ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1990
Seria: AUTOMATYKA z.100 Nr kol.1082

Ewa Dudek-Dyduch
Akademia Górniczo-Hutnicza

DEKOMPOZYCJA W OPTYMALIZACJI STEROWANIA DYSKRETNYCH PROCESÓW PRZEMYSŁOWYCH

Streszczenie. Artykuł związany jest z metodologią optymalizacji sterowania dyskretnych procesów przemysłowych. Dotyczy zastosowania metody dekompozycji. W pracy scharakteryzowano algorytmy oparte na dekompozycji, podano dyskusję dotyczącą złożoności obliczeniowej oraz klasyfikację algorytmów heurystycznych. Wprowadzone pojęcia zilustrowano przykładami.

1. Wstęp

Artykuł związany jest z metodologią optymalizacji sterowania dyskretnych procesów przemysłowych. Jak wiadomo, rozważaniom podlega wiele bardzo różnorodnych procesów, optymalizowanych przy różnych kryteriach. Algorytmy częstokroć prezentowane są w sposób mało sformalizowany, zaś uzasadniane są za pomocą dowodów nie wprost lub są heurystyczne. Często zatem, przy analizie zaproponowanego algorytmu optymalizacji konkretnego problemu powstają pytania:

- na czym polega zastosowana metoda rozwiązywania problemu,
- jakie własności problemu sprawiły, że zastosowane podejście było możliwe,
- przy jakich uogólnieniach bądź zmianach w problemie zastosowane podejście będzie nadal aktualne,
- na czym polega uproszczenie powodujące zaliczenie danego algorytmu do heurystycznych.

Artykuł związany jest z próbą odpowiedzi na te pytania. Dotyczy konkretnego podejścia stosowanego bardzo często w optymalizacji dyskretnych procesów produkcyjnych (DPP), a mianowicie zastosowania dekompozycji. Dekompozycja stosowana jest w różnych dziedzinach wchodzących w skład ogólnej tematyki zwanej "zarządzaniem produkcją", takich jak optymalizacja planów, optymalizacja struktury systemu (optymalizacja systemów elastycznych) i innych. Niniejszy artykuł związany jest z dekompozycją w problemach optymalizacji sterowania procesami.

Dla problemów przedstawianych za pomocą modeli w przestrzeniach liczbowych (zarówno statycznych, jak i dynamicznych, liniowych i nieliniowych) istnieje bogata literatura. Do klasycznych pozycji należą między innymi [6],[7],[8]. Jednakże sposoby rozwiązywania problemów sterowania DPP naogół nie wykorzystują modeli całkowito - liczbowych. Dlatego też konieczne jest odmienne podejście do analizy tych problemów oraz ich algorytmów. Ponieważ w celu porównywania i klasyfikacji algorytmów konieczne jest przedstawienie poszczególnych problemów za pomocą modeli jednakowego typu, w niniejszej pracy wykorzystywany będzie formalny, rekurencyjny model dyskretnego procesu produkcyjnego zaproponowany w [2],[3].

2. Formalna postać problemu optymalizacji sterowania DPP

Przez określenie "dyskretny proces produkcyjny" będziemy rozumieli zbiór trajektorii jednoznacznie wyznaczony przez czwórkę $P = (s_o, f, S_N, S_F)$, gdzie

$f : U \times S \rightarrow S$ jest funkcją częściową zwaną funkcją przejścia.

U jest zbiorem decyzji sterujących,

$S = X \times T$ jest zbiorem stanów uogólnionych,

X jest zbiorem stanów właściwych, $T \subset \mathbb{R}^+$ jest podzbiorem nieujemnych liczb rzeczywistych reprezentujących chwile czasowe,

$s_o = (x_o, t_o)$ - uogólniony stan początkowy, S_N, S_F - odpowiednio zbiór uogólnionych stanów niedopuszczalnych i końcowych, S_F - zbiór uogólnionych stanów końcowych.

