

Zbigniew Banaszak, Andrzej Drzazga, Marek Wnuk

Politechnika Wrocławska  
Instytut Cybernetyki Technicznej

## KOORDYNACJA DZIAŁAŃ W ZŁOŻONYCH PRZEMYSŁOWYCH SYSTEMACH ROBOTYCZNYCH

**Streszczenie.** W pracy przedstawiono zasadnicze trendy warunkujące rozwój systemów sterowania w złożonych systemach robotycznych. Zakres rozważań ograniczony został do zagadnień modelowania i symulacji dyskretnych procesów produkcyjnych oraz technik budowy systemów sterowania koordynujących ich przebieg. Przedstawiono koncepcję budowy systemu sterowania procesami dyskretnymi opartą na wyznaczanych w trybie interakcyjnym modelach sieciowych procesu technologicznego.

### 1. Wstęp

Rozwój w zakresie konstrukcji i zastosowań robotów przemysłowych wiąże się z potrzebami w zakresie budowy i analizy działania systemów robotycznych takich, jak zrobotyzowane gniazda produkcyjne, linie produkcyjne oraz elastyczne systemy produkcyjne. Szczególnego znaczenia w tym względzie nabierają problemy budowy procesorowych systemów sterowania obiektami rozłożonymi w dyskretnych procesach produkcyjnych. Zadania takich sterujących systemów sprowadzają się m.in. do koordynacji przepływów detali i narzędzi, rozdziału zasobów, monitorowania i diagnostyki procesu, itp. Spośród zagadnień występujących zarówno na etapie projektowania, jak i na etapie implementacji systemów sterowania, należy wymienić zagadnienie modelowania i symulacji procedur sterowania, planowania i harmonogramowania procesu produkcyjnego, koordynacji procesów przebiegających współbieżnie oraz stosowania technik zapewniających elastyczność struktury i adaptacyjność funkcjonowania sprzętowej realizacji systemu.

W pracy przedstawiono koncepcję komputerowo wspomaganego projektowania procedur koordynujących współdziałanie urządzeń systemu produkcyjnego w toku obsługi realizowanych w nim zadań. Celem koordynacji jest zapewnienie określonej, np. bezblokadowej realizacji procesu technologicznego. Koncepcja ta oparta jest na wyznaczanych w trybie interakcyjnym modelach sieciowych procesów technologicznych. Przyjęcie sieciowej reprezentacji modelu procesu pozwala na wykorzystanie środków grafiki komputerowej /na etapie projektowania/ oraz daje się łatwo implementować w mikroprocesorowych sterownikach procesów technologicznych.

Prezentowane podejście będąc zasadniczo odmiennym /w sposobie uzyskiwania modelu sieciowego/ od prezentowanego w pracach [4,6] pozwala w sposób szybki i prosty uzyskiwać modele sieciowe rozważanych procesów technologicznych.

## 2. Przemysłowe systemy robotyczne - systemy sterowania

Rozwój automatyzacji dyskretnych systemów produkcyjnych uwarunkowany jest m.in. postępem w zakresie komputerowych systemów sterowania oraz programowanych środków produkcji. Przesłanką takiego postępu jest rozwój robotów przemysłowych umożliwiających elastyczne powiązanie procesów obróbki z procesami obsługi, np. transportu operacyjnego i międzyoperacyjnego przedmiotów. Inną przesłanką jest rozwój systemów sterowania, które w coraz większym stopniu spełniają wymagania elastyczności /przestrajanie systemu produkcyjnego z obsługi jednego procesu technologicznego na drugi/, niezawodności /automatyczne podnoszenie systemu z upadku/ oraz wieloprocusorowości /obsługa procesów przebiegających współbieżnie/. Przykładami systemów integrujących wymienione kierunki rozwojowe są elastyczne systemy produkcyjne, a w szczególności zrobotyzowane stanowiska natryskowego nakładania powłok ochronnych, obróbki wykończającej, kontroli wymiarów, linie zgrzewania karoserii samochodów, itp.

