

Zbigniew Banaszak
Robert Podsiadły
Instytut Cybernetyki Technicznej
Politechniki Wrocławskiej

MODELE SYNCHRONIZACJI PROCESÓW WSPÓLBIEŻNYCH W ZASTOSOWANIU DO PROCESÓW
PRODUKCYJNYCH

Streszczenie. Problem rozważany w niniejszej pracy sprowadza się do wyznaczania takich ogólnych mechanizmów synchronizacji obiektów systemu produkcyjnego, które zapewniają bezblokadową, współbieżną realizację przebiegających w tym systemie procesów technologicznych. Wyznaczone mechanizmy pozwalają określić warunki wystarczające dla syntezy modeli sieciowych reprezentujących wszystkie bezblokadowe realizacje przebiegu procesów.

1. Wstęp

Podstawowy problem z zakresu programowania systemów sterowania w czasie rzeczywistym wiąże się z wyznaczaniem poprawnie działających algorytmów sterowania. Budowa algorytmów określających zadany sposób funkcjonowania sterowanego systemu jest zadaniem złożonym, trudnym dla metod jego formalnej reprezentacji i weryfikacji. Stosowane metody budowy algorytmów sterowania w czasie rzeczywistym sprowadzają się do iteracyjnego wyznaczania kolejnych wariantów procedury sterowania i badania ich określonych własności w kontekście zadanych wymagań na działanie sterowanego systemu. Uzyskanie wariantu procedury spełniającego zadany test poprawności kończy zwykle proces poszukiwania algorytmu.

Na problem poszukiwania algorytmu sterowania można również spojrzeć jak na zadanie wyznaczania modelu określonego typu zachowania. Podejście takie umożliwia zastąpienie iteracyjnego procesu poszukiwania, postępowaniem mającym na celu wyznaczenie warunków wystarczających dla budowy modelu reprezentującego określoną klasę zachowań. Podanie tego typu warunków stanowi podstawę dla opracowania metod automatycznego modelowania algorytmów sterowania w czasie rzeczywistym.

Rozważany w pracy problem sprowadza się do wyznaczenia mechanizmów warunkujących bezblokadową realizację procesów współbieżnych przebiegających w systemach o ograniczonych zasobach. Zakłada się, że rozważane procesy mają charakter potokowy i realizowane są w sposób asynchroniczny.

Dla rozwiązania tak sformułowanego problemu wykorzystany został formalizm teorii sieci Petriego. Uzyskane wyniki pozwalają na automatyczną syntezę modeli sieciowych reprezentujących procedury bezblokadowego sterowania procesami współbieżnymi. W szczególności, rezultaty te umożliwia-

ją również wyznaczanie minimalnej długości kolejek tworzących się w systemie, w trakcie przebiegu procesów.

Po zestawieniu, w rozdziale 2, podstawowych określeń z zakresu teorii sieci Petriego przedstawione zostało, w rozdziale 3, sformułowanie rozważanego w pracy problemu oraz ilustrujący go przykład. W rozdziale 4, dla wybranej klasy systemów produkcyjnych wyznaczone zostały warunki wystarczające dla syntezy procedur bezblokadowego sterowania przebiegiem procesów współbieżnych. Krótkie uwagi wskazujące na zakres zastosowań uzyskanych rezultatów, jak również na kierunki dalszych badań zebrane zostały w rozdziale 5.

2. Podstawowe definicje i oznaczenia

Znakowana sieć Petriego jest dogodnym narzędziem modelowania procesów dyskretnych, w szczególności procesów przebiegających współbieżnie [4]. Możliwość jej wykorzystania do modelowania zarówno przepływu informacji i/lub materiałów jak i sterowania na różnych poziomach szczegółowości stanowią o jej dużej atrakcyjności.

Definicja 1

Znakowaną siecią Petriego nazywamy uporządkowaną czwórkę $PN=(P, T, E, M_0)$, gdzie:

P - niepusty skończony zbiór miejsc,

T - niepusty skończony zbiór przejść taki, że $P \cap T = \emptyset$,

$E \subset (P \times T) \cup (T \times P)$ - relacja przepływu, $\text{dom}(E) \cup \text{cod}(E) = T \cup P$

$M_0 : P \rightarrow \{0, 1\}$ - funkcja znakowania początkowego.

Obrazem sieci PN jest graf, w którym wierzchołki w postaci okręgów, odpowiadają miejscom (określają stan modelowanego procesu) podczas gdy wierzchołki, w postaci kresiek, przejściom (odpowiadają zdarzeniom występującym w modelowanym procesie).