Funkcja przejścia zdefiniowana jest za pomocą dwóch funkcji

$f = (f_x, f_t)$, gdzie $f_x : U \times X \times T \rightarrow X$ określa następny stan właściwy, zaś $f_t : U \times X \times T \rightarrow T$ następny moment czasu; funkcja f_t spełnia następujący warunek:

$$\Delta t = f_t(u,x,t) - t > 0 \text{ i ma wartość skończoną.} \quad (1)$$

Funkcję f zdefiniowano jako funkcję częściową, co pozwala uwzględnić wszystkie ograniczenia dotyczące decyzji sterujących za pomocą tzw. zbiorów sterowań możliwych w stanie s oznaczonych $U_p(s)$ i zdefiniowanych następująco: $U_p(s) = \{ u \in U : (u,s) \in \mathrm{Dom}\, f \}$.

W najogólniejszym przypadku zbiory U oraz X można przedstawić jako iloczyny kartezjańskie $U = U^1 \times U^2 \times .. U^m$ $X = X^1 \times X^2 \times .. X^n$, Funkcję f_x z kolei można przedstawić jako wektor $f_x = (f^o, f^1, f^2, .. f^m)$, gdzie $f^j : U \times X \times T \rightarrow X^j$ dla $j = 0,1,..m$.

Procesy, dla których przynajmniej jeden zbiór U^i lub X^j jest zbiorem nazw (w szczególności nazwy mogą być numerami) lub obiektów kombinatorycznych, będziemy nazywać procesami niezupełnie liczbowymi i dalej zajmować będziemy się tylko takimi procesami.

Zadanie optymalizacji sterowania polega na znalezieniu takiego ciągu sterowań $\tilde{u}$, który spełnia ograniczenia określone w definicji procesu P

(jest sterowaniem dopuszczalnym) i ekstremalizuje pewien funkcjonał Q.

Wspólne ograniczenia w zaprezentowanym modelu mają odmienną postać niż w modelach analitycznych. Wyrażają się w tym, że:

- poszczególne funkcje f^j $j = 1,2,\ldots m$ zależą od wspólnych współrzędnych sterowania i stanu,
- zbiór sterowań możliwych U_p nie da się przedstawić jako iloczyn kartezjański zbiorów $U_p^1 \times U_p^2 \times \ldots U_p^m$ zależnych tylko od rozłącznych podzbiorów współrzędnych stanu,
- ograniczenia dotyczące zbiorów stanów końcowych lub niedopuszczalnych zdefiniowane są za pomocą wspólnych zmiennych.

Rodzaj i stopień tych zależności obok własności kryterium będą decydowały o możliwości zastosowania dekompozycji i ostatecznej postaci algorytmu.

3. Charakterystyka algorytmów wykorzystujących dekompozycję

Najogólniej rozumiana metoda dekompozycji polega na rozwiązywaniu problemu (globalnego) za pomocą szeregu specjalnie skonstruowanych problemów, które nazywane są zazwyczaj lokalnymi (lub podproblemami), a tutaj ze względu na pewną odmienną specyfikę ich wykorzystywania nazywane będą problemami częściowymi.

Można wyróżnić dwie główne przyczyny, dla których stosuje się dekompozycję problemu:

