ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1996
Seria: AUTOMATYKA z. 119 Nr kol.1339

Antoni KORCYL, Tadeusz SAWIK
Akademia Górniczo-Hutnicza, Kraków

WYBÓR SEKWENCJI MONTAŻOWYCH I OPTYMALIZACJA ROZDZIAŁU ZADAŃ W ELASTYCZNYM GNIEŹDZIE MONTAŻOWYM

Streszczenie. W pracy przedstawiono algorytm heurystyczny dla optymalizacji wyboru sekwencji montażowych i pracy elastycznego systemu montażowego. Algorytm oparty na heurystyce Tabu Search pozwala na jednoczesną optymalizację pracy elastycznego gniazda montażowego wraz z wyborem sekwencji montażowych montowanych w systemie wyrobów. W pracy zamieszczono przykład liczbowy oraz omówiono wyniki eksperymentów obliczeniowych, potwierdzające korzystne własności zaproponowanego algorytmu.

SELECTION OF ASSEMBLY SEQUENCE AND TASK ASSIGNMENT IN A FLEXIBLE ASSEMBLY CELL

Summary. The paper presents a new heuristic algorithm for optimisation of assembly sequences and station workloads in a flexible assembly cell. The algorithm based on metaheuristic Tabu Search allow to simultaneously optimisation of station workloads and assembly sequences selection. A simple numerical example is included to illustrate application of the algorithm proposed in the paper.

1. Wstęp

Elastyczne gniazdo montażowe [5] jest systemem produkcyjnym składającym się ze zautomatyzowanych stanowisk montażowych i urządzeń transportowych, przeznaczonych do jednoczesnego montażu w krótkich seriach różnych typów wyrobów z różnych części składowych. Optymalne wykorzystanie zasobów gniazda polega na zrównoważeniu obciążeń stacji montażowych oraz minimalizacji przemieszczeń wyrobów pomiędzy stacjami. Optymalizacja pracy gniazda wymaga wcześniejszego ustalenia sekwencji montażowych. Z zadaniem opisanym powyżej ściśle jest związane zadanie wyboru sekwencji montażowej ze zbioru wielu sekwencji montażowych. Wybór sekwencji montażowej powinien prowadzić do jednoczesnej optymalizacji wykorzystania zasobów gniazda. Zaproponowany algorytm umożliwia dokonanie wyboru optymalnej sekwencji, dla której osiągniemy zrównoważenie obciążeń stacji montażowych wraz z minimalizacją przemieszczeń wyrobów pomiędzy stacjami.

2. Algorytm heurystyczny dla wyboru optymalnej sekwencji montażowej

W zaproponowanym algorytmie ze zbioru sekwencji montażowych, dla każdego montowanego wyrobu, wybierana jest sekwencja optymalna za pomocą metody Tabu Search [2,3,4]. Kryterium optymalności wyboru sekwencji jest zrównoważenie obciążeń stacji montażowych wraz z minimalizacją przemieszczeń wyrobów między nimi.

Rozważmy elastyczne gniazdo montażowe składające się z M stacji montażowych $i = 1,2,..M$. Elastyczne gniazdo montażowe jest systemem montażowym, w którym montowane wyroby odwiedzają poszczególne stacje z możliwością powrotu.

W systemie z N różnych typów części montowanych jest K różnych typów wyrobów $k = 1,...,K$. Dla każdego wyrobu k zadany jest zbiór $J_K \subset \{1,...,N\}$ typów części składowych, oraz zbiór R_k par kolejno montowanych części (j,r) takich, że dla wyrobu k montaż części j bezpośrednio poprzedza montaż części r. Zbiory R_k tworzy się na podstawie zbioru sekwencji montażowych dla każdego montowanego wyrobu. Znane jest zapotrzebowanie d_k dla każdego typu wyrobu k. Znane są także czasy montażu p_{jk} poszczególnych części j ($j = 1,...,N$) w każdym wyrobie k ($k = 1,...,K$) oraz czasy transportu wyrobów q_{il} pomiędzy stacjami i,l ($i,l = 1,...,M$). Każda stacja montażowa i ($i = 1,...,M$) jest wyposażona w ograniczoną liczbę podajników części s_i. Model matematyczny zadania równoważenia łącznego czasu montażu i transportu w elastycznym gnieździe montażowym ma postać (por. [6]):