Należy również podkreślić wpływ układów mikroprocesorowych na zmianę podejścia w projektowaniu systemów sterowania. Sprowadza się ono do algorytmów i programów jako zasadniczych elementów w projektowaniu systemów. Sprzęt dobierany jest w dalszej kolejności i traci swoje dotychczasowe, dominujące znaczenie.

Rozwój nowej metodologii projektowania prowadzi do pojawiania się zunifikowanych programowanych środków produkcji o strukturze modułowej. Z kolei tak rozumiana struktura systemu produkcyjnego warunkuje modularność oprogramowania jego systemu sterowania. Własność ta pozwala tworzyć hierarchicznie zorganizowane struktury systemów sterowania integrujące poszczególne poziomy sterowania, tzn. sterowania parametrami obróbki, poszczególnymi urządzeniami, współpracę urządzeń tworzących gniazda lub linie produkcyjne, procesem produkcyjnym w całości.

Sterowanie przemysłowym systemem robotycznym obejmuje m.in. zadanie planowania operatywnego sprowadzające się do określenia dla każdego urządzenia kolejności zadań i czasu ich realizacji /harmonogramu zmianowego/ oraz zadanie sterowania operatywnego związane z nadzorowaniem realizacji harmonogramu i jego korygowanie w warunkach różnorodnych zakłóceń procesu produkcyjnego.

Zadanie planowania operatywnego związane jest z rozdziałem zadań i zasobów /określanie porządku operacji technologicznych realizowanych na określonych maszynach przy użyciu odpowiednich zasobów/ oraz z wyznaczeniem kolejności wykonywanych zadań /sterowanie przepływem materiałów w dyskretnym procesie produkcyjnym/. Wymienione elementy realizowane są na ogół w trybie interakcyjnym korzystającym z technik CAD. Systemy komputerowo wspomaganego projektowania systemów robotycznych umożliwiają wyznaczanie optymalnego rozmieszczenia stanowiska, dobór robotów przemysłowych i urządzeń towarzyszących, a także dokonanie obliczeń efekty-



ności poszczególnych wariantów stanowiska. Coraz częściej też obejmują swym zasięgiem zagadnienie planowania i symulacji procesów oraz wyznaczania programów sterowania przebiegiem procesów.

Zadanie sterowania operatywnego wiąże się m.in. z programową implementacją procedur sterowania nadzorujących wykonanie harmonogramów w autonomicznych blokach sterowników oraz z budową adaptacyjnych, wykorzystujących redundancję funkcjonalną elementów systemu, układów sterowania. Wspólnym dla obu wymienionych zadań jest problem reprezentacji procesu produkcyjnego obejmującego podsystemy odpowiednio sterowania i realizacji produkcji. Reprezentacja taka winna umożliwiać zarówno modelowanie systemu na różnych poziomach jego szczegółowości, jak i planowanie przebiegających w nim procesów, jak również winna pozwalać na prostą programową implementację modelowanych w niej procedur sterowania. Duże możliwości w tym zakresie stwarza język sieci Petriego [3] umożliwiający modelowanie systemów, których komponenty pozostając we wzajemnych uwarunkowaniach mogą funkcjonować w sposób współbieżny. Koncepcje wykorzystania tej reprezentacji do modelowania i analizy dyskretnych procesów produkcyjnych, jak również do wyznaczania procedur koordynujących pracę maszyn i sterujących operacjami przez nie wykonywanymi, przedstawione zostaną w dalszej części opracowania.

### 3. Modelowanie i symulacja procesów technologicznych

Problem modelowania i symulacji procedur sterowania dla potrzeb koordynacji przebiegu procesów w złożonych systemach produkcyjnych stał się aktualny wraz z powstaniem elastycznych systemów produkcyjnych. Postęp w tym zakresie związany jest z jednej strony z budową dialogowych systemów CAD wykorzystywanych na etapie przygotowania produkcji /harmonogramowanie pracy maszyn, rozdział zadań i zasobów/ z drugiej zaś strony związany jest z rozwojem wieloprocesorowych systemów sterowania warunkujących sposób eksploatacji systemu produkcyjnego /sterowanie przepływem materiałów, narzędzi, itp./. Wymienione uwarunkowania winny być rozważane łącznie. Oznacza to, że działanie systemu interakcyjnego winno posiadać zapewnioną możliwość fizycznej realizacji, wyznaczonych z jego pomocą, optymalnych lub tylko dopuszczalnych procedur sterowania procesem /sposobem koordynacji realizacji procesu/.

