



A. ADRABIŃSKI, J. GRABOWSKI, M. WODECKI

Wrocław

Optimalizacja pracy elastycznego centrum obróbczego tokarsko-frezarskiego

(Praca wpłynęła do Redakcji 22.09.1993)

Zasadniczym celem pracy jest prezentacja przykładu zastosowania metod optymalizacji z teorii szeregowania do rozwiązania problemu optymalnego wykorzystania automatu tokarsko-frezarskiego. Praca składa się z dwóch części. W pierwszej zajmujemy się zagadnieniem wyznaczania optymalnej kolejności obróbki detali przez automat. Przedstawiamy opis zagadnienia, model matematyczny oraz algorytm jego rozwiązywania. Dowodzimy także, że zagadnienie to należy do klasy problemów *NP-trudnych*. Druga część pracy dotyczy problemu optymalizacji procesu obróbki detali. Zajmujemy się zagadnieniem optymalizacji współdziałania obrabiarki z magazynami narzędzi oraz wykorzystania narzędzi w trakcie wykonywania operacji na detalach.

1. Wstęp. W pracy przedstawiamy problem optymalizacji procesu produkcyjnego nowej generacji automatów tokarskich, zwanych elastycznymi centrami tokarsko-frezarskimi. Umożliwiają one pełną automatyzację produkcji małoseryjnej i jednostkowej. Prace konstrukcyjne nad tego typu centrami były prowadzone, między innymi, w Fabryce Automatów Tokarskich "RAFAMET" w Kuźni Raciborskiej. Ze względu na dużą kapitałochłonność automatów, wymagany jest wysoki stopień ich wykorzystania. Jest to możliwe dzięki optymalnemu planowaniu procesu produkcyjnego. W trakcie produkcji, detale poddawane są procesowi obróbki polegającemu na wykonywaniu operacji tokarskich, frezarskich i pomiarowych. Narzędzia potrzebne do wykonania operacji pobierane są automatycznie, z odpowiednich magazynów wchodzących w skład centrum, bez ingerencji pracownika. Przed obróbką każdy detal mocowany jest na specjalnej palecie.

Optimalizacja pracy centrum obejmuje zasadniczo dwa problemy:

1) wyznaczanie optymalnej kolejności wykonywania detali, (harmonogra-

- mowanie procesu załadunku na palety, obróbki i rozładunku detali),
- 2) sterowanie wykonywaniem operacji na detalach, w tym:
- (i) przydział i optymalizacja wykorzystania narzędzi,
 - (ii) wyznaczenie terminów wymiany narzędzi w magazynach.

Praca składa się z dwóch części. Część pierwsza, Rozdział 2–4, zawiera opis centrum tokarsko-frezarskiego, model matematyczny i algorytmy heurystyczne służące do harmonogramowania produkcji. Natomiast część druga, Rozdział 5 i 6, poświęcona jest problemowi współdziałania obrabiarki z magazynami narzędzi podczas obróbki detali. W Rozdziale 7 zamieszczamy wyniki eksperymentów obliczeniowych, potwierdzające w pełni prawidłowy wybór metod rozwiązania problemu. Przedstawione w pracy algorytmy zostały zaprogramowane w języku Turbo Pascal i stanowią pakiet oprogramowania, który może być wykorzystywany do sterowania procesem produkcji w dyskretnych systemach produkcyjnych.

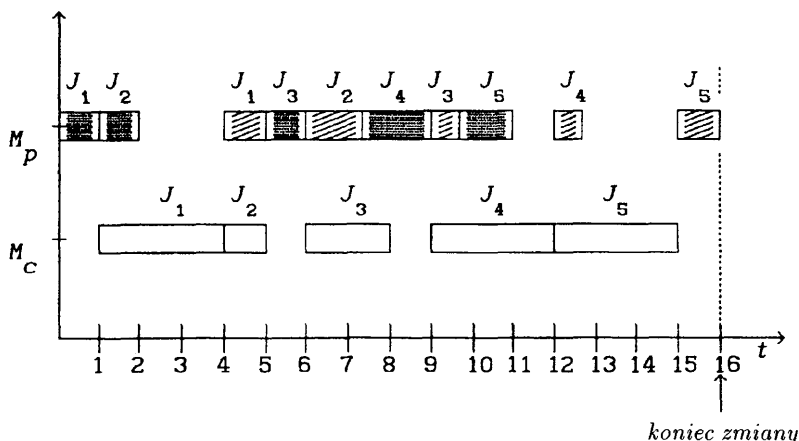
2. Opis zagadnienia. Poniżej przedstawimy zagadnienie harmonogramowania procesu obróbki detali przez centrum tokarsko-frezarskie.

