

275/2004

**Raport Badawczy**

**RB/21/2004**

**Research Report**

**Automatyczna synteza  
modeli wydajnościowych  
współbieżnych procesów  
cyklicznych**

**R. Wójcik, P. Majdzik, Z. Banaszak**

**Instytut Badań Systemowych  
Polska Akademia Nauk**

**Systems Research Institute  
Polish Academy of Sciences**



# **POLSKA AKADEMIA NAUK**

## **Instytut Badań Systemowych**

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 8373578

fax: (+48) (22) 8372772

Kierownik Pracowni zgłaszający pracę:  
Prof. dr hab. inż. Zdzisław Bubnicki

Warszawa 2004

# AUTOMATYCZNA SYNTEZA MODELI WYDAJNOŚCIOWYCH WSPÓLBIEŻNYCH PROCESÓW CYKLICZNYCH

Robert Wójcik, Paweł Majdzik, Zbigniew Banaszak

W pracy rozważany jest problem budowy modelu analitycznego systemów dyskretnych procesów cyklicznych współdzielących zasoby zgodnie z zasadą wzajemnego wykluczania. Poprawne działanie systemów tego typu wymaga uwzględniania w procedurach sterowania warunków wystarczających, gwarantujących bezblokadową oraz bezzagłodzeniową pracę systemu. Celem pracy jest przedstawienie, dla pewnej klasy systemów współbieżnych procesów cyklicznych, metody automatycznej syntezy procedur sterowania procesami ze szczególnym uwzględnieniem etapu budowy analitycznego modelu wydajnościowego w oparciu o formalizm  $(\max, +)$  algebry. Formalizm ten daje możliwość analitycznego wyznaczenia wielu charakterystyk funkcjonowania systemu, takich jak harmonogram przebiegu procesów czy okres pracy systemu. Głównym osiągnięciem pracy jest określenie warunków wystarczających, gwarantujących wyznaczenie minimalnego okresu pracy systemu w

oparciu o formalizm  $(\max, +)$  algebry, dla przyjętych reguł rozstrzygania konfliktów zasobowych oraz przyjętego stanu początkowego.

**Słowa kluczowe:** systemy współbieżnych procesów cyklicznych, wzajemne wykluczanie, modelowanie, formalizm  $(\max, +)$  algebry

## 1 WPROWADZENIE

Przedmiotem rozważań jest klasa Systemów Współbieżnych Procesów Cyklicznych będących zbiorem współbieżnych procesów dzielących wspólne zasoby [4], [5]. Przyjmując, że w danym systemie wykonywane procesy są koordynowane w oparciu o protokół wzajemnego wykluczania, poszukiwana jest efektywna obliczeniowo metoda syntezy procedur sterowania rozproszonego gwarantująca m.in. poprawne rozwiązywanie konfliktowych żądań zasobowych procesów (konfliktów zasobowych) oraz bezblokadowe funkcjonowanie systemu. W proponowanym podejściu wykorzystano formalizm  $(\max, +)$  algebry [2], będący naturalnym formalizmem stosowanym w modelowaniu systemów, w których współpraca procesów jest oparta na protokole rendez-vous. Formalizm  $(\max, +)$  algebry daje możliwość analitycznego wyznaczenia wielu charakterystyk funkcjonowania systemu, takich jak harmonogram przebiegu procesów czy okres pracy systemu. W rozważanych w pracy systemach podstawą współpracy procesów jest protokół wzajemnego wykluczania. Stąd jednym z celów pracy jest weryfikacja tezy, że

formalizm (max, +) algebry umożliwia modelowanie systemów, w których procesy wykorzystują zasoby w trybie wzajemnego wykluczania.

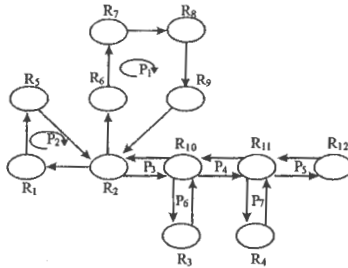
## 2 SYSTEMY PROCESÓW CYKLICZNYCH

System Współbieżnych Procesów Cyklicznych (SWPC) stanowi zbiór wykonujących się współbieżnie procesów cyklicznych, które są związane ze sobą poprzez korzystanie ze wspólnych zasobów (Rys. 1). Przyjęto następujące założenia dotyczące współpracy procesów:

- procesy dzielą wspólne zasoby w trybie wzajemnego wykluczania,
- rozpoczęcie kolejnej operacji procesu następuje natychmiast po zakończeniu operacji bieżącej, pod warunkiem, że istnieje możliwość wykorzystania kolejnego zasobu występującego w sekwencji danego procesu,
- w czasie oczekiwania na zajęty zasób, proces nie zwalnia zasobu przydzielonego do wykonywania poprzedniej operacji,
- proces jest niewyłączalny, tzn. procesowi nie można odebrać zasobu przed ukończeniem operacji, którą wykonuje przy użyciu tego zasobu.

Z powyższych założeń wynika, że ograniczenie uniwersalności rozważanych systemów dotyczy w szczególności modelowania systemów niedeterministycznych (np. sieci komputerowych) ma jednak szerokie zastosowanie w systemach o znanej liczbie procesów, określonej strukturze połączeń procesów i wykorzystywanych przez nie zasobów, w których czasy

wykonania operacji przez poszczególne procesy można wyznaczyć z dużą dokładnością (np. synchroniczne systemy czasu rzeczywistego, Elastyczne Systemy Produkcyjne).



Rys. 1 Reprezentacja graficzna SWPC

Poniżej zdefiniowano pojęcia związane z SWPC, takie jak proces cykliczny, reprezentacja czasowa procesu, sekwencje wykorzystania zasobów dzielonych, stan początkowy systemu.

