ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2002
Seria: AUTOMATYKA z. 135 Nr kol. 1555

Damian KRENCZYK, Bożena SKOŁUD
Politechnika Śląska

METODA IDENTYFIKACJI STRUKTURY SYSTEMU*

Streszczenie. W artykule przedstawiono metodę identyfikacji struktury systemu (cykli podstawowych). Znajomość struktury systemu jest niezbędna do podejmowania decyzji o konstrukcji lokalnych reguł rozstrzygania konfliktów zasobowych w fazie wybiegu, rozbiegu systemu i zmiany przepływów w związku ze zmianą produkowanego asortymentu. Odpowiednia ich konstrukcja gwarantuje żywotność globalną. W przedstawionej metodzie identyfikacji cykli podstawowych wykorzystano teorię grafów. Przedstawiono procedurę identyfikacji cykli oraz przykład ilustrujący.

THE METHOD OF SYSTEM STRUCTURE IDENTIFICATION

Summary. In the paper, the problem of the system synchronisation is provided. The method of the system structure identification is presented. The system structure (especially closed loop) identification is needed for dispatching rule construction that is used in the phase of system starting-up, cease and transition between two different known production flows. The method of identification base on graph theory. The identification procedure and the illustrative example are presented.

1. Wprowadzenie

We wcześniejszych pracach autorów [3,9] rozważano problemy planowania i sterowania wieloasortymentową produkcją rytmiczną w systemie dla przebiegów ustalonych. Wykazano, że spełnienie warunku bilansu sytemu (liczba procesów wchodzących do systemu w jednym cyklu jest równa liczbie procesów opuszczających system w tym okresie) oraz warunku pojemności magazynów międzyoperacyjnych (pojemność magazynu międzyoperacyjnego jest nie mniejsza niż krotność wystąpienia danego procesu w jednym cyklu) zapewnia żywotność globalną systemu (brak blokad) w przebiegu ustalonym [8].

* Pracę wykonano częściowo w ramach projektu badawczego KBN 8 T11A 020 18

Jednakże rosnące wymagania rynku na coraz to bardziej zróżnicowane wyroby powodują, że produkcja realizowana jest w krótkich seriach na zlecenie klienta (make-to-order - MTO), a co za tym idzie, przebiegi procesów często ulegają zmianie, gdyż pewne procesy opuszczają system, z kolei inne są do niego przyjmowane [1]. Istotnym problemem jest zatem przeprowadzenie systemu z jednego przebiegu ustalonego w inny.

Rozważane będą systemy o sterowaniu rozproszonym [8, por.6], bazującym na lokalnych regułach rozstrzygania konfliktów zasobowych (LRRKZ). Reguła taka określa liczbę oraz kolejność procesów realizowanych na zasobie i zapewnia przynajmniej jednokrotną realizację operacji należącej do każdego z procesów, przebiegających przez wspólny zasób, tym samym gwarantuje żywotność lokalną (brak zagłodzeń).

Doprowadzenie do oczekiwanego przebiegu ustalonego wymaga przyjęcia odpowiedniej kolejności procesów w regule lub wykonania pewnej sekwencji procesów (innej niż w przepływie ustalonym), którą nazwano regułą rozruchu [3]. Wykonanie reguły rozruchu ma na celu wstępne zapełnienie magazynów międzyoperacyjnych, co z kolei gwarantuje bezblokadowe funkcjonowanie systemu dla zadanego zbioru LRRKZ. Wybór sposobu postępowania przy konstruowaniu LRRKZ jest zależny od występowania cykli podstawowych w strukturze systemu produkcyjnego.