Definicja 2

Przejście $t \in T$ jest przygotowane do palenia przez znakowanie M w sieci PN jeżeli: $(\forall p \in {}^*t)(M(p) = 1)$ i $(\forall p \in t^*)(M(p) = 0)$ gdzie:

${}^*t = \{p \mid (p, t) \in E\}$, $t^* = \{p \mid (t, p) \in E\}$.

Definicja 3

Funkcja przejścia $\delta(M, t)$ jest określona dla przejścia $t \in T$ przygotowanego do palenia przez znakowanie M , a jej wartość wyznacza zależność

$$M'(p) = \begin{cases} M(p) - 1 & \text{dla } p \in {}^*t \\ M(p) + 1 & \text{dla } p \in t^* \\ M(p) & \text{w pozostałych przypadkach.} \end{cases}$$

Zapalanie przejść w sieci określa dynamikę zmian stanów modelu. Zbiór

wszystkich znakowań osiągalnych ze znakowania początkowego M_0 , oznaczany jest przez $R(M_0)$. Własności tego zbioru warunkują określone cechy sieci.

Definicja 4

Sieć PN jest siecią bezblokadową jeżeli dla każdego $M \in R(M_0)$ istnieje takie $t \in T$, że t jest przygotowane do palenia przez znakowanie M .

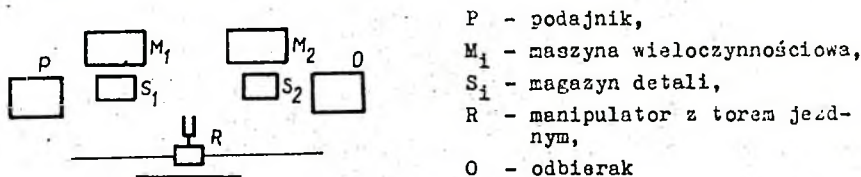
Badanie powyższej, jak i innych szczegółowo opisanych w [2,4,5] własności sieci umożliwia prostą analizę odpowiednich własności, modelowanych za ich pomocą, systemów.

3. Sformułowanie problemu

Rozważmy system produkcyjny złożony ze zbioru maszyn wieloczynnościowych $M = \{M_i \mid i \in N_1^V\}$ wyposażonych w magazyny detali. W systemie tym realizowany jest zbiór procesów potokowych $P = \{P_i \mid i \in N_1^F\}$ przebiegających współbieżnie. Porządki zdarzeń (operacji o niestalonych czasach trwania) każdego procesu zadane są marszrutami lub ciągami technologicznymi. Obiekty systemu mogą występować jednocześnie (również wielokrotnie) w różnych marszrutach.

Rozważany problem sprowadza się do określenia mechanizmów synchronizacji zapewniających bezblokadowy przebieg procesów przy zadanych ograniczeniach na pojemności magazynów detali. Problem ten staje się szczególnie istotny w zadaniu automatycznej syntezy zbioru sterowań dopuszczalnych.

Celem ilustracji problemu rozważmy system przedstawiony na rys. 1



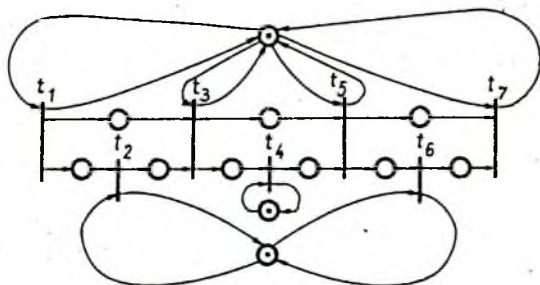
Rys. 1. Gniazdo produkcyjne
Fig. 1. The machining cell

składający się z dwóch maszyn M_1, M_2 wyposażonych odpowiednio w magazyny detali S_1, S_2 ; każdy o pojemności równej 1. Realizowany proces potokowy zadany jest ciągiem technologicznym postaci $CT = t_1, t_2, t_3, t_4, t_5, t_6, t_7$. Przyjmuje się, że operacje transportu międzyoperacyjnego t_1, t_3, t_5, t_7 wykonywane są przez manipulator R, a operacje technologiczne t_2, t_6 i t_4 wykonywane są odpowiednio na maszynie M_1 i M_2 . Wprowadzanie detalu na maszynę M_1 następuje poprzez odpowiedni magazyn S_1 .

Jak łatwo zauważyć, przy niestalonych czasach trwania operacji może dojść do sytuacji, w której oba magazyny są zajęte, przy czym detal znaj-

dujący się w S_2 oczekuje na zwolnienie magazynu S_1 , a detal znajdujący się w magazynie S_1 oczekuje na zwolnienie S_2 . Dla uniknięcia tego typu blokad, mechanizm synchronizacji algorytmu sterowania winien uwzględniać istniejące ograniczenia na pojemności magazynów detali. Model takiego mechanizmu przedstawiony został w rozdziale 4.

