

Franoiszek Marecki
Politechnika Śląska

STEROWANIE ELASTYCZNYM SYSTEMEM PRODUKCYJNYM^{*/}

Streszczenie. W pracy przedstawiony jest problem sterowania elastycznym systemem produkcyjnym. System składa się z robota i magazynu obrotowego. Podano algorytm harmonogramowania procesu.

1. Wstęp - sformułowanie problemu

Problematyka sterowania elastycznymi systemami produkcyjnymi stanowi przedmiot rozważań wielu prac [1, 2, 5]. W ogólnym przypadku w elastycznych systemach produkcyjnych wyróżnia się: agregaty, magazyny oraz transportery. Strumień obiektów jest przemieszczany pomiędzy tymi elementami za pomocą robotów.

W niniejszym artykule analizowany jest dydaktyczny elastyczny system ROBKO [6]. System ten w minimalnej konfiguracji składa się z: robota, magazynu obrotowego i transportera. Ponadto robot może przyjmować 8 sygnałów /binarnych/ o stanie otoczenia i wysyłać 8 sygnałów informacyjnych do otoczenia. Dla przykładu jednym z sygnałów wejściowych jest informacja z optycznego sensora w chwytaku. System ROBKO jest sterowany przez mikrokomputer PRAWEC. Programy do sterowania są umieszczane na dyskach elastycznych.

W maksymalnej konfiguracji system ROBKO może zawierać dwa roboty, dwa magazyny i dwa transportery oraz podwójną liczbę sygnałów wejściowych i wyjściowych. System jest sterowany z jednego mikrokomputera PRAWEC.

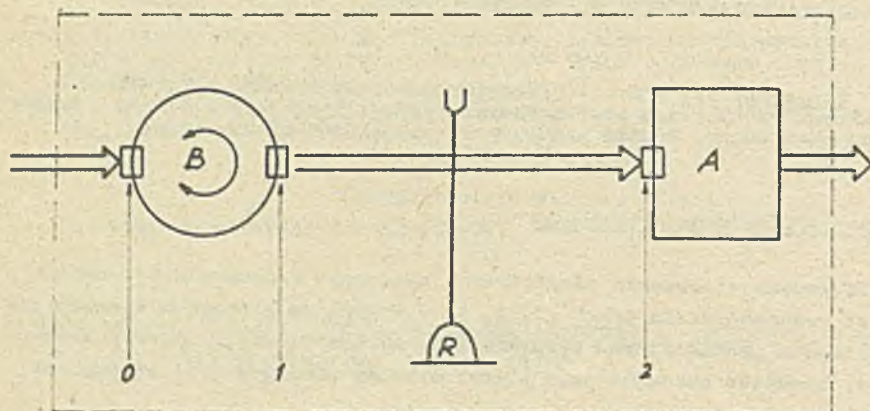
Sterowanie systemem ROBKO może być oparte na uczeniu lub projektowaniu trajektorii [3, 6]. Podstawowe znaczenie ma określenie operacji dla robota. Operacje magazynu obrotowego oraz transportera /dwukierunkowe ruchy o zadanej wartości/ stanowią uzupełnienie sekwencji operacji robota. Ponadto przejście do wykonania kolejnej operacji przez robot może być warunkowane stanem otoczenia /sygnałów wejściowych/.

W dalszej części artykułu ograniczymy rozważania do systemu składającego się z robota i magazynu obrotowego oraz jednego agregatu. Przenalizujemy problem wyznaczenia harmonogramu, który pozwala wykonać wszystkie zadania w najkrótszym czasie.

^{*/} Praca była częściowo finansowana przez problem RP.I.02 i CPBR 7.4.

2. Analiza problemu

Rozważmy system pokazany na rys. 1.



Rys. 1. System robot - magazyn obrotowy
/A-agregat, B-magazyn obrotowy, R-robot,
0-punkt załadunkowy magazynu, 1-punkt
wyladunkowy magazynu, 2-punkt wejściowy agregatu/

Fig. 1. Robot - rotary store system
/A-a machine, B-a rotary store, R-a robot,
0-a store loading point, 1-a store unloading
point, 2-a machine input point.

Na rysunku tym wyróżniono: buforowy magazyn obrotowy B, agregat A oraz robot R. W agregacie A są obsługiwane obiekty dostarczane przez robot R z magazynu B. Przyjmujemy, że zmiana typu obiektu obsługiwanego w agregacie A wymaga dokonania zmiany narzędzia. Zatem przy pewnej kolejności obsługi obiektów sumaryczny czas wymiany narzędzi będzie najmniejszy. Jednakże przestoje agregatu A mogą wynikać również z terminów dostarczenia obiektów z magazynu B.