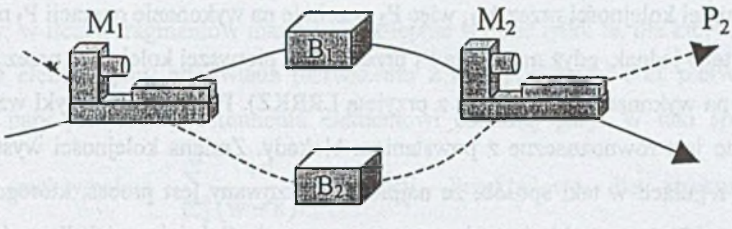
W szczególności oznacza to opracowanie metody umożliwiającej przechodzenie od arbitralnie wybranego stanu systemu do jednego ze stanów występujących w docelowym przebiegu ustalonym. Każdy przebieg ustalony charakteryzuje cyklicznie powtarzająca się sekwencja stanów systemu (alokacji procesów do zasobów). Przyjmuje się, że arbitralnie wybrany stan początkowy nie należy do tej sekwencji. Ponieważ rozważane przebiegi ustalone generowane są przez zbiory reguł alokowanych do zasobów dzielonych systemu, więc naturalną metodą wydaje się propozycja wyznaczania zbioru reguł rozruchowych, które umożliwią przejście z wybranego stanu początkowego do jednego ze stanów sekwencji przebiegu ustalonego. W pracy tej pokazane będzie, że budowa reguł rozruchowych może zostać ograniczona do reguł przypisanych do zasobów pojawiających się w cyklach zamkniętych.

2. Analiza przypadku

Topologia systemu w znacznym stopniu wymusza jego zachowanie. Wyróżnia się trzy podstawowe struktury systemów produkcyjnych: liniową, otwartą oraz zamkniętą [5,7].

W systemach, w których występują jedynie struktury liniowe i/lub otwarte, nie tworzą się cykle wzajemnych oczekiwań. Nie jest zatem spełniony jeden z warunków koniecznych powstania blokady. W przypadku struktury zamkniętej cykl wzajemnych oczekiwań może wystąpić, tym samym powstanie blokada, co ilustruje poniższy przykład.

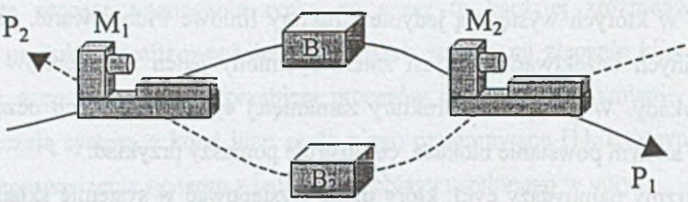
Rozpatrzmy najprostszy cykl, który może występować w systemie składającym się z dwóch zasobów M_1 i M_2, przez który przebiegają marszruty procesów P_1 i P_2 (rys. 1). Marszruta P_1 oraz P_2 przebiegają przez M_1 i M_2.



Rys. 1. Cykl zamknięty (I)
Fig. 1. The closed cycle (I)

Jeżeli do zasobów z tego cyklu przydzielone zostaną takie LRRKZ, że spełniają warunek bilansu systemu i warunek pojemności magazynów, w systemie nie może dojść do powstania blokady. Spowodowane jest to tym, że niezależnie od kolejności występowania procesów w regule przydzielonej do zasobu M_1, bądź R_1=(2,1), co oznacza, że w pierwszej kolejności jest realizowany P_2, a następnie P_1, bądź też R_1=(1,2), co oznacza, że najpierw realizowany jest P_1, a później P_2, to procesy zostaną zrealizowane na tym zasobie, ponieważ oba procesy rozpoczynają na nim swoje marszruty. Również na zasobie M_2 procesy mogą być realizowane niezależnie od ich kolejności w regule. Zatem nie wystąpi cykl wzajemnych oczekiwań.

Jeżeli zaś marszruty procesów będą przebiegać w sposób pokazany na rysunku 2, to w systemie może wystąpić blokada.



Rys. 2. Cykl zamknięty (II)
Fig. 2. The closed cycle (II)

Dochodzi do niej w przypadku gdy do zasobów zostaną przydzielone następujące LRRKZ: $R_1=(2,1)$ i $R_2=(1,2)$ to w pierwszej kolejności na zasobie M_1 musi zostać wykonana operacja procesu P_2. Ponieważ jednak marszruta P_2 najpierw przebiega przez zasób M_2, a dopiero w drugiej kolejności przez M_1, więc P_2 oczekuje na wykonanie operacji P_1 na M_2. Nie dochodzi do tego jednak, gdyż marszruta P_1 przebiega w pierwszej kolejności przez M_1, a tam ciągle czeka na wykonanie P_2 (zgodnie z przyjętą LRRKZ). Powstaje więc cykl wzajemnych oczekiwań, co jest równoznaczne z powstaniem blokady. Zmiana kolejności występowania procesów w regułach w taki sposób, że najpierw realizowany jest proces, którego operacja realizowana na danym zasobie jest pierwszą w marszrucie (lub jej części dla ogólniejszego przypadku), a następnie realizowany jest proces, którego operacja jest ostatnią w marszrucie, co zapewnia realizację przepływu bez blokad. Dla przypadku z rysunku 2 takimi regułami są: $R_1=(1,2)$ i $R_2=(2,1)$.

