ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1986
Seria: AUTOMATYKA z. 84 Nr kol. 894

Józef Goetz
Politechnika Wrocławska

MODELOWANIE SZEREGOWANIA OPERACJI Z WYKORZYSTANIEM CZASOWYCH SIECI PETRI

Streszczenie. W pracy przedstawiono sformułowanie i model matematyczny dwóch zagadnień optymalizacji szeregowania operacji wykorzystując czasową sieć Petri. Jedno dotyczy przypadku, gdy zadania mogą być wykonywane cyklicznie co najmniej raz, w drugim - zadania mają być wykonane nieskończoną liczbę razy.

1. Wstęp

Zagadnienia szeregowania operacji w warunkach deterministycznych są w literaturze formułowane i rozwiązywane w sposób statyczny np. [1,5,6,7] Przykładowo wykorzystując do tego celu graf dysjunktywny [5,6,7] nie można na nim zamodelować i rozwiązać zagadnienia cyklicznego wykonywania zadań i zobaczyć aktualny stan wykonania zadań.

W pracy podjęto po raz pierwszy oryginalną próbę sformułowania i zamodelowania dwóch zagadnień optymalizacji szeregowania częściowo uporządkowanych operacji niepodzielnych pochodzących od niezależnych cyklicznych zadań. W pierwszym zagadnieniu jest określone ile razy ma wykonać się każde zadanie. W drugim zagadnieniu każde zadanie wykonuje się nieskończoną liczbę razy. Kryteriami optymalizacji są odpowiednio minimalizacja maksymalnego albo sumarycznego czasu wielokrotnego wykonania zadań i minimalizacja czasu cyklu.

Do osiągnięcia powyższego celu są wykorzystane czasowe sieci Petri (w szczególności czasowe grafy markowane), które służą do opisu cyklicznego działania współbieżnych systemów oraz dają dobrą wizualizację stanu wykonania zadań. W pracy też udowodniono kilka własności i twierdzeń w celu uzasadnienia zastosowanego modelu. Z przedstawioną treścią artykułu mają związek prace [2,3,4,9,10,11].

2. Sformułowanie zagadnienia

Zagadnienie szeregowania operacji P sformułowane jest następująco. Dane są:

(i) zbiór niezależnych różnych typów zadań (zleceń) J składających się ze zbioru częściowo uporządkowanych operacji N^i, i J (pierwotnym modelem każdego typu zadania jest spójny graf skierowany, bezkonturowy),

(ii) funkcja $l : J \longrightarrow \{1,2,\ldots\}$, która przyporządkowuje każdemu typo-

wi zadania liczbę takich samych zadań do wykonania,

(iii) zbiór niepodzielnych operacji $N = \{1,2,\ldots,n\}$ pochodzących od wszystkich zadań,

(iv) relacja $RT \subset N \times N$ reprezentująca porządek technologiczny wykonywania operacji; $\langle i,j\rangle \in RT$ pociąga, że operacja i musi być zakończona przed rozpoczęciem operacji j (operacja j korzysta z wyników operacji i),

(v) zbiór maszyn różnych typów (procesorów wraz z potrzebnymi zasobami dodatkowymi) $M = \{1,2,\ldots,m_1\}$,

(vi) funkcja $F : M \longrightarrow R^N$, która przyporządkowuje każdej maszynie do wykonania podzbiór operacji, $R^N = \{N_1,\ldots,N_{n_1}\}$, $N_p \subset N$, $p = 1,\ldots,n_1$, $n_1 \leqslant m_1$ oraz $N_k \cap N_l = \emptyset$, $k,l \in M$, $k \neq l$,

(vii) funkcja $d : N \longrightarrow \mathcal{R} \times M$, która przyporządkowuje każdej operacji czas przydziału (wykonania) na maszynie, $\mathcal{R}$ - zbiór liczb rzeczywistych.

Z (vi) wynika, że operacja jest przyporządkowana tylko jednej maszynie.

Uszeregowaniem będziemy nazywać takie przyporządkowanie w czasie maszyn do wszystkich operacji, dla którego są spełnione warunki:

(a) każda maszyna może wykonywać w danej chwili co najwyżej jedną operację,

(b) dla każdej operacji $j \in N$ może być przydzielona maszyna $m \in M$ na odcinek czasu $e_j^n - s_j^n \geqslant d_j$, gdzie s_j^n, e_j^n - termin rozpoczęcia i zakończenia operacji j po raz n-ty,

(c) spełnione są ograniczenia technologiczne RT.