Robot R przenosi kolejne obiekty od punktu 1 do punktu 2. Załóżmy, że robot R rozpocznie operację przeniesienia obiektu w punkcie 1, jeśli może ją zakończyć w punkcie 2 bez przerwania. Oznacza to, że agregat A musi być przygotowany do rozpoczęcia operacji obsługi obiektu. Poza tym robot R może rozpocząć operacje, jeśli w punkcie 1 znajduje się odpowiedni obiekt.

Obiekty są dostarczane do punktu 1 w wyniku odpowiednich obrotów magazynu B. Załóżmy, że magazyn B przystępuje do operacji podstawiania kolej-

nego obiektu bezpośrednio po usunięciu przez robotę poprzedniego obiektu. Oznacza to, że magazyn B może oczekiwać na usunięcie obiektu z punktu 1 /w tym czasie można załadować obiekt w punkcie 0, jeśli odpowiednia pozycja jest wolna/.

Stan magazynu może przedstawiać obiekty, które znajdują się na określonych pozycjach w wyróżnionej chwili. Z kolei stan agregatu można opisać podając numer obiektu i termin rozpoczęcia jego obsługi. Tak więc założymy, że dla magazynu i agregatu istotne są chwile, w których rozpoczynane są odpowiednie operacje /podstawiania i obsługi obiektu/. Dla robota istotne znaczenie ma chwila rozpoczęcia operacji transportu obiektu.

2.1. Opis matematyczny

Założmy, że dany jest zbiór obiektów, które mają być obsłużone w systemie

$$\Omega = \{ \omega_n \} \quad /1/$$

$$n = 1, \dots, N$$

gdzie: ω_n - obiekt n-tego typu, N - liczba obiektów.

Dany jest magazyn obrotowy zawierający M pozycji. Założmy, że w stanie początkowym w punkcie wyladunkowym znajduje się pierwsza pozycja.

Def. 1 Stan magazynu jest wektorem

$$S = [s_m] \quad /2/$$

$$m = 1, \dots, M$$

Elementy tego wektora określamy następująco

$$s_m = \begin{cases} n & \text{jeśli na } m\text{-tej pozycji znajduje się obiekt } \omega_n, \\ 0 & \text{w przypadku przeciwnym.} \end{cases} \quad /2a/$$

Przyjmując, że wszystkie obiekty znajdują się w magazynie, tzn.

$$N \leq M \quad /3/$$

pominiemy określenie położenia punktu załadunkowego magazynu. Niech czas obrotu magazynu o jedną pozycję wynosi σ .

Założmy, że dane są czasy obsługi obiektów w agregacie

$$\Theta = [\psi_n] \quad /4/$$

$$n = 1, \dots, N$$

gdzie: ψ_n - czas obsługi obiektu ω_n w agregacie A.

Ponadto dane są czasy wymian narzędzi w agregacie

$$T = [\tau_{\nu, n}] \quad /5/$$

$$\nu = 1, \dots, N$$

$$n = 1, \dots, N$$

gdzie: $\tau_{\nu, n}$ - czas wymiany narzędzia w agregacie A po obiekcie ω_{ν} i przed obiektem ω_n .

Przyjmujemy, że bezpośrednio przed rozpoczęciem obsługi obiektów ze zbioru Ω w agregacie A był obsługiwany obiekt ω_{κ} , $1 \leq \kappa \leq N$.

Oznaczmy przez:

- ϑ_n^A - chwilę rozpoczęcia obsługi obiektu ω_n w agregacie A,
- ϑ_n^B - chwilę rozpoczęcia podstawienia obiektu ω_n do punktu wyladunkowego w magazynie B,
- ϑ_n^R - chwilę rozpoczęcia przez robotę transportu obiektu ω_n od magazynu do agregatu.

Chwile te /dla $n = 1, 2, \dots, N/$ określają harmonogramy pracy agregatu, magazynu i robota.

Def. 2 Stan agregatu jest wektorem

$$P = [p_n] \quad /6/$$

$$n = 1, \dots, N$$

Elementy tego wektora określamy następująco

$$p_n = \begin{cases} \vartheta_n, & \text{jeśli obsługę obiektu } \omega_n \text{ w agregacie A rozpoczęto} \\ & \text{w chwili } \vartheta_n, \\ 0, & \text{w przypadku przeciwnym /tzn. nie rozpoczęto jeszcze} \\ & \text{obsługi obiektu/.} \end{cases} \quad /6a/$$

Definicja stanu /6/ zostanie wykorzystana do wyznaczenia harmonogramów metodą programowania wieloetapowego [4].