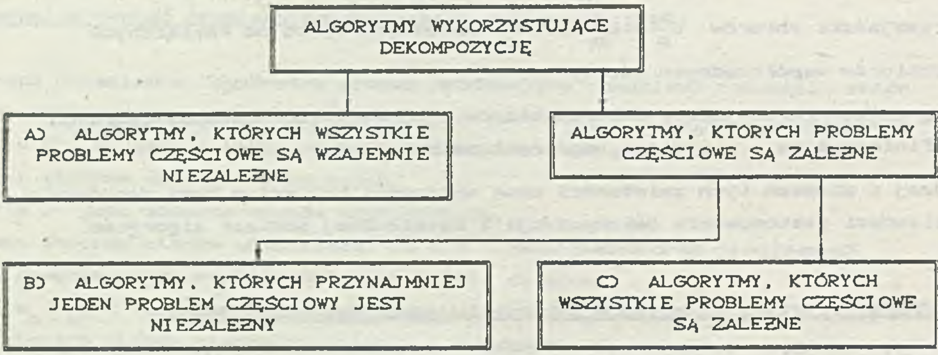
- oszczędność obliczeniową - możliwość rozwiązania problemu o znacznym rozmiarze drogą rozwiązywania problemów częściowych o znacznie mniejszych rozmiarach (np. dekompozycja programowania liniowego),
- względy algorytmiczne - np. konieczność stosowania odmiennych algorytmów dla kolejnego obliczania pewnych wielkości.

Wymienione przyczyny determinują wyróżnianie problemów częściowych. W optymalizacji sterowania DPP dominuje przypadek drugi. W celu porównywania i analizy koncepcji algorytmów wyróżnimy następujące cechy:

1. rodzaj powiązań między wyróżnionymi problemami częściowymi, determinujący ich kolejność rozwiązywania i określony przez:
   - hierarchię zależności pomiędzy wyróżnionymi problemami częściowymi,
   - cel, dla którego rozwiązuje się problemy częściowe.
2. własności algorytmów rozwiązywania problemów częściowych wynikające z:
   - typów problemów częściowych (statyczne lub dynamiczne, liczbowe lub nie),
   - złożoności obliczeniowej problemów częściowych.

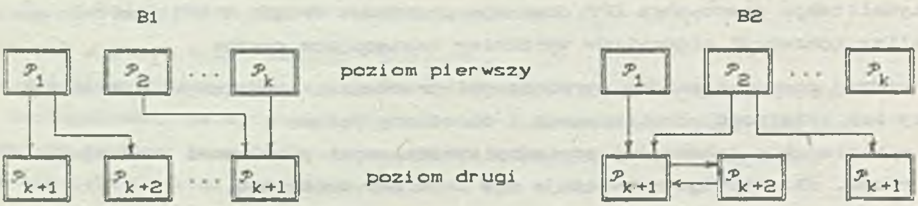
3.1. Rodzaj powiązań

Powiemy, że dany problem częściowy zależy od drugiego problemu, jeśli dla rozwiązania go wykorzystywane są informacje uzyskane w wyniku rozwiązania tego drugiego problemu. Podział algorytmów dokonany ze względu na wzajemną zależność problemów częściowych przedstawiają rys 1,2 i 3.

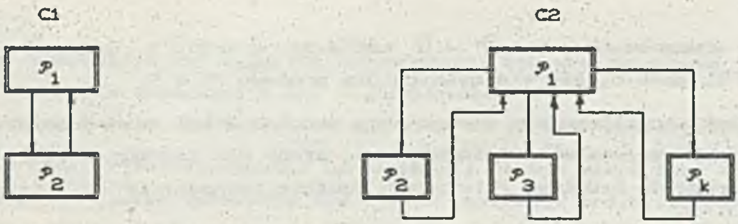


Rys.1. Podział algorytmów ze względu na wzajemną zależność problemów częściowych. Repartition of algorithms due to the mutual dependences of partial problems.

W grupie B możemy wyróżnić dwie dalsze grupy B1 i B2, dla których zależności pomiędzy problemami częściowymi przedstawia rysunek 2. Literą $\mathcal{P}$ oznaczony jest problem częściowy. Strzałka wskazuje kierunek przekazywania informacji. W grupie B1 informacje przekazywane są tylko od problemów wyższego poziomu do niższego. Problemy w takim układzie będziemy nazywać jednostronnie zależnymi. W grupie B2 możliwa jest wymiana informacji między pewnymi problemami tego samego poziomu. Na rysunku zaznaczono tylko dwa poziomy, jakkolwiek poziomów (o analogicznym sposobie powiązań) może być więcej.