Wprowadźmy jeszcze następujące oznaczenia:

$l_i = l(i)$ - liczba zadań do wykonania,

$C_i^{l_i}, (C_i^n)$ - termin zakończenia wykonania zadania $i \in J$ po raz l_i-ty (n-ty).

Sformułujemy dwa nowe zagadnienia szeregowania operacji. Zagadnienie P1 wtedy, gdy zadanie każdego typu może być wykonywane co najmniej 1 raz oraz zagadnienie P2 wtedy, gdy zadania mogą być wykonywane nieskończoną liczbę razy.

Rozwiązanie zagadnienia P1 szeregowania operacji polega na wyznaczeniu wartości zmiennych s_j^n, e_j^n, $j \in N^i$, C_i^n, $n \in l_i$, $i \in J$ minimalizujących funkcję celu F_c przy ograniczeniach:

$$e_j^n - s_j^n \geqslant d_j, \qquad j \in N^i,\ n \in l_i,\ i \in J, \tag{2.1}$$

$$s_j^n - e_k^n \geqslant 0, \qquad \langle k,j\rangle \in RT,\ k,\ j \in N^i,\ n \in l_i,\ i \in J, \tag{2.2}$$

$$(s_j^{n'} - e_k^{n''} \geqslant 0) \vee (s_k^{n''} - e_j^{n'} \geqslant 0),\ n',n'' \in \{n,n+1\},\ i \neq k,\ j \in N^i \cap N_p,$$

$$k \in N^u \cap N_p,\ n \in \{1,2,\ldots,\min\{l_i,l_u\} - 1\},\ u,i \in J,\ N_p \in R^N \tag{2.3}$$

$$C_i^n - e_j^n \geqslant 0, \qquad j \in N^i,\ n \in l_i,\ i \in J, \tag{2.4}$$

$$s_j^n, e_j^n \geqslant 0, \qquad j \in N^i,\ n \in l_i,\ i \in J, \tag{2.5}$$

Sformułowanie zagadnienia P2 szeregowania operacji jest analogiczne za wyjątkiem, że n przyjmuje wartości od 1 do nieskończoności.

Ograniczenie (2.1) wyraża wymaganie, aby różnica między terminem zakończenia i rozpoczęcia operacji $j \in N$ nie była mniejsza niż czas, na jaki została przydzielona maszyna do wykonania tej operacji po raz n-ty. Warunek (2.2) zapewnia zachowanie kolejności wykonania operacji i oraz j w przypadku, gdy są wykonywane po raz n-ty. Ograniczenie (2.3) zapewnia "szeregowe w czasie" wykonanie operacji i oraz j wykorzystujących tę samą maszynę, w przypadku gdy operacje są wykonywane po raz n-ty, lub jedna wykonywana jest po raz n-ty a druga po raz (n+1)-szy. Warunek (2.4) przedstawia naturalny warunek, aby termin zakończenia wykonania zadania po raz n-ty był nie mniejszy od terminu zakończenia jakiejkolwiek operacji tego zadania po raz n-ty.

Ograniczenie (2.1) wiąże się z warunkiem (b) na uszeregowanie operacji, ograniczenie (2.2) z warunkiem (c) a ograniczenie (2.3) z warunkiem (a).

Funkcja celu F_c dla zagadnienia P1 ma jedną z następujących postaci:

$$H_{max} \triangleq \max_{i \in J} C_i^{l_i}, \tag{2.7}$$

$$\sum H \triangleq \sum_{i \in J} C_i^{l_i}. \tag{2.8}$$

Funkcja celu H_{max}, $\sum H$ przedstawia odpowiednio maksymalny i sumaryczny czas wykonania l_i-krotnego zadań typu $i \in J$. Funkcja celu F_c dla zagadnienia P2 jest czasem cyklu γ (szczegółowo jest przedstawiona w dalszej części).

Modelowanie przedstawionych w pracy zagadnień realizowane jest za pomocą żywych i ograniczonych czasowych sieci Petri, w szczególności czasowych grafów markowanych, dlatego najpierw przedstawimy definicje.

3. Definicje

Siecią Petri jest 5 [8]:

$$PN = \langle P, T, F, W, M_0 \rangle,$$

gdzie: S jest zbiorem miejsc, T jest zbiorem przejść ($P \cap T = \emptyset$, $P \cup T \neq \emptyset$); $G = \langle P, T, F \rangle$ jest grafem dwudzielnym, gdzie $F \subset P \times T \cup T \times P$ odpowiada zbiorowi łuków,

$W : F \to \mathbb{N}$ funkcja krotności łuków, $\mathbb{N} = \{0, 1, \ldots\}$;

$M_0 : P \to \mathbb{N}$ jest funkcją markowania początkowego.