Zminimalizować

$$Q_{max} \tag{1}$$

przy ograniczeniach

$$\sum_{k=1}^{K}\sum_{j\in J_k} d_k p_{jk} u_{ij} + \sum_{k=1}^{K}\sum_{l\neq i}\sum_{j\in J_k} d_k q_{li} v_{lijk} \le Q_{max}\,; \quad \forall i \tag{2}$$

$$\sum u_{ij} = 1\,; \quad \forall j \tag{3}$$

$$\sum u_{ij} \le s_i\,; \quad \forall i \tag{4}$$

$$u_{ij} + u_{lr} - v_{iljk} \le 1\,; \quad \forall k, \forall i,l, l\neq i, \forall (j,r) \in R_k \tag{5}$$

$$-u_{ij} - u_{lr} + 2v_{iljk} \le 0\,; \quad \forall k, \forall i,l\, l\neq i, \forall (j,r) \in R_k \tag{6}$$

$$u_{ij} \in \{0,1\}\,; \quad \forall i,j \tag{7}$$

$$v_{iljk} \in \{0,1\}\,; \quad \forall i,l,k \tag{8}$$

gdzie:

$u_{ij} = \begin{cases} 1, & \text{jeżeli część typu } j \text{ jest przydzielona do podajnika przy stacji i;} \\ 0, & \text{inaczej} \end{cases}$

$v_{iljk} = \begin{cases} 1, & \text{jeżeli wyrób typu } k \text{ po wykonaniu operacji } j \text{ przechodzi ze stacji } i \text{ do stacji } l\,; \\ 0, & \text{inaczej} \end{cases}$

Funkcja celu Q_{max} reprezentuje sumę czasów montażu i transportu wyrobów wyznaczonych dla stacji będącej wąskim gardłem w gnieździe i określonej przez ograniczenie (2). Ograniczenie (3) zapewnia przydział części każdego typu do tylko jednej stacji, zaś (4) uwzględnia ograniczoną liczbę podajników części przy każdej stacji. Ograniczenia (5) i (6) zapewniają wybór marszrut montażu zgodnych z ograniczeniami kolejnościowymi i przydziałami części składowych do stacji montażowych.

Na podstawie powyższego modelu dokonywane są w algorytmie poszukiwania stacji "wąskiego gardła" oraz sprawdzana możliwość przydzielenia części składowych do poszczególnych stacji, przy których są wolne podajniki. W zaproponowanym algorytmie rozwiązaniem x jest macierz $\left[u_{ij}\right]$ przydziałów części składowych j do stacji i.

Ruchem g jest zmiana przydziału pewnej części j do innej stacji i w stosunku do rozwiązania początkowego x_p, któremu odpowiadała macierz $\left[u^p_{ij}\right]$. Otrzymane w wyniku wykonania ruchu g nowe rozwiązanie $\left[u'_{ij}\right]$ z sąsiedztwa $N\left(\left[u^p_{ij}\right]\right)$ generuje nową wartość funkcji celu $f\left(\left[u'_{ij}\right]\right) = Q_{max}$ oznaczoną przez *New_Solution*. Na *listę tabu LT* trafiają takie typy części, którym w trakcie kilku poprzednich iteracji zmieniono przydziały do stacji; posiadają one więc *status tabu*. *Status tabu* jest utrzymywany w ciągu określonej liczby iteracji, a liczba ta zależy od rozmiaru problemu, podobnie jak parametr *Max_Iteration* określający maksymalną liczbę iteracji poszukiwania rozwiązania.

Wybór części j, która zostanie przydzielona do innej stacji i w danej iteracji, polega na znalezieniu spośród wszystkich części składowych j takiej, która po zmianie przydziału pozwoli na otrzymanie najmniejszej wartości Q_{max} w danej iteracji. W każdej iteracji spośród wszystkich części składowych j nie mających *statusu tabu* (tzn. które można przydzielić do innej stacji) kolejno dokonywane są zamiany przydziału części składowych j do kolejnych stacji i, do których można je przydzielić (mają wolne podajniki). Na podstawie każdej kolejnej zmiany przydziału części składowej j wybierane są dla każdego typu montowanego wyrobu sekwencje montażowe. Sekwencje te wybierane są spośród zbioru sekwencji montażowych dla każdego wyrobu. Wybierana jest taka sekwencja, która zostanie zrealizowana w najkrótszym czasie (łączny czas montażu części składowych i transport wyrobu pomiędzy stacjami) dla danej macierzy przydziałów $\left[u_{ij}\right]$ części składowych do poszczególnych stacji. Na podstawie tak wybranej sekwencji dokonywane jest obliczanie wartości Q_{max}, sumy czasów montażu i transportu wyrobów wyznaczonych dla stacji będącej wąskim gardłem w gnieździe. Następnie spośród wszystkich możliwych przemieszczeń części składowych j do stacji i wybierane jest takie, dla którego wartość Q_{max} jest najmniejsza w danej iteracji.