Rys. 2. Schematy zależności problemów częściowych dla grupy algorytmów B). Schema of the partial problems of the type B) algorithms



Rys 3. Schemat najprostszych powiązań problemów częściowych w algorytmach typu C . Schema of the simplest dependences of the type C partial problems.

Jeśli rozwiązanie problemu $\mathcal{P}$ można uzyskać w wyniku rozwiązywania problemów częściowych $\mathcal{P}_1, \mathcal{P}_2, \ldots \mathcal{P}_k$, które są wzajemnie niezależne lub jednostronnie zależne, to powiemy, że problem $\mathcal{P}$ jest całkowicie dekomponowalny na problemy częściowe $\mathcal{P}_1, \mathcal{P}_2, \ldots \mathcal{P}_k$. W pozostałych przypadkach powiemy, że problem nie jest całkowicie dekomponowalny (B2, C). Dla tych problemów mówimy o koordynacji problemów częściowych, zaś algorytm jest iteracyjny.

W przypadku problemów całkowicie dekomponowalnych (A, B1) należy wyróżnić dwa przypadki:

- ilość problemów częściowych jest taka sama dla wszystkich zadań problemu globalnego i rozwiązywane są jednokrotnie,
- ilość problemów częściowych zależy od danych i rozmiaru zadania; przykładem jest dekompozycja wykorzystywana w programowaniu dynamicznym, gdzie rozwiązywanych jest szereg problemów częściowych odpowiadających optymalizacji lokalnej i otrzymane wyniki wykorzystywane są dla rekurencyjnego wyznaczania rozwiązania globalnego. Ze względu jednak na własną specyfikę tej metody programowanie dynamiczne nie będzie tutaj omawiane.

W dalszym ciągu, mówiąc o problemach całkowicie dekomponowalnych, będziemy mieli na myśli tylko problemy pierwszego typu.

Analogicznie jak dla modeli liczbowych [6], możemy teraz podać warunki na to, aby problem był całkowicie dekomponowalny na problemy częściowe wzajemnie niezależne. Rozpoczniemy od określenia zadania związanego.

Zadanie pierwotne (P,Q): znaleźć taki ciąg sterujący $\tilde{u}^*$, który spełnia ograniczenia procesu P i minimalizuje Q.

Jeśli zadanie (P',Q') ma postać: znaleźć takie $\tilde{u}'^*$, które spełnia ograniczenia procesu P' i minimalizuje Q' oraz spełnione są poniższe warunki 1) i 2), to zadanie (P',Q') nazywamy zadaniem związanym z (P,Q).

Wówczas możliwe jest użycie rozwiązania $\tilde{u}'^*$ do wyznaczenia $\tilde{u}^*$.

1) Rozwiązanie $\tilde{u}'^*$ istnieje wtedy i tylko wtedy, gdy istnieje rozwiązanie zadania (P,Q).

2) Istnieje odwzorowanie $\phi : \tilde{U}' \to \tilde{U}$ takie, że $\tilde{u} = \phi(\tilde{u}')$, gdzie $\tilde{U}'$, $\tilde{U}$ są odpowiednio zbiorem ciągów sterujących dla procesu P' i P.

Warunkiem wystarczającym, aby stacjonarny problem $\mathcal{P}$ był dekomponowalny na dwa problemy wzajemnie niezależne, jest, ażeby dla każdego zadania (P,Q) należącego do problemu $\mathcal{P}$ istniało zadanie związane (P',Q') takie, że:

- nowy wektor decyzji u' oraz stanów x' można podzielić na dwa rozłączne podzbiory współrzędnych $u' = (u'^{I}, u'^{II})$, $x' = (x'^{I}, x'^{II})$ takie, że zbiór $U'_p = U'^{I}_p \times U'^{II}_p$ oraz definicja zbioru U'^{I}_p zależy tylko od współrzędnych x'^{I}, zaś zbioru U'^{II}_p tylko od x'^{II}
- funkcja $f'_x = (f'^{I}, f'^{II})$, gdzie $f'^{I} : U'^{I} \times X'^{I} \to X'^{I}$, zależy tylko od pierwszego wyróżnionego podzbioru współrzędnych, zaś funkcja f'^{II} : $U'^{II} \times X'^{II} \to X'^{II}$ tylko od pozostałych współrzędnych,
- formuły definiujące zbiory stanów końcowych i niedopuszczalnych mogły być przedstawione odpowiednio jako koniunkcje i alternatywy formuł zależnych tylko od wyróżnionych podzbiorów współrzędnych,
- kryterium mogło być przedstawione jako suma $Q' = Q^{I}(\tilde{u}'^{I}) + Q^{II}(\tilde{u}'^{II})$ lub ogólniej jako pewna funkcja niemalejąca członów Q^{I} i Q^{II}.

Informacje otrzymywane w wyniku rozwiązywania problemów częściowych podzielmy na dwa typy:
- wartości pewnego podzbioru wielkości szukanych w problemie globalnym,
- inne informacje, wykorzystywane do rozwiązywania dalszych problemów częściowych.

W przypadku modeli liczbowych dekompozycja rozumiana jest właściwie jako podział mający na celu otrzymanie informacji pierwszego typu. Jednakże wydaje się, że, zwłaszcza w kombinatoryce, pojęcie to należy rozumieć szerzej.

3.2. Aspekty obliczeniowe

Przedstawimy teraz dyskusję dotyczącą złożoności obliczeniowej problemu globalnego i algorytmu optymalizacji; mówiąc o wielomianowym problemie częściowym, będziemy mieli na myśli wielomianową zależność od odpowiednich danych problemu globalnego.

Fakt 1.

Jeśli problem globalny jest całkowicie dekomponowalny na wielomianowe problemy częściowe $P_1, P_2 \ldots P_k$, to problem globalny jest wielomianowy.
Prawdziwość twierdzenia wynika z faktu, że każdy problem częściowy jest rozwiązywalny jednokrotnie.
Zauważmy, że nie jest to warunek konieczny dla wielomianowości problemu globalnego.

Fakt 2.

Warunkiem koniecznym, aby algorytm dekompozycji był wielomianowym, jest, aby każdy z problemów częściowych był wielomianowym.
Dowód jest oczywisty; zauważmy, że nie jest to warunek wystarczający.

Możemy teraz scharakteryzować uproszczenia (a tym samym typy algorytmów heurystycznych), jakie wprowadza się w celu uzyskania przybliżonych algorytmów efektywnych.

1. Dla problemów całkowicie dekomponowalnych można stosować przybliżone algorytmy dla jednego lub więcej niewielomianowych problemów częściowych.

2. Dla problemów niecałkowicie dekomponowalnych można:
a) wprowadzać arbitralne zmiany w modelu (ustalenia) mające na celu otrzymanie zmodyfikowanego problemu całkowicie dekomponowalnego lub o słabszych zależnościach między problemami częściowymi,
b) ograniczać ilość iteracji (w przypadku algorytmu zbieżnego); ograniczenie może być uzależnione od wyników obliczeń lub narzucone arbitralnie,
c) stosować algorytmy przybliżone dla niewielomianowych problemów częściowych i ewentualne ograniczenie ilości iteracji.

4. Przykłady

Dla ilustracji podanych pojęć przedstawimy analizę trzech algorytmów.

Przykład 1.