Należy ponadto zauważyć, że jeżeli magazyn S_1 posiada pojemność równą 2, wówczas w przebiegu procesu blokady wyżej wymienionego typu nie występują. Założenie to nie jest warunkiem wystarczającym dla zapewnienia bezblokadowej pracy systemu. Albowiem, przy asynchronicznej pracy elementów gniazda może powstać sytuacja, w której np. pierwszy z serii elementów znajdujący się na S_2 oczekuje na przeniesienie do S_1 , podczas gdy kolejny element (transportowany przez R) oczekuje na zwolnienie miejsca w S_2 . Przedstawiona sytuacja ilustruje inny typ blokady procesu.



rys. 2. Model sieciowy przebiegu procesu

Fig. 2. Net model of the process flow

Uniknięcie blokad obu typów zapewnia mechanizm synchronizacji współpracy elementów gniazda, warunkujący pracę manipulatora od stanów maszyn i magazynów, których pojemności zadane są przez liczby powtórzeń związanych z nimi maszyn w ciągu technologicznym. Przykładem takiego rozwiązania jest model sieciowy procesu przedstawiony na rys. 2. Algorytm syntezy tego typu modeli przedstawiony został w pracy [1].

Prezentowane rozwiązanie pozwala na automatyczne wyznaczanie algorytmów sterowania obiektami pracującymi w czasie rzeczywistym [3].

Próbie rozszerzenia uzyskanych wyników na systemy zawierające magazyny o dowolnych pojemnościach podjęto w kolejnym rozdziale.

4. Modele sieciowe procesów współbieżnych

W ostatnim z omawianych w poprzednim rozdziale przykładów rozważany był system zawierający maszyny wieloczynnościowe o zadanej pojemności magazynów. Przyjęto tam implícite, że w magazynach detali wydzielone są obszary zajmowane przez detale znajdujące się w różnych fazach obróbki technologicznej.

Zakładano ponadto, że detal może być wprowadzony w dany obszar magazynu tylko wówczas, gdy jest on zwolniony i maszyna nie pracuje lub obrabia detal pobrany z innego obszaru magazynu. Przy czym detal jest od-

kładany na ten sam obszar magazynu, z którego został pobrany.

Liczba wszystkich możliwych stanów systemu, opisujących położenie detali w obszarach magazynów w różnych fazach obróbki (przed lub po wykonaniu określonej operacji na danej maszynie) określona jest poniższą zależnością:

$$S = 3^a \quad (1)$$

gdzie: a - liczba operacji występujących w ciągach technologicznych.

Łatwo zauważyć, że pokazany na rys. 2 model sieciowy przebiegu procesu odwzorowuje każdy z tych stanów. Oznacza to tym samym, że przyjęty mechanizm synchronizacji obiektów systemu zapewnia odwzorowanie w modelu sieciowym wszystkich bezblokadowych realizacji procesu.

W naturalny sposób powstaje pytanie: jak wyznaczać podobnie działający mechanizm synchronizacji dla systemów o odmiennych sposobach funkcjonowania.

Jako pierwszy, rozważmy system składający się ze zbioru maszyn wieloczynnościowych $M = \{M_i \mid i \in N_1^V\}$ wyposażonych w magazyny detali. Każdy z magazynów S_i zawiera k_i obszarów związanych z odpowiednimi operacjami wykonywanymi na maszynie M_i . W każdym z obszarów mogą być umieszczone co najwyżej dwa detale - jeden przed, drugi po danej fazie obróbki. Detale po określonej fazie obróbki technologicznej mogą być umieszczone w określonym miejscu obszaru tylko wówczas, gdy miejsce to jest zwolnione. W systemie realizowany jest zbiór procesów potokowych $P = \{P_i \mid i \in N_1^F\}$ przebiegających współbieżnie. Porządki operacji wykonywanych nad detalami w każdym z procesów P_i zadane są przez ciągi technologiczne $CT_i = t_{j_1}, t_{j_2},$

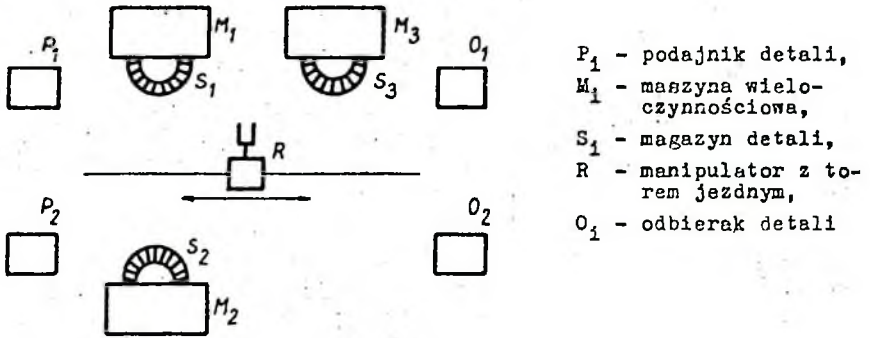
$\dots, t_{j_1}, \dots, t_{j_{g_i}}$, $g_i = 4n+1$, $n \geq 1$. Dla każdego ciągu CT_i elementy zbiorów