Czasową siecią Petri jest:

$$TPN = \langle PN, d \rangle$$

gdzie $d : T \longrightarrow \mathbb{R}^+$ jest funkcją czasu palenia przejść.

Zbiór wejściowych (wyjściowych) przejść dla miejsca $p \in P$ jest

$^{\bullet}p = \{t \in T : W(t,p) \neq 0\}$, $\quad (p^{\bullet} = \{t \in T : W(p,t) \neq 0\})$.

Zbiór wejściowych (wyjściowych) miejsc dla przejścia $t \in T$ jest

$^{\bullet}t = \{p \in P : W(p,t) \neq 0\}$, $\quad (t^{\bullet} = \{p \in P : W(t,p) \neq 0\})$.

Niech $M : P \longrightarrow \mathbb{N}$ jest funkcją markowania przyporządkowującą kropki do miejsc.

Przejście $t \in T$ jest przygotowane, jeśli dla każdego miejsca $p \in {}^{\bullet}t$ markowanie $M(p) \geqslant W(p,t)$. Zapalenie przejścia $t \in T$ może nastąpić, gdy ono jest przygotowane i polega na usunięciu kropek z miejsc wejściowych a dodaniu kropek do miejsc wyjściowych zgodnie z wyrażeniem

$$M'(p) = \begin{cases} M(p) - W(p,t), & (s,t) \in F \text{ w chwili } \tau, \\ M(p) + W(t,p), & (t,s) \in F \text{ w chwili } \tau + d, \end{cases}$$

gdzie: d jest czasem palenia przejścia t.

Oznaczymy $M_j \xrightarrow{t_i} M_{j+1}$ jako przejście od markowania M_j do markowania M_{j+1} po zapaleniu $t_i \in T$. Sekwencja palenia $\sigma = t_{i_1}, \ldots, t_{i_g}$ jest to sekwencja, dla której istnieją markowania $M_r, M_{r+1}, \ldots, M_{r+g}$ spełniające warunek $M_r \xrightarrow{t_i} M_{r+1} \cdots M_{r+g-1} \xrightarrow{t_{i_g}} M_{r+g}$. M_{r+g} nazywa się markowaniem osiągalnym z M_r.

$M \xrightarrow{t}$ może być zapalone z M przejście t.

Sieć Petri jest żywa jeżeli $\forall M' \in [M]\ \forall t\ \exists \sigma\ \ M' \xrightarrow{\sigma} M''$ i $M'' \xrightarrow{t}$, gdzie $[M]$ jest zbiorem osiągalnych markowań z markowania M, tzn. zawsze istnieje sekwencja palenia, która może zapalić każde przejście sieci. Jeżeli sieć Petri jest żywa to jest pozbawiona blokady.

Sieć jest ograniczona, jeżeli $\exists k \in \mathbb{N}\ \forall M \in [M_0]\ \forall p \in P : M(p) \leq k$. W przypadku, gdy markowanie nie przekracza wartości k=1 dla każdego miejsca, wówczas sieć jest bezpieczna.

Czasową sekwencją palenia jest sekwencja palenia z określeniem dla każdego kolejnego elementu (przejścia) tej sekwencji liczby oznaczającej, które jest to palenie wraz z czasem rozpoczęcia tego palenia.

(Czasowa) zwykła siecią Petri jest PN (TPN) taka, że $W(x) = 1$, $x \in F$. (Czasowym) grafem markowanym MG (TMG) jest czasowa zwykła sieć Petri taka, że $\forall p \in P : |{}^{\bullet}p| = |p^{\bullet}| = 1$.

Z definicji dla (czasowego) grafu markowanego wynika, że można jednoznacznie zastąpić każdą czwórkę elementów $\langle {}^{\bullet}p, p \rangle, \langle p, p^{\bullet} \rangle$, p, M(p) pojedynczym łukiem $\langle {}^{\bullet}p, p^{\bullet} \rangle$ z wagą M(p), $p \in P$.

Czysta (czasowa) siecia Petri jest PN taka, że $^{\bullet}t \cap t^{\bullet} = \emptyset$ dla każdego $t \in T$. Dla czystej (czasowej) sieci Petri definiuje się macierz incydencji

$$C = [c_{ij}]_{|P| \times |T|},$$

$$c_{ij} = \begin{cases} -W(p_i,t_j) & \text{jeśli } (p_i,t_j) \in F, \\ +W(t_j,p_i) & \text{jeśli } (t_i,p_j) \in F, \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

Stąd czysta PN może być wyrażona przez $PN = \langle C, M_0 \rangle$. Dla czasowego grafu markowanego, z definicji wynika, że $c_{ij} \in \{-1,1,0\}$, $p \in P$, $t \in T$.