Dany jest zbiór detali, które mają być wykonane przez centrum tokarsko-frezarskie (w skrócie CTF). W skład centrum wchodzi maszyna (obrabarka) wykonująca operacje: tokarskie, frezarskie i pomiarowe, magazyny narzędzi oraz palety, na których mocowane są obrabiane detale. W zależności od rodzaju obróbki, maszyna pobiera automatycznie odpowiednie narzędzie z magazynu. Detal, przed wykonaniem na nim pierwszej operacji, musi być umocowany za pomocą odpowiedniego narzędzia na palecie, a po wykonaniu ostatniej – zdjęty. Każde narzędzie, którym wykonujemy operację na detailu, może być użyte wielokrotnie, jednak łączny czas jego pracy nie może przekroczyć z góry zadanej żywotności. Centrum obsługiwane jest przez pracownika, który m.in. mocuje oraz zdejmuje detale z palet oraz dokonuje wymiany narzędzi w magazynach. Dany jest także porządek technologiczny, tj. kolejność wykonywania pewnych detali, określony na zbiorze detali oraz, dla każdego detailu, czasy i kolejność wykonywania operacji.

Optymalizacja procesu wykonywania detali polega na ustaleniu, dla każdej zmiany pracy centrum, kolejności ich obróbki i kolejności wykonania operacji dla każdego detailu, zgodnie z wymogami porządku technologicznego, oraz na wyznaczeniu zbiorów narzędzi i momentów wymiany narzędzi w magazynach tak, aby stosunek sumy czasów wykonania wszystkich operacji, tzw. czasu pracy pod wiórem, do całkowitego czasu obróbki wszystkich detali był maksymalny.

Na Rys. 1 przedstawiamy diagram Gantta ilustrujący harmonogram czynności wykonywanych przez maszynę i pracownika przy obróbce pięciu detali J_1, J_2, J_3, J_4, J_5 w czasie 16 godzinnej pracy centrum przy założeniu, że mamy do dyspozycji dwie palety. Symbolem M_p oznaczono czynności

wykonywane przez pracownika: mocowanie detalu na palecie (pola zakropkowane) oraz zdejmowanie z palety (linie ukośne). Natomiast, M_c oznacza obrabiarkę. Dla tego przykładu, stosunek czasu pracy obrabiarki pod wiórem do całkowitego czasu pracy wynosi 4/5.



Rys. 1.

3. Sformułowanie problemu. W paragrafie tym sformalizujemy opisane wyżej zagadnienie optymalizacji pracy centrum tokarsko-frezarskiego. Udowodnimy także, że nawet jego uproszczona wersja należy do klasy problemów *NP-trudnych*, tj. klasy problemów, dla których nie jest obecnie możliwe skonstruowanie algorytmów o wielomianowej złożoności obliczeniowej.

W dalszej części pracy, *zadaniem* nazywać będziemy zbiór operacji wykonywanych przez maszynę w czasie jednego zamocowania detalu na palecie. Detal może być obrabiany wielokrotnie, przy różnych zamocowaniach.

Założmy, że dany jest zbiór n zadań

$$J = \{J_1, J_2, \dots, J_n\},$$

które mają być wykonane przez obrabiarkę (maszynę) wchodzącą w skład centrum. Przed rozpoczęciem wykonywania zadania, detal z nim związany mocowany jest na palecie, a po wykonaniu, zdejmowany. Paleta może być wykorzystana wielokrotnie. "Ładowanie i zdejmowanie wykonuje obsługujący maszynę pracownik (operator).

Dla każdego zadania $J_i \in J$, niech z_i , p_i oraz w_i będą odpowiednio: *czasem załadunku* (tj. czasem załadunku i mocowania detalu na palecie), *czasem wykonywania* oraz *czasem wyładunku* detalu z palety. Zakładamy, że centrum pracuje w systemie *zmianowym*. Oznaczmy przez dz *długość zmiany*, liczbę godzin pracy centrum w każdym dniu ($dz \leq 24$). Detal,

w dniu obróbki, musi być załadowany, wykonany i zdjęty z palety. W dalszej części pracy będziemy także rozpatrywać problem, gdy centrum pracuje w systemie *ciągłym*, tj. bez wyłączania maszyny.

Rozpatrywany przez nas problem harmonogramowania procesu obróbki detali polega na wyznaczeniu w zbiorze J podzbiorów zadań, które będą wykonane w kolejnych dniach pracy maszyny oraz kolejności ich wykonywania w każdym dniu tak, aby stosunek

$$(1) \quad K = \frac{T_w}{T_c},$$

zwany *współczynnikiem wykorzystania maszyny*, osiągnął wartość maksymalną, gdzie:

T_w — jest *efektywnym* (tzw. pod wiórem) *czasem* pracy maszyny, czyli

$$T_w = \sum_{i=1}^n p_i, \text{ a}$$

T_c — jest *całkowitym czasem* wykonania zadań, tj. czasem pracy centrum jaki upłynął od momentu rozpoczęcia mocowania pierwszego detalu na palecie do momentu zakończenia wykonywania ostatniego zadania przez maszynę.