Jeśli obiekt ω_{ν} został obsługowany przed obiektem ω_n , to dopuszczalny harmonogram pracy agregatu musi spełniać warunek

$$\vartheta_n^A \geq \vartheta_{\nu}^A + v_{\nu}^B + \tau_{\nu, n} \quad /7/$$

Analizując pracę magazynu przyjmujemy, że może się on obracać w dwóch kierunkach. Czas v_n^B dostarczenia obiektu ω_n do punktu wyladunkowego wyznaczamy ze wzoru:

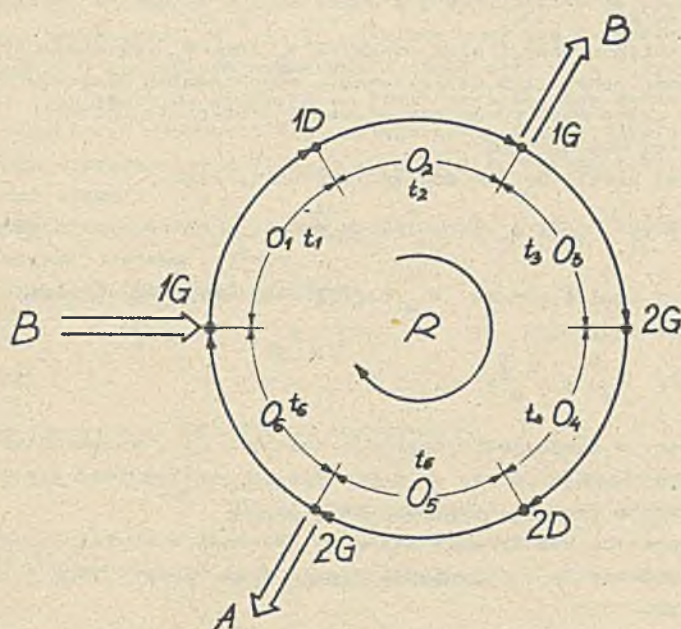
$$v_n^B = \min(m - \mu, M - m + \mu) \cdot c, \quad /8/$$

gdzie: n - numer pozycji, na której znajduje się obiekt. ω_n ,
 μ - numer pozycji znajdującej się w punkcie wyladunkowym,
 σ - czas obrotu magazynu o jedną pozycję.

Zatem dopuszczalny harmonogram pracy magazynu musi spełniać warunek

$$\vartheta_v^B + \vartheta_v^B \leq \vartheta_n^B \quad /9/$$

Cykl pracy robota przedstawiono na rys. 2.



Rys. 2. Cykl pracy robota

O_1, \dots, O_6 - operacje elementarne robota

t_1, \dots, t_6 - czasy operacji elementarnych

1G - górne położenie chwytaka w punkcie 1

1D - dolne położenie chwytaka w punkcie 1

2G - górne położenie chwytaka w punkcie 2

2D - dolne położenie chwytaka w punkcie 2

\Rightarrow - sygnały wejściowe lub wyjściowe agregatu i magazynu

Fig. 2. Cycle of a robot work

O_1, \dots, O_6 - elementary operations of a robot

t_1, \dots, t_6 - times of elementary operations

1G - upper position of a robot at the point 1

1D - lower position of a robot at the point 1

2G - upper position of a robot at the point 2

2D - lower position of a robot at the point 2

\Rightarrow - input or output signals of a machine or storage

W cyklu pracy robota można wyróżnić 6 elementarnych operacji /zakładając położenie początkowe w punkcie wyladunkowym 1G/:

- O_1 - przejście od górnego położenia /1G/ do dolnego położenia /1D/ w punkcie załadunkowym i zamknięcie chwytaka.
- O_2 - przejście od 1D do 1G' /powrót do punktu 1G/.
- O_3 - przejście od punktu 1G' do punktu 2G /górnego położenie w punkcie wejściowym agregatu/.
- O_4 - przejście od 2G do 2D i otwarcie chwytaka.
- O_5 - przejście od 2D do 2G' /powrót do punktu 2G/.
- O_6 - przejście od 2G' do 1G.

Czasy wszystkich operacji elementarnych t_n /dla O_n / są dano. Wynikają one z fazy uczenia robota lub projektowania odpowiednich trajektorii.

