przez wózek. Łuki należące do tych zbiorów oznaczone są za pomocą linii przerywanej.



Rys. 2. Uszeregowanie zgodne z permutacją π



Rys. 3. Graf $G(\pi)$. Numerację węzłów podano z lewej, obciążenia w węzłach. Krawędzie bez podanego obciążenia mają wagę zero

Łuki ze zbioru E^V określają kolejność wykonania operacji w poszczególnych cyklach pracy wózka. Po każdej sekwencji łuków ze zbioru E^V znajduje się jeden obciążony łuk ze zbioru E^S reprezentujący czasy przejazdów bez załadunku. W zbiorach E^V, E^S znajduje się odpowiednio 8 i 6 elementów.

Wózek wykonuje $lc = 4 + 5 - 2 = 7$ cykli. W pierwszym cyklu pracy, wózek wykonuje tylko jedną operację transportową (5,1). Polega na przetransportowaniu palety z zadaniem 1 pomiędzy maszynami produkcyjnymi 1 i 2. Następnie wózek wykonuje przejazd pusty pomiędzy maszynami 2 i 1 w czasie $s((5,1), (5,2)) = t(5 - 4 + 1, 5 - 4) = t(2,1) = 3$ i rozpoczyna wykonanie operacji transportowej (5,2). Przejazd pusty jest reprezentowany w grafie $G(\pi)$ przez obciążony łuk $((5,1), (5,2)) \in E^S$. W drugim cyklu pracy wózka, po wykonaniu operacji (5,2) wykonywana jest operacja (6,1). Kolejność tą określa łuk $((5,2), (6,1)) \in E^V$. Po zakończeniu operacji (6,1) i przed rozpoczęciem operacji (5,3) następuje kolejny, kończący cykl 2 pracy wózka, przejazd bez załadunku reprezentowany przez łuk $((6,1), (5,3)) \in E^S$ o obciążeniu $s((6,1), (5,3)) = t(56 - 4 + 1, 5 - 4) = t(3,1) = 2$. W trzecim cyklu pracy wózka wykonywane są kolejno operacje (5,3), (6,2), (7,1). Kolejność ta odnotowana jest przez łuki $((5,3), (6,2)), ((6,2), (7,1)) \in E^V$. Po zakończeniu cyklu 3 wózek wykonuje kolejne cykle aż do momentu wykonania ostatniej operacji transportowej (7,5).

Graf $G(\pi)$ skonstruowany jest w ten sposób, że każdy z wierzchołków posiada co najwyżej dwa bezpośrednie poprzedniki i następniki, dzięki czemu stosunkowo łatwo można wyznaczyć wartości $r^\pi(i)$, $q^\pi(i)$ oraz, w konsekwencji, długość ścieżki krytycznej. Długość ścieżki krytycznej w grafie (równa wartości funkcji kryterialnej) wynosi $C_{\max}(\pi) = 33$. Na rysunku 3 łuki i wierzchołki należące do ścieżki krytycznej zostały wyróżnione.

Rozważmy permutację $\delta = (1, 2, 5, 4, 3)$ powstałą z permutacji π po zamianie miejscami elementów $\pi(3)$ i $\pi(5)$. Aby przejść do grafu $G(\delta)$ z grafu $G(\pi)$ wystarczy jedynie zmienić obciążenie części wierzchołków. Obciążenie wierzchołków $(l,3)$, $(l,5)$, $1 \leq l \leq 7$, w grafie $G(\delta)$ wyniesie odpowiednio $p(l, \pi(5))$ i $p(l, \pi(3))$. Warto zauważyć, że przy poszczególnych modyfikacjach grafu nie ma konieczności dokonywania zmian w zbiorze łuków E .

6. Podsumowanie

W niniejszej pracy rozważano elastyczny system produkcyjny o strukturze przepływowej permutacyjnej, w którym maszyny zorganizowane zostały w układ typu pętla. Do transportu zadań pomiędzy maszynami wykorzystano jeden jednokierunkowy wózek AGV. Za kryterium optymalizacji przyjęto moment zakończenia wykonywania procesu technologicznego.

Dla problemu skonstruowano model matematyczny oraz wprowadzono reprezentację permutacyjno-grafową, popartą wyczerpującym przykładem obliczeniowym. Niewątpliwymi zaletami zdefiniowanego grafu jest jego acykliczność oraz łatwość modyfikacji, towarzysząca modyfikacjom reprezentowanych permutacji. Wprowadzony graf umożliwia

również stosunkowo szybkie i „wygodne” wyznaczenie wartości funkcji kryterialnej. Dzięki reprezentacji permutacyjno-grafowej możliwe jest łatwe zaadaptowanie niektórych algorytmów dedykowanych klasycznemu problemowi przepływowemu i zastosowanie do przybliżonego rozwiązania poszczególnych instancji rozważanego problemu.

Literatura

- [1] Ganesharajah T., Hall N., Sriskandarajah C.: *Design and operational issues in AGV-served manufacturing systems*. Annals of Operations Research, 76, 1998, 109
- [2] Hurink J., Knust S.: *Makespan minimization for flow-shop problems with transportation times and a single robot*. Discrete Applied Mathematics, 112, 2001, 199
- [3] Suri R., Desiraju R.: *Performance analysis of flexible manufacturing systems with a single discrete material-handling device*. The International Journal of Flexible Manufacturing Systems, 9, 1997, 223
- [4] Hall N., Sriskandarajah C., Ganesharajah T.: *Operational decisions in AGV-served flowshop loops: scheduling*. Annals of Operations Research, 107, 2001, 161
- [5] Bozer Y.A., Srinivasan M.M.: *Tandem configurations for automated guided vehicle systems and the analysis of single vehicle loops*. IIE Transactions, 23, 1991, 72
- [6] Bozer Y.A., Srinivasan M.M.: *Tandem AGV systems: a partitioning algorithm and performance comparison with conventional AGV systems*. European Journal of Operational Research, 63, 1992, 173
- [7] Nowicki E., Smutnicki C.: *A fast tabu search algorithm for the permutation flow-shop problem*. European Journal of Operational Research, 9, 1996, 160