Z przeprowadzonej analizy wynika, że w celu budowy fazy startowej LRRKZ konieczne jest opracowanie metody identyfikacji takich cykli w systemie, które powodują wystąpienie cyklu wzajemnych oczekiwań na zasobach.

W rozważanej klasie systemów (patrz przykład ilustrujący) można zastosować metodę *zapobiegania blokadom* polegającą na sekwencyjnym uporządkowaniu żądań - zapobiegającą powstaniu cyklu wzajemnych oczekiwań. Nowy przebieg ustalony procesów w systemie może charakteryzować się innym okresem powtarzalności lub inną kolejnością realizacji procesów w cyklu.

W systemach o złożonej strukturze, przed przydziałem LRRKZ, istotna jest automatyczna identyfikacja cykli zamkniętych.

Rozważany dalej problem sprowadza się do pytania: czy możliwa jest automatyczna identyfikacja cykli podstawowych w systemie współbieżnych procesów? Identyfikacja ta umożliwi budowę procedury sterującej w warunkach rozbiegu systemu.

3. Identyfikacja cykli podstawowych

Cykl podstawowy w systemie współbieżnych procesów produkcyjnych jest tworzony przez zbiór marszrut przebiegających przez skończoną liczbę zasobów w taki sposób, że startując z dowolnego zasobu i poruszając się wzdłuż marszrut lub ich fragmentów, zgodnie z kierunkiem ich przebiegu, można powrócić do zasobu startowego, bez przechodzenia dwukrotnie przez tę samą marszrutę.

W celu wyznaczenia zbiorów zasobów, pomiędzy którymi fragmenty marszrut tworzą cykle podstawowe, należałoby ze wszystkich fragmentów marszrut występujących w strukturze systemu utworzyć wszystkie k-elementowe wariacje bez powtórzeń, gdzie k=2,3,...,w; w-liczba fragmentów marszrut. Następnie wybrać tylko te, dla których dla każdej pary drugi element pary odpowiada pierwszemu z następnej pary, oraz pierwszy element pierwszej pary jest równy ostatniemu elementowi ostatniej pary. W taki sposób można utworzyć maksymalnie $\sum_{k=2}^{w} \frac{w!}{(w-k)!}$ wariacji. Przykładowo dla struktury systemu składającego się ze 100 fragmentów marszrut czas obliczeń wyniósłby około $80 \cdot 10^{146}$ lat obliczeń przy założeniu, że jedna wariacja zajmuje nam 0,001 sekundy.

Zdecydowano się na wykorzystanie teorii grafów [2] do wyznaczania cykli podstawowych. Graf jest modelem umożliwiającym opis struktury systemów produkcyjnych. W przyjętym modelu węzły grafu identyfikowane są przez przyporządkowane im numery. Zbiorem poprzedników ustalonego węzła x w grafie $\langle N,A \rangle$ nazywamy zbiór $P(x)$ złożony z takich węzłów y grafu, dla których para (y,x) jest łukiem grafu, czyli (y,x) należy do A. Zbiorem następników ustalonego węzła x w grafie $\langle N,A \rangle$ nazywamy zbiór $S(x)$ złożony z takich węzłów y grafu, dla których para (x,y) jest łukiem grafu, czyli (x,y) należy do A. Dla struktury systemu produkcyjnego węzłami grafu są zasoby produkcyjne, łuki grafu symbolizują fragmenty marszrut produkcyjnych zgodnie z kierunkiem ich przebiegu pomiędzy każdymi dwoma sąsiadującymi zasobami dla wszystkich procesów realizowanych w systemie.

Grafem zorientowanym G nazywamy parę $\langle N,A \rangle$, gdzie N - dowolny zbiór reprezentujący węzły grafu, A - zbiór par uporządkowanych (x,y) reprezentujących łuki grafu, x, y elementy zbioru N. Konturem w grafie $G = \langle N,A \rangle$ nazywamy ciąg $x_1, x_2, \ldots ,x_k,x_1$ taki, że (x_i, x_{i+1})) należy do zbioru łuków A dla (i = 1,2,....k-1) oraz (x_k,x_1) należy do A. Konturem elementarnym nazywamy kontur, w którym $x_i \neq x_j$, dla wszystkich $i \neq j$.

