NIETYPOWE WŁASNOŚCI PERMUTACYJNEGO PROBLEMU PRZEPŁYWOWEGO Z OGRANICZENIEM BEZ PRZESTOJÓW

Mariusz MAKUCHOWSKI

Streszczenie: W pracy rozważa się permutacyjny problem przepływowy z kryterium będącym momentem wykonania wszystkich zadań z dodatkowym ograniczeniem na to, aby żadna z maszyn nie miała przestojów w pracy. Praca przedstawia sformułowanie problemu, model permutacyjno-grafowy omawianego zagadnienia oraz pokazuje pewne bardzo ciekawe własności tego zagadnienia. Przedstawione własności poparte są odpowiednimi przykładami numerycznymi oraz ilustracjami odpowiadających im harmonogramów.

Słowa kluczowe: problem przepływowy, ograniczenie bez przestojów.

1. Wprowadzenie

Występujący w teorii szeregowania zadań, permutacyjny problem przepływowy jest od dawna badanym zagadnieniem [1,2,3]. Modeluje on szereg procesów produkcyjnych w szczególności linie produkcyjne w których montowane elementy przechodzą przez kolejne gniazda montażowe. W procesie tym można wybrać sekwencję wykonywania zadań, co skutkuje uzyskiwaniem różnych harmonogramów pracy maszyn. Problem polega na wybraniu harmonogramu minimalizującego wybrane kryterium optymalizacji, np. całkowity czas produkcji. Dla tego zagadnienia stworzono wiele dedykowanych algorytmów, zarówno bardzo szybkich algorytmów konstrukcyjnych [2], jak i bardzo wydajnych algorytmów popraw [3]. Duża efektywność algorytmów uzyskana jest głównie poprzez umiejętne wykorzystanie różnych własności rozwiązywanego problemu.

W niektórych procesach produkcyjnych może zaistnieć sytuacja wymuszająca ciągłą pracę maszyny [4]. Przyczyn takiego wymogu pracy może być kilka. Jedną z nich jest stosunkowo kosztowny rozruch i/lub wygaszanie maszyny, inny powód to chęć maksymalnej redukcji przedziału czasu wykorzystywania maszyny. Ograniczenie bez przestojów może dotyczyć wszystkich bądź tylko wybranych maszyn w systemie.

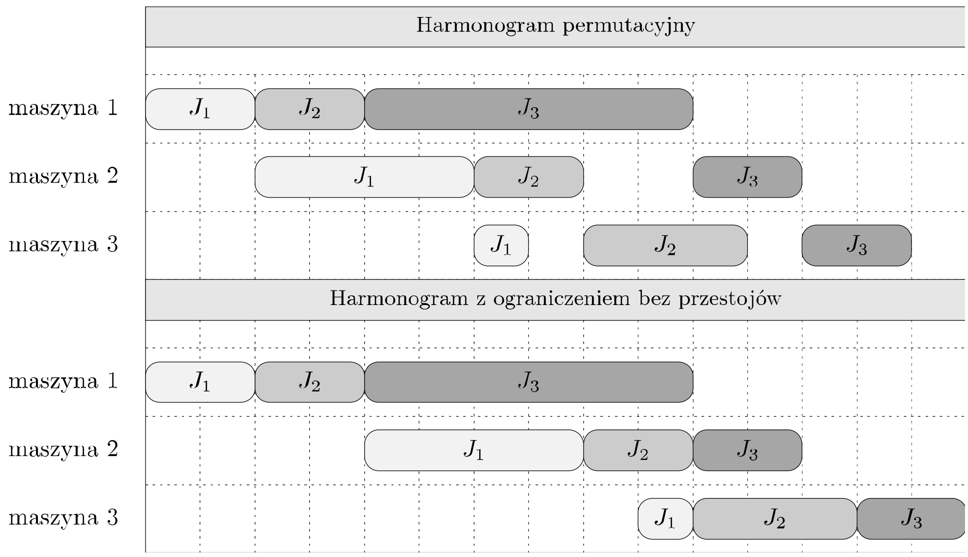
Ograniczenie bez przestojów, jak każde inne, zwiększa lub pozostawia bez zmian długość harmonogramów oraz powoduje pojawienie się nowych własności. W przypadku wymuszenia pracy maszyn bez przestojów, problem charakteryzuje się nietypowymi, wręcz paradoksalnymi własnościami. Przykładowo wymiana w systemie maszyny na wolniejszą może skutkować skróceniem pracy całej produkcji. Dodanie do istniejącego harmonogramu jeszcze dodatkowych zadań do wykonania może także skrócić czas całej produkcji. Ponadto w problemie tym nie istnieją typowe własności występujące w klasycznej odmianie problemu dające bazę do budowania szybkich akceleratorów przyśpieszających wyliczanie wartości długości uszeregowania.