Z punktu 1G robot może rozpocząć operację transportu obiektu ω_n , jeśli spełnione są dwa warunki:

a/ W odpowiedniej chwili będzie dostępny agregat, tzn.

$$\rho_n^R + t_1 + t_2 + t_3 \geq \rho_v^A + v_n^A + \tau_{v,n} \quad /10a/$$

b/ W odpowiedniej chwili obiekt ω_n będzie podstawiony do punktu wyladunkowego magazynu, tzn.

$$\rho_n^R \geq \rho_n^B + v_n^B \quad /10b/$$

Sygnal informujący o zakończeniu operacji magazynu jest wprowadzany na wejście robota, gdy znajduje się on w punkcie 1G. Analogiczny sygnal o gotowości agregatu - robot odbiera w punkcie 2G.

Magazyn rozpoczyna wykonywanie kolejnej operacji, w chwili gdy robot znajdzie się w punkcie 1G'. Odpowiedni sygnal jest wyprowadzany z wyjścia robota do magazynu.

Agregat rozpoczyna obsługę obiektu ω_n w chwili gdy robot znajdzie się w punkcie 2G' - po wyprowadzeniu odpowiedniego sygnalu wyjściowego.

Spośród harmonogramów dopuszczalnych należy wyznaczyć optymalny harmonogram pracy systemu. Jako kryterium optymalizacji przyjmujemy

$$Q = \max_{1 \leq n \leq N} (\rho_n^A + v_n^A) \rightarrow \min, \quad /11/$$

czyli minimalizację czasu obsługi wszystkich obiektów w systemie.

3. Algorytm

Sformułowany problem zostanie rozwiązany metodą programowania wieloetapowego. Problem w istocie polega na wyznaczeniu najlepszej kolejności obsługi obiektów w agregacie. Jednakże z uwagi na ograniczenia czasowe nie jest to wprost problem komiwojżera /analogia między czasami $\tau_{\nu,n}$ i odległościami między miastami/.

Rozwiązanie problemu polega na wyznaczeniu najlepszej trajektorii /ciągu N stanów/ lub N kolejnych decyzji. Przed podjęciem pierwszej decyzji /w chwili zerowej/ znany jest stan systemu /położenie obiektów w magazynie, położenie robota w punkcie 1G, zakończenie obsługi obiektu ω_n w agregacie/. Wychodząc z tego stanu i przydzielając do obsługi obiekt ω_n można wyznaczyć: g_n^B , g_n^R i g_n^A . Należy przy tym respektować ograniczenia czasowe. W ten sposób wychodząc z etapu $e-1$, / $e = 1, \dots, N$ /, można otrzymać stan etapu e . Ponieważ ze stanu początkowego można wygenerować różne trajektorie dopuszczalne, dlatego stany tego samego etapu zostaną ponumerowane przez 1, $l = 1, \dots, L_e$, /gdzie: L_e - dopuszczalna liczba stanów e -tego etapu/.

Uogólniając definicje: stanu magazynu /2/ i stanu agregatu /6/, przyjmujemy definicję stanu systemu:

Daf. 3 Stan systemu jest macierzą

$$P^{e,l} = [P_{n,j}^{e,l}] \quad /12/$$

$$n = 1, \dots, N$$

Elementy tego stanu definiujemy następująco:

$$P_{n,1}^{e,l} = \begin{cases} g_n^B, & \text{jeśli na etapie } \eta \leq e \text{ została podjęta decyzja} \\ & \text{o wydaniu z magazynu obiektu } \omega_n \\ -1, & \text{w przypadku przeciwnym} \end{cases} \quad /12a/$$

$$P_{n,2}^{e,l} = \begin{cases} g_n^R, & \text{jeśli na etapie } \eta \leq e \text{ została podjęta decyzja} \\ & \text{o przeniesieniu przez robota obiektu } \omega_n \text{ z maga-} \\ & \text{zynu do agregatu} \\ -1, & \text{w przypadku przeciwnym} \end{cases} \quad /12b/$$

oraz

$$P_{n,3}^{e,l} = \begin{cases} g_n^A, & \text{jeśli na etapie } \eta \leq e \text{ została podjęta decyzja} \\ & \text{o obsłudze w agregacie obiektu } \omega_n \\ -1, & \text{w przypadku przeciwnym} \end{cases} \quad /12c/$$

Zatem stan $P^{0,1}$ jest macierzą o elementach równych - 1, natomiast stan końcowy $P^{N,1}$ ma wszystkie współrzędne dodatnie. Ze stanu końcowego można odczytać wprost dopuszczalne harmonogramy pracy: magazynu, robota i agregatu.