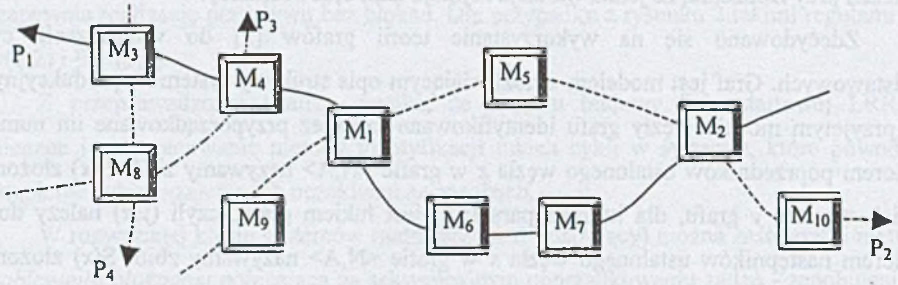
Pojęcie konturu elementarnego w teorii grafów odpowiada pojęciu cyklu podstawowego w strukturze systemu produkcyjnego. Zatem identyfikacja cykli podstawowych w strukturze systemu sprowadza się do identyfikacji konturów elementarnych w grafie.

Przykład ilustrujący

Dany jest system produkcyjny składający się z 10 zasobów. W systemie jednocześnie realizowane są 4 procesy produkcyjne. Macierze procesów realizowanych w systemie są następujące:

$$M_{P1} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 7 & 6 & 1 & 4 & 3 \\ 2 & 5 & 4 & 3 & 1 & 2 \end{bmatrix}, M_{P2} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 9 & 1 & 5 & 2 & 10 \\ 2 & 5 & 3 & 4 & 1 \end{bmatrix}, M_{P3} = \begin{bmatrix} 1 & 2 \\ 8 & 4 \\ 3 & 8 \end{bmatrix}, M_{P4} = \begin{bmatrix} 1 & 2 \\ 3 & 8 \\ 2 & 2 \end{bmatrix}$$

Elementy pierwszego wiersza macierzy oznaczają numery kolejne operacji w marszrucie. Elementy drugiego wiersza odpowiadają numerom zasobów, zaś elementom trzeciego wiersza odpowiadają czasy jednostkowe tych operacji. Strukturę systemu pokazuje rysunek 3.



Rys. 3. System o strukturze zamkniętej
Fig. 3. The system with closed structure

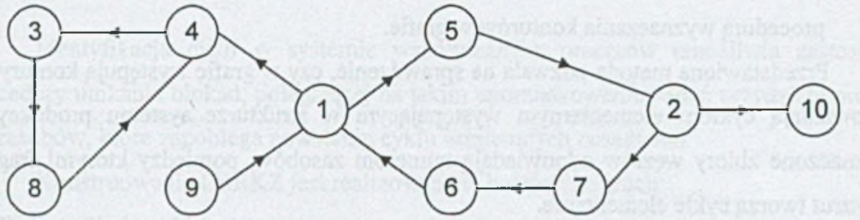
Dla struktury systemu produkcyjnego węzłami grafu G są zasoby produkcyjne, do opisu których zostaną użyte wartości liczbowe numerów zasobów. Węzeł symbolizujący zasób M_1 mieć będzie wartość 1. Łuki grafu symbolizujące fragmenty marszrut produkcyjnych zgodnie z kierunkiem ich przebiegu pomiędzy każdymi dwoma sąsiadującymi zasobami dla wszystkich procesów realizowanych w systemie zapisane będą jako zbiór par uporządkowanych (x,y), gdzie x i y są numerami wierzchołków. Dane te zostaną pobrane z drugiego wiersza macierzy procesów. Graf został zapisany następująco:

G=<N,A>;

N={1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

A= {(2, 7), (7, 6), (6, 1), (1, 4), (4, 3), (9, 1), (1, 5), (5, 2), (2, 10), (8, 4), (3, 8) };

Ilustrację graficzną przedstawia rys.4.