$$\{t_{j_{4n+1}} \mid n \in N_0^{(g_i-1)/4-1}\}, \{t_{j_{4n+2}} \mid n \in N_0^{(g_i-1)/4-1}\},$$

$$\{t_{j_{4n+3}} \mid n \in N_0^{(g_i-1)/4-1}\}, \{t_{j_{4n+4}} \mid n \in N_0^{(g_i-1)/4-1}\}$$

interpretowane są odpowiednio jako: operacje transportu międzystanowiskowego i/lub międzyoperacyjnego, operacje wprowadzania detalu w przestrzeń roboczą maszyny, operacje załączania obrabiarki oraz operacje wyprowadzania detalu z przestrzeni roboczej obrabiarki. Przyjmuje się, że obrabiarki wyposażone są w czujniki rozpoznające rodzaj detalu i fazę jego obróbki technologicznej, umożliwiające automatyczny wybór bieżącego programu obróbki.

Dla przykładu rozważmy gniazdo produkcyjne o strukturze jak na rys. 3, w którym realizowane są procesy zadane ciągami technologicznymi $CT_1 = t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}, t_{13}$, $CT_2 = t_{14}, t_{15}, t_{16}, t_{17}, t_{18}, t_{19}, t_{20}$. Operacje t_3, t_7 i t_{11} wiążą się z załączeniami odpowiednich maszyn M_1, M_2 i M_3 .



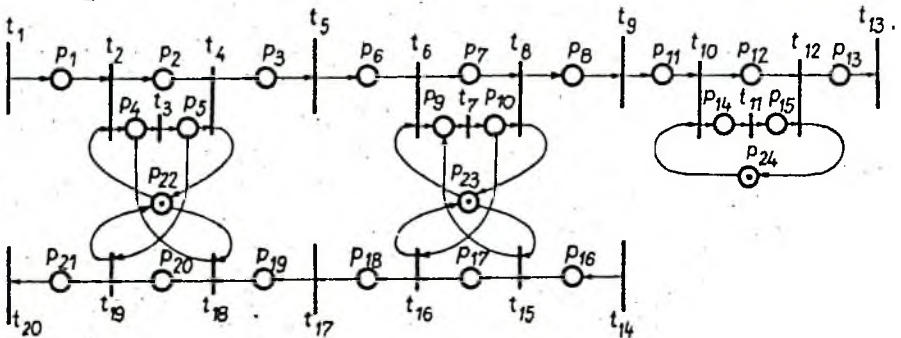
Rys. 3. Gniazdo produkcyjne
 Fig. 3. The machining cell

Modelem wszystkich dopuszczalnych realizacji przebiegu procesu jest sieć przedstawiona na rys. 4. Przejścia sieci interpretowane są jako zdarzenia odnoszące się do odpowiednich operacji transportu detali i załączeń maszyn. Zbiór znakowań osiągalnych ilustruje możliwe stany przebiegu procesu, a m.in. możliwe rozmieszczenia obrabianych detali oraz aktualne fazy ich obróbki technologicznej.

Liczność wszystkich możliwych stanów SP przebiegu procesu określona jest wyrażeniem

$$SP = \prod_{i=1}^v (u_i + 1) 2^{2u_i} \quad (2)$$

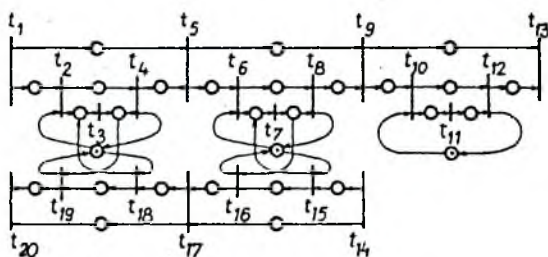
gdzie: v - liczba obrabiarek uczestniczących w realizacji procesu,
 u_i - liczba wszystkich operacji wykonywanych na i -tej maszynie.