W kolejnym kroku następuje aktualizacja wielkości związanych z *listą tabu* (*status tabu*, *liczba iteracji* , *Best_Solution* itd.). Procedura jest powtarzana dopóki nie ulegnie poprawie wartość funkcji celu *Best_Solution* lub zostanie osiągnięta maksymalna liczba iteracji *Max_Iteration*.

W trakcie wykonywania algorytmu, w przypadku gdy nie następuje poprawa wartości funkcji celu, liczba iteracji nie osiągnęła liczby maksymalnej, a osiągnęła określoną liczbę iteracji *FK_Iteration*, używa się *funkcji kryterium kwalifikacji*. Funkcja ta pozwala na zmianę *statusu tabu* części składowej i możliwość przydziału do innej stacji pod warunkiem polepszenia wartości funkcji celu *Best_Solution*.

W zaproponowanym algorytmie może być użyta *pamięć długoterminowa* w celu intensyfikacji i urozmaicenia poszukiwań rozwiązania optymalnego. *Funkcją pamięci długoterminowej* jest częstość przydzielania części składowych do poszczególnych stacji w trakcie wykonywania algorytmu.

Użycie tej funkcji polega na przydzieleniu części składowych do stacji, zgodnie z informacjami zawartymi w tym module, i powrocie do linii głównej algorytmu. Procedura ta jest uruchamiana w algorytmie, w momencie gdy nie istnieje możliwość poprawienia wartości funkcji celu *New_Solution* > *Best_Solution*, a liczba iteracji nie osiągnęła wartości maksymalnej *Max_Iteration*, a osiągnęła wartość *LT_Iteration*. Algorytm jest wykonywany dopóki nie zostanie osiągnięte rozwiązanie bliskie optymalnemu lub znalezione rozwiązanie nie ulega poprawie w ciągu liczby iteracji równej *Max_Iteration*.

Algorytm wyboru sekwencji montażowej

Krok 1 Przydziel kolejno części składowe *j* do kolejnych stacji montażowych *i*. Utwórz macierz $\left[u^{p}_{ij}\right]$.

Zdefiniuj: *Funkcję kryterium kwalifikacji - FK, pamięć długoterminową.*

Ustal: *Listę_Tabu, Status_Tabu, Max_Iteration, FK_Iteration, LT_Iteration.*

Krok 2 Oblicz wartości Q dla każdej stacji *i* oraz znajdź stację będącą "wąskim gardłem", dla której Q przyjmuje wartość maksymalną Q_{max}.

Podstaw: *New_Solution* = Q_{max}.

Krok 3 Znajdź część składową *j*, która nie posiada *Statusu Tabu* i przydzielaj ją do wszystkich pozostałych stacji, jeżeli są przy nich wolne podajniki. Dla każdej zmiany przydziału utwórz macierz $\left[u'_{ij}\right]$ i na jej podstawie znajdź dla wszystkich typów montowanych wyrobów sekwencje montażowe o najkrótszym czasie realizacji. Dla wybranych sekwencji oblicz wartości Q dla każdej stacji *i*. Znajdź stację, dla której Q przyjmuje wartość największą. Zapamiętaj tę wartość jako element wektora Q_{TEMP} oraz skojarzoną z nią macierz $\left[u'_{ij}\right]$.

Powtórz dla każdej części składowej *j* nie mającej *Statusu Tabu*.

Uszereguj elementy wektora niemalejąco. Weź pierwszy element.

Podstaw *New_Solution* = Q_{TEMP}

$\left[u_{ij}\right] = \left[u'_{ij}\right]$

Krok 4 Jeżeli *New _Solution* > *Best_Solution* i *liczba iteracji* = *FK_Iteration* i *liczba iteracji* < *Max_Iteration*, zastosuj *funkcję kryterium kwalifikacji - FK*.
Wróć do **Kroku 2**.