2. Opis problemu

Permutacyjny problem przepływowy z kryterium będącym długością harmonogramu z dodatkowym ograniczeniem bez przestojów oznaczany jest w notacji Graham'a [5] jako $F|no-idle, perm|C_{max}$.

W permutacyjnym problemie przepływowym, poszukiwany jest harmonogram wykonywania operacji na maszynach. Wśród możliwych harmonogramów można wyróżnić harmonogramy aktywne, czyli takie w których wszystkie operacje dosunięte są możliwie maksymalnie do lewej strony na osi czasu. Ponieważ poszukiwane rozwiązanie optymalne zawsze można znaleźć wśród harmonogramów aktywnych w dalszej części pracy tylko takie harmonogramy będą brane pod uwagę. Każdy harmonogram aktywny można jednoznacznie przedstawić w postaci sekwencji wykonywania zadań w systemie. Permutacyjny problem przepływowy z kryterium będącym czasem wykonania wszystkich zadań, sprowadza się do wyznaczenia takiej sekwencji wykonywania zadań na maszynach, aby moment zakończenia wszystkich zadań był możliwie najkrótszy.

Ograniczeniem bez przestojów, w permutacyjnym problemie przepływowym odnosi się bezpośrednio do sposobu pracy maszyn. Wymusza ono ciągła pracę każdej z maszyn bez możliwości chwilowego jej wyłączenia. Dla każdej sekwencji wykonywania zadań, można zawsze utworzyć harmonogram spełniające powyższe ograniczenie. Ograniczenie te wymusza zazwyczaj konieczność opóźnienia rozpoczęcia niektórych operacji w celu uzyskania ciągłej pracy maszyn. Powoduje więc zazwyczaj wydłużenie harmonogramu odpowiadającego danej sekwencji wykonywania zadań. Przykład permutacyjnego harmonogramu przepływowego bez i z ograniczeniem bez przestojów przedstawiony został na rysunku 1.



Rys. 1. Przykład permutacyjnego harmonogramu przepływowego bez dodatkowych ograniczeń i z ograniczeniem bez przestojów

2.1. Sformułowanie problemu

Mamy dany zbiór zadań $J = \{1,2,\dots,n\}$ i zbiór maszyn $M = \{1,2,\dots,m\}$. Każde zadanie $j \in J$ musi być wykonane sekwencyjnie na każdej maszynie w kolejności wynikającej z numeru maszyny. Proces wykonywania zadania $j \in J$ na maszynie $l \in M$ nazywamy operacją i notujemy jako parę (j,l). Dla każdej operacji (j,l) dany jest czas $p_{j,l}$ jej wykonywania. Klasyczne ograniczenia wymagają aby (*i*) operacje wykonywały się bez przestojów, (*ii*) maszyna może wykonywać tylko jedną operację w danej chwili, (*iii*) w danym momencie można wykonywać tylko jedną operację danego zadania. Dodatkowe ograniczenie bez przestojów wymusza, ciągłą pracę maszyny, to znaczy po zakończeniu wykonywania się danej operacji na maszynie, maszyna od razu zaczyna wykonywać kolejną operację. Wyjątkiem są oczywiście operacje wykonywane jako ostatnie na maszynach. Akceptowalnym uszeregowaniem nazywamy momenty rozpoczęcia $S(j,l)$ i/lub momenty zakończenia $C(j,l)$ wykonywania się operacji (j,l); $C(i,l) = S(i,l) + p_{j,l}$. Problem polega znalezieniu uszeregowania dopuszczalnego minimalizującego:

$$C_{max} = max_{j\in J} C(j,m). \tag{1}$$

2.1. Model permutacyjno–grafowy

Ponieważ, podobnie jak w klasycznym permutacyjnym problemie przepływowym bez dodatkowych ograniczeń tak i problemie z ograniczeniem bez przestojów, dopuszczalny harmonogram może być jednoznaczne definiowany przez sekwencję wykonywania zadań. Z tego powodu w obu przypadkach problemu wygodnie jest stosować model permutacyjno-grafowy [6], w którym zmienną decyzyjną jest permutacja zbioru zadań $\pi = \big(\pi(1),\pi(2),\dots,\pi(n)\big)$. Zbiór wszystkich możliwych takich permutacji tworzy przestrzeń rozwiązań oznaczanych przez Π. Kryterium optymalizacji jest długość najdłuższej ścieżki w skierowanym grafie:

$$G(\pi) = (J \times M, E^T \cup E^K(\pi) \cup E^{NI}(\pi)). \tag{2}$$

Wierzchołek $(j,l), j \in J, l \in M$ reprezentuje operację, tak samo indeksowaną, i ma obciążeni $p_{j,l}$. Zbiór nieobciążonych łuków

$$E^T = \bigcup_{j\in J} \bigcup_{l=2}^{m} \{(j,l-1),(j,l)\} \tag{3}$$

modeluje ograniczenia technologiczne. Zbiór nieobciążonych łuków

$$E^K(\pi) = \bigcup_{i=2}^{n} \bigcup_{l\in M} \big\{\big((\pi(i-1),l),(\pi(i),l)\big)\big\} \tag{4}$$

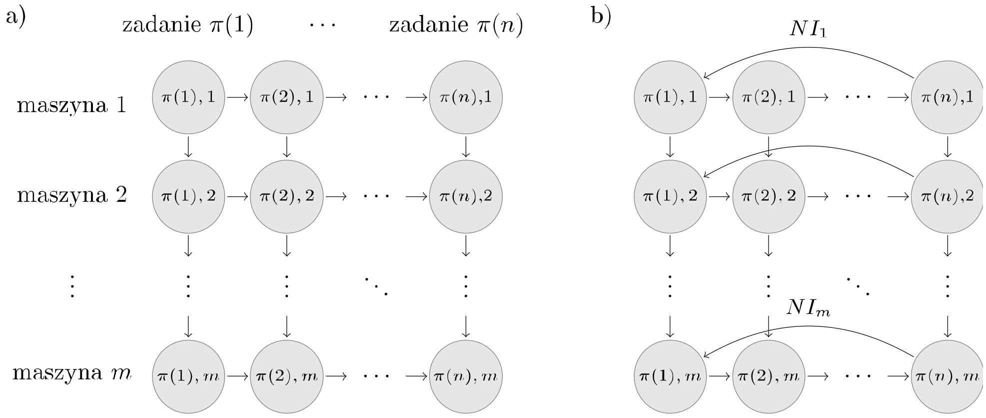
modeluje ograniczenia kolejnościowe wynikające z przyjętej kolejności wykonywania zadań. Dodatkowe ograniczenie bez przestojów modelowane jest poprzez obciążone łuki

$$E^{NI} = \bigcup_{l\in M} \big\{\big((\pi(n),l),(\pi(1),l)\big)\big\}. \tag{5}$$

Łuk łączący ostatnią i pierwszą operację wykonywaną na maszynie $l \in M$ obciążony jest sumą czasów trwania operacji wykonywanych na tej maszynie;

$$NI_l = -\sum_{j\in J} p_{j,l}. \tag{6}$$

Grafy modelujące dopuszczalne rozwiązania odpowiadające permutacji π w permutacyjnym problemie przepływowym bez dodatkowych ograniczeń oraz z ograniczeniem bez przestojów przestawione są na rysunku 2.



Rys. 2. Modele grafowe harmonogramów problemu permutacyjnego: a) bez dodatkowych ograniczeń, b) z dodatkowym ograniczeniem bez przestojów

Ponieważ długość najdłuższej ścieżki w grafie $G(\pi)$ oznaczonej przez $C_{max}(\pi)$ odpowiada dokładnie momentowi wykonania wszystkich zadań, badany problem sprowadza się do znalezienia sekwencji zadań minimalizującej długość najdłuższej ścieżki w grafie;

$$\pi^* \in \arg\min_{\pi\in\Pi} C_{max}(\pi). \tag{7}$$

3. Własności

Na podstawie własności problemu, pokazanych w niniejszej pracy, można wnioskować o możliwości wystąpienia pewnych nietypowych zjawisk. Możliwość wystąpienia takiego zjawiska będzie udowodniona za pomocą pokazania odpowiedniego przykładu numerycznego oraz zobrazowana odpowiednią ilustracją harmonogramu.