Proces cykliczny  $P_i$  jest wektorem, którego elementy określają numery zasobów wykorzystywanych przez proces:

$$P_i = (p_{i,1}, p_{i,2}, \dots, p_{i,n}) \quad (1)$$

gdzie:  $p_{i,j}$  oznacza numer zasobu wykorzystywanego przez  $i$ -ty proces w  $j$ -tej operacji.

Reprezentację czasową procesu cyklicznego stanowi wektor, którego elementy określają czasy wykonania poszczególnych operacji przez  $i$ -ty proces:

$$T_i = (t_{i,1}, t_{i,2}, \dots, t_{i,n}) \quad (2)$$

gdzie:  $t_{i,j}$  oznacza czas wykonania przez  $i$ -ty proces  $j$ -tej operacji.

Sekwencję dostępu procesów do zasobów dzielonych stanowi wektor:

$$\Theta = (\sigma_j, \sigma_k, \dots, \sigma_n) \quad (3)$$

gdzie:  $\sigma_i$  jest wektorem  $\sigma_i = (P_j, P_l, \dots, P_r)$ , którego elementy określają kolejność dostępu procesów do  $i$ -tego zasobu dzielonego.

Stan początkowy systemu  $S_0$  jest wektorem, którego elementy określają alokacje procesów na poszczególnych zasobach:

$$S_0 = (P_j, P_k, A, \dots, P_l) \quad (4)$$

gdzie:  $\|S_0\|$  równa jest liczbie zasobów w systemie,  $crd_i S_0 = P_j$  wówczas gdy, na  $i$ -tym zasobie jest w stanie początkowym realizowany  $j$ -ty proces,  $crd_i S_0 = A$  wówczas gdy, na  $i$ -tym zasobie w stanie początkowym nie jest zaalokowany żaden proces.

Z powyższego wynika, że SWPC może być opisany przez dwie  $n$ -ki:  $P^n = (P_1, P_2, \dots, P_n)$ ,  $T^n = (T_1, T_2, \dots, T_n)$ , które określają strukturę systemu oraz parametry procesów składowych, tzn. czasy i kolejność wykonywania operacji przez procesy z wykorzystaniem określonych zasobów. System współbieżnych procesów cyklicznych będzie oznaczany  $S(P^n, T^n)$ .

### **3 SFORMUŁOWANIE PROBLEMU**

Tematyka pracy obejmuje zagadnienia związane z algorytmami syntezy modeli procedur sterowania rozproszonego z wykorzystaniem metod analitycznych oraz wyznaczaniem charakterystyk wydajnościowych przebiegu procesów w systemie. W procedurze budowy modelu analitycznego zastosowano formalizm  $(\max, +)$  algebry [2], który umożliwia wyznaczenie wielu charakterystyk funkcjonowania systemu, takich jak harmonogram przebiegu procesów czy okres pracy systemu. Formalizm  $(\max, +)$  algebry jest naturalnym formalizmem w modelowaniu systemów, w których współpraca procesów jest oparta na protokole *randez-vous*. W rozważanych w pracy SWPC podstawą współpracy procesów jest protokół wzajemnego wykluczania. W wielu pracach [1], [2] pokazano możliwość reprezentacji protokołu wzajemnego wykluczania w oparciu o sekwencje protokołów *randez-vous*. W pracach tych realizacja protokołu wzajemnego wykluczania w oparciu o sekwencje protokołów *randez-vous* sprowadzała się do odpowiedniej specyfikacji procesu koordynującego. Proces koordynujący alokowany na określonym zasobie dzielonym realizuje spotkania z procesami współzawodniczącymi o dostęp do tego zasobu. Liczba operacji realizowanych przez proces koordynujący jest zależna od liczby procesów oraz krotności ich dostępu do zasobu dzielonego w jednym cyklu pracy systemu. Wzajemne wykluczanie w dostępie do danego zasobu dzielonego gwarantuje



sekwencyjność procesu koordynującego. Oznacza to, że w tym samym czasie proces koordynujący może realizować spotkanie tylko z jednym z procesów, co gwarantuje wzajemne wykluczanie procesów w dostępie do zasobu dzielonego. Powyższy model koordynacji procesów wprowadza dodatkowe operacje w systemie związane z reprezentacją w modelu procesów koordynujących dostęp do zasobów dzielonych, co z kolei zwiększa rozmiary macierzy będących składowymi równania stanu. Jednym z celów pracy jest weryfikacja tezy, że formalizm (max, +) algebry przy pewnych założeniach daje możliwość modelowania SWPC, w których procesy wykorzystują zasoby w trybie wzajemnego wykluczania bez konieczności stosowania procesów koordynujących.

Głównym celem pracy jest omówienie w szczegółowy sposób metody budowy modelu analitycznego będącego jednym z etapów algorytmu syntezy procedur sterownia. Stąd problem pracy został sformułowany następująco: Dany jest SWPC, w którym procesy są koordynowane w oparciu o protokół wzajemnego wykluczania. Poszukiwana jest efektywna obliczeniowo automatyczna procedura syntezy jego modelu analitycznego.

#### 4 METODA SYNTEZY PROCEDUR STEROWANIA ROZPROSZONEGO

Metoda syntezy procedur sterowania rozproszonego składa się z następujących etapów:

1. *Specyfikacja parametrów wejściowych procedury.* Wielkościami wejściowymi procedury syntezy są: struktura systemu oraz parametry procesów.
2. *Określenie stanu początkowego i reguł rozstrzygnięcia konfliktów zasobowych (RRKZ).* Warunki wystarczające, gwarantujące bezblokadową i bezagłodziową pracę systemu zostały przedstawione w pracach [3], [4]. W oparciu o te warunki można wyznaczyć dopuszczalne stany początkowe oraz RRKZ, gwarantujące bezblokadową i bezagłodziową pracę systemu.
3. *Budowa modelu analitycznego.* Realizacja tego etapu sprowadza się do wyznaczenia parametrów przebiegu systemu w oparciu o formalizm  $(\max, +)$  algebry, dla określonych w poprzednich etapach RRKZ oraz stanu początkowego.
4. *Ocena jakości funkcjonowania.* Ocena jakości funkcjonowania systemu jest dokonywana w oparciu o wskaźniki ilościowe, takie jak: wskaźnik wykorzystania zasobów, wskaźnik aktywności procesów.
- 5.