Rys. 4. Graf G

Fig. 4. The graph G

Rozwiązanie problemu identyfikacji konturów w grafie rozpoczyna się od wyznaczenia numeracji topologicznej węzłów. Numeracja topologiczna węzłów grafu G=<N,A> określona jest wówczas, gdy każdemu węzłowi (x) grafu przyporządkowana jest liczba naturalna *tnr(x)* ze zbioru (1, 2, ..., n) (n to liczba węzłów grafu) o tej własności, że dla każdego łuku (x,y) grafu zachodzi warunek *tnr(x)<tnr(y)*. Liczbę *tnr(x)* nazywa się numerem topologicznym węzła (x).

Metoda rozwiązania sprowadza się do realizacji kolejnych kroków:

- budujmy listę *T* węzłów. Konstrukcję listy *T* rozpoczynamy od umieszczenia na niej wszystkich węzłów, dla których liczba poprzedników jest równa zero;
- rozpatrujemy kolejne węzły z listy *T*, aż do jej wyczerpania;
- dla kolejnego węzła *x* z listy *T* "wykreślamy" wszystkie łuki *(x,y)* wychodzące z *x*, przy czym jeśli po "wykreśleniu" łuku *(x,y)* liczba poprzedników węzła *y* będzie równa zero, to węzeł *y* dołączamy na koniec listy *T*;
- jeśli na liście *T* dadzą się umieścić wszystkie węzły grafu, to graf jest bezkonturowy i numeracja topologiczna jest określona przez kolejność występowania węzłów na liście;
- jeśli natomiast na liście *T* nie dadzą się umieścić wszystkie węzły grafu, to graf zawiera kontury. Kontury te na pewno nie przechodzą przez węzły znajdujące się na liście *T* i należy ich poszukiwać w zbiorze węzłów *N-T*, ze względu na fakt, że graf jest bezkonturowy wtedy i tylko wtedy, gdy istnieje numeracja topologiczna jego węzłów;

- dalej stosujemy metodę przeszukiwania w głąb (ang. DFS) grafu G z nawrotami, znacząc drogi przemieszczania się. Wychodząc z ustalonego węzła *x* posuwamy się od węzła do węzła wzdłuż pewnego drzewa generowanego przez listę następników S(i) (poprzedników P(*x*)) kolejno rozpatrywanych węzłów. Jeśli przyjąć, że z każdym węzłem grafu związany jest atrybut sprawdzony lub niesprawdzony, dysponujemy procedurą wyznaczania konturów w grafie.

Przedstawiona metoda pozwala na sprawdzenie, czy w grafie występują kontury, które odpowiadają cyklom elementarnym występującym w strukturze systemu produkcyjnego. Wyznaczone zbiory węzłów odpowiadają numerom zasobów, pomiędzy którymi fragmenty marszrut tworzą cykle elementarne.

Do identyfikacji konturów elementarnych w grafie zastosowano program *Kontur* wykorzystujący algorytm przeszukiwania w głąb z powracaniem. Kod źródłowy programu oraz jego opis został zamieszczony w [2].

Przykład identyfikacji

Dla grafu G z rys.4 należy wyznaczyć wszystkie kontury elementarne. W tym celu dane o grafie G:

G=<N,A>;

N={1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

A= {(2, 7), (7, 6), (6, 1), (1, 4), (4, 3), (9, 1), (1, 5), (5, 2), (2, 10), (8, 4), (3, 8) };

zostały zapisane w pliku wejściowym programu *Kontur*. Poniżej podano zawartość pliku wejściowego i wyjściowego programu.

Plik wejście:	Plik wyjście:	
GRAF G:	GRAF POCZATKOWY :	LUKI :
A	WEZLY :	X Y
A X Y	X	/ 2 7
/ 2 7	. 2	/ 7 6
/ 7 6	. 7	/ 6 1
/ 6 1	. 6	/ 1 4
/ 1 4	. 1	/ 4 3
/ 4 3	. 4	/ 9 1
/ 9 1	. 3	/ 1 5
/ 1 5	. 9	/ 5 2
/ 5 2	. 5	/ 2 10
/ 2 10	. 10	/ 8 4
/ 8 4	. 8	/ 3 8
/ 3 8		***
***		KONTUR : 1 5 2
		7 6 1
		KONTUR : 3 8 4
		3
		LICZBA KONTUROW = 2

W wyniku przeprowadzonej analizy zidentyfikowano dwa kontury, co jest równoważne z istnieniem cykli zamkniętych.