Krok 5 Uaktualnij wszystkie wielkości (*Status Tabu, liczba iteracji*).
Jeżeli wartość *New_Solution* < *Best_Solution* i *liczba iteracji* < *Max_Iteration*, podstaw *Best_Solution* = *New_Solution* .
Wróć do **Kroku 2**

Krok 6 Jeżeli *liczba iteracji* < *Max_Iteration* i *liczba iteracji* = *LT_Iteration* i *New_Solution* ≥ *Best_Solution*, użyj modułu pamięci długoterminowej i wróć do **Kroku 2**
W innym przypadku STOP.

3. Przykład liczbowy

Dla zilustrowania zaproponowanego algorytmu rozważmy przykład elastycznego gniazda montażowego składającego się z $M = 5$ stacji montażowych. Każda stacja dysponuje jednakową liczbą $s_i = 7$ podajników części. W systemie z $N = 20$ typów części są montowane $K = 4$ typy wyrobów. Wykonanie wyrobu typu k wymaga montażu części różnego typu $j \in J_k$. W tablicy 1 przedstawiono zbiory części wymagane do montażu poszczególnych wyrobów.

Należy wyprodukować jednakowe liczby wyrobów wynoszące $d_1 = d_2 = d_3 = d_4 = 20$ sztuk. Czasy p_{jk} montażu części dla poszczególnych typów wyrobów są podane w tablicy 2. Czas transportu wyrobu pomiędzy sąsiednimi stacjami wynosi 2 jednostki czasu, a stacje rozmieszczone są tak, że wartości q_{il} czasów przemieszczeń pomiędzy stacjami wynoszą : $q_{12} = q_{15} = q_{23} = q_{34} = q_{45} = 2$, oraz $q_{13} = q_{24} = q_{35} = q_{14} = 4$.

Tablica 1

Części składowe oraz sekwencje montażowe dla różnych typów wyrobów

Wyrób k	Zbiory J_k części składowych	Sekwencje montażowe
1	1,2,3,4,5,6,7,8	7,6,2,3,1,4,5,8; 5,1,4,3,2,6,7,8;
		2,3,6,7,1,4,5,8;
2	2,4,5,7,8,10,12,14,16,18,20	4,5,8,14,12,16,18,7,10,2,20,
		10,12,14,2,7,4,5,8,16,18,20
		2,7,10,4,5,8,14,12,16,18,20
3	1,3,4,5,6,9,11,13,14,15,16,18,19	1,4,5,19,11,15,16,17,18,3,6,9
		3,6,1,9,4,5,11,13,15,16,17,18,19
		11,13,14,15,3,6,1,4,5,9,16,18,19
4	2,4,5,6,7,8,10,11,14,15,16,17,18,20	4,5,8,20,11,16,17,18,14,15,2,6,7,10
		10,11,14,15,1,2,6,7,4,5,8,16,18,20,17
		2,6,7,10,11,16,17,18,4,5,8,14,15,20

W tablicy 3 podano przydziały części do stacji (zmienne u_{ij}) wyznaczone poprzez rozwiązanie zadania programowania całkowitoliczbowego przy użyciu pakietu LINGO [7](u_{ij}^{*}) oraz przy zastosowaniu algorytmu heurystycznego u_{ij}^{H}.

Tablica 2

Czasy P_{jk} montażu części składowych dla różnych typów wyrobów

Wyrób k	Typ części j																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	5	2	5	2	2	5	2	0	0	0	0	0	0	0	0	0	0	0	0
2	0	4	0	4	2	0	2	3	0	3	0	5	0	3	0	3	0	3	0	5
3	4	0	1	2	2	2	0	0	2	0	1	0	2	5	4	2	0	4	4	0
4	0	3	0	2	2	2	4	2	0	1	3	0	0	4	2	3	4	2	0	3

Tablica 3

Przydziały typów części do stacji u_{ij}^{*} / u_{ij}^{H}

Stacja i	Typ części j																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1		1/0								1/1	1/1	1/1	1/1	1/1	0/1					
2		0/1	0/1			0/1	0/1								1/0					
3	1/1		1/0	0/1	0/1	1/0	1/0													
4				1/0	1/0			1/1	1/1							0/1		0/1		
5																1/0	1/1	1/0	1/1	1/1

Otrzymano następujące wyniki: łączny czas montażu i transportu dla stacji $i = 2$, będącej wąskim gardłem, uzyskany poprzez rozwiązanie zadania programowania całkowitoliczbowego przy użuciu pakietu LINGO wynosi $Q_{max} = 720$ jednostek. Dla pozostałych stacji wynosi: $Q_{max} = 700$ dla $i = 1$, $Q_{max} = 700$ dla $i = 3$, $Q_{max} = 720$ dla $i = 4$ oraz $Q_{max} = 680$ dla $i = 5$. Czas obliczeń CPU = 3godz.12min.