Jako pierwszy przeanalizujemy wielomianowy problem szeregowania niezależnych zadań na jednakowych maszynach równoległych, przy kryterium średniego czasu przepływu (przy jednakowych czasach dostępności). Zauważmy, że arbitralne rozdzielenie zbioru zadań na rozłączne podzbiory $Z^1, Z^2 \dots Z^m$ (m - ilość maszyn) i przydzielenie podzbiorów do poszczególnych maszyn zamienia problem na m niezależnych problemów częściowych, z których każdy odpowiada szeregowaniu zadań na jednej maszynie. Z braku miejsca nie przytoczymy pełnego przekształcenia modelu (które jest zresztą oczywiste), zauważając jedynie, że w wyniku arbitralnego podziału nowy zbiór sterowań $U' = U'^1 \times U'^2 \dots \times U'^m$, gdzie $U'^1 \cap U'^2 \cap \dots U'^m = \emptyset$ $(U'^j = Z^j)$, zaś brak wspólnych ograniczeń pozwala zmodyfikować definicję stanu oraz funkcji przejścia tak, że $x'^j_{i+1} = f'^j(u'^j_i, x'^j_i)$ $j = 1, 2, \dots m$, gdzie x'^j_i oznacza zbiór zadań wykonanych na j-tej maszynie do stanu i-tego.
$f' = (f'^1, f'^2, \dots f'^m)$, wielkości primowane oznaczają odpowiednie zmienne zmodyfikowane.

Kryterium przyjmuje postać $Q = Q^1(\tilde{u}'^1) + Q^2(\tilde{u}'^2) + .. Q^m(\tilde{u}'^m)$, gdzie $\tilde{u}'^j$ jest ciągiem sterującym dla j-tego autonomicznego problemu częściowego, odpowiadającym ciągowi zadań (permutacji) wykonywanych na tej maszynie. Znany algorytm SPT [1] wynika z rozwiązania problemu rozdziału zadań pomiędzy maszyny oraz m identycznych problemów częściowych odpowiadających szeregowaniu zadań na jednej maszynie.

Przykład 2.

Jako przykład problemu całkowicie dekomponowalnego, lecz nie na problemy częściowe wzajemnie niezależne, rozważymy problem, którego uogólnienie jest dyskutowane w [4], a które jest niemal identyczne z zagadnieniem chińskiego listonosza.
Dany jest spójny unigraf $H = (V,J)$ reprezentujący plan chodników, które mają być wydrążone. Zbiór wierzchołków V reprezentuje skrzyżowania, zaś zbiór $J \subset V \times V$ reprezentuje połączenia (chodniki), które mają być wykonane. Jeśli drążenie chodnika ma się rozpocząć od skrzyżowania innego niż to, w którym się skończyło drążenie poprzedniego, transport maszyny pomiędzy tymi skrzyżowaniami musi być wzięty pod uwagę; oczywiście transport ten może odbywać się tylko chodnikami uprzednio już wydrążonymi. Dana jest funkcja $d : J \to \mathbb{R}^+$ określająca długość chodników oraz wyróżnione dwa wierzchołki: początkowy v_o, z którego można rozpocząć pracę oraz końcowy v_f, w którym ją należy zakończyć. Zakłada się, że zarówno czasy drążenia, jak i transportu są proporcjonalne do długości chodników. Należy określić kolejność drążenia chodników i ewentualne drogi transportowe tak, aby czas wykonania całej sieci był minimalny (kryterium C_{max}).
Główna różnica w stosunku do problemu chińskiego listonosza polega na tym, że w rozważanym zadaniu należy nie tylko podać kolejność przechodzenia przez krawędzie grafu, lecz też ciąg krawędzi stanowiący rozwiązanie rozdzielić na krawędzie przechodzone po raz pierwszy (drążone) oraz podciągi stanowiące drogi transportowe.
Wielomianowy algorytm, przedstawiony w terminach teorii grafów, został podany w [5]. Poniżej przedstawimy uzasadnienie algorytmu oparte na modelu przestrzeni stanów i wykorzystujące dekompozycję problemu.