#### 4.1 Formalizm (max,+) algebry

##### Definicja 1

(Max, +) algebraiczna struktura  $(R_{max}, \oplus, \otimes)$  jest definiowana następująco:

$R_{max} = R \cup \{-\infty\}$ , gdzie  $R$  jest zbiorem liczb rzeczywistych;

$\oplus$  operator maksimum zgodnie z porządkiem liczb w  $R_{max}$ ,

$\otimes$  operator multiplikatywny - suma w konwencjonalnej algebrze.

##### Definicja 2

Grafem  $G(V,E)$  stowarzyszonym z macierzą kwadratową  $A \in R_{max}^{n \times n}$  nazywamy graf skierowany o  $n$  wierzchołkach, jeżeli krawędź od wierzchołka  $j$  do wierzchołka  $i$  należy do zbioru  $E$ . Krawędź od wierzchołka  $j$  do wierzchołka  $i$  należy do zbioru  $E$  wtedy i tylko wtedy, gdy  $A_{ij} > \varepsilon$ , gdzie  $\varepsilon = -\infty$ , nie należy, gdy  $A_{ij} = \varepsilon$ . Wartość  $A_{ij}$  jest wagą krawędzi  $(j,i)$ .

##### Definicja 3

Wagę  $w_\rho$  ścieżki  $\rho = (e_1, e_2, \dots, e_n)$  nazywamy sumę wag poszczególnych krawędzi.

Ponizej przedstawiono opartą na teorii grafów interpretację  $(\max, +)$  algebraicznego potęgowania macierzy. Niech  $A \in R_{\max}^{n \times n}$ ,  $\rho_{i,j} = \{\rho^1_{i,j}, \rho^2_{i,j}, \dots, \rho^m_{i,j}\}$  będzie zbiorem wszystkich ścieżek w grafie stowarzyszonym z macierzą  $A$ , o długości  $k$ , z wierzchołka  $j$  do wierzchołka  $i$ , a  $(w^1_{i,j}, w^2_{i,j}, \dots, w^1_{i,j})$  wagami tych ścieżek. Jeżeli  $k \in N_0$ , to

$$(A^k)_{i,j} = \max(w^1_{i,j}, w^2_{i,j}, \dots, w^1_{i,j}) \quad \text{dla } i, j \in \{1, 2, \dots, n\} \quad (5)$$

Z powyższego wynika, że  $(A^k)_{i,j}$  jest maksymalną wagą wszystkich ścieżek w grafie  $G(V, E)$  stowarzyszonym z macierzą  $A$ , o długości  $k$ , o wierzchołku początkowym  $j$  oraz wierzchołku końcowym  $i$ . Stąd, jeżeli nie istnieje ścieżka o długości  $k$  z wierzchołka  $j$  do wierzchołka  $i$ , to element  $(A^k)_{i,j}$  jest równy  $\varepsilon$ .

#### Własność 4

Niech  $A \in R_{\max}^{n \times n}$ . Jeżeli istnieje liczba  $\lambda \in R$  i wektor  $x \in R_{\max}^n$ ,  $x \neq [\varepsilon, \varepsilon, \dots, \varepsilon]$  taki, że

$$A \otimes x = \lambda \otimes x \quad (6)$$

to  $\lambda$  jest  $(\max, +)$  algebraiczną wartością własną macierzy  $A$ , a  $x$  odpowiada  $(\max, +)$  algebraicznemu wektorowi własnemu macierzy  $A$ .

#### Definicja 5

Każda macierz kwadratowa  $A \in R_{\max}^{n \times n}$  posiada co najmniej jedną wartość własną.

(Max, +) algebraiczna wartość własna posiada następującą interpretację w teorii grafów. Niech  $A \in R_{max}^{n \times m}$ . Jeżeli  $\lambda_{max}$  jest maksymalną średnią wagą spośród wszystkich elementarnych cykli grafu  $G(V, E)$ , stowarzyszonego z macierzą  $A$ , to  $\lambda_{max}$  jest (max, +) algebraiczną wartością własną macierzy  $A$ .

#### **4.2 Budowa modelu analitycznego**

Formalizm (max, +) algebry jest naturalnym formalizmem w modelowaniu systemów, w których współpraca procesów jest oparta na protokole rendez-vous. W punkcie 3 opisano możliwość reprezentacji protokołu wzajemnego wykluczania w oparciu o sekwencje protokołów rendez-vous. Ten model koordynacji procesów wprowadza dodatkowe operacje w systemie związane z reprezentacją procesów koordynujących, zwiększając jednocześnie rozmiary macierzy będących składowymi równania stanu. W przedstawionej metodzie protokół wzajemnego wykluczania jest reprezentowany w modelu bez alokacji procesów koordynujących. Wielkościami wejściowymi procedury budowy modelu analitycznego są: struktura systemu, parametry procesów oraz stan początkowy i reguły rozstrzygnięcia konfliktów zasobowych. Wyznaczenie tych wielkości jest przedmiotem dwóch początkowych etapów algorytmu syntezy procedur sterowania rozproszonego w oparciu o warunki wystarczające, gwarantujące bezblokadową i bezzagłodzeniową pracę systemu.