Z uwagi na specyfikę rozważanych systemów produkcyjnych /systemy czasowo uwarunkowanych operacji procesów przebiegających współbieżnie/ oraz ich systemów sterowania /sterowanie obiektami przestrzennie rozłożonymi/ jako aparat formalny reprezentacji modeli przyjęto język sieci Petriego. Przyjęta reprezentacja pozwala w sposób jednorodny rozważać zarówno etap syntezy procedur sterowania, jak i etap ich programowej implementacji w procesie sterowania.

Z uwagi na ograniczoną objętość pracy pominięte zostały teoretyczne podstawy sieci Petriego. Obszerną literaturę w tym zakresie stanowią prace [1, 2, 3, 5].

Model procesu technologicznego tworzony jest w oparciu o informację zawartą w zbiorach: zbiorze warunków  $W$  określających stany urządzeń systemu produkcyjnego i zbiorze zdarzeń  $Z$  określających czynności realizowane na poszczególnych maszynach. Z każdym zdarzeniem  $z \in Z$  związane są zbiory: zbiór warunków  $W_1 \subset W$  koniecznych dla zajścia zdarzenia i zbiór warunków  $W_2 \subset W$  osiąganych w wyniku zajścia zdarzenia.

Ustalenie takiego przyporządkowania pozwala wyznaczyć diagram wzajemnie uwarunkowanych zdarzeń i zbiorów warunków dla zadanego ciągu technologicznego  $CT$ . Pozwala to dalej rozważać zbiór trójek postaci

$$\left\{ (W_1, z, W_2) : z \in Z' \subset Z \right\}, W_1, W_2 \subset W \quad /1/$$

będący systemem wzajemnie uwarunkowanych operacji rozpatrywanego ciągu technologicznego.

W ogólnym przypadku, gdy w systemie produkcyjnym realizowanych jest kilka ciągów technologicznych, wyznaczany jest zbiór

$$\left\{ (W_1^x, z, W_2^x) : z \in Z^x \subset Z \right\}, Z^x = \cup \{ Z^i : i \in N_1 \} \quad /2/$$

będący złożeniem systemów wzajemnie uwarunkowanych operacji odpowiadających poszczególnym ciągom technologicznym, przy czym zbiory  $W_1^x$  i  $W_2^x$  są sumami odpowiednich zbiorów  $W_1$ ,  $W_2$  będących warunkami koniecznymi dla zajścia zdarzenia  $z$  i osiąganych w wyniku jego zajścia.

Systemy wzajemnie uwarunkowanych operacji postaci /1/ lub /2/ będą dalej rozumiane jako modele procesów technologicznych.

Należy zauważyć, że z tak przyjętą reprezentacją procesu technologicznego wiąże się również odpowiednia reprezentacja sieciowa [1]. Przyporządkowanie to określone jest następująco: Każdemu zdarzeniu procesu /2/ odpowiada jedno przejście sieci  $PN = (P, T, K, M_0)$ , każdemu warunkowi procesu odpowiada jedno miejsce sieci, przy czym różnym warunkom i różnym zdarzeniom odpowiadają odpowiednio różne miejsca i przejścia. Miejsca odpowiadające warunkom koniecznym dla zajścia danego zdarzenia połączone są, wychodzącymi z tych miejsc, łukami z przejściem określonym przez zdarzenie. Z kolei miejsca odpowiadające warunkom osiąganym w wyniku zajścia danego zdarzenia połączone są, dochodzącymi do nich łukami, z przejściem określonym przez rozważane zdarzenie. Niezerowe współrzędne wektora stanu początkowego  $M_0$  określone są przez warunki oznaczające stany oczekiwania poszczególnych urządzeń systemu produkcyjnego. Uzyskana w powyższy sposób sieć  $PN$  interpretowana będzie dalej jako sieciowy model procesu technologicznego. Model sieciowy pozwala na przeprowadzenie analizy procesu technologicznego sprowadzającej się do symulacji palen poszczególnych przejść sieci. Symulacja taka, np. wykorzystująca systemy grafiki komputerowej, pozwala określić poszczególne fazy i/lub warianty przebiegu