Zdefiniujmy model przestrzeni stanów dla przedstawionego procesu.
Zbiór stanów właściwych X zdefiniowany jest jako iloczyn kartezjański $X = X^o \times X^1$, $x = (x^o, x^1)$, gdzie $X^o \subset 2^J$, x^o jest zbiorem krawędzi odpowiadających chodnikom już wykonanym w stanie x, zaś $X^1 \subset V$, x^1 jest numerem węzła (skrzyżowania), w którym znajduje się maszyna.
Stan początkowy $s_o = (x_o, t_o)$, gdzie $x_o = (\emptyset, v_o)$.
Zbiór uogólnionych stanów końcowych $S_F = X_F \times \mathbb{R}^+$ ($\mathbb{R}^+$ - zbiór liczb

rzeczywistych dodatnich), gdzie $X_F = \{(x^o, x^1): x^o = J,\ x^1 = v_f\}$.

Zbiór stanów niedopuszczalnych jest pusty $S_N = \emptyset$.

Zbiór decyzji $U = U^1 \times U^2$, $u = (u^1, u^2)$, gdzie $U^1 = J$ - współrzędna u^1 określa numer krawędzi reprezentującej chodnik, którego drążenie ma się rozpocząć, $U^2 = L^o$, gdzie L^o jest zbiorem wszystkich łańcuchów prostych w grafie H powiększonym o łańcuch pusty oznaczany symbolem λ; współrzędna u^2 określa drogę transportową (lub jej brak).

Z braku miejsca pominiemy dokładny model, podsumowując tylko jego istotne własności:

- zbiór decyzji możliwych U_p zależy tylko od stanu właściwego i można go przedstawić jako $U_p(x) = U_p^1(x) \times U_p^2(x, u_1)$, przy czym zbiór U_p^1 zależy tylko od x, zaś definicja U_p^2 zależy od decyzji u^1 - nie są to więc zbiory niezależne.
- modyfikacja współrzędnej x^o zależy tylko od wartości pierwszej współrzędnej sterowania u^1. $x_{i+1}^o = x_i^o \cup \{u^1\}$, (1)

$- \Delta t_i = f_t(u_i, x_i, t_i) - t_i = \Delta t_i = c_2 * dł(u_i^2) + c_1 * d(u_i^1)$, (2)

gdzie c_1, c_2 są stałymi współczynnikami proporcjonalności, zaś funkcja $dł : L^o \to \mathbb{R}^+$ określa długość drogi; w przypadku drogi pustej $dł(\lambda) = 0$.

Zauważmy, że założenie odnośnie spójności grafu H oraz brak ograniczeń odnośnie stanów implikuje, że $U_p(x) = \emptyset \Leftrightarrow x \in X_F$; rozwiązanie dopuszczalne zostanie zatem zawsze znalezione za pomocą algorytmu wielomianowego.

Kryterium optymalizacji C_{max} należy do klasy regularnych. Z zależności (1) oraz definicji zbioru S_F wynika, że długość każdej trajektorii $d(\tilde{s})$ jest równa m, gdzie $m = |J|$. Zatem wartość kryterium można wyrazić $C_{max} = \sum_{i=1}^{m} \Delta t_i$. (3)

Z (3),(2) wynika, że $\min C_{max} = \min \sum_{i=1}^{m} \Delta t_i = \min c_2 \sum_{i=1}^{m} dł(u_i^2) + const$, (4)

gdzie $const = c_1 \sum_{j \in J} d(j)$.

Zauważmy, że kryterium zależy tylko od wartości współrzędnych u^2 oraz jest funkcją addytywną. Zadanie optymalizacji można zatem zdekomponować następująco:

1. Szukanie zbioru optymalnych wartości dla współrzędnych u^2.
2. Szukanie ciągu $\tilde{u}^1$ takiego, że sterowanie $\tilde{u} = (\tilde{u}^1, \tilde{u}^2)$ wyznacza

trajektorię dopuszczalną (kończącą się w zbiorze S_F) oraz odpowiednie współrzędne u^2 przyjmują wartości wyznaczone w etapie pierwszym. Jeśli taki ciąg istnieje, to-jak wynika z (4)-sterowanie $\tilde{u}$ jest na pewno optymalne. Dowód istnienia tego ciągu oraz algorytm (wielomianowy) jego generowania musimy z braku miejsca pominąć.