(Czasowa) sieć Petri jest T-inwariantem sieci PN (TPN) wtedy i tylko wtedy, jeżeli istnieje kolumnowy wektor I o wszystkich jego składowych dodatnich taki, że $C \cdot I = 0$. I jest nazywane T-inwariantem. I_i oznacza i-tą składową $I_{|T| \times 1}$.

Niech M i M′ są markowaniami (wektorami kolumnowymi). Dla (czasowej) sekwencji palenia $\sigma = t_{i1},\ldots,t_{iq}$, niech $f(\sigma)$ będzie wektorem palenia, której i-ta składowa jest krotnością palenia się przejścia t_i w sekwencji (za okres $\tau - \tau_0$). Nowe markowanie M′ oblicza się ze wzoru $M' = M + CI$ [8].

Dla T-inwariantu I zachodzi własność (z definicji)

$$M' = M + CI = M,$$

tj. powrotu do markowania wyjściowego. Dla czasowej sieci Petri jest $M' = M(\tau)$, $M = M(\tau_0)$ i $I = I(\tau)$ jest wektorem częstotliwości palenia się poszczególnych przejść za okres $\tau - \tau_0$ [11]. T-inwariant odpowiada wektorowi $f(\sigma)$ dla sekwencji, po której otrzymuje się takie same markowanie. Dla grafu markowanego przestrzeń rozwiązań $C \cdot I = 0$ jest 1 i wektor $I^T = [11\ldots1]$ jest bazą tej przestrzeni.

W grafie markowanym (sieci Petri) sekwencja wierzchołków takich, że każda para sąsiadujących wierzchołków w tej sekwencji należy do łuków grafu markowanego (sieci Petri) skierowanych w jedną stronę i wszystkie wierzchołki tej sekwencji są różne oprócz pierwszego i ostatniego (dla sieci Petri pierwszym i ostatnim jest to samo miejsce) nazywa się obwodem prostym.

Graf markowany (sieć Petri) jest silnie spójna jeśli każda para wierzchołków (miejsc) jest połączona obwodem prostym. Dalsze rozważania będą dotyczyć grafów markowanych silnie spójnych.

Liczba γ jest nazwana czasem cyklu [9] sieci TPN dla T-inwariantu I jeśli istnieje czasowa sekwencja palenia cs taka, że

$$\gamma = \lim_{n \to \infty} \frac{s_i^{nI_i}}{n}, \forall t_i \in T,$$

gdzie $s_i^{nI_i}$ jest czasem, w którym przejście t_i rozpoczyna nI_i palenie w sekwencji cs.

Nas interesuje określenie minimalnego czasu cyklu dla czasowego grafu markowanego.

Twierdzenie 1 [9,10]. Dla czasowego grafu markowanego minimalny czas cyklu jest

$$\gamma = \max_{L_j \in L^*} \frac{T_j}{K_j},$$

taki, że

$$s_i^{n_i} = s_i^1 + (n_i - 1)\cdot\gamma,$$

gdzie:

- $T_j = \sum_{t_i \in T(L_j)} d_i$ jest sumą czasów palenia przejść w obwodzie prostym L_j,
- $K_j = \sum_{p_i \in P(L_j)} M_0(p_i)$ jest liczbą kropek w miejscach w obwodzie prostym L_j,
- L^* jest zbiorem wszystkich obwodów w TMG,
- $s_i^{n_i}$ - jest czasem, w którym przejście t_i rozpoczyna swoje n_i-te palenie.

W [9] jest podana metoda wielomianowa obliczenia minimalnego czasu cyklu γ oraz $s_1^1, \ldots, s_{|T|}^1$ przez sprowadzenie zagadnienia do problemu programowania liniowego.

Dla czasowego grafu markowanego dla T-inwariantu $I = [1,1,\ldots,1]$, $1 \in \mathbb{N}$ minimalny czas cyklu wynosi $\gamma(1) = 1\cdot\gamma$. Zatem funkcje celu (2.7) i (2.8) mają postać $H_{max} = \max_{i \in J} \gamma(1_i)$, $\sum H = \sum_{i \in J} \gamma(1_i)$.

4. Modelowanie i własności modelu

Niech

$$\Gamma_i^{-1} = \{j \in N^i : \forall k \in N^i \wedge \langle k,j\rangle \in RT\},$$

$\Gamma_i = \{j \in N^i : \forall k \in N^i \wedge \langle j,k\rangle \in RT\}$ odpowiednio zbiór operacji nie posiadających poprzedników i następników dla zadania typu $i \in J$.