Własność 1.A.
Usuwając pewne zadania z uszeregowania, nowo powstały dopuszczalny harmonogram może być dłuższy niż harmonogram pierwotny.

Własność 1.B.
Usuwając pewne zadanie ze zbioru zdań do uszeregowania w instancji pierwotnej, powstaje nowa instancja problemu w której długość optymalnego uszeregowania jest dłuższa niż w przypadku instancji pierwotnej.

Własność 2.A.
Skracając czasy wykonywania wszystkich operacji pewnego zadania w danym uszeregowaniu, nowo powstały dopuszczalny harmonogram może być dłuższy niż harmonogram pierwotny.

Własność 2.B.
Istnieją takie instancje problemu dla których skrócenie wszystkich czasów wykonywania operacji pewnych zadań powoduje powstanie nowej instancji o optymalnym uszeregowaniu dłuższym niż w przypadku instancji pierwotnej.

Własność 3.A.
Skracając czasy wszystkich operacji wykonywanych na pewnych maszynach w danym uszeregowaniu, nowo powstały dopuszczalny harmonogram może być dłuższy niż harmonogram pierwotny.

Własność 3.B.
Istnieją takie instancje problemu dla których skrócenie wszystkich czasów wykonywania operacji wykonywanych na pewnych maszynach powoduje powstanie nowej instancji o optymalnym uszeregowaniu dłuższym niż w przypadku instancji pierwotnej.

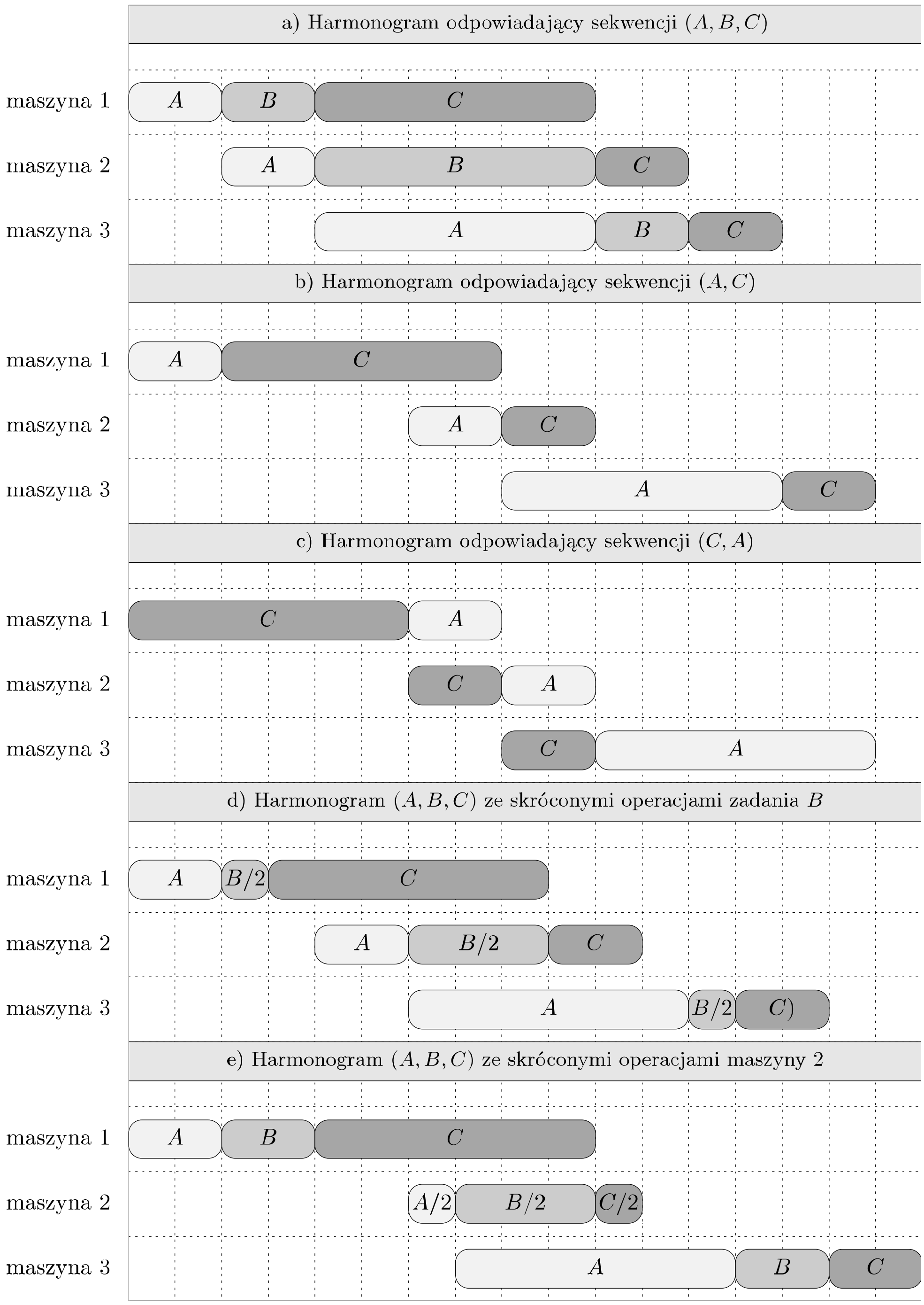
Własność 3.A oraz 3.B występuje także dla problemu z ograniczeniem bez czekania [7]. Pozostałe z wymienionych własności są cechą charakterystyczną dla analizowanego problemu. Wszystkie własności będą wyjaśnione na przykładzie numerycznym w którym należy wykonać 3 zadania $J = \{A, B, C\}$ na 3 maszynach $M = \{1,2,3\}$. Czasy trwania zadań na kolejnych maszynach wynoszą odpowiednio:

dla zadania A: $p_{A,1} = 2, p_{A,2} = 2, p_{A,3} = 6$,
dla zadania B: $p_{B,1} = 2, p_{B,2} = 6, p_{B,3} = 2$,
dla zadania C: $p_{C,1} = 6, p_{C,2} = 2, p_{C,3} = 2$.

Przykład ten będzie dalej nazywany instancją pierwotną. Optymalny w sensie długości uszeregowania, harmonogram odpowiada sekwencji wykonywania zadań $\pi = (A, B, C)$. Długość tego optymalnego uszeregowania wynosi $C_{max}\big((A, B, C)\big) = 14$, a dla pozostałych 5 sekwencji wynosi odpowiednio 18 i 22, tabela 1. Wykres przedstawiający optymalne uszeregowanie przedstawiony został na rysunku 3.a.

Tabela 1. Długość harmonogramów dla wszystkich 6 sekwencji zadań $\{A, B, C\}$ z modyfikowanymi czasami trwania operacji

Sekwencja harmonogramu	Instancja pierwotna	Instancja z skróconymi o połowę operacjami zadania B	Instancja z skróconymi o połowę operacjami maszyny 2
(A, B, C)	14	15	17
(A, C, B)	18	17	19
(B, A, C)	18	17	19
(B, C, A)	18	17	19
(C, A, B)	18	17	19
(C, B, A)	22	19	19
średnia	18,0	17,0	18,6



Rys. 3. Harmonogramy permutacyjnego problemu przepływowego bez przestojów

Dowód własności 1.A.
Dla instancji pierwotnej, usuwając z sekwencji zadań (A, B, C) zadanie B, otrzymujemy sekwencję (A, C) oraz odpowiadający jej nowy harmonogram przedstawiony na rysunku 3.b. Nowo powstały harmonogram jest dłuższy niż harmonogram pierwotny. Dowodzi to poprawności własności 1.A.

Dowód własności 1.B.
Usuwając z instancji pierwotnej zadanie B, przestrzeń możliwych rozwiązań zmniejsza się do dwóch harmonogramów. Harmonogram dla sekwencji (A, C) został przedstawiony na rysunku 3.b, a dla sekwencji (C, A) na rysunku 3.c. Każdy z tych harmonogramów jest dłuższy niż harmonogram dla sekwencji (A, B, C) instancji pierwotnej. Dowodzi to poprawności własności 1.B.

Dowód własności 2.A.
W instancji pierwotnej, dla uszeregowania (A, B, C), skracając o połowę czas trwania wszystkich operacji zadania B, uzyskujemy nowy harmonogram dłuższy niż w przypadku pierwotnych danych. Harmonogram ten przedstawiony jest na rysunku 3.d. Dowodzi to poprawności własności 2.A.