Dodatkowo, muszą być spełnione następujące ograniczenia:

- a) w każdej chwili maszyna wykonuje co najwyżej jedno zadanie,
- b) wykonywanie zadania nie może być przerwane przed jego zakończeniem,
- c) detale należy zamocować na palecie, a po wykonaniu wyładować w dniu, w którym jest obrabiany,
- d) w każdej chwili na palecie może się znajdować co najwyżej jeden detal,
- e) pracownik, w każdej chwili, może wykonywać czynność mocowania albo zdejmowania detalu z palety.

Jak łatwo zauważyć, wielkość T_w występująca w liczniku ilorazu (1) nie zależy od kolejności wykonywania zadań, a zatem maksymalizacja tego wyrażenia jest równoważna minimalizacji wartości mianownika, tj. T_c . Wobec tego, w dalszej części pracy, za kryterium optymalizacji przyjmujemy czas wykonania wszystkich zadań T_c , który należy minimalizować.

Oznaczmy przez $\mathbf{P}$ wyżej opisany problem optymalizacji kolejności wykonywania zadań. Sformułujemy pewien podproblem $\mathbf{P}_1$ problemu $\mathbf{P}$ przyjmując dodatkowe ograniczenia.

Założmy obecnie, że czasy załadunku i wyładunku detalu są równe zero, tj. $z_i = w_i = 0$, dla zadania $J_i \in J$, oraz za kryterium optymalności przyjmijmy liczbę dni pracy maszyny potrzebną do wykonania wszystkich zadań, którą należy minimalizować. Jak łatwo zauważyć, $\mathbf{P}_1$ jest pewnym szczególnym przypadkiem problemu $\mathbf{P}$. Bez utraty ogólności, problem ten można sformułować w postaci decyzyjnej, tj. wymagającej jedynie podania odpo-

wiedzi *tak* lub *nie*. Oznaczmy więc przez P_1D problem P_1 sformułowany jako problem decyzyjny.

Problem P_1D

Dane: zbiór n zadań $J = \{J_1, J_2, \dots, J_n\}$, czasy wykonywania zadań $p_1, p_2, \dots, p_n$, długość zmiany dz , ($p_i \leq dz$) oraz ustalona liczba naturalna L .

Cel: czy istnieje przyporządkowanie zadań poszczególnym dniom pracy maszyny spełniające ograniczenia (a)-(b) i takie, że liczba dni pracy jest nie większa niż L ?

W dalszej części tego rozdziału pokażemy, że problem P_1D należy do klasy problemów *NP-zupełnych*. W tym celu udowodnimy, że problem pakowania (ang. *bin packing*) jest wielomianowo transformowalny do P_1D .

Problem pakowania BP

Dane: zbiór n ponumerowanych elementów $N = \{1, 2, \dots, n\}$ o wielkościach $\{b_1, b_2, \dots, b_n\}$, rozmiar pudełka C , ($b_i \leq C$) oraz pewna liczba naturalna L .

Cel: czy istnieje L elementowa partycja zbioru N , tj. rodzina parami rozłącznych podzbiorów $I_1, I_2, \dots, I_L$ taka, że

$$\sum_{i \in I_j} b_i \leq C \quad \text{dla } 1 \leq j \leq L?$$

Dowód, że problem BP należy do klasy problemów *NP-zupełnych* zamieszczony jest w pracy [2].

Twierdzenie 1. *Problem pakowania BP jest wielomianowo transformowalny do problemu P_1D .*

Dowód. Poniżej pokażemy, że dowolny przykład problemu BP jest wielomianowo transformowalny do pewnego przykładu problemu P_1D mającego rozwiązanie (*tak*) wtedy i tylko wtedy, gdy odpowiedni przykład problemu BP ma rozwiązanie (*tak*).

Niech $b_1, b_2, \dots, b_n, C, L$ będzie układem danych pewnego przykładu problemu BP. Skonstruujemy dla niego odpowiedni przykład problemu P_1D . Zdefiniujemy zbiór zadań $J = \{J_1, J_2, \dots, J_n\}$. Czas wykonania zadania J_i określamy jako $p_i = b_i$ ($i = 1, 2, \dots, n$) oraz $dz = C$. Pytamy, czy istnieje przydział zadań do poszczególnych dni pracy maszyny spełniający ograniczenia (a)-(b) i taki, że liczba dni jest nie większa niż L . Przypuścimy, że przykład problemu BP ma rozwiązanie. Istnieje więc L elementowa partycja $I_1, I_2, \dots, I_L$ zbioru N taka, że $\sum_{i \in I_j} b_i \leq C$, ($j = 1, 2, \dots, L$). Oznaczmy przez H_i podzbiór zadań ze zbioru J , które należy wykonać w

i -tym dniu pracy maszyny i określmy $H_i = \{J_j : j \in I_i\}$. Jak łatwo zauważyć $\sum_{j \in H_i} p_j \leq dz$ dla $i = 1, 2, \dots, L$, liczba dni pracy maszyny jest nie większa niż L oraz spełnione są ograniczenia (a)-(b). Udowodniliśmy więc, że jeśli przykład problemu **BP** ma rozwiązanie, to także odpowiadający mu przykład problemu **P₁D** ma rozwiązanie.