/dynamiki/ procesu technologicznego. Tak rozumiana analiza symulacyjna modelu sieciowego pozwala określić sytuacje prowadzące do powstawania blokad w fizycznej realizacji procesu oraz sytuacje związane ze współbieżną realizacją ciągów technologicznych. Pozwala również analizować różne warianty obsługi zadań.

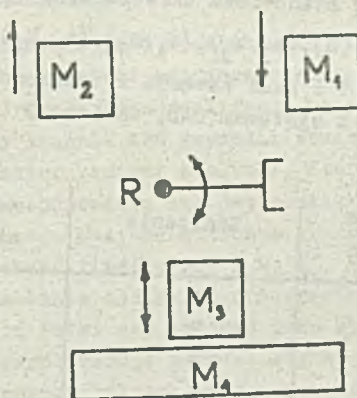
Przeprowadzając analizę w systemach interakcyjnych możliwym się staje wprowadzanie, do modelu sieciowego procesu, pewnych dodatkowych warunków zapewniających pożądaną przebieg procesu. Warunki te modyfikują wzajemne powiązania występowania zdarzeń, koordynując ich zachodzenie spełniające wymagania, np. bezblokadowej czy też zdeterminowanej realizacji procesu /obsługi spływających do niego zadań/.

Tak zmodyfikowany model sieciowy procesu zawiera model systemu sterującego go jako swoją integralną część.

Model ten jest modelem dynamicznym realizacji /obsługi zadań/ procesu technologicznego i będzie dalej interpretowany, jako jego procedura sterowania. W implementacji programowej procedura ta określa kolejność operacji wykonywanych na maszynach, w szczególności, oraz koordynację współpracy poszczególnych maszyn systemu, w ogólności.

Przedstawione rozważania ilustruje poniższy przykład.

Rozważane jest zrobotyzowane stanowisko technologiczne rys. 1 o następującym cyklu pracy: Detal z podajnika  $M_1$ , odbierany jest przez robot  $R$  i ustawiany na transporterze  $M_3$ , który przekazuje go do maszyny  $M_4$ . Po wykonaniu operacji obróbki na maszynie  $M_4$ , detal podawany jest na transporter  $M_3$ , który przekazuje go do robota  $R$ . Robot  $R$  odbiera detal z transportera  $M_3$  i przekazuje go do odbieraka  $M_2$ .



Rys. 1. Konfiguracja stanowiska

Przyjmuje się, że realizacja kolejnych operacji poszczególnych urządzeń stanowiska wymuszane są stanem pozostałych urządzeń, np. zachodzenie operacji przenoszenia przez robot R detalu z  $M_1$  do  $M_4$  uzależnione jest jedynie obecnością detalu na  $M_1$ . W sytuacji takiej może wystąpić stan blokady: np. jeden detal obrabiany jest na  $M_4$ , podczas gdy drugi oczekuje na  $T_3$ . Konieczność zabezpieczenia się przed występowaniem blokad zmusza do wyznaczenia dodatkowych uwarunkowań koordynujących w odpowiedni sposób pracę urządzeń stanowiska przy obsłudze swobodnie spływających zadań /detali/.

Niech w bazie danych systemu CAD zawarta jest informacja o rozważanym stanowisku reprezentowana zawartością tablic 1 i 2.