Dla oceny rozwiązania, które reprezentuje stan, wprowadzimy definicję wartości stanu $V^{e,1}$.

Def. 4 Wartość stanu jest skalarem

$$V^{e,1} = \max_{\omega_n \in \Omega^{e,1}} (P_{n,3}^{e,1} + v_n), \quad /12/$$

przy czym

$$\Omega^{e,1} = \{ \omega_n : P_{n,3}^{e,1} > 0 \} \quad /13a/$$

Dla stanu końcowego otrzymamy

$$V^{N,1} = \max_{1 \leq n \leq N} (P_{n,3}^{N,1} + v_n), \quad /14/$$

co koresponduje z przyjętym kryterium optymalizacji /11/. Zatem stan końcowy o najmniejszej wartości jest stanem optymalnym.

Ze stanu $P^{e,1}$ można wyznaczyć pewne wielkości pomocnicze w algorytmie: - numer ostatniego z obsługiwanych obiektów $K^{e,1}$

$$K^{e,1} = \begin{cases} k, & \text{jeśli } \max_{1 \leq n \leq N} P_{n,3}^{e,1} = P_{k,3}^{e,1} > 0 \\ \mu, & \text{w przypadku przeciwnym} \end{cases} \quad /15/$$

Jeśli $K^{e,1} = \mu$, tzn. $e = 0$ - wówczas przyjmujemy $\tau_{\mu,n}$ dla agregatu.

- chwile zakończenia obsługi obiektu $K^{e,1}$ przez: magazyn ($T_B^{e,1}$), robot ($T_R^{e,1}$) i agregat ($T_A^{e,1}$)

$$T_B^{e,1} = \varrho_{K^{e,1}}^B + v_{K^{e,1}}^B \quad /16a/$$

$$T_R^{e,1} = \varrho_{K^{e,1}}^R + \sum_{i=1}^{i=6} t_i \quad /16b/$$

oraz

$$T_A^{e,1} = \varrho_{K^{e,1}}^A + v_{K^{e,1}}^A \quad /16c/$$

- stan $S^{e,1}$ magazynu

Na podstawie /2/ i /13a/ otrzymamy S^e , przy czym

$$s'_m = \begin{cases} s_m, & \text{jeśli } \omega_n \notin \Omega^{e,1} \\ 0, & \text{w przypadku przeciwnym} \end{cases} \quad /17/$$

Obiekt $K^{e,1}$, który w stanie $P^{0,1}$ /tzn. S/, był na pozycji μ , w stanie $S^{e,1}$ będzie na pierwszej pozycji magazynu. Zatem otrzymamy

$$s_m^{e,1} = \begin{cases} s_{m-\mu+1}^{e,1}, & \text{dla } m \geq \mu \\ s_{N-\mu+m+1}^{e,1}, & \text{dla } m < \mu \end{cases} \quad /18/$$

Generowanie trajektorii polega na przejściu od wybranego stanu $P^{e-1,1}$ do nowego stanu $P^{e,1}$ poprzez przydzielenie do obsługi obiektu ω_n . Ogólna procedura generowania stanów ma postać:

$$\forall_n \exists_{1 \leq m \leq M} (s_m^{e,1} = n) \Rightarrow (P^{e,1} = P^{e-1,1} + \Delta P) \quad /19/$$

Elementy macierzy ΔP mają postać:

$$\Delta P_{i,1} = \begin{cases} \rho_n^B, & \text{dla } i = n \\ 0, & \text{dla } i \neq n \end{cases} \quad /20a/$$

$$\Delta P_{i,2} = \begin{cases} \rho_n^R, & \text{dla } i = n \\ 0, & \text{dla } i \neq n \end{cases} \quad /20b/$$

oraz

$$\Delta P_{i,3} = \begin{cases} \rho_n^A, & \text{dla } i = n \\ 0, & \text{dla } i \neq n \end{cases} \quad /20c/$$

Chwile: ρ_n^B , ρ_n^R i ρ_n^A wyznaczamy znając: $T_B^{e-1,1}$, $T_R^{e-1,1}$, $T_A^{e-1,1}$ i $K^{e-1,1}$ /obliczenia prowadzimy od stanu $P^{0,1}$ /.