Załóżmy teraz, że przykład problemu **P₁D** ma rozwiązanie. Istnieje więc przydział zadań do poszczególnych dni pracy maszyny spełniający warunki (a)-(b). Niech H_i będzie zbiorem zadań wykonywanych w i -tym dniu. Oznaczmy przez $I_i = \{j : J_j \in H_i\}$ zbiór elementów, które należy umieścić w i -tym pudełku. Ponieważ $\sum_{j \in H_i} p_j \leq dz$, zatem zachodzi $\sum_{j \in I_i} b_j \leq C$, ($i = 1, 2, \dots, L$) oraz $I_1, I_2, \dots, I_L$ jest partycją pokrywającą zbiór N . Udowodniliśmy więc, że odpowiedni przykład problemu **BP** ma rozwiązanie.

Dowód wielomianowości przedstawionej transformacji pomijamy, ponieważ jest on oczywisty. ■

Z udowodnionego twierdzenia wynika, że obecnie nie jest możliwa konstrukcja algorytmu, o wielomianowej złożoności obliczeniowej, wyznaczania optymalnego rozwiązania problemu **P₁**, a więc i problemu **P**. Fakt ten usprawiedliwia stosowanie algorytmów heurystycznych, o wielomianowej złożoności obliczeniowej, do rozwiązywania tego typu zagadnień. Algorytmy takie przedstawiamy w następnym rozdziale.

Załóżmy, że na zbiorze zadań J określona jest relacja częściowego porządku $RT \subset J \times J$, wyrażająca wymagania *porządku technologicznego*, tj. kolejność wykonywania pewnych zadań. Jeśli para $(J_i, J_j) \in RT$ to oznacza, że rozpoczęcie ładowania na paletę detalu zadania J_j może nastąpić dopiero po rozładunku detalu zadania J_i (ma to miejsce między innymi wtedy, gdy detal obrabiany jest wielokrotnie przy różnych zamocowaniach). Relację tę można przedstawić jednoznacznie w postaci acyklicznego grafu skierowanego $G = (V, E)$ zwanego *grafem pierwszeństwa*, którego zbiorem wierzchołków jest $V = \{1, 2, \dots, n\}$, a E zbiorem łuków. Odpowiednikiem zadania $J_i \in J$ jest w grafie G wierzchołek i . Para różnych wierzchołków $(u, v) \in V$ tworzy łuk w G , tj. $(u, v) \in E$ wtedy i tylko wtedy, gdy $(J_u, J_v) \in RT$. W dalszej części pracy, wszędzie tam gdzie nie prowadzi to do nieporozumień, będziemy zamiennie pisali wierzchołek i oraz zadanie J_i . Można łatwo udowodnić następujące własności:

Własność 1. W grafie pierwszeństwa istnieje wierzchołek, który nie jest końcem żadnego łuku.

Własność 2. Każdy podgraf grafu pierwszeństwa nie zawiera cyklu.

Wprowadzimy teraz pewne dodatkowe pojęcia i oznaczenia. Na zbiorze wierzchołków V oraz łuków E grafu pierwszeństwa G określamy nieujemne funkcje wagowe. *Wagą wierzchołka* $i \in V$ jest czas wykonywania p_i zadania J_i . Przez f_{ij} oznaczamy *wagę łuku* $(i, j) \in E$. Jeśli $f_{ij} > 0$ to znaczy, że roz-

poczęcie ładowania detalu zadania J_j na paletę może nastąpić dopiero po upływie f_{ij} czasu od momentu zakończenia wyładunku detalu zadania J_i . Jest to czas potrzebny np. do wymiany zamocowania lub ostygnięcia, gdy oba zadania dotyczą dwóch różnych stron tego samego detalu. Dalej, niech r_i będzie *najwcześniejszym możliwym terminem*, licząc od momentu rozpoczęcia pracy centrum, w którym można rozpocząć ładowanie detalu zadania J_i na paletę. Ograniczenie to wynika z faktu, że np. pewne detale przybywają już po rozpoczęciu pracy przez centrum. Korzystając z wyżej przedstawionych pojęć i oznaczeń wprowadzamy dodatkowo następujące ograniczenia:

- g) zachowany musi być porządek technologiczny wykonywania zadań oraz uwzględnione wagi wierzchołków i łuków grafu pierwszeństwa,
- h) termin rozpoczęcia załadunku detalu nie może być mniejszy niż najwcześniejszy możliwy termin załadunku.

W dalszej części pracy rozważać będziemy problem **P** łącznie z ograniczeniami **g** i **h**.

4. Algorytmy optymalizacji procesu obróbki detali. W pierwszej kolejności przedstawimy algorytm, który może być stosowany w przypadku gdy centrum pracuje w systemie zmianowym a następnie, algorytm stosowany dla pracy ciągłej.