4. Konstrukcja reguł

Identyfikacja cykli w systemie współbieżnych procesów umożliwia zastosowanie procedury unikania blokad, polegającej na takim uporządkowaniu żądań przydziału procesów do zasobów, które zapobiega powstaniu cyklu wzajemnych oczekiwań.

Konstruowanie LRRKZ jest realizowane w kolejnych fazach:

1. Zidentyfikuj cykle podstawowe w systemie.
2. Jeśli cykle podstawowe występują, to utwórz listę L1 zasobów, które tworzą te cykle.
3. Utwórz listę L2 zasobów dzielonych tworzących cykle.
4. Dla LRRKZ przydzielonych do zasobów z listy L2 uporządkuj procesy w regułach zgodnie z następującą zasadą: w pierwszej kolejności przydziel procesy, których kolejne operacje w marszrucie realizowane są na innych zasobach w cyklu, następnie pozostałe procesy. Tak konstruowane reguły zapewniają rozruch systemu i jego pracę bez blokad.
5. W celu zsynchronizowania systemu do zasobu krytycznego należy dodatkowo wyznaczyć krotność realizacji każdego z procesów w regule rozruchu [3,9].

5. Podsumowanie

W artykule przedstawiono metodę automatycznej identyfikacji cykli podstawowych w strukturze systemu wytwórczego. Znajomość tych cykli umożliwia generowanie reguł w fazie rozruchu, dla LRRKZ, które spełniają warunek bilansu systemu. Znajomość sekwencji procesów w fazie rozruchu i w stanie ustalonym umożliwia z kolei precyzyje określenie terminu realizacji zadania i sterowanie pracą systemu wytwórczego bez konieczności ciągłego odwoływania się do operatora ludzkiego.

LITERATURA

1. Hendry L.C., Kingsman B.G.: Production planning systems and their applicability to make to order companies. European Journal of Operational Research, nr 40, 1989, s.1-15.

2. Jankowski B.: Programowanie w praktyce. Mikom, Warszawa 1999.
3. Krenczyk D., Skołud B.: Samosynchronizacja systemów rozproszonych: zastosowanie dynamicznych reguł wyboru priorytetu. (XII KKAPD, Zakopane, 2000) Zeszyty naukowe Politechniki Śląskiej, ser. Automatyka, z.129, s. 243-252.
4. Majdzik P.: Algorytmy synchronizacji systemów sekwencyjnych procesów cyklicznych. Praca doktorska . Wydział Elektryczny, Politechnika Poznańska, Poznań 1998.
5. Perkins J.R., Humes C., Kumar P.R.: Distributed Scheduling of Flexible Manufacturing Systems. IEEE Transactions on Robotics and Automation, vol.10, nr 2,1994, s. 133-141.
6. Pinedo M.: Scheduling, Algorithms and Systems. Prentice Hall, New Jersay 1995.
7. Sawik T.: Optymalizacja dyskretna w elastycznych systemach produkcyjnych, WNT, Warszawa 1992.
8. Skołud B.: Planowanie wieloasortymentowej produkcji rytmicznej. Zeszyty Naukowe Politechniki Śląskiej, seria Mechanika, z.136, Gliwice 2000.
9. Skołud B., Krenczyk D.: Flow synchronisation of the production systems: the distributed control approach. The 6th IFAC Workshop on Intelligent Manufacturing Systems IMS'2001, Poznań, 2001, s. 127- 132.

Recenzent: Prof. dr hab. inż. Zbigniew Banaszak

Abstract

The ability to quickly validate market demands and react to them decides about the competitive advantage. The manufacturer should take the decision about the order acceptance in the moment when the production order is placed. Being competitive involves the organisation method of production flow, and, first and foremost, the time at which the method is chosen and applied. The speed of decision making in the area of production flow and creating control procedures for this flow has a crucial sense. In the case when the production flow is changeable the very important problem is a system transition from one known steady state to another, expected one. The concept of the distributed control based on the dispatching rule allocation to the resources is discussed. In the paper, the problem of the system synchronisation is provided. The behaviour of the system in such situation is depending on the system structure. In the paper the method of the system structure identification is presented. The system structure (especially closed loop) identification is needed for decision making about dispatching rule construction that is used in the case of system starting-up, cease or transition between two different known production flows. The method of identification based on graph theory and illustrative example are presented.