Rys. 4. Model sieciowy przebiegu procesu
 Fig. 4. Net model of the process flow

Należy zauważyć, że w zbiorze wszystkich stanów istnieją również stany blokady przebiegu procesu. Przykładem tego jest sytuacja opisana mar-

kowaniem $M = (0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1)$. Jednym ze sposobów zapobieżenia powstawaniu blokad jest przyjęcie podanego na wstępie rozważań ograniczenia, które dopuszcza do umieszczenia detalu w odpowiednich obszarach magazynu tylko wówczas gdy obszar ten jest zwolniony i maszyna nie pracuje lub obrabia detal pobrany z innego obszaru magazynu. Przedstawiony mechanizm synchronizacji pozwala skonstruować bezcukadkowy model przebiegu procesów współbieżnych zadanych przez dowolne zbiory ciągów technologicznych [2]. Dla rozważanego przykładu, postać odpowiedniego modelu sieciowego przedstawiona jest na rys. 5.



Rys. 5. Model sieciowy procesu
Fig. 5. Net model of the process

Łatwo zauważyć, że przyjęcie przedstawionego mechanizmu synchronizacji powoduje dwukrotne zmniejszenie wykorzystania pojemności magazynów, prowadząc do istotnego zmniejszenia liczby realizowanych w systemie stanów. Liczba tak rozważanych stanów określona jest poniższą zależnością

$$SP^I = \prod_{i=1}^v (2^{u_i} + u_i 2^{(u_i-1)}) \quad (3)$$

gdzie: v i u_i określone są w sposób analogiczny jak w wyrażeniu (2).

Przedstawiony mechanizm synchronizacji, jakkolwiek łatwy w implementacji, zbytnio ogranicza liczbę stanów systemu. Już pewna jego modyfikacja dopuszczająca do jednoczesnego zajmowania maszyny (wykonującej tylko jedną operację i tylko w jednym ciągu technologicznym) i obszaru magazynu przez detale znajdujące się w tej samej fazie obróbki (dokładniej mówiąc: znajdujące się tuż przed, lub tuż po danej fazie obróbki) znacznie zwiększa liczbę rozważanych stanów. Odpowiednią zależność przedstawia wyrażenie

$$SP^{II} = 2^{3v'} \prod_{i=1}^{v''} (2^{u_i} + u_i 2^{(u_i-1)}) \quad (4)$$

gdzie: v' - liczba obrabiarek, które wykonują tylko po jednej operacji w tylko jednym ciągu technologicznym,

$v'' = v - v'$, v - liczba obrabiarek występujących w realizacji procesu.

Dalsze modyfikacje mechanizmu synchronizacji dopuszczające do jednoczesnego występowania dwóch lub trzech detali (znajdujących się w tej

sanej fazy obróbki technologicznej) na maszynie i/lub odpowiednich miejscach obszaru magazynu określone są następującymi regułami.

1. Jeżeli w danym ciągu technologicznym występuje z operacji załączania jednej i tej samej maszyny, wówczas dla ostatniej z tych operacji (w sensie uporządkowania w ciągu) dopuszczalna jest sytuacja, w której dwa lub trzy detale zajmują związany z tą operacją obszar magazynu i/lub samą maszynę.

2. Jeżeli w każdym z ciągów CT_1, CT_2 występują operacje załączania maszyn M_k i M_m oraz występowanie operacji załączania maszyny M_k (oznaczenie A) poprzedza w ciągu CT_2 występowanie operacji załączania maszyny M_m (oznaczenie B) w sytuacji gdy występowanie operacji załączania maszyny M_m (oznaczenie C) poprzedza w ciągu CT_1 występowanie operacji załączania maszyny M_k (oznaczenie D), wówczas dla operacji ze zbioru A, B, D lub zbioru B, C, D dopuszczalna jest sytuacja, w której dwa lub trzy detale zajmują związane z tymi operacjami obszary magazynów i/lub same maszyny. Postępowanie takie pozwala związać z każdą z czterech operacji dopuszczalną liczbę elementów mogących jednocześnie zajmować odpowiednie obszary magazynów i samych maszyn. Rac ustalone, wynikające z zastosowania tej reguły, przyporządkowania nie mogą ulegać zmianom w kolejnych etapach jej stosowania.

Wprowadzenie mechanizmu synchronizacji opartego na przyjętych regułach pozwala uwzględnić jeszcze większą liczbę stanów przebiegu procesu. W przypadku ogólnym jest ona określona poniższą zależnością:

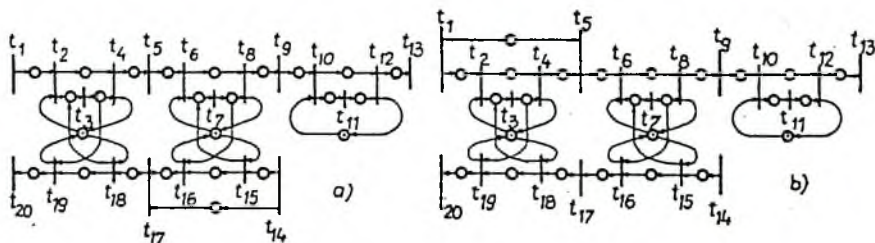
$$SP^{III} = \prod_{i=1}^v 2^{a_i + 2b_i} (1 + a_i/2 + b_i) \quad (5)$$

dzie: u_i - liczba wszystkich operacji wykonywanych na i-tej maszynie,
 b_i - liczba operacji wykonywanych na i-tej maszynie, dla których mechanizm synchronizacji dopuszcza do jednoczesnego występowania dwóch lub trzech detali na odpowiednich obszarach magazynu detali i/lub samej maszyny,

$$a_i = u_i - b_i.$$

Przykładem alternatywnych modeli przebiegu procesu uwzględniających wyżej przedstawione modyfikacje mechanizmu synchronizacji są sieci przedstawione na rys. 6.

Każda z przedstawionych na rys. 6 sieci stanowi pewien alternatywny, bezbłokadowy model przebiegu procesów współbieżnych. Oznacza to m.in., że dla rozważań uwzględniających wszystkie możliwe bezbłokadowe realizacje przebiegu procesu wystarczy wziąć pod uwagę jego wszystkie alternatywne modele sieciowe. Przy tym każdy z modeli odwzorowuje tylko pewien podzbiór zbioru wszystkich stanów systemu obserwowanych przy w pełni asynchronicznym przebiegu procesów.



Rys. 6. Modele sieciowe procesu
Fig. 6. Net models of the process

Przedstawione uwagi wynikają bezpośrednio z poniższych twierdzeń.

Lemat 1 [2]

Dla dowolnego, skończonego zbioru ciągów technologicznych

$$CT = \{CT_i \mid i \in N_1^r\}, CT_i = (t_{j_1}^i \mid 1 \in N_1^{g_i}), g_i = 4n + 1, n \geq 1,$$

istnieje bezblokowa sieć Petriego, o strukturze jak na rys. 7, będąca modelem potokowych procesów współbieżnych określonych przez liniowe porządki operacji ciągów zbioru CT.

Twierdzenie 1

Sieć Petriego o strukturze jak na rys. 7, w której zgodnie z poniższymi warunkami, usunięte zostały pewne miejsca $p_{j_n}^i$ jest siecią bezblokową.

(i) miejsce $p_{j_n}^i$ może być usunięte z sieci, gdy zachodzi

$$(\forall k \in N_1^r)(\forall n' \in N_0^{(g_k-1)/4-1})(k \neq i \ \& \ n' \neq n \longrightarrow t_{4n+3}^i \neq t_{4n'+3}^k) \tag{6}$$

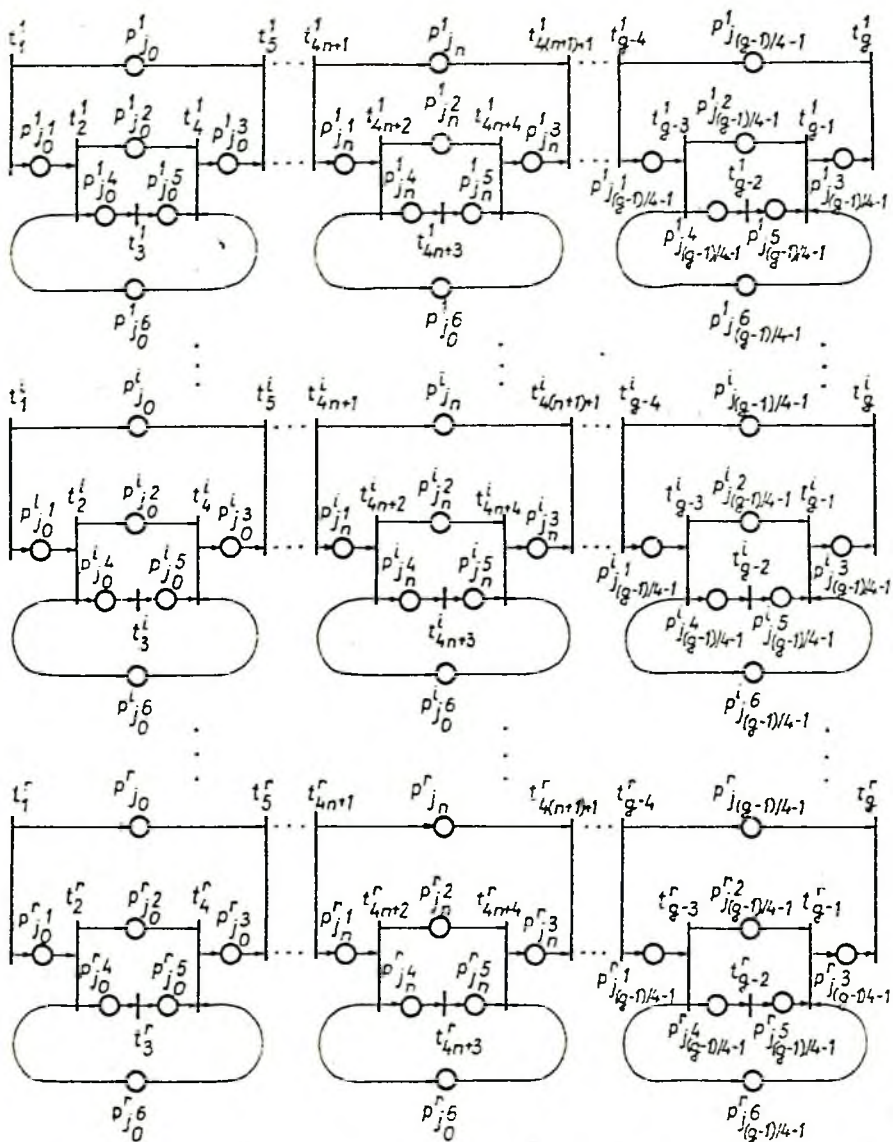
(ii) miejsce $p_{j_n}^i$ może być usunięte z sieci, gdy zachodzi