Procedura budowy modelu analitycznego składa się z następujących etapów:

**Etap 1. Specyfikacja wektora operacji i czasów oraz wektora stanu.**

Wektor operacji, którego rozmiar jest równy liczbie operacji wykonywanych w systemie jest definiowany:  $p = (p_{1,1}, p_{1,2}, \dots, p_{1,m_1}, p_{2,1}, p_{2,2}, \dots, p_{2,m_2}, \dots, p_{n,1}, p_{n,2}, \dots, p_{n,m_n})$ , gdzie  $p_{i,j}$  - oznacza  $j$ -tą operację  $i$ -tego procesu. Wektor czasów wykonywania operacji jest równy:  $t = (t_{1,1}, t_{1,2}, \dots, t_{1,m_1}, t_{2,1}, t_{2,2}, \dots, t_{2,m_2}, \dots, t_{n,1}, t_{n,2}, \dots, t_{n,m_n})$ , gdzie  $t_{i,j}$  oznacza czas wykonywania  $j$ -tej operacji przez  $i$ -ty proces,  $m_i = L(p_i)$ , natomiast  $L(p)$  - oznacza rozmiar wektora  $p$ . Wektor stanu jest definiowany:  $x(k) = (x_1(k), x_2(k), \dots, x_i(k), \dots, x_{L(p)}(k))$ , gdzie  $x_i(k)$  reprezentuje operacje  $crd p$ , natomiast wartość  $x_i(k)$  oznacza moment rozpoczęcia wykonywania operacji  $crd p$  w  $k$ -tej iteracji. Określony na tym etapie porządek operacji będzie obowiązującym porządkiem w następnych krokach procedury.

**Etap 2. Wyznaczenie okresu pracy systemu  $T$  oraz wektora momentów rozpoczęć operacji w zerowej iteracji  $x_0$ .**

Określenie składowych równania stanu:

$$\begin{aligned} x(k) &= Ax(k-1) \oplus Bx(k) \\ x(0) &= x_0 \end{aligned} \quad (7)$$

gdzie:  $A$  i  $B$  - macierze o rozmiarach  $n \times n$  ( $n$  - rozmiar wektora stanu).

Warunki konstrukcji macierzy  $A$  i  $B$  przedstawiono w punkcie 4.3. Równanie stanu (7) jest równoważne równaniu [2]:

$$\begin{aligned} x(k) &= Dx(k-1) \\ x(0) &= x_0 \end{aligned} \quad (8)$$

gdzie:

$$D = \left( \bigoplus_{i=0}^{\infty} B^i \right) A \quad (9)$$

Jeżeli przez  $x(k)$  oznaczono poszukiwany rozkład rozpoczęć operacji w  $k$ -tej iteracji, to podlega on równaniu:

$$\begin{aligned} x(k) &= Tx(k-1) \\ x(0) &= x_0 \end{aligned} \quad (10)$$

Jednocześnie wektor  $x(k)$  musi spełniać równanie stanu (7), co prowadzi do równania spektralnego macierzy  $D$

$$x(k) = Dx(k-1) = Tx(k-1) \quad (11)$$

gdzie:  $x(k)$  - wektor własny macierzy  $D$ ,  $T$  - wartość własna macierzy  $D$ .

### 4.3 Warunki konstrukcji macierzy

W niniejszym punkcie przedstawiono algorytmy konstrukcji składowych macierzy równania stanu (7). Etapem poprzedzającym wyznaczenie elementów macierzy  $A$  i  $B$  jest konstrukcja macierzy  $C$ . Struktura macierzy  $C$  zależy od struktury systemu i reprezentuje warunki jakie muszą być spełnione, aby poszczególne operacje mogły się wykonywać zgodnie z przyjętymi RRKZ. Macierze  $A$  i  $B$  powstają w wyniku podziału macierzy  $C$ . Podział ten i w konsekwencji struktura macierzy  $A$  i  $B$  zależy od przyjętego stanu początkowego systemu. Oznacza to, że mając wyznaczoną macierz  $C$  dla danej struktury systemu oraz danych RRKZ można wyznaczyć parametry równania

stanu dla różnych stanów początkowych. Należy podkreślić, że konstrukcja macierzy zgodnie z przedstawionymi warunkami gwarantuje spełnienie założeń dotyczących współpracy procesów przedstawionych w punkcie 2 oraz wyznaczenie przebiegu cyklicznego o najkrótszym okresie dla przyjętych sekwencji dostępu do zasobu dzielonego.

$$C_{i,j} = \begin{cases} t_{l,k} & \text{if } * \\ e & \text{if } ** \\ \varepsilon & \text{w pozostałych przypadkach} \end{cases}$$

$$*) \exists p_{l,k} \in p \qquad **) \forall r \in \{1,2,3,\dots,L(\Theta)\}, \quad q = \text{idx}(\text{crd}_r, \Theta)$$

$$i = j + 1 \quad \text{if } (j \bmod \sum_{a=1}^l L(P_a)) \neq 0 \qquad i = \sum_{l=1}^{a-1} L(P_l) + b,$$

$$i = \sum_{a=1}^l L(P_a) + 1 \quad \text{if } (j \bmod \sum_{a=1}^l L(P_a)) = 0 \qquad a = \text{idx}(\text{crd}_{k+1} \sigma_q), \text{crd}_b(\text{crd}_{k+1} \sigma_q) = q$$

$$j = \sum_{l=1}^{c-1} L(P_l) + d + l,$$

$$c = \text{idx}(\text{crd}_k \sigma_q), \text{crd}_d(\text{crd}_k \sigma_q) = q$$

gdzie:  $k = l \bmod m$ ,  $m = L(\sigma_q)$ ,  $l \in \{1, 2, 3, \dots, m\}$ ,  $\text{idx}(\text{crd}_i K)$  - wartość indeksu  $i$ -tego elementu wektora  $K$ .

Warunek \*) związany jest z porządkiem operacji wykonywanych przez dany proces. Z warunku tego wynika, że  $C_{i,j} = t_{i,k}$  wtedy i tylko wtedy, gdy  $j$ -ta operacja poprzedza wykonanie  $i$ -tej operacji, gdzie  $t_{i,k}$  - czas wykonywania  $j$ -tej operacji. Innymi słowy dany proces może rozpocząć  $i$ -tą operację jeśli zakończy wykonywanie  $j$ -tej operacji trwającej  $t_{i,k}$ .