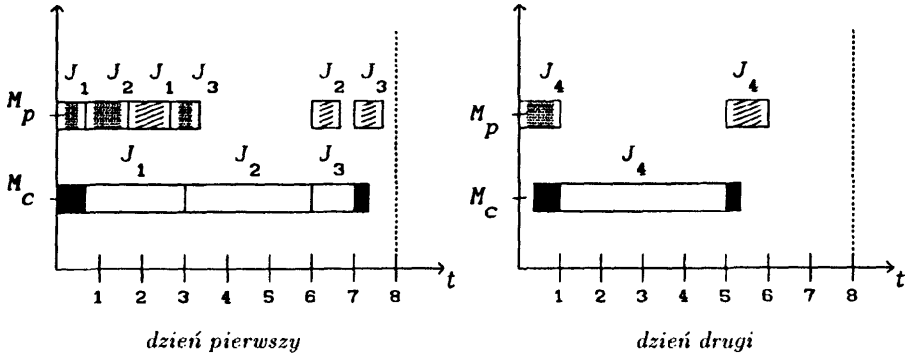
Niech S_m będzie *czasem rozruchu*, a Z_m *czasem zatrzymania* maszyny. W trakcie rozruchu i zatrzymania maszyna nie może wykonywać żadnej operacji. Rozruch następuje na początku zmiany, w każdym dniu pracy, a zatrzymanie przed końcem zmiany, po obróbce ostatniego detalu.

Oznaczmy przez H_i całkowicie uporządkowany podzbiór zbioru zadań wykonywanych przez maszynę w i -tym dniu pracy, tj. zbiór, w którym ustalona jest kolejność wykonywania wszystkich zadań uwzględniająca również relację porządku technologicznego. Z kolei, niech t_i (*zajętość maszyny*) będzie terminem zakończenia wykonywania przez maszynę ostatniego zadania znajdującego się aktualnie w H_i oraz T_i (*zajętość pracownika*) terminem zakończenia przez pracownika załadunku (albo wyładunku) detalu ostatniego w i -tym dniu zadania. Zadania należące do zbiorów H_i nazywać będziemy *zadaniami ustalonymi*, a pozostałe z J *zadaniami wolnymi*.

Dzień k -ty pracy maszyny jest *dopuszczalnym* dla zadania J_j , jeżeli w dniu tym zadanie to może być *całkowicie wykonane*, tj.

- (i) od momentu rozpoczęcia wykonywania przez centrum zadań z J do momentu T_k , w którym (w k -tym dniu) pracownik może rozpocząć ładowanie detalu zadania J_j , upłynęło co najmniej r_j czasu,
- (ii) załadunek, wykonanie i rozładunek zadania J_j nastąpi w k -tym dniu,
- (iii) od momentu zakończenia rozładunku każdego zadania J_i będącego poprzednikiem J_j (w relacji porządku technologicznego, tj. $(i, j) \in E$) do chwili T_k upłynęło co najmniej f_{ij} czasu.

Na Rys. 2 przedstawiamy przykładowy diagram Gantta dla dwóch ośmiodzinnych ($dz = 8$) dni pracy centrum. Oznaczenia są takie same jak na Rys. 1, a ponadto zaciemniony jest czas rozruchu i zatrzymania maszyny odpowiednio na początku i końcu każdej zmiany. Dla tego przykładu $H_1 = \{1, 2, 3\}$, $H_2 = \{4\}$ oraz $t_1 = 7$, $T_1 = 7.5$, $t_2 = 5$, $T_2 = 6$.



Rys. 2.

Oznaczmy przez A_i oraz B_i zbiory *następników* i *poprzedników* wierzchołka $i \in V$ w grafie G . Jeżeli z wierzchołka i do wierzchołka j istnieje droga w G , to $i \in B_j$ oraz $j \in A_i$.

Dla zadania $J_i \in J$, niech

$$q_i = \max \left\{ \sum_{j \in A_i} p_j, \max \{d_{ij} : j \in A_i\} \right\},$$

gdzie d_{ij} jest długością drogi w grafie pierwszeństwa G z wierzchołka i do j , tj. sumą wag wierzchołków i łuków tworzących tę drogę. Jeśli $A_i = \emptyset$, to przyjmujemy, że $q_i = 0$. Ponadto, niech

$$(2) \quad b_i = r_i + p_i + q_i.$$

Jak łatwo zauważyć, wielkość b_i jest najwcześniejszym terminem zakończenia wykonywania zadań z J_i . Może być ona wykorzystana do obliczania dolnego ograniczenia na liczbę dni pracy centrum.