Zgodnie z cyklem pracy robota /przedstawionym na rys. 2/, jeśli robot zakończył operację dla obiektu $K^{e-1,1}$ w chwili $T_R^{e-1,1}$, to magazyn może podstawić obiekt ω_n od chwili

$$\rho_n^B = T_R^{e-1,1} - (t_6 + t_5 + t_4 + t_3) \quad /21/$$

Czas operacji ρ_n^B wyznaczamy z /8/, a następnie $T_B^{e,1}$ z /16a/.

Chwila ρ_n^R jest najpóźniejszą z trzech chwil:
- podstawienia obiektu w magazynie: $T_B^{e,1}$,

- dostępności robota: $T_R^{e-1,1}$,
 - względnej dostępności agregatu: $T_A^{e-1,1} + T_{K^{e-1,1,n}}^{e-1,1} - (t_1 + t_2 + t_3)$,
- czyli

$$\vartheta_n^R = \max [T_B^{e,1}; T_R^{e-1,1}; T_A^{e-1,1} + T_{K^{e-1,1,n}}^{e-1,1} - (t_1 + t_2 + t_3)] / 22/$$

Chwilę ϑ_n^A wyznaczamy jako

$$\vartheta_n^A = \vartheta_n^R + (t_1 + t_2 + t_3 + t_4 + t_5) \quad /23/$$

Tak więc wychodząc ze stanu $P^{0,1}$, można wygenerować kolejne stany $P^{e,1}$ i rozwiązanie dopuszczalne, które reprezentuje stan końcowy $P^{N,1}$.

W metodzie programowania wieloetapowego można wyznaczyć trajektorie wg różnych algorytmów:

- konwersacyjnych /decyzje krok po kroku podejmuje operator na podstawie oceny stanu systemu/,
- heurystycznych /decyzja jest podejmowana programowo na podstawie tzw. reguły heurystycznej np. wybór obiektu ω_n o najkrótszym czasie $T_{p,n}$ lub najmniej odległego w magazynie od punktu wyładunkowego/,
- najbliższego sąsiedztwa /wyznacza się trajektorie sąsiednie do wybranej trajektorii/.

Jeśli wybrana jest pewna trajektoria, to odpowiada jej kolejność obsługiwanego obiektów zapisana wektorem

$$X = [x_n] \quad /24/$$

$n = 1, \dots, N$

gdzie: x_n - nr n-tego z kolei obiektu.

Trajektoria $P^{N,1^1}$, której odpowiada wektor X^1 , jest leksykograficznie wcześniejsza od trajektorii $P^{N,1^2}$, której odpowiada wektor X^2 , jeśli jest spełniony warunek:

$$\forall_{i < j} \exists_j (x_i^1 < x_i^2) \vee (x_i^1 = x_i^2) \wedge (x_j^1 < x_j^2) \quad /25/$$

Trajektorie $P^{N,1^1}$ i $P^{N,1^2}$ są sąsiednie, jeśli nie istnieje taka trajektoria, która w sensie leksykograficznego porządku mieści się pomiędzy nimi.

- przeglądowy /podziału i ograniczeń/

W tym algorytmie stany są generowane w oparciu o reguły wyboru i podziału /na ogół jednocześnie zapamiętywanych jest wiele stanów każdego etapu/. Dla zapewnienia efektywności /w sensie czasu obliczeń/ algorytmu, formułowane są reguły eliminacji stanów nieperspektywnych.

W rozwiązywanym problemie jako dolne ograniczenie czasu obsługi wszyst-

kich zadań można wykorzystać znane oszacowanie dla problemu komiwojażera. Ponadto dla stanów tego samego etapu można udowodnić twierdzenie: Tw. Stan $P^{e,1_1}$ dominuje nad stanem $P^{e,1_2}$, jeśli jest spełniony warunek:

$$(\Omega^{e,1_1} = \Omega^{e,1_2}) \wedge (K^{e,1_1} = K^{e,1_2}) \wedge (V^{e,1_1} \leq V^{e,1_2}) \quad /26/$$

Stan zdominowany $P^{e,1_2}$ można w obliczeniach pominąć, ponieważ nie pozwala uzyskać rozwiązania optymalnego.