Warunek \*\*) związany jest z reprezentacją w macierzy  $C$  protokołów wzajemnego wykluczania, tzn. z reprezentacją przyjętych sekwencji dostępu do zasobów dzielonych. Dany proces może rozpocząć operację na zasobie dzielnym jeśli proces poprzedzający go w sekwencji dostępu do tego zasobu rozpocznie wykonywanie operacji z wykorzystaniem kolejnego zasobu. Z warunku tego wynika, że  $C_{i,j} = e = 0$  wtedy i tylko wtedy, gdy rozpoczęcie  $i$ -tej operacji jest warunkiem koniecznym rozpoczęcia  $j$ -tej operacji.

W Twierdzeniu 1 określono warunki konstrukcji macierzy  $A$  i  $B$  w oparciu o macierz  $C$  oraz stan początkowy systemu  $S_0$ .

### Twierdzenie 1

Dana jest macierz  $C$  oraz stan początkowy  $S_0$ . Jeżeli są spełnione poniższe warunki konstrukcji macierzy  $A$  i  $B$ , to wyznaczony z równania (11) okres  $T$  jest najkrótszy.

$$A_{i,j} = \begin{cases} C_{i,j} & \text{if } * \\ C_{i,j} & \text{if } ** \\ \varepsilon & \text{w pozostałych przypadkach} \end{cases} \quad B_{i,j} = \begin{cases} \varepsilon & \text{dla } A_{i,j} = C_{i,j} \\ C_{i,j} & \text{dla pozostałych} \end{cases}$$

\*)  $\forall q, \text{crd}_q S_0 \neq A$ ,  
 $j = \sum_{l=1}^{a-1} L(P_l) + b$ ,  
 $i = \sum_{k=1}^{a-1} L(P_k) + b \bmod L(P_k) + 1$ ,  
 $a = i \text{ dx}(\text{crd}_q S_0), \text{crd}_b(\text{crd}_q S_0) = q$

\*\*)  $\exists q, \text{crd}_q S_0 = \Lambda \cap R_q \in RD$   
 $i = \sum_{l=1}^{a-1} L(P_l) + b$ ,  
 $a = \text{idx}(\text{crd}_1 \sigma_q), \text{crd}_b(\text{crd}_1 \sigma_q) = q$   
 $j = \sum_{l=1}^{c-1} L(P_l) + d \bmod L(P_l) + 1$ ,  
 $c = \text{idx}(\text{crd}_m \sigma_q), m = L(\sigma_q)$ ,  
 $\text{crd}_d(\text{crd}_m \sigma_q) = q$

gdzie:  $RD$  oznacza zbiór zasobów dzielonych w systemie.

Z warunku \*) wynika, że macierz  $A$  zawiera elementy związane z operacjami wykonywanymi w stanie początkowym. Stąd  $A_{i,j} = C_{i,j}$  wtedy i tylko wtedy gdy  $j$ -ta operacja należy do stanu początkowego, gdzie  $i$  jest indeksem następnej operacji wykonywanej przez dany proces.

Warunek \*\*) jest związany ze stanem początkowym reguł. Np. Sekwencja  $\sigma_1 = (P_1, P_2)$  oznacza, że zasób  $R_1$  będzie najpierw wykorzystywany przez proces  $P_1$ . Innymi słowy protokół „oczekuje” na proces  $P_1$ . Stąd krawędź  $e$  (krawędzie o wadze  $e$  reprezentują w macierzy  $C$  przyjęte sekwencje dostępu do zasobów dzielonych), reprezentująca warunek rozpoczęcia operacji procesu  $P_1$  na zasobie  $R_1$ , będzie przeniesiona z macierzy  $C$  do  $A$ . Z warunku tego wynika, że rozważane są tylko sekwencje, dla których zasób w stanie początkowym jest niezaalokowany. Wynika to z faktu, że dla zasobu dzielonego, na którym w stanie początkowym jest zaalokowany  $i$ -ty proces stan sekwencji jest jednoznacznie określony.

Z równania (7) wynika, że momenty rozpoczęć operacji  $k$ -tej iteracji zależą od operacji wykonywanych w  $k$ -tej iteracji oraz od operacji wykonywanych w poprzedniej iteracji. Macierz  $A$  zawiera elementy reprezentujące operacje wykonywane w poprzedniej iteracji. Powyższe warunki zakładają, że do operacji wykonywanych w poprzedniej iteracji będą należeć operacje wykonywane w stanie początkowym. Natomiast z warunku tworzenia macierzy  $B$  wynika, że są w niej reprezentowane operacje wykonywane w  $k$ -tej iteracji.

Graf stowarzyszony z macierzą  $B$  musi być grafem acyklicznym. Wynika, to z równania (9), gdzie macierz będąca sumą potęg macierzy  $B$  zawiera elementy (zgodnie z interpretacją graficzną potęgowania macierzy w  $(\max, +)$  algebrze) równe maksymalnym wagom wszystkich ścieżek w grafie  $G(V, E)$  stowarzyszonym z macierzą  $B$ . Nie można arbitralnie określić długości ścieżki o maksymalnej wadze ponieważ, zależy ona od przyjętego stanu początkowego, który determinuje podział macierzy  $C$  na  $A$  i  $B$ . W przypadku gdy, graf stowarzyszony z macierzą  $B$  byłby grafem posiadającym cykl suma macierzy kolejnych potęg  $B^i$  od  $i=0$  do  $n-1$  mogłaby zawierać wagi ścieżek reprezentujących wykonywanie tej samej operacji. Oznacza to, że macierz  $B$  zawierałaby wagi ścieżek obejmujących wykonywanie operacji w różnych iteracjach. Macierz  $D$  z równania (9) jest iloczynem rozważanego wyrażenia i macierzy  $A$ , tzn. macierzy, której elementy reprezentują operacje wykonywane w stanie początkowym. Z powyższego wynika, że graf z nią stowarzyszony zawiera ścieżki o wadze powiększonej o czas operacji wykonywanych w stanie początkowym. Z kolei  $(\max, +)$  algebraiczna wartość własna zgodnie z Definicją 5 stanowi maksymalną średnią wagą spośród wszystkich elementarnych cykli grafu  $G(V, E)$  stowarzyszonego z macierzą  $D$ . Z powyższego wynika, że wartość własna macierzy  $D$  z równania (11), będąca okresem pracy systemu, posiada najmniejszą wartość dla przyjętych reguł dostępu do zasobu, jeśli graf stowarzyszony z macierzą  $B$  będzie acykliczny oraz w macierzy  $A$  będą reprezentowane operacje poprzedzające bezpośrednio