Zagadnieniu P1 i P2 możemy przyporządkować czasowy graf dysjunktywnie markowany TMDG:

$$D = \langle N', U \cup V;\ d,\ F_u,\ F_v\rangle,$$

gdzie:

$N' = N \cup \{a_i : |\Gamma_i^{-1}| \geq 2,\ i \in J\} \cup \{f_i : |\Gamma_i| \geq 2,\ i \in J\}$ - zbiór wierzchołków, pozostałe wierzchołki $a_i = \{\Gamma_i^{-1} : |\Gamma_i^{-1}| = 1\} \in N$, $f_i = \{\Gamma_i : |\Gamma_i| = 1\} \in N$,

$U = U_1 \cup U_2 \cup U_3 \cup U_4 \cup U_5$ - zbiór łuków,

$U_1 = \{\langle k,j\rangle : \langle k,j\rangle \in RT\}$,

$$U_2 = \bigcup_{i \in J} U_2^i = \bigcup_{i \in J} \{\langle a_i, j\rangle : j \in \Gamma_i^{-1}, |\Gamma_i^{-1}| \geq 2\},$$

$$U_3 = \bigcup_{i \in J} U_3^i = \bigcup_{i \in J} \{\langle j, f_i\rangle : j \in \Gamma_i, |\Gamma_i| \geq 2\},$$

$$U_4 = \{\langle f_i, a_i\rangle : i \in J\},$$

$$U_5 = \{\langle j, j\rangle : j \in N\},$$

$$V = \bigcup_{p \in M} V_p = \bigcup_{p \in M} \{\langle i, j\rangle : i, j \in N_p \times N_p \wedge \sim\exists\, i\, RT^* j \wedge \sim\exists\, j RT^* i\}$$

- zbiór markowanych dysjunktywnie łuków kolejnościowych, gdzie $iRT^*j \leftrightarrow \exists i_1, i_2, \ldots, i_l : iRTi_1, i_1RTi_2, \ldots, i_lRTj$,

$d : N' \longrightarrow \mathcal{R} \times (M \cup \lambda)$, λ - element pusty,

$F_U : U \longrightarrow \mathbb{N}$,

$F_V : V \longrightarrow \{0\} \times \{1\}$.

Czasowy graf dysjunktywnie markowany posiada zbiór wierzchołków odpowiadający "właściwym" operacjom oraz odpowiednio zbiór fikcyjnych wierzchołków początkowych lub końcowych dla każdego typu zadania $i \in J$ w przypadku, gdy były więcej niż 2 wierzchołki początkowe lub więcej niż 2 wierzchołki końcowe w zadaniu.

Łukami TMDG są technologiczne łuki "właściwe" - U_1 oraz łuki "początkowe" - U_2 łączące wierzchołek początkowy (jeżeli sam nie odpowiadał "właściwej" operacji początkowej zadania $i \in J$) z wierzchołkami stanowiącymi operacje początkowe każdego zadania i łuki "końcowe" - U_3 łączące wszystkie wierzchołki stanowiące operacje końcowe i-tego zadania (jeżeli nie istnieje pojedyncza "właściwa" operacja końcowa zadania $i \in J$) z wierzchołkiem f_i, $i \in J$. Łuki sprzężenia zwrotnego - U_4-zamykające obwód kolejnego wykonywania zadań tego samego typu. Łuki U_5 są to pętle własne (modelujące cykl własny).

Łuki V są dysjunktywnie markowanymi łukami kolejnościowymi - występują parami pomiędzy wierzchołkami, dla których nie zachodzi relacja tranzytywna RT^* i operacje będą wykonywane na tej samej maszynie.

Funkcja d przyporządkowuje każdemu wierzchołkowi (operacji) $j \in N$ czas przydziału (wykonania) na maszynie $p \in M$, przy czym fikcyjnym wierzchołkom początkowym wartość 0. Funkcja F_U jest taka, że przyporządkowuje łukom liczby w następujący sposób:

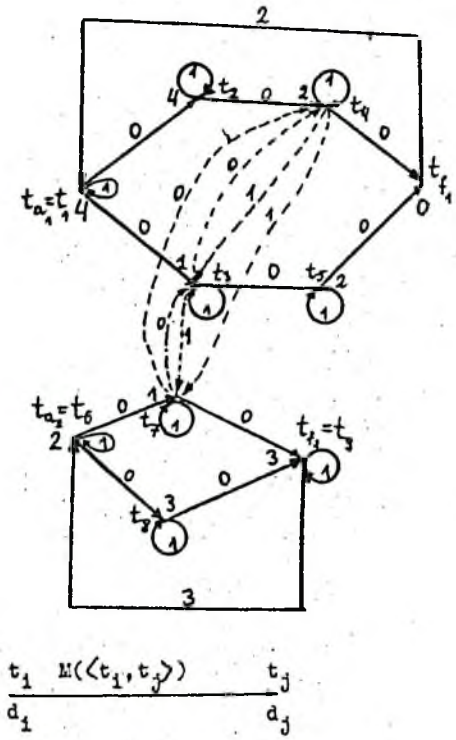
$$F_U(\langle i, j\rangle) = 0, \ \langle i, j\rangle \in U \setminus (U_4 \cup U_5),$$

$$F_U(\langle i, j\rangle) = l_i, \ \langle i, j\rangle \in U_4,$$

$$F_U(\langle j, j\rangle) = 1, \ \langle j, j\rangle \in U_5.$$

Funkcja F_V przyporządkowuje każdemu łukowi V parę liczb $\langle 0, 1\rangle$.

Rysunek 1 modeluje dwa różne zadania, które mają być wykonane $l_1 = 2$, $l_2 = 3$ razy. Zbiór operacji pochodzący od tych zadań $N = \{1,2,\ldots,9\}$, które będą wykonane na zbiorze maszyn $M = \{1,\ldots,7\}$. Kolejnym maszynom zostały przyporządkowane następujące zbiory operacji do wykonania $N_1 = \{1\}$, $N_2 = \{2\}$, $N_3 = \{3,4,7\}$, $N_4 = \{5\}$, $N_5 = \{6\}$, $N_7 = \{8\}$, $N_9 = \{9\}$. Zatem pomiędzy operacjami zbioru N_3 poprowadzone są pary dysjunktywnie markowanych łuków kolejnościowych (linie przerywane). Rys. 1 będzie przedstawiał czasowy graf dysjunktywnie markowany, jeżeli do każdego łuku z V wpiszemy obciążenie $\langle 0,1\rangle$. Markowanie łuków $\langle j,f_i\rangle \in U_3$ jest $M(\langle t_{f_1},t_1\rangle) = l_1 = 2$, $M(\langle t_{f_2},t_6\rangle) = l_2 = 3$. Wierzchołkom są przyporządkowane wartości czasów przydziału d_j, $j \in N$ (z jednoznacznie przyporządkowanym numerem maszyny nieuwidocznionym na rysunku dla przejrzystości).



Rys. 1. Czasowy graf markowany $G = \langle N', U \cup V; d, F_U, F_V^S\rangle$.

Fig. 1. Timed marked graph $G = \langle N', U \cup V; d, F_U, F_V^S\rangle$.

Twierdzenie 2. (Czasowy) graf markowany posiada stałą liczbę kropek w swoich obwodach po każdej sekwencji palenia [10].

Dowód. Kropki w obwodach mogą tylko być dostarczane lub pobierane przez przejścia w swoich obwodach. Jeżeli przejście pobiera kropkę, to ją z powrotem dostarcza do obwodu, co wynika z def. grafu markowanego; dlatego liczba kropek pozostaje taka sama po każdej sekwencji palenia. ж ж ж

Twierdzenie 3. (Czasowy) graf markowany silnie spójny jest żywy wtedy i tylko wtedy, gdy każdy jego obwód prosty zawiera przynajmniej jedną kropkę.

Dowód: $\rightarrow$ (Czasowy) graf markowany silnie spójny jest żywy. Ponieważ jest silnie spójny więc każda para wierzchołków należy do obwodu prostego.

Graf markowany jest żywy, więc każde z jego przejść może być zapalone nieograniczoną liczbę razy. Z powyższego oraz z twierdzenia 2 wynika, że każdy jego obwód prosty zawiera przynajmniej jedną kropkę. Z twierdzenia 2 wynika od razu, że każde przejście obwodu prostego z przynajmniej jedną kropką może zapalać się nieskończoną liczbę razy. ж ж ж

Własność 1. Czasowy graf markowany $G = \langle N', U; d, F_U \rangle$ jest żywy. Z własności (spójny graf skierowany, bezkonturowy) pierwotnego modelu każdego typu zadania oraz uzupełnienia go łukami $U \setminus U_1$ wynika, że otrzymuje się czasowy graf markowany silnie spójny. Stąd na podstawie twierdzenia 2 wynika własność 1.

Własność 2. Dla czasowego grafu markowanego jest

$$s_j^{n+M_{ij}} \geqslant s_i^n + d_i, \quad n \geqslant 1,$$

gdzie: $M_{ij} = M(\langle t_i, t_j \rangle)$, $\langle t_i, t_j \rangle \in E$, d_i - czas trwania palenia przejścia, t_i, $E = \{\langle {}^\bullet p, p^\bullet \rangle : p \in P\}$ - zbiór wszystkich zastępczych łuków.