Tablica 1

Zbiór warunków

- a/  $M_1$  oczekuje na pobranie detalu
- b/ R oczekuje na detal
- c/  $M_3$  oczekuje
- d/  $M_4$  oczekuje
- e/  $M_2$  oczekuje
- f/  $M_1$  został obsłużony przez R
- g/  $M_2$  został obsłużony przez R
- h/  $M_3$  został obsłużony przez R
- i/  $M_4$  otrzymał detal z  $M_3$
- j/  $M_3$  otrzymał detal z  $M_4$
- k/  $M_3$  przesłał detal do R

Tablica 2

Zbiór zdarzeń

- 1/ detal przybywa do  $M_1$
- 2/ R odbiera detal z  $M_1$  i przekłada go na  $M_3$
- 3/  $M_3$  przesuwa detal do  $M_4$
- 4/  $M_4$  obrabia detal i oddaje go na  $M_3$
- 5/  $M_3$  przesuwa detal do R
- 6/ R odbiera detal z  $M_3$  i przekłada go na  $M_2$
- 7/  $M_2$  wysyła detal

Rozważany cykl pracy stanowiska określa cykl technologiczny postaci:

$$CT = \langle (z_1, M_1), (z_2, R), (z_3, M_3), (z_4, M_4), (z_5, M_3), (z_6, R), (z_7, M_2) \rangle$$

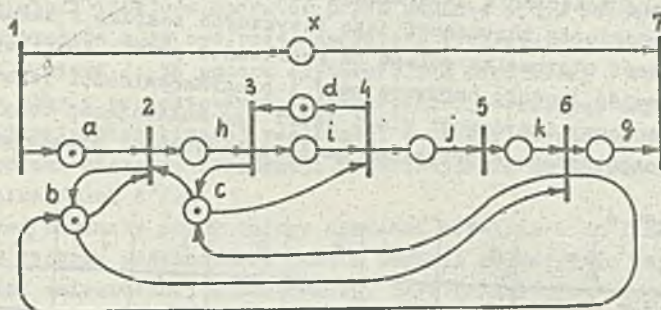
W oparciu o zadany CT i dane zawarte w tablicach 1 i 2 wyznaczana jest, w trybie współpracy z systemem CAD, tablica 3.

Tablica 3

Warunki konieczne zajścia zdarzenia	Zdarzenie	Warunki określone w wyniku zajścia zdarzenia
żaden	1	a
a, b, c	2	h, b
d, h	3	i, c
i, c	4	j, d
j	5	k
k, b	6	g, c, b
g	7	żaden



Dane zawarte w tablicy przetwarzane są automatycznie na postać graficzną reprezentującą model sieciowy procesu technologicznego, rys. 2. Na rysunku tym zaznaczono także warunek X uniemożliwiający powstawanie blokady /np. zachodzą jednocześnie warunki b,h,i/ w realizacji procesu.



Rys. 2. Model procedury sterowania

Przedstawiony w przykładzie tok postępowania może być odniesiony do różnych poziomów szczegółowości opisu systemu produkcyjnego, tj. poziomu koordynacji gniazd i linii produkcyjnych, koordynacji urządzeń współpracujących w gniazdach czy wreszcie operacji realizowanych na poszczególnych maszynach. Podejście takie zapewnia również, przy określonej strukturze systemu sterowania, możliwość modyfikacji on-line procedury sterowania realizacją procesu technologicznego, np. w wypadku wystąpienia częściowych lub całkowitych uszkodzeń maszyn systemu produkcyjnego.

#### 4. Struktura systemu sterowania

Opisana metoda realizowana jest w strukturze dwuwarstwowej układu sterowania. Pierwszą warstwę stanowi wspomagane komputerowo projektowanie systemu sterowania, a drugą - jego realizacja na procesie. Do wspomaganie projektowania służy konsola ADC /specjalizowany  $\mu C$  wyposażony w monitor graficzny i programator pamięci stałych/. Sterownik realizujący przygotowany program jest rozbudowanym układem kombinacyjnym. Najlepiej wykonać go w postaci modułowego systemu mikrokomputerowego wyposażonego w odpowiednią do danego procesu ilość wejść i wyjść, którego jednostka centralna posiada oprogramowanie zdolne do interpretacji macierzy incydencji warunków i zdarzeń sieci Petri'ego zadanej jako struktura programu dla sterownika. Sterownik współpracuje poprzez swe wejścia i wyjścia z lokalnymi sterownikami CNC poszczególnych maszyn realizujących proces. Przyjmuje się, że układy CNC dysponują oprogramowaniem wymaganym do realizacji operacji wchodzących w skład procesu. Sygnały ze sterownika służą jedynie do inicjacji odpowiednich procedur sterowania poszczególnych maszyn zgod-

nie z wprowadzonym do sterownika programem. Sygnalizacja zakończenia operacji jest interpretowana w sterowniku jako spełnienie odpowiedniego warunku.