Stosowanie reguły dominacji wymaga sprawdzenia pierwszych dwóch części warunku /26/. Jeżeli liczba stanów e-tego etapu jest duża, to zajmuje to wiele czasu /N porównań dla zbioru $\Omega^{e,1}$ /. Dlatego dla zbioru $\Omega^{e,1}$ /lub stanu $P^{e,1}$ / można wprowadzić klucz

$$F^{e,1} = \sum_{n=1}^{n=N} f_n^{e,1} \cdot 2^n, \quad /27/$$

przy czym

$$f_n^{e,1} = \begin{cases} 1, & \text{jeśli } P_{n,3}^{e,1} > 0 \\ 0, & \text{w przypadku przeciwnym} \end{cases} \quad /27a/$$

Klucz $F^{e,1}$ można obliczać w trakcie generowania stanu, tzn.

$$F^{e,1} = F^{e-1,1} + 2^n \quad /27b/$$

W ten sposób warunek /26/ sprowadza się do postaci

$$(F^{e,1_1} = F^{e,1_2}) \wedge (K^{e,1_1} = K^{e,1_2}) \quad /28/$$

Jeśli warunek ten jest spełniony, to jeden ze stanów $P^{e,1_1}$ lub $P^{e,1_2}$ zostanie wyeliminowany.

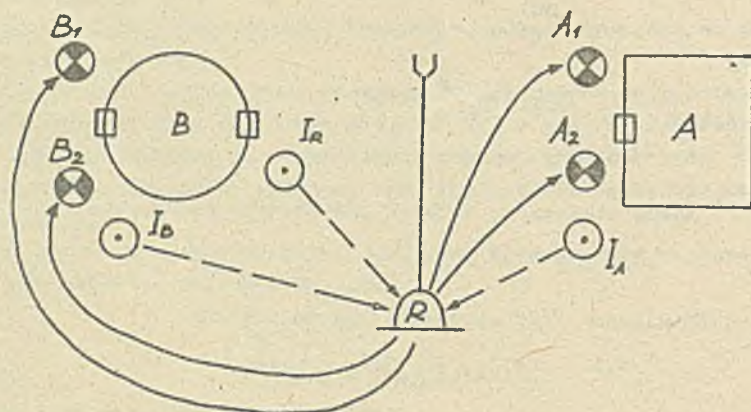
Dalsze skrócenie czasu obliczeń otrzymujemy dokonując podziału zbioru stanów e-tego etapu na podzbiory - ze względu na klucz $K^{e,1}$. Wówczas dla stanu $P^{e,1_2}$ znany jest klucz $K^{e,1_2}$, tym samym poszukiwania zostają ograniczone do pełnego podzbioru /dla $K^{e,1_2}$ /. Z kolei podzbiór ten jest uporządkowany z uwagi na klucz $F^{e,1}$, zatem stosując podział połówkowy można znacznie skrócić czas obliczeń.

Przeprowadzona dyskusja ma istotne znaczenie, gdy liczba obiektów przeznaczonych do obsługi jest duża.

4. Realizacja praktyczna

Przedstawiony sposób sterowania systemem ROEKO został zrealizowany w Laboratorium Robotów Przemysłowych w Instytucie Automatyki Politechniki Śląskiej. Operacje elementarne wyznaczono poprzez uczenie oraz projektowanie trajektorii. Obliczenia przeprowadzone zostały w języku ROBASIC na minikomputerze PRAWEC. W systemie ROEKO magazyn ma $M = 8$ pozycji. W związku z tym liczba rozwiązań dopuszczalnych /dla $N = M/$ wynosi 40320. Dla większych wartości N optymalne harmonogramy wyznaczono na mikrokomputerze IBM PC/XT. Sterowanie systemem dla realizacji tych harmonogramów przeprowadzono na mikrokomputerze PRAWEC.

Dla realizacji sterowania, układ pokazany na rys. 1 został rozbudowany do postaci pokazanej na rys. 3.



Rys. 3. Schemat realizacyjny systemu

B-magazyn buforowy, R-robot, A-agregat, B_1 -sygnalizacja zajętości /czerwona/, B_2 -sygnalizacja dostępności magazynów /zielona/, A_1 -sygnalizacja zajętości agregatu /czerwona/, A_2 -sygnalizacja dostępności agregatu /zielona/, I_B -czujnik magazynu, I_R -czujnik robota, I_A -czujnik agregatu.

Fig. 3. Realization Scheme of a system

B-a buffer rotary store, A-a machine, R-a robot, B_1 -a busy store signal /red/, B_2 -a free store signal /green/, A_1 -a busy machine signal /red/, A_2 -a free store signal /green/, I_B -a store sensor, I_R -a robot sensor, I_A -a machine sensor.