Własność wynika z następującego rozważania. Aktualne $(n+M_{ij})$-te zapalenie przejścia t_j rozpoczęło się w momencie $s_j^{n+M_{ij}}$ jeśli markowanie aktualne jest nieujemne. Tak więc jeśli q jest liczbą osiągniętych zapaleń przejścia t_i w czasie $s_j^{n+M_{ij}}$ to musi być spełniony warunek $M_{ij}+q - n-M_{ij} \geqslant 0$, stąd $q \geqslant n$. Ale q jest liczbą całkowitą.

Zatem $s_j^{n+M_{ij}} \geqslant s_i^q + d_i$ pociąga własność 2.

Posługując się czasowym grafem dysjunktywnie markowanym można na nim zamodelować wcześniej sformułowane zagadnienia. Ograniczenie (2.1) $e_j^n \geqslant s_j^n + d_j$, $n \geqslant 1$ jest w naturalny sposób zamodelowane (z definicji działania czasowej sieci Petri). Ograniczenie (2.2) jest spełnione na podstawie własności 2 dla $M_{ij}=0$ i ograniczeniu temu odpowiadają łuki U_2.

Ograniczenie (2.3) ma postać dysjunkcji rozłącznych warunków w przypadku wykonywania operacji na tej samej maszynie. Zatem dla par wierzchołków (operacji) pomiędzy którymi nie zachodzi relacja tranzytywna RT^* możemy przyporządkować parę łuków kolejnościowych $\langle k,j \rangle, \langle j,k \rangle$ dysjunktywnie markowanych wektorem $\langle 0,1 \rangle$. W skład zbioru V wchodzą wszystkie dysjunktywnie markowane pary łuków, gdyż jeżeli $\langle k,j \rangle \in V$, to także $\langle j,k \rangle \in V$. Warunkowi (2.3) odpowiada zbiór V taki, że dla każdej pary $\langle k,j \rangle, \langle j,k \rangle$ wybrane jest markowanie ze zbioru $\{0,1\}$ takie, że $M(k,j) + M(j,k) = 1$. Taki zbiór łuków o dopełniającym markowaniu będziemy nazywać reprezentacją markowania S dla zbioru V a funkcję dokonującą wyboru reprezentacji S oznaczymy przez $F_V^S : V \to \{0,1\}$.

Rozwiązaniu dopuszczalnemu (warunkowi (2.3)) odpowiada taka reprezentacja markowania S dla zbioru V, że włączenie zbioru V z jej reprezentacją S do czasowego grafu markowanego (modelującego strukturę zbioru zadań) $G = \langle N', U; d, F_u \rangle$ daje model żywy (czasowy)-graf markowany

$G = \langle N', U \cup V; d, F_U, F_V^S \rangle$. Włączenie innej reprezentacji markowania daje rozwiązanie niedopuszczalne.

Na podstawie twierdzenia 3 wynika, że reprezentacja markowania S musi być tak wybrana, żeby nie utworzyć obwodów prostych o sumie markowania równej Ø. Warunek o dopełniającym markowaniu dla każdej pary przeciwnych łuków należących do V jest warunkiem koniecznym ale niedostatecznym. Niech R_S będzie rodziną wszystkich takich reprezentacji markowań S, które dają żywy graf markowany.

Rys. 1 przedstawia czasowy graf markowany o reprezentacji markowania wybranej tak, że dla $N_3 = \{3,4,7\}$ operacje będą wykonane na maszynie 3 w kolejności 7,3,4. Minimalny czas cyklu wynosi $\gamma = (d_1 + d_2 + d_4 + d_3 + d_5)/3 = 13/3$, d_j - czas wykonania operacji $j \in N$. Dla obwodu t_2, t_2, który stanowi pętlę własną cykl wynosi 4. Przełączenie markowania tak, że $M(\langle t_4, t_3 \rangle)=1$ i $M(\langle t_3, t_4 \rangle)=0$ spowoduje, że minimalny cykl będzie wynosić 6.5.

Rozwiązanie zagadnienia P2 polega na znalezieniu takiej reprezentacji markowania $S \in R_S$ w czasowym grafie dysjunktywnie markowanym $D = \langle N', U \cup V; d, F_U, F_V \rangle$, że czasowy graf markowany $G = \langle N', U \cup V; d, F_U, F_V^S \rangle$ jest żywy i minimalizuje funkcję celu $F_c = \gamma$ przy ograniczeniach (2.1)-(2.5) przy $n \in \{1,2,\ldots\}$.