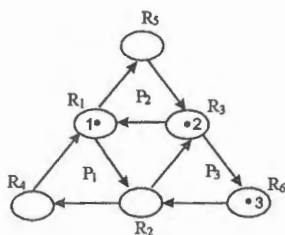
początek iteracji (np. operacje wykonywane w stanie początkowym). Stąd należy pokazać, że warunki Twierdzenia 1 gwarantują acykliczność grafu stowarzyszonego z macierzą  $B$ . W tym celu rozważmy przykład systemu z Rys. 2, składającego się z trzech procesów:  $P_1 = (1, 2, 4)$ ,  $P_2 = (3, 1, 5)$ ,  $P_3 = (2, 3, 6)$ , dla następujących reguł:  $\sigma_1 = (P_1, P_2)$ ,  $\sigma_2 = (P_1, P_3)$ ,  $\sigma_3 = (P_2, P_3)$ , oraz stanu początkowego  $S_0 = (P_1, A, P_2, A, A, P_3)$ .

Ponieważ macierz  $B$  powstaje w wyniku podziału macierzy  $C$  rozważmy cykle występujące w grafie stowarzyszonym z macierzą  $C$  (Rys. 3). W każdym z systemów należących do klasy SWPC występują dwa rodzaje cykli.

- Cykl związany z operacjami wykonywanymi przez procesy składowe. Ponieważ każdy z procesów jest cykliczny i po wykonaniu ostatniej operacji wykonuje pierwszą operację w swoim cyklu, np. krawędzie od wierzchołka  $x_7$  do  $x_8$ , od  $x_8$  do  $x_9$  oraz od wierzchołka  $x_9$  do  $x_7$  związane z operacjami procesu  $P_3$  (cykl 1 z Rys. 3).
- Cykl związany z reprezentacją protokołu wzajemnego wykluczenia. Wynika to z faktu, że sekwencja dostępu do zasobu posiada również cechy cykliczności. Krawędzie w grafie stowarzyszonym z macierzą  $C$ , odwzorowujące protokół dostępu do zasobu dzielonego, wraz z krawędziami reprezentującymi operacje wykonywane przez procesy na zasobie dzielonym tworzą w grafie stowarzyszonym z macierzą  $C$  cykl,

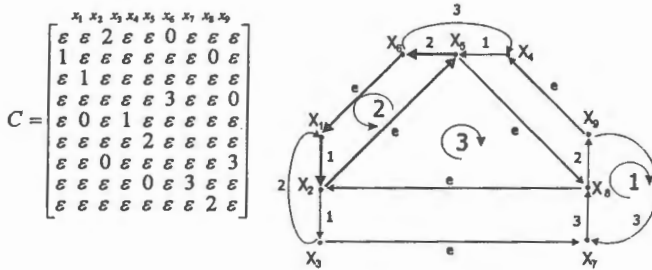
np. krawędzie od wierzchołka  $x_5$  do  $x_6$ , następnie od  $x_6$  do  $x_7$ , od  $x_7$  do  $x_2$  oraz od  $x_2$  do  $x_5$  (cykl 2 z Rys. 3).

Ponadto, mogą istnieć cykle związane z strukturą systemu, tzn. w systemie może występować zamknięta pętla żądań zasobowych procesów, np. krawędzie od wierzchołka  $x_2$  do  $x_5$ , od  $x_5$  do  $x_8$ , od  $x_8$  do  $x_2$  (cykl 3 z Rys 3).

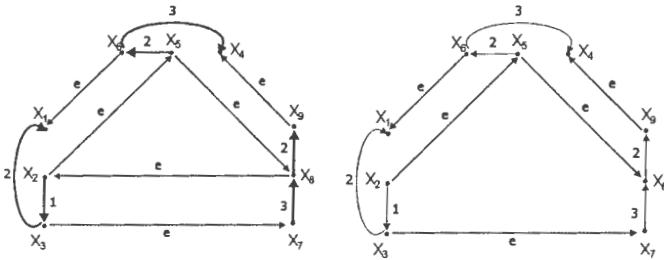


Rys. 2 System Współbieżnych Procesów Cyklicznych

Z warunków \*) zdefiniowanych w Twierdzeniu 1 wynika, że graf stowarzyszony z macierzą  $B$  nie zawiera krawędzi związanych z operacjami wykonywanymi w stanie początkowym. Oznacza to, że konstrukcja macierzy  $B$  zgodnie z warunkiem \*) gwarantuje, że w grafie stowarzyszonym z macierzą  $B$  nie istnieją cykle generowane przez operacje wykonywane przez dany proces (Rys. 4a).