Wyznaczając kolejność wykonywania zadań, spośród zadań wolnych, których wszystkie poprzedniki w grafie pierwszeństwa G są już ustalone, wybieramy zadanie o maksymalnej watości wyrażenia (2). Dla ustalenia uwagi założymy, że wybraliśmy zadanie J_k . Następnie, spośród dni dopuszczalnych dla tego zadania wybieramy pierwszy (o najmniejszym numerze). W dniu tym zadanie J_k będzie wykonywane. Jeśli wybraliśmy dzień l , to:

- ustalamy zadanie J_k w zbiorze H_l , tj. $H_l \leftarrow H_l \cup \{J_k\}$ przy czym, zadanie to będzie wykonywane jako ostatnie, tj. po wszystkich zadaniach znajdujących się aktualnie w H_l ,

- uaktualniamy czas zajętości pracownika i maszyny uwzględniając fakt, że zadanie J_k będzie wykonywane w l -tym dniu,
- modyfikujemy zbiory poprzedników zadania J_k :

$$\forall i \in A_k, \quad B_i \leftarrow B_i \setminus \{k\},$$

- uaktualniamy najwcześniejsze możliwe terminy rozpoczęcia wykonywania zadań uwzględniając termin wykonania J_k , tj.

$$\forall j \in A_k, \quad r_j := \max\{r_j, (l-1)dz + t_l + w_l + f_{kj}\}.$$

Po wykonaniu wyżej opisanych modyfikacji wszystkie zbiory poprzedników zawierają jedynie zadania wolne, tj. zadania, które nie zostały jeszcze ustalone.

Algorytm AHZ {Wyznaczenie rozwiązania heurystycznego dla problemu harmonogramowania produkcji, tj. przydzielenie oraz wyznaczenie kolejności wykonywania zadań w poszczególnych dniach pracy centrum}

KROK 1 {Zapoczątkowanie obliczeń}

$$H_j \leftarrow \emptyset, t_j := Sm, T_j := 0, \text{ dla } j = 1, 2, \dots$$

Wyznaczyć zbiór poprzedników B_i i następników A_i oraz obliczyć b_i zgodnie z (2), dla każdego zadania $J_i \in J$.

KROK 2 {Iteracja}

Wykonać n razy następujące polecenia:

{Określenie kolejnego zadania do wykonania (ustalenia)}

Wyznaczyć zadanie J_k takie, że

$$b_k = \max\{b_i : B_i = \emptyset \text{ oraz } J_i \text{ jest zadaniem wolnym}\}.$$

{Określenie dnia, w którym zadanie J_k będzie wykonywane}

Wyznaczyć dzień l dla zadania J_k taki, że

$$l = \min\{j : j \text{ jest dniem dopuszczalnym dla zadania } J_k\}$$

{Modyfikacja parametrów}

Ustalić zadanie J_k w dniu l przyjmując $H_l \leftarrow H_l \cup \{J_k\}$,

uaktualnić czasy zajętości t_l i T_l , oraz $B_i \leftarrow B_i \setminus \{J_k\}$ i

$$r_j := \max\{r_j, (l-1)dz + t_l + w_l + f_{kj}\}, \quad \forall j \in A_k$$

■

Złożoność obliczeniowa algorytmu AHZ jest równa $O(n^2)$. Można łatwo wykazać, że otrzymane rozwiązanie jest dopuszczalne, czyli spełnia wszystkie ograniczenia (a)-(h). Modyfikacja zbiorów poprzedników wykonywana w Kroku 2 algorytmu jest równoważna usunięciu z grafu pierwszeństwa G wierzchołka k oraz wszystkich łuków, których początkiem jest ten wierzchołek. Na podstawie Własności 1 i 2 zmodyfikowany graf G zawiera wierzchołek nie będący końcem żadnego łuku, a więc wierzchołek, którego zbiór poprzedników jest pusty. Wobec tego wielkość b_k jest dobrze określona. Strategia,

którą stosujemy w Kroku 2 (wyznaczając numer zadania i odpowiedni dzień) jest oparta na idei algorytmu *najlepszego dopasowania uporządkowanych zadań* (ang. first-fit decreasing) wyznaczającego rozwiązanie przybliżone dla problemu pakowania BP (zobacz [4]).

TWIERDZENIE 2. *Jeżeli L_{OP} jest minimalną liczbą dni potrzebnych do wykonania zadań ze zbioru J , a L_{AHZ} liczbą dni wyznaczoną przez algorytm AHZ dla problemu P_1D , to wówczas*

$$L_{AHZ} \leq \frac{11}{9} L_{OP} + 4.$$

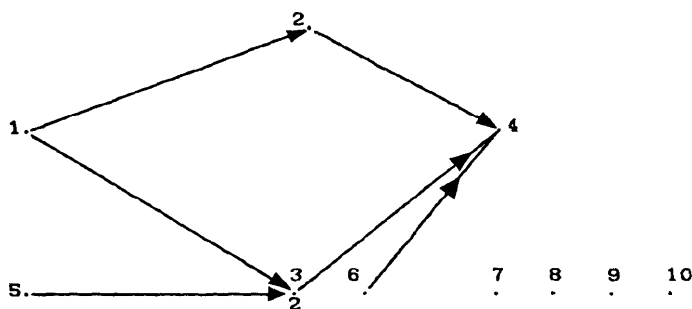
Powyższe twierdzenie zostało udowodnione w pracy [4]. ■

Dla uproszczenia opisu algorytmu przyjęliśmy, że długość zmiany dz , tj. liczba godzin pracy centrum, w każdym dniu, jest stała. W ogólnym przypadku długości zmian mogą być różne.