Rozwiązanie zagadnienia P2 daje rozwiązanie dopuszczalne zagadnienia P1, które może być wyjściowym przy jego optymalnym rozwiązaniu.

Wyznaczenie najwcześniejszych wartości $s_j^n, e_j^n, C_i^n, i \in J, n \geq 1$ dokonuje się ze wzoru tw. 1 albo z grafu zapaleń [4] lub z grafu najwcześniejszych stanów [2].

5. Uwagi końcowe

Przedstawione w pracy nowe zagadnienia optymalizacji wymagają zastosowania metody podziału i ograniczeń na czasowym grafie dysjunktywnie markowanym. Zatem, aby osiągnąć ten cel należy znaleźć efektywne dolne ograniczenie oraz opracować cały algorytm rozwiązania. Wymaga to prowadzenia dalszych badań.

Rozwiązanie przedstawionych problemów oraz uogólnienie ich dla przypadku ograniczonych zasobów powinno być bardzo użyteczne zarówno dla badania efektywności elastycznych systemów produkcyjnych jak i systemów oraz sieci komputerowych.

LITERATURA

[1] Błażewicz J.; Złożoność obliczeniowa algorytmów i problemów szeregowania zadań, Rozprawy Nr 104, Polit. Poznańska, 1979.

[2] Carlier J., Chretienne P.; Studies of Solutions of Scheduling Problems Associated with Timed Petri Nets, 6th European Workshop on Applications and Theory of Petri Nets, June 26-28, 1985, Espoo, Finland, 1-22.

[3] Carlier J., Chretienne Ph., Girault C.; Modeling Scheduling Problems with Timed Petri Nets, 4th European Workshop on Applications and Theory of Petri Nets, Toulouse, France, September 26-29, 1983, 1-19.

[4] Chretienne P., Timed Event Graphs: A Complete Study of their Controlled Executions, IEEE Computer Society Press, International Workshop on Timed Petri Nets, Torino, Italy, July 1-3, 1985.

[5] Grabowski J.; Uogólnione zagadnienia optymalizacji kolejności operacji w dyskretnych systemach produkcyjnych, Prace Naukowe Inst. Cybern. Techn. Polit. Wrocławskiej, Monografie Nr 9, 1979.

[6] Goetz J., Makuchowski M.; Metody przybliżone dla ogólnych zagadnień przydziału i szeregowania operacji, Postępy Cybernetyki, Nr 3, 1985.

[7] Goetz J.; Modelowanie na grafie dysjunktywnym ogólnych zagadnień przydziału zasobów i szeregowania operacji, Zeszyty Naukowe Polit. Śląskiej, Z54, Gliwice 1980.

[8] Jantzen M., Valk R.; Formal Properties of Place/Transition Nets, Lecture Notes in Computer Science, Net Theory and Applications, vol, 84, 1980, 165-212.

[9] Magott J.; Performance Evaluation of Concurrent Systems Using Petri Nets, Information Processing Letters 18 (1984), 7-13.

[10] Ramamoorthy C.V., Ho G.S.; Performance Evaluation of Asynchronous Concurrent Systems Using Petri Nets, IEEE Trans. Software Engineering, SE-6, No 5, 1980, 440-449.

[11] Sifakis J., Performance Evaluation of Systems Using Nets, in: W. Brauer, ed., Net Theory and Applications, Lecture Notes in Computer Science, vol. 84, Springer, Berlin 1980, 307-319.

Recenzent: Prof.dr Tadeusz Puchałka

Wpłynęło do Redakcji do 1986.04.30

МОДЕЛИРОВАНИЕ ПРОБЛЕМЫ РАСПИСАНИЯ ОПЕРАЦИЙ С ИСПОЛЬЗОВАНИЕМ СЕТИ ПЕТРИ

Р е з ю м е

В статье представлена формулировка и математическая модель проблем оптимизации расписания операций с использованием временной сети Петри. Одна проблема касается случая, когда задачи могут быть реализованы циклически, по крайней мере один раз, а во второй задача реализуется бесконечное число раз.

SCHEDULING PROBLEMS MODELING USING TIMED PETRI NETS

S u m m a r y

The formulation and the mathematical model of two optimization problems of operations scheduling with timed Petri nets are presented in the paper. New scheduling optimization problems are repetitive problem (jobs have to be excuted more than once) and recycling problem (each job has to be excuted an infinite number of times). Jobs consist of partially ordered set of operations. The first problem is to find operations schedule minimizing maximum or sum of completion times of repetitive jobs ;the second one is to find minimal cycle time.