Zastosowanie heurystyki doprowadziło do uzyskania łącznego czasu obciążenia oraz czasu transportu montowanych wyrobów dla stacji $i = 3$, będącej wąskim gardłem $Q_{max} = 800$ jednostek. Dla pozostałych stacji wynosi: $Q_{max} = 780$ dla $i = 1$, $Q_{max} = 440$ dla $i = 2$, $Q_{max} = 800$ dla $i = 4$ oraz $Q_{max} = 780$ dla $i = 5$.

Otrzymany rezultat jest gorszy od osiągniętego w wyniku zastosowania pakietu LINGO o ok.11 %. Czas obliczeń CPU = 4min.13sek., czyli ok. 45 - krotnie krótszy niż w przypadku pakietu LINGO.

W obu przypadkach zostały wybrane te same sekwencje montażowe dla każdego z wyrobów montowanych w gnieździe:

$J_1 = \{2,3,6,7,1,4,5,8;\}$,

J_2= {10,12,14,2,7,4,5,8,16,18,20},
J_3 = {11,13,14,15,3,6,1,4,5,9,16,18,19}
J_1 = {10,11,14,15,1,2,6,7,4,5,8,16,18,20,17}.
Obliczenia przeprowadzono na komputerze typu PC P100.

4. Uwagi końcowe

W celu sprawdzenia poprawności działania zaproponowanego algorytmu przeprowadzono szereg symulacji komputerowych. Dane generowane były losowo. Wartości funkcji celu były bliskie otrzymywanym dla modelu programowania całkowitoliczbowego. Odchylenie nie przekraczało 15%, przy istotnym skróceniu czasu obliczeń od 20 do 120 razy.

Zaproponowane podejście umożliwia wybór sekwencji montażowych z jednoczesnym zrównoważeniem obciążeń stacji w elastycznym gnieździe montażowym. Wyznaczone obciążenia stacji mogą stanowić bazę dla harmonogramowania pracy całego systemu. Dalsze badania będą prowadzone w kierunku rozszerzenia algorytmu o część dotyczącą harmonogramowania pracy gniazda i wprowadzenia sprzężenia zwrotnego pozwalającego na etapie ustalania harmonogramu na dokonywanie zmian obciążeń stacji i modyfikacji sekwencji montażowych.

LITERATURA

1. Ghosh S., Gagnon R.J.: A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. International Journal of Production Research, vol.27, 1989, pp.637-670.
2. Glover F.: Tabu Search, Part I. ORSA Journal of Computing, vol. 1, no 3, 1989, s.190-206.
3. Glover F.: Tabu Search, Part II. ORSA Journal of Computing, vol. 2, no 1, 1990, s.4-32.
4. Glover F.: Tabu Search: A Tutorial. Interfaces, vol. 20, no 4, 1990, s.74-90.
5. Sawik T.: Optymalizacja dyskretna w elastycznych systemach produkcyjnych. WNT, Warszawa 1992.
6. Sawik T.: Modele zadań równoważenia obciążeń maszyn w elastycznych systemach montażowych. Zeszyty Naukowe Politechniki Śląskiej,s. Automatyka z. 115, Gliwice 1994, ss. 113-124
7. Schrage L., Cunningham K.: LINGO, Optimization Modeling Language. LINDO Systems Inc., Chicago 1991.

Recenzent: Dr hab.inż. Jan Kałuski, prof. Pol. Śl.

Wpłynęło do Redakcji do 30.06.1996 r.

Abstract

The paper presents an integer programming formulation and a tabu search algorithm for balancing of a flexible assembly system. A flexible assembly system is made up of a set of assembly stations (usually robots) linked with an automated handling system. Each station has a limited working space, where limited number of part feeders can be placed. As a result each station is capable of performing a limited number of different assembly operations.
The objective of the FAS balancing problem is to assign parts and assembly operations to stations for a set of different assembly types so as to balance workloads of machines and material handling system. The above objective is associated with a problem of selection assembly sequences for all part types assembled in a flexible assembly cell.
A complex heuristic is presented for solving the balancing and assembly sequence selection problem. The heuristic is based on the Tabu Search metaheuristic modified appropriatelly.
Simulation studies have been performed to evaluate the performance of the algorithm and to compare the solution results with those obtained by using LINGO discrete optimizer. The results have indicated that the proposed algorithm yields good solutions in short CPU time.