Obecnie, zilustrujemy działanie algorytmów **AHZ** na przykładzie niewielkiego zagadnienia. Dane liczbowe przedstawione są w Tabeli 1.

TABELA 1

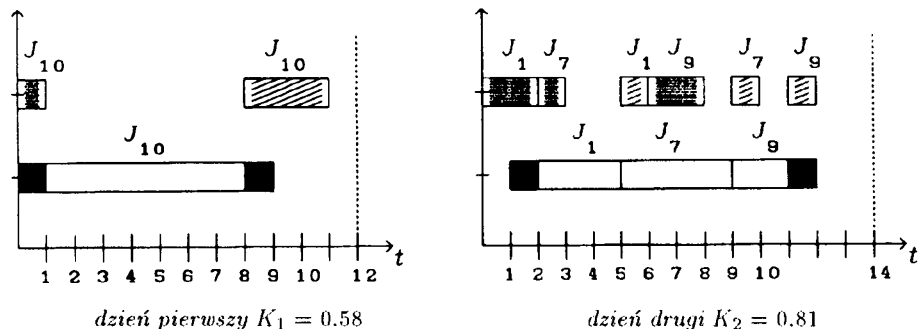
nr zad. →	1	2	3	4	5	6	7	8	9	10
r_i	6	5	4	3	2	1	8	0	4	0
p_i	3	6	8	5	9	2	4	3	2	7
z_i	2	1	2	1	2	2	1	1	2	1
w_i	1	1	1	1	2	3	1	1	1	3



Rys. 3.

Relację porządku technologicznego RT reprezentuje graf pierwszeństwa $G = (V, E)$ przedstawiony na Rys. 3. Zbiór wierzchołków $V = \{1, 2, \dots, 10\}$ a łuków $E = \{(1, 2), (1, 3), (5, 3), (2, 4), (3, 4), (6, 4)\}$. Ponadto, czas rozruchu oraz zatrzymania maszyny $Sm = Zm = 1$. Długości zmian, w kolejnych dniach, wynoszą odpowiednio $dz = (12, 14, 14, 16, 5, 7, 20, 14)$.

W wyniku działania algorytmu **AHZ**, wyznaczone zostały uporządkowane zbiory zadań: $H_1 = (10)$, $H_2 = (1, 7, 9)$, $H_3 = (5)$, $H_4 = (2, 3)$, $H_5 = (8)$, $H_6 = (6)$, $H_7 = (4)$, $H_8 = \emptyset$ wykonywanych w kolejnych dniach pracy. Współczynnik wykorzystania maszyny $K = 0.72$. Na Rys. 4 przedstawiono diagramy Gantta dla pierwszych dwóch dni pracy centrum. Wykorzystanie maszyny w i -tym dniu określa współczynnik K_i . Pozostałe oznaczenia są takie same, jak na Rys. 2.



Rys. 4.

Do tej pory rozpatrywaliśmy problem obróbki detali, gdy centrum pracuje w systemie zmianowym. Obecnie zajmiemy się krótko przypadkiem pracy centrum w systemie ciągłym. Rozruch maszyny następuje przed rozpoczęciem obróbki pierwszego detalu, a zatrzymanie dopiero po wykonaniu wszystkich zadań. Czynności ładowania i zdejmowania detali z palet mogą być wykonane w dowolnym momencie. Oczywiście muszą być spełnione ograniczenia (a)-(h). Ten system pracy można traktować jako pracę zmianową, w której długość zmiany dz jest równa nieskończoności. Ze względu jednak na specyfikę tego problemu skonstruowaliśmy oddzielny, bardziej efektywny, algorytm oznaczany dalej w skrócie przez **AHC**. Wyznaczając kolejności wykonywania zadań korzystamy ze znanej w literaturze zasady Jacksona stosowanej w wielu algorytmach dla problemów szeregowania zadań. Dokładnie jest ona przedstawiona w pracy [3]. Poniżej zamieszczamy krótki opis algorytmu oraz sam algorytm.

Na początku przyjmujemy, że zajętość pracownika $T=0$, maszyny $t = Sm$ oraz zbiór zadań ustalonych $H = \emptyset$. Niech

$$K = \{J_i \in J : J_i \notin H, B_i = \emptyset, r_i \leq t\},$$

będzie zbiorem zadań wolnych, których wykonywanie może się rozpocząć w chwili t . Jeżeli zbiór $K \neq \emptyset$ wówczas wybieramy do wykonywania zadanie z K , dla którego wartości wyrażenia (2) jest minimalna. Gdy $K = \emptyset$, wtedy wyznaczamy zbiór

$$K' = \{J_i \in J : J_i \notin H, B_i = \emptyset\},$$

i do wykonywania wybieramy zadanie z K' o najmniejszym możliwym terminie rozpoczęcia. Łatwo wykazać, że jeżeli zbiór $K = \emptyset$, to $K' \neq \emptyset$. Po wyznaczeniu zadania do wykonywania, modyfikujemy zajętości pracownika i maszyny oraz zbiory poprzedników i najwcześniejsze możliwe terminy rozpoczęcia wykonywania zadań tak, jak w algorytmie **AHZ**.