$$(\forall n' \in N_0^{(g_i-1)/4-1})(n' > n \longrightarrow t_{4n+3}^i \neq t_{4n'+3}^i), \tag{7}$$

(iii) jeżeli $(\exists i, k \in N_1^r)(\exists n, n' \in N_1^{(g_i-1)/4-1})(n^2, n'^2 \in N_1^{(g_k-1)/4-1})$

$$(t_{4n+3}^i = t_{4n^2+3}^k \ \& \ t_{4n'+3}^i = t_{4n'^2+3}^k \ \& \ n < n^1 \ \& \ n^2 < n^3), i \neq k, \tag{8}$$

oraz miejsca należące do zbioru $\{p_{j_n^2}^k, p_{j_n^3}^k, p_{j_n}^i, p_{j_n^1}^i\}$ spełniają warunek (ii) wówczas usunięte z sieci mogą być albo miejsca należące do zbioru $\{p_{j_n^2}^k, p_{j_n^3}^k, p_{j_n}^i\}$ albo miejsca należące do zbioru $\{p_{j_n}^i, p_{j_n^1}^i, p_{j_n^3}^k\}$. Działanie tej reguły jest ograniczone warunkiem mówiącym, że jeżeli miejsce $p_{j_n}^i$ (lub alternatywnie, miejsce $p_{j_n^2}^k$) pozostało nie usunięte (w wyniku wcześniejszego zastosowania warunku (iii)), wówczas każde następane stosowanie tej reguły nie może spowodować usunięcia tego miejsca.



gdzie: $t_{4n+3}^i = t_{4n'+3}^k$, $i, k \in \mathbb{N}_1^T$, $n \in \mathbb{N}_0^{(\mathcal{E}_1-1)/4-1}$, $n' \in \mathbb{N}_0^{(\mathcal{E}_X-1)/4-1}$

implikuje $p_{j_n}^i = p_{j_n'}^k$ & $p_{j_n}^i = p_{j_n'}^k$ & $p_{j_6}^i = p_{j_6'}^k$.

Rys. 7. Struktura beblokadowej sieci Petriego
 Fig. 7. The structure of the deadlock-free Petri net

Dowód:

Dla naszkicowania dowodu powyższego twierdzenia zauważmy co następuje:
Zachodzenie zdarzeń $t_{4n+2}^i, t_{4n+3}^i, t_{4n+4}^i$ w podsieci

$$\{t_{4n+1}^i, t_{4n+2}^i, t_{4n+3}^i, t_{4n+4}^i, t_{4(n+1)+1}^i\}, \{p_{j_n}^1, p_{j_n}^2, p_{j_n}^3, p_{j_n}^4, p_{j_n}^5, p_{j_n}^6\}$$

wyznaczonej przez t_{4n+3}^i spełniający warunek (6) nie wpływa na zachodzenia innych zdarzeń sieci.

Usunięcie z sieci miejsca $p_{j_n}^i$, dla którego zachodzi warunek (7) nie prowadzi do sytuacji, w której zajdzie zdarzenia $t_{4n'+2}^i$ dla $n' > n$, jest bezpośrednio uzależnione od zmiany stanu $M(p_{j_n}^i)$.

Rozważmy miejsca $p_{j_n}^i, p_{j_n}^1, p_{j_n}^k, p_{j_n}^k$ spełniające warunki (7), (8).