Dowód własności 2.B.
Dla instancji pierwotnej, w przypadku skrócenia o połowę czasów trwania wszystkich operacji zadania B, dla każdej możliwej sekwencji wykonywania zadań, odpowiadające im harmonogramy (niezamieszczone w pracy w postaci rysunku) są dłuższe niż optymalny harmonogram instancji pierwotnej, tabela 1. Dowodzi to poprawność własności 2.B.

Dowód własności 3.A.
W instancji pierwotnej, dla uszeregowania (A, B, C), skracając o połowę czas trwania wszystkich operacji wykonywanych na maszynie 2, uzyskujemy nowy harmonogram dłuższy niż w przypadku pierwotnych danych. Harmonogram ten przedstawiony jest na rysunku 3.e. Dowodzi to poprawności własności 3.A.

Dowód własności 3.B.
Dla instancji pierwotnej, w przypadku skrócenia o połowę czasów trwania wszystkich operacji wykonywanych na maszynie 2, dla każdej możliwej sekwencji wykonywania zadań, odpowiadające im harmonogramy (niezamieszczone w pracy w postaci rysunku) są dłuższe niż optymalny harmonogram instancji pierwotnej, tabela 1. Dowodzi to poprawność własności 3.B.

4. Podsumowanie

Przedstawione własności permutacyjnego problemu przepływowego z ograniczeniem bez przestojów są dość nietypowe. Projektując wszelkiego rodzaju algorytmy dedykowane temu problemowi należy mieć świadomość istnienia wykazanych własności (anomalii). Przykładowo konstruując harmonogram, należy mieć świadomość, iż długość budowanego harmonogramu może ulec zmniejszeniu przy dodawaniu kolejnego zadani. Przełożenie zadania w uszeregowaniu także może być nietypowe. Przykładowo wyjęte zadanie paradoksalnie może wydłużyć bieżący harmonogram, a dopiero ponowne włożenie wyciągniętego zadania w inne (lepsze) miejsce może spowodować znaczne skrócenie harmonogramu.

Dodatkową własnością problemu z ograniczeniem bez przestojów, jest brak stabilnej części harmonogramu przy jakichkolwiek jego zmianach. Dokonując modyfikacji dowolnego fragmenty harmonogramu, modyfikacja ta wpływa na ułożenie wszystkich operacji w tym harmonogramie. Utrudnia to projektowanie efektywnych algorytmów konstrukcyjnych, gdyż misternie zbudowanych częściowy harmonogram ulegnie cały zaburzeniu nawet w przypadku dodania kolejnego zadania na jego końcu. Dla porównania w klasycznym problemie przepływowym zmiany w harmonogramie propagowały się tylko od momentu jej wystąpienia. W problemie z ograniczeniem bez czekania harmonogram jest jeszcze bardziej stabilny. Zmiana sekwencji wykonywania operacji w tym wypadku wywołuje tylko zmianę fragmentu harmonogramu.

Literatura

1. Garey M.R., Johnson D.S., Sethi R., The complexity of flowshop and jobshop scheduling, Mathematics of Operations Research 1, 1976, 117-129.
2. Nawaz M., Enscore Jr. E.E., Ham I., A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. OMEGA International Journal of Management Science, 11, 1983, str. 91-95.
3. Nowicki E., Smutnicki C.: A fast taboo search algorithm for the permutation flow-shop problem. European Journal of Operational Research, 91, 1996, str. 160-175.
4. Goncharov Y., Sevastyanov S., The flow shop problem with no-idle constraints: A review and approximation, European Journal of Operational Research, vol. 196(2), 2009, str. 450–456.
5. Graham R., Lawler E., Lenstra J., Rinnooy Kan A., Optimization and approximation in deterministic sequencing and scheduling: a survey, Annals of Discrete Mathematics, 1979, 5, 287.
6. Makuchowski M., Problem przepływowy: permutacyjny, bez czekania, bez przestojów, Automatyzacja Procesów Dyskretnych, Teoria i Zastosowania, tom I., Gliwice 2014, str. 143-152.
7. Frits C.R. Spieksma, Gerhard J. Woeginger, The no-wait flow-shop paradox, Operations Research Letters, vol. 33(6), 2005, str. 603–608.

Dr inż. Mariusz MAKUCHOWSKI
Katedra Automatyki, Mechatroniki i Systemów Sterowania
Politechnika Wrocławska
50-372 Wrocław, ul. Janiszewskiego 11/17
tel./fax: (0-71) 320-29-61
e-mail: mariusz.makuchowski@pwr.edu.pl