W układzie tym uwzględniono cztery lampki sygnalizacyjne, dla magazynu / B_1 i B_2 / oraz dla agregatu / A_1 i A_2 / . Wprowadzenie sygnalizacji na wejściu do magazynu pozwala analizować przypadki dla $N > M$. Wówczas w trakcie postoju magazynu można do niego wprowadzać dalsze objekty. W tym celu wyróżnia się punkt załadunkowy /pozycja wektora S / . Zakładając, że czas załadunku wynosi Δo , dopuszcza się załadunek obiektu, jeśli

$$T_R^{e,1} - T_B^{e,1} \geq \Delta o, \quad /29/$$

oznacza to, że z chwilą gdy robot znajdzie się w punkcie 1G /rys. 2/, załadunek musi być zakończony. Sygnalizację zajętości punktu załadunkowego otrzymujemy przy wyłączonym B_2 i załączonym B_1 . Otwarcie punktu załadunkowego może nastąpić, gdy kolejny obiekt /po otwarciu magazynu/ znajdzie się w punkcie wyładunkowym. Sytuację tę sygnalizuje przełącznik I_B .

Robot rozpoczyna transport obiektu po sprawdzeniu jego obecności w punkcie wyładunkowym 1D /za pomocą optycznego sensora chwytaka I_R / . Następnie robot będąc w punkcie 1G' wysyła sygnał zezwolenia na obrót magazynu.

W analizowanym systemie działanie agregatu jest symulowane, tzn. w określonych chwilach usuwane są objekty z punktu 2. Robot po dojściu do punktu 2G blokuje dostęp do agregatu - sygnalizacja lampką czerwoną A_1 . Z kolei po złożeniu obiektu w punkcie 2D, robot wracając do 2G' wysyła sygnał zezwolenia /sygnalizacja lampką zieloną/ na obsługę obiektu. Dostępność agregatu jest sygnalizowana przełącznikiem I_A . Jeżeli agregat nie jest dostępny, to operacja elementarna robota nie zostanie wykonana. Przedstawiony układ jest testowany w warunkach laboratoryjnych.

5. Wnioski

Przedstawione w referacie zagadnienia wskazują na komplikacje, które pojawiają się przy próbie harmonogramowania procesów dyskretnych w elastycznych systemach produkcyjnych. W ogólnym przypadku trzeba wyróżnić warstwy: wyznaczenia parametrów operacji /np. uczenie lub projektowanie trajektorii/, wyznaczenia harmonogramu /np. na IBM PC/ oraz realizacji sterowania /uwzględniając sygnały z procesu rzeczywistego/.

W rozpatrywanym problemie założono, że wszystkie objekty znajdują się w magazynie. W przypadku ogólnym należy rozpatrywać strumień obiektów dostarczanych do magazynu. Jeżeli sekwencja obiektów jest dana, to problem można rozwiązać w taki sam sposób - jak w niniejszym artykule. Pewne komplikacje powodują jedynie terminy dostępności obiektów. W przypadku gdy sekwencja obiektów nie jest dana, należy w sposób jawny zdefiniować stan magazynu.

Perspektywę dalszych prac w przedstawionym zakresie stanowi harmonogramowanie procesów dyskretnych w systemach elastycznych o różnych strukturach, składających się z: agregatów, magazynów buforowych, linii transportowych oraz robotów przemysłowych.

LITERATURA

- [1] Automatyzacja dyskretnych procesów produkcyjnych, Materiały V KK, Instytut Automatyki, Politechniki Śląskiej, Gliwice 1986.
- [2] Kowalowski H. i inni: Automatykacja dyskretnych procesów przemysłowych, WNT, Warszawa 1986.
- [3] Kowalowski H. i inni: Laboratorium robotów przemysłowych, Skrypt Pol. Śl. /w druku/.
- [4] Marecki F.: Metoda programowania wieloetapowego, ZN Pol. Śl. s. Automatyka, nr 74, ss. 159-178.
- [5] Marecki F.: Draft of Theory of Discrete Multi-stage Processes, Dysertacja doktorska, WzSzE, Praga, 1988.
- [6] ROBKO - dokumentacja techniczna, Instytut Automatyki, Pol. Śl., Gliwice, 1988.

Recenzent: Doc. dr hab. inż. K. Wala

Wpłynęło do Redakcji do 1988-04-30.

УПРАВЛЕНИЕ ГИБКОЙ ПРОИЗВОДСТВЕННОЙ СИСТЕМОЙ

Резюме

В работе представлена проблема управления гибкой производственной системой. Система состоит из робота и поворотного накопителя. Дан алгоритм календарного планирования для этого процесса.

FLEXIBLE MANUFACTURING SYSTEMS CONTROL

Summary

A control problem for flexible manufacturing system is presented. The system consists of a robot and a rotary store. Scheduling algorithm for the process is presented.