Wykonane połączenia determinują interpretację poszczególnych wierszy i kolumn macierzy incydencji jako fizycznych zdarzeń i warunków procesu. Elastyczność sterownika polega na możliwości prawie dowolnej rozbudowy wejść i wyjść /moduły hardware'owe/ i programowalności jego funkcji za pomocą macierzy incydencji sieci Petri'ego modelującej dany proces.

Dla rozpatrywanego przykładu macierz incydencji przedstawiono w tab.4.

Tablica 4

	a	b	c	d	g	h	i	j	k	x
1	1;0									1;0
2	1;1	1;1	0;1			1;0				
3			1;0			0;1	1;0			
4			0;1				0;1	1;0		
5								0;1	1;0	
6		1;1	1;0		1;0				0;1	
7					0;1					0;1

Zauważmy, że przedstawienie modelu w postaci macierzy incydencji nie jest dla operatora konieczne. Jest natomiast wewnętrzną daną niosącą informację o strukturze programu koordynującego przebieg procesu. Informacja ta pozwala wraz z zadaniem stanem początkowym  $M_0$  w oparciu o zależność  $M' = M + U_t^T C^+ - U_t^T C^-$  określić kolejne stany modelu dla badanego procesu /gdzie  $C = C^+; C^-$  jest macierzą incydencji [1],  $M$  - stanem wyjściowym,  $U_t$  - wektorem jednostkowym odpowiadającym przygotowanemu przejściu  $t$ /.

Oprogramowanie konsoli CAD zawiera następujące bloki funkcjonalne:

- 1<sup>o</sup> edycja macierzy incydencji;
- 2<sup>o</sup> graficzna interpretacja sieci na monitorze, lub pisaku X-Y;
- 3<sup>o</sup> symulacja działania sieci;
- 4<sup>o</sup> generacja programu dla sterownika.

Tworzenie i poprawianie macierzy incydencji można uzyskać przez wprowadzenie dodatkowych warunków do modelu sieciowego procesu produkcyjnego w formacie zgodnym z tab. 3, bądź poprzez wprowadzenie warunków koordynujących bezpośrednio na graficznym obrazie modelu sieciowego. Postać graficzna modelu sieciowego ładowana jest do pamięci operacyjnej w postaci macierzy o dwubitowych elementach. Każda macierz incydencji może być zobrazowana graficznie na ekranie monitora, lub pisaku X-Y. Ograniczenia tanowią rozmiary ekranu i czytelność obrazu sieci. Można tych ograniczeń



uniknąć stosując segmentację obrazu /wyświetlanie wybranych fragmentów sieci/, oraz agregację sieci polegającą na zastępowaniu pewnych jej fragmentów /dobrze zbadanych poprzednio/ symbolem graficznym przejścia /interpretowanym jako zdarzenie złożone/.

Symulację sieci inicjuje operator przez podanie stanu początkowego  $Z_0$ . Wyniki eksperymentów mogą być przez operatora wykorzystane do wprowadzania poprawek w sieci / za pomocą edytora/. Tak więc bloki 1<sup>o</sup>-3<sup>o</sup> są wykorzystywane w czasie projektowania interakcyjnie, pozwalając uzyskać kolejne przybliżenia poszukiwanej realizacji procesu. Otrzymaną dopuszczalną sieć w postaci macierzy incydencji przekazuje się do opisanego wcześniej sterownika /blok 4<sup>o</sup>/.