Rys. 3 Macierz  $C$  oraz graf stowarzyszony z macierzą  $C$ 

Alokacja początkowa procesu na  $i$ -tym zasobie dzielnym implikuje przeniesienie krawędzi związanej z operacją na tym zasobie dzielnym do macierzy  $A$  gwarantując, że krawędzie związane z protokołem wzajemnego wykluczania na tym zasobie oraz krawędzie związane z operacjami procesów, które współzawodniczą o dostęp do  $i$ -tego zasobu nie tworzą cyklu w grafie stowarzyszonym z macierzą  $B$ . Np. alokacja procesu  $P_1$  na zasobie dzielnym  $R_1$  implikuje brak krawędzi w grafie stowarzyszonym z macierzą  $B$  od wierzchołka  $x_1$  do wierzchołka  $x_2$ , likwidując cykl  $x_1 \rightarrow x_2 \rightarrow x_5 \rightarrow x_6 \rightarrow x_1$  z Rys. 4a.



Rys. 4 Graf stowarzyszony z macierzą B a) powstały zgodnie z warunkami \*) Twierdzenia 1; b) powstały zgodnie z warunkami \*) oraz \*\*) Twierdzenia 1

W przypadku gdy, w stanie początkowym na  $i$ -tym zasobie dzielnym nie jest zaalokowany żaden z procesów, w grafie stowarzyszonym z macierzą B istnieje cykl tworzony przez krawędzie związane z operacjami procesów oraz krawędzie reprezentujące sekwencje dostępu do  $i$ -tego zasobu. Np. zasób  $R_2$ , wolny w stanie początkowym, generuje w grafie stowarzyszonym z macierzą B krawędzie cyklu  $x_7 \rightarrow x_8 \rightarrow x_2 \rightarrow x_3 \rightarrow x_7$  z Rys. 4a. Spełnienie warunku \*\*) Twierdzenia 1, gwarantuje brak powyższego cyklu w grafie stowarzyszonym z macierzą B (Rys. 4b).

Jako kontrprzykład rozważmy macierz B, której konstrukcja została zrealizowana zgodnie z warunkiem \*) Twierdzenia 1. Pominięcie warunku \*\*) Twierdzenia 1 prowadzi do istnienia cyklu w grafie stowarzyszonym z macierzą B (Rys. 4a). Wyznaczona macierz D oraz wartość własna T dla konstrukcji macierzy A i B uwzględniającej jedynie warunek \*) Twierdzenia 1 jest następująca:

$$D = \begin{bmatrix} 12 & \varepsilon & \varepsilon & 8 & \varepsilon & \varepsilon & \varepsilon & 12 \\ 9 & \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon & \varepsilon & 14 \\ 10 & \varepsilon & \varepsilon & 6 & \varepsilon & \varepsilon & \varepsilon & 11 \\ 11 & \varepsilon & \varepsilon & 10 & \varepsilon & \varepsilon & \varepsilon & 15 \\ 9 & \varepsilon & \varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & 10 \\ 11 & \varepsilon & \varepsilon & 7 & \varepsilon & \varepsilon & \varepsilon & 12 \\ 10 & \varepsilon & \varepsilon & 6 & \varepsilon & \varepsilon & \varepsilon & 11 \\ 9 & \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon & \varepsilon & 14 \\ 11 & \varepsilon & \varepsilon & 11 & \varepsilon & \varepsilon & \varepsilon & 16 \end{bmatrix}$$

Wartość własna będąca okresem pracy systemu jest równa  $T=16$ . Z przykładów zawartych w następnym punkcie dotyczących rozważanego powyżej systemu wynika, że najkrótszy okres pracy dla przyjętych reguł jest równy 8. Z przyjętych reguł wynika, że dany proces w danej iteracji wykonuje każdą operację jednokrotnie. Istnienie cyklu w grafie stowarzyszonym z macierzą  $B$  powoduje, że kolejne sumy potęg macierzy  $B$  obejmują ścieżki uwzględniające wielokrotne wykonywanie operacji przez ten sam proces.

## 5 PRZYKŁADY ZASTOSOWANIA

W poniższych przykładach skupiono się na pokazaniu realizacji etapu algorytmu syntezy procedur sterowania rozproszonego dotyczącego budowy modelu analitycznego SWPC zakładając, że wielkości wejściowe modelu wydajnościowego (reguły dostępu do zasobów oraz stan początkowy), gwarantują bezblokadową i bezagłodziową pracę systemu.



**Przykład 1.**

Rozważmy system z poprzedniego punktu (Rys 2) dla dwóch wariantów stanów początkowych. Specyfikacja struktury systemu, parametrów procesów składowych:

$$P_1 = (1, 2, 4), P_2 = (3, 1, 5), P_3 = (2, 3, 6), T_1 = (1, 1, 2), T_2 = (1, 2, 3), T_3 = (3, 2, 3).$$

Specyfikacja reguł rozstrzygnięcia konfliktów zasobowych:

$$R_1 = (P_1, P_2), \quad R_2 = (P_1, P_3), \quad R_3 = (P_2, P_3).$$

**Wariant 1.** Stan początkowy:

$$(R_1, R_2, R_3, R_4, R_5, R_6)$$

$$S_0 = (P_1, A, P_2, A, A, P_3).$$

**Etap 1.** Specyfikacja wektora stanu oraz wektora operacji.

Wektor operacji:  $p = (p_{1,1}, p_{1,2}, p_{1,3}, p_{2,1}, p_{2,2}, p_{2,3}, p_{3,1}, p_{3,2}, p_{3,3})$ .

Wektor stanu:  $x(k) = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9)$ ,  $x_1$  oznacza wykonywanie operacji  $p_{1,1}$ ,  $x_2 = p_{1,2}$ ,  $x_3 = p_{1,3}$ ,  $x_4 = p_{2,1}$ ,  $x_5 = p_{2,2}$ ,  $x_6 = p_{2,3}$ ,  $x_7 = p_{3,1}$ ,  $x_8 = p_{3,2}$ ,  $x_9 = p_{3,3}$ .