Zauważmy, że zdefiniowana powyżej zmienna decyzyjna u^2 nie jest zmienną indywiduową, lecz jej wartością jest pewien ciąg skończony. Stąd etap 1 realizowany jest za pomocą dwóch różnych algorytmów optymalizacji:

a) wyznaczenie minimalnych łańcuchów pomiędzy wszystkimi węzłami nie spełniającymi warunku Eulera ,

b) wyznaczenie minimalnego skojarzenia par.

Przykład 3.

Jako przykład problemu nie całkowicie dekomponowalnego, lecz rozwiązywanego metodą dekompozycji może służyć przepływowy problem szeregowania zadań na dwóch maszynach przy możliwości ustalania w pewnym zakresie czasów przetwarzania i uwzględnieniu w kryterium dodatkowego kosztu związanego ze skróceniem tych czasów. Algorytm optymalizacji jest opublikowany w [9]. Algorytm dokładny nie jest wielomianowy, chociaż algorytmy problemów częściowych są wielomianowe. Ponieważ jednak jest zbieżny, to przy ograniczonej ilości iteracji otrzymuje się rozwiązanie przybliżone.

LITERATURA

[1] E.G.Coffman,jr (red) i in.: Teoria szeregowania zadań. WNT, Warszawa 1980.

[2] E.Dudek-Dyduch: Scheduling some class of discrete processes. Proc.of 12th IMACS World Congress, Paryż 1988.

[3] E.Dudek-Dyduch: Formal bases of Classification of Discrete Production Processes Control Problems. Zeszyty Naukowe AGH, Automatyka z.49, 1989, 8 Sympozjum Teorii Systemów.

[4] E.Dudek-Dyduch, K.Wala : Programowanie robót przygotowawczych w kopalni jako uogólnione zagadnienie chińskiego listonosza.Zeszyty Nauk. Polit. Śląskiej, s.Automatyka z.100, Gliwice 1990.

[5] J.Edmonds, L.Johnson:Ellis Matching, Euler tours and the Chinese postman. Math. Prog. vol.5, pp.88-124, 1973.

[6] W.Findeisen: Wielopoziomowe układy sterowania. PWN, Warszawa 1974.

[7] H.Górecki: Synteza struktur hierarchicznych wielkich systemów. Skrypty AGH Nr 662, Kraków 1979.

[8] R.Kulikowski: Sterowanie w wielkich systemach. WNT, Warszawa 1974.

[9] E.Nowicki, S.Zdrzałka: A two-machine flow shop scheduling problem with controllable job processing times. Europ.Journ.of Oper.Research, vol.34 no 2, pp.208-220, March 1988.

Recenzent: Prof.dr inż.H.Kowalowski

Wpłynęło do Redakcji do 1990-04-30.

ON DECOMPOSITION IN CONTROL OPTIMIZATION OF DISRETE MANUFACTURING PROCESSES

S u m m a r y

The paper deals with methodology of control optimization of discrete manufacturing processes. It concerns the decomposition method. Repartition of algorithms and computational complexity has been discussed. Some examples has been given.

ДЕКОМПОЗИЦИЯ В ОПТИМИЗАЦИИ УПРАВЛЕНИЯ ДИСКРЕТНБІМИ ПРОМБІШЛЕННЕІМИ ПРОЦЕССАМИ

Р е з ю м е

Статья связана с методологией оптимиэации управления дискретными промышленными процессами и вопросом применения метода декомпозуции. В работе дана характеристика алгоритмоз, использующих декомпозуцию, представлена их вычислительная сложность, дана классификация эвристических алгоритмов. Введенные понятия иллустрирыются примерами.