Możliwe są dwa warianty pracy całego systemu: off-line i on-line. W trybie off-line program przekazywany jest w postaci odpowiednio zapisanej pamięci stałej umieszczonej w sterowniku. Tryb on-line jest wygodniejszy przy uruchamianiu systemów i polega na umieszczeniu w sterowniku modułu pamięci RAM z układem transmisji danych i autonomicznym zasilaniem. Urządzenie wspomagające łączy się z tym modułem przez odpowiednią linię transmisyjną. Raz przesłany program pozostaje w pamięci RAM do następnej transmisji. Operator może zmodyfikować zadany program dla sterownika. Warto zwrócić uwagę, że w obu trybach jedno urządzenie /konsola CAD/ może obsługiwać wiele sterowników. W trybie on-line konieczne jest jedynie komutowanie linii transmisyjnych. Przy pracy on-line szczególnie istotne jest wyposażenie urządzenia w pamięć zewnętrzną /dyski elastyczne, pamięci kasetowe, itp./ umożliwiającą zorganizowanie biblioteki programów dla różnych sterowników. Dodatkowe wyposażenie urządzenia może stanowić oprogramowanie umożliwiające dokumentowanie opracowanych programów, np. na drukarce mozaikowej.

## 5. Zakończenie

Przedstawiona koncepcja budowy systemu sterowania procesami dyskretnymi umożliwia konwersacyjny tryb konstruowania tegoż systemu. Aparat sieci Petri'ego okazał się wygodnym narzędziem interakcyjnego modelowania i automatycznej symulacji dyskretnych procesów produkcyjnych. Pozwala on na koordynację procesów przebiegających współbieżnie, eliminując sytuacje związane z występowaniem blokad, i na podejmowanie decyzji w przypadkach występowania konfliktów, już na etapie projektowania.

Ostateczny kształt procedur koordynacji i sterowania uzyskiwany w trakcie badania modelu może zostać wprowadzony do pamięci sterownika zapewniając pracę systemu zgodną z wynikami eksperymentów przeprowadzonych na modelu.

Przyjęta struktura systemu sterowania jest nadmiarowa. Wynika to z uniwersalności przyjętej koncepcji. Sterownik umożliwia bowiem nie tylko

realizację sterowania i koordynacji procesem, ale także testowanie poprawności modelu, jego rozszerzanie i modyfikację, a także formułowanie procedur sterowania. Proces ten nie stawia istotnych wymagań operatorowi. Rozwiązanie wykorzystujące graficzne przedstawienie modelu pozwala na prostą interpretację zachodzących w trakcie procesu zjawisk.

Recenzent: Doc dr inż. Andrzej Keczmarczyk

Wpłynęło do Redakcji do 30.03.1984r.

#### LITERATURA

- [1] Peterson J.L.: Petri net theory and the modelling of systems, Prentice-Hall, Englewood Cliffs, New York 1981.
- [2] Starke P.H.: Petri - netze, VEB Deutscher Verlag der Wissenschaften Berlin 1980.
- [3] Brauer W., red: Net theory and applications, w: Lecture notes in computer science B4, Springer Berlin 1980.
- [4] Masuda R., Hasegawa K.: Mark Flow Graph and its applications to complex sequential system, Proc. of the 13th Hawaii Int. Conf. on System Science 1980, s. 194-203.
- [5] Girault C., Reisig W. red.: Application and theory of Petri nets, Informatik - Fachberichte 52, Berlin 1982.
- [6] Hasegawa K., Masuda R.: Mark Flow Graph and its application to programming of industrial robot system, The 5th Symp. on Industrial Robots, Tokyo 1978.

#### КООРДИНАЦИЯ ДЕЙСТВИЙ В СЛОЖНЫХ РОБОТОТЕХНИЧЕСКИХ СИСТЕМАХ

##### Р е з ю м е

В статье представлены основные тенденции развития систем управления сложными робототехническими системами. Рассматриваются вопросы моделирования дискретных процессов, а также методы построения систем управления, координирующих протекание процессов. Представлено концепцию построения систем управления дискретными процессами, основанную на применении в интерактивных системах сетевых моделей технологических процессов.

#### FUNCTIONAL COORDINATION IN COMPLEX INDUSTRIAL ROBOTIC SYSTEMS

##### С и ж м а р у

The paper is concerned with simulation and modelling of discrete production processes, and control of these processes. Basic trends in development of control systems for complex robotic systems are discussed. A conception is proposed of the control system based upon the interactivel established net model of the technological process.