**Etap 2.** Wyznaczenie okresu pracy systemu  $T$  oraz wektora momentów rozpoczęć operacji w zerowej iteracji  $x_0 = x(0)$  (diagramy Gantt'a – przebiegu zmian stanów, pokazano na Rys. 5). Określenie elementów macierzy  $C$  oraz  $A$  i  $B$ .

$$C = \begin{bmatrix} \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & 0 \\ \varepsilon & 0 & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & 3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon \end{bmatrix} \quad A = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix} \quad B = \begin{bmatrix} \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & 0 \\ \varepsilon & 0 & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon \end{bmatrix}$$

Wyznaczenie macierzy  $D$  oraz wartości własnej  $T$  i wektora własnego  $x_0 = x(0)$  macierzy  $D$ .

$$D = \begin{bmatrix} 4 & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon \\ 7 & \varepsilon & \varepsilon & 6 & \varepsilon & \varepsilon & \varepsilon & 6 & 8 \\ 1 & \varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 3 & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & 3 \\ 5 & \varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & 4 & 6 \\ 7 & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & 6 & 8 \end{bmatrix} \quad T = 8, \quad x_0 = (5, 0, 1, 2, 3, 5, 5, 0, 2)$$

**Wariant 2.** Zmiana stanu początkowego:

$$(R_1, R_2, R_3, R_4, R_5, R_6)$$

$$S_0 = (P_1, A, A, A, P_2, P_3).$$

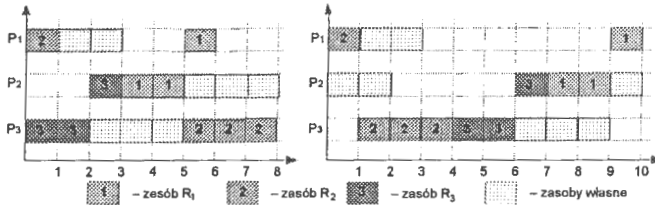
Ponieważ zmiana stanu początkowego ma wpływ tylko na strukturę macierzy  $A$  i  $B$ , dlatego też macierz  $C$  posiada strukturę taką jak w wariantcie 1 rozważanego przykładu.

$$A = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}$$

$$B = \begin{bmatrix} \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}$$

$$D = \begin{bmatrix} 10 & \varepsilon & \varepsilon & \varepsilon & 5 & 6 & \varepsilon & 9 & 11 \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon \\ 7 & \varepsilon & \varepsilon & 2 & 3 & \varepsilon & 6 & 8 & \varepsilon \\ 8 & \varepsilon & \varepsilon & 3 & 4 & \varepsilon & 7 & 9 & \varepsilon \\ 10 & \varepsilon & \varepsilon & 5 & 6 & \varepsilon & 9 & 11 & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & 3 & \varepsilon \\ 5 & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & 4 & 6 & \varepsilon \\ 7 & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & 6 & 8 & \varepsilon \end{bmatrix}$$

Okres systemu  $T = 10$ . Wektor stanu  $x_0 = (9, 0, 1, 6, 7, 9, 1, 4, 6)$ .



Rys. 5 Diagramy Gantt'a: a) wariant 1, b) wariant 2

## 6 PODSUMOWANIE

W pracy przedstawiono metodę syntezy procedur sterowania rozproszonego gwarantującą bezblokadowe i bezgłodzeniowe funkcjonowanie Systemu Współbieżnych Procesów Cyklicznych. W proponowanym podejściu wykorzystano formalizm  $(\max, +)$  algebry - naturalny formalizm wykorzystywany w modelowaniu systemów, w których współpraca procesów jest oparta na protokole rendez-vous. Określono warunki oraz przedstawiono

metodę budowy analitycznego modelu wydajnościowego, który pozwala wyznaczyć różne parametry ilościowe SWPC, np. czas cyklu, czasy aktywności procesów, dla systemów, w których procesy współpracują w trybie wzajemnego wykluczania. Zdefiniowano warunki umożliwiające automatyzację konstrukcji macierzy  $A$  i  $B$ , będących składnikami równania stanu. Ponadto pokazano, że budowa modelu analitycznego zgodna z przedstawionymi warunkami gwarantuje wyznaczenie z równania (11) najkrótszego okresu pracy systemu, dla przyjętych reguł rozstrzygania konfliktów zasobowych oraz przyjętego stanu początkowego.

#### Literatura

1. Zaremba M.B., Banaszak Z.A., Majdzik P., Jędrzejek K.J.: *Distributed Flow control design for repetitive manufacturing processes. Proc. of the 14th Triennial Worlds Congress of IFAC, July 5-9, Beijing, P.R. China, 1999, Vol. A, pp.1-6*
2. Baccelli F.L. et al.: *Synchronization and linearity. An algebra for discrete event systems. John Wiley & Sons, N.Y., Toronto 1992*
3. Banaszak Z.A., Polak M.: *Deadlock-free distributed control for repetitive flows. Proc. of the 6th Int. Workshop on Discrete Event Systems, 2002, Zaragoza, Spain, pp.273-278*

4. Polak M., Wójcik R., Majdzik P., Banaszak Z.: *Prototyping of Distributed Control Procedures in Concurrent Cyclic Processes Systems. 9th IEEE International Conference on Emerging Technologies and Factory Automation, Lisbon, Portugal, September 2003, pp.261-268*
5. Wójcik R.: *Towards strong stability of concurrent repetitive processes sharing resources. Systems Science, 2001, Vol. 27, No. 2, pp. 37-47*

#### AN AUTOMATIC SYNTHESIS OF PERFORMANCE MODELS OF CONCURRENT CYCLIC PROCESSES

Systems composed of a set of concurrently executed cyclic processes, which share resources in the mutual-exclusion mode are considered. The purpose of the paper is to present an approach to automated prototyping of distributed control procedures, i.e. a sets of pairs (an initial state, a set of dispatching rules). The  $(\max,+)$  algebra formalism is used as a modeling framework. The procedure enabling to determine components of a state equation, representing a performance model for a given set of processes and their distributed control, is provided. The illustrative example of the state equation synthesis and then its application to the system performance evaluation are considered in deep.