Usunięcie z sieci miejsc należących do zbioru $\{p_{j_n}^i, p_{j_n}^1, p_{j_n}^k\}$ nie prowa-

dzi do sytuacji, w której zachodzi $M(p_{j_n^2}^k) = M(p_{j_n^2}^k) = 1$, podczas gdy

$M(p_{j_n^3}^k) = M(p_{j_n}^i) = M(p_{j_n}^i) = M(p_{j_n^1}^i) = 1$. Analogiczne spostrzeżenie odnosi

się do miejsc należących do zbioru $\{p_{j_n^2}^k, p_{j_n^3}^k, p_{j_n^1}^i\}$.

c.b.d.o.

Modele sieciowe przedstawione na rys. 6 reprezentują alternatywne sposoby synchronizacji procesów współbieżnych. Każdy z tych modeli reprezentuje pewien algorytm bezblokadowego sterowania przebiegiem procesów. Uzyskana w ten sposób algorytmy pozwalają na sterowanie obiektami w czasie rzeczywistym [3].

Inny obszar zastosowań stwarzają metody symulacji komputerowej. Przy-
porządkowanie każdemu przejściu w modelu sieciowym odpowiedniego czasu trwania związanej z nim operacji umożliwia symulacyjną analizę dynamiki procesów.

Przedstawiony wyżej mechanizm synchronizacji dotyczył systemów, w których występują maszyny wyposażone w magazyny detali o minimalnych pojemnościach, zadanych liczbami wszystkich operacji wykonywanych na danej maszynie w rozważanych procesach technologicznych. Założenie to nie uwzględnia rzeczywistych ograniczeń nakładanych na pojemności magazynów w systemach produkcyjnych. Można pokazać, że maksymalne zmniejszenie tych wymagań umożliwi mechanizm polegający na tym, że obecność w systemie detalu znajdującego się w dowolnej fazie obróbki technologicznej określonej danym ciągiem technologicznym - pomiędzy pierwszą i ostatnią fazą obróbki detalu na danej maszynie - wstrzymuje wprowadzanie w ten obszar innych detali. Przedstawiony mechanizm ogranicza minimalną pojemność magazynu do liczby współbieżnie realizowanych procesów, w których występuje maszyna związana z danym magazynem. Przykładem ilustrującym działanie tego

LITERATURA

- [1] Banaszak Z., Coordination of concurrent processes: automatic program synthesis, Mat. Konf., 8th European Meeting on Cybernetics and Systems Research, Wiedeń, 1-4 kwietnia, 1986.
- [2] Banaszak Z., Automatic modelling of pipeline concurrent processes, Mat. Konf. 4th IFAC/IFORS Symposium on Large Scale Systems: Theory and Applications, Zurich, 26-29 sierpnia, 1986.
- [3] Banaszak Z., Mazur M., Self-programmable controller of concurrent processes, Prace Naukowe Inst. Cybernetyki Techn. Pwr. Nr 69, seria: Konf., Nr 28, Wrocław 1985, s. 5-12.
- [4] Peterson J.L., Petri net theory and the modelling of systems. Prentice-Hall, New York 1981.
- [5] Reisig W., Petrinetze, Springer-Verlag, Berlin 1982.

Recenzent: Prof.dr h.inż. Stanisław Piasecki

Wpłynęło do Redakcji do 1986.04.30

МОДЕЛИ СИНХРОНИЗАЦИИ ПАРАЛЛЕЛЬНО ПРОТЕКАЮЩИХ ПРОЦЕССОВ

Р е з ю м е

В статье даны достаточные условия для синтеза сетевых моделей взаимосвязанных, одновременно протекающих процессов. Полученные результаты позволяют исследовать, в виде сетевых моделей типа Петри, все возможные беступиковые протекания потоковых технологических процессов. Приведённые примеры иллюстрируют практические применения разработанных механизмов.

MODELS OF CONCURRENT PROCESSES SYNCHRONIZATION

S u m m a r y

The Petri net approach to the automatic synthesis of control procedures ensuring deadlock-free cooperation of concurrently flowing processes is discussed. Net models of the control flows are derived with the help of an algorithm transforming the given process specification into the relevant PT-net model. The main purpose of conducted considerations lies in the formal investigation of the conditions sufficient for the design of a class of a priori regular and deadlock-free PT-nets. Our main problem is stated as follows: Given a finite set concurrent processes executed asynchronously on a system consisting of a finite number of components which may function concurrently. Each process realizes the pipeline-like flow of workpieces along the given routes. Some of the system components may be shared among different processes. Then our task is to prove the synchronization mechanism useful for the automatic design of net models reflecting all admissible, i.e. dead-lock-free controls.

The results obtained allow to cope with the problem of automatic conversion of any process specification reflecting the order operations occurring in the course of process flow, into the corresponding net model of the control flow ensuring a pipeline -like execution of asynchronously flowing processes.