Algorytm AHC

{ Wyznaczanie harmonogramu produkcji dla ciągłej pracy centrum }

KROK WSTĘPNY: *{ Zapoczątkowanie obliczeń }*

$H \leftarrow \emptyset$; $U \leftarrow J$; $t := Sm$; $T := 0$;

for $i := 1$ **to** n **do**

 Wyznaczyć wartość wyrażenia b_i zgodnie z (2);

KROK ITERACYJNY: *{ Ustalenie kolejności wykonywania zadań }*

while $U \neq \emptyset$ **do**

begin *{ Wyznaczenie kolejnego zadania do wykonania }*

if $K \neq \emptyset$ **then**

 Wybrać zadanie $J_l \in K$ takie, że $b_l = \max\{b_i: J_i \in K\}$

else

 Wybrać zadanie $J_l \in K'$ takie, że $r_l = \min\{r_i: J_i \in K'\}$;

{ Modyfikacja parametrów }

$U \leftarrow U \setminus \{J_l\}$; $T := t := r_l$;

 Ustalić zadanie J_l przyjmując $H \leftarrow H \cup \{J_l\}$;

 Uaktualnić czasy zajętości t i T oraz

$B_i \leftarrow B_i \setminus \{l\}$ i $r_j := \max\{r_j, t_l + w_l + f_{lj}\}$, $\forall j \in A_l$.

end

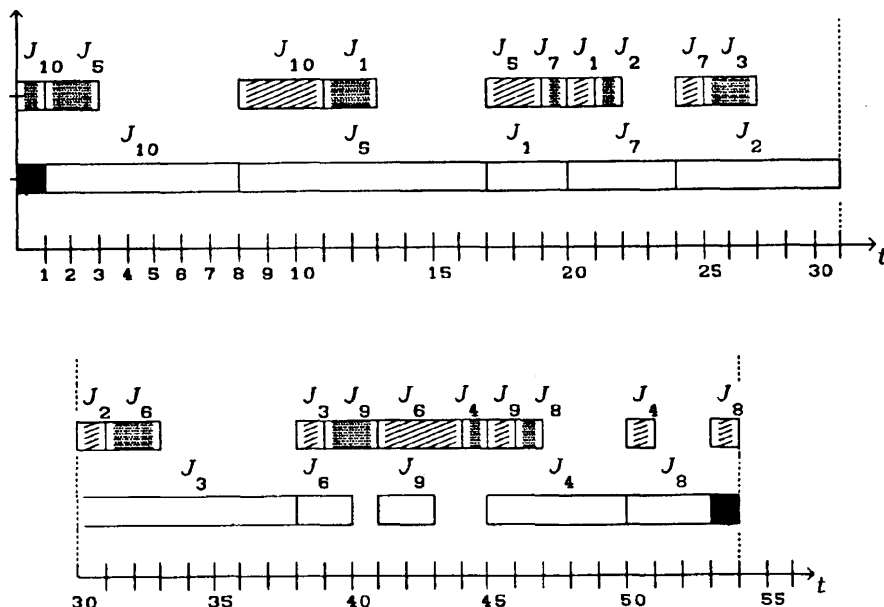
■

Złożoność obliczeniowa algorytmu **AHC** jest równa $O(n^2)$.

Zamieszczony w Tabeli 1 przykład zagadnienia został rozwiązany przez algorytm **AHC** dla centrum pracującego w systemie ciągłym. Wyznaczona została kolejność wykonywania zadań (10, 5, 1, 7, 2, 3, 6, 9, 4, 8), dla której współczynnik wykorzystania maszyny $K = 0.92$. Diagram Gantta, dla tego uszeregowania zadań, przedstawiony jest na Rys. 5. Wszystkie oznaczenia są takie same jak na Rys. 2.

Wyniki obliczeniowe algorytmów przedstawionych w tej części pracy, dla wielu losowo generowanych przykładów, zamieszczamy w Rozdziale 7. Wyniki te, oraz obliczenia wykonane dla dużej ilości praktycznych przykładów, w pełni potwierdzają słuszność założeń dotyczących modelu procesu obróbki detali jak i wyboru algorytmów.

5. Współdziałanie obrabiarki z magazynami narzędzi. W rozdziale tym zajmujemy się współdziałaniem obrabiarki z magazynami narzędzi podczas procesu obróbki detali.



Rys. 5.

Wchodząca w skład centrum tokarsko-frezarskiego obrabiarka wyposażona jest w magazyny narzędzi. Umieszczone w nich, na początku zmiany, narzędzia mogą nie wystarczyć do wykonania wszystkich operacji przewidzianych na dany dzień. Dlatego, w trakcie obróbki, operator dokonuje wymian. Usuwa niepotrzebne lub zużyte narzędzia i w ich miejsce wstawia nowe.

Każde zadanie $J_i \in J$ jest ciągiem m_i ponumerowanych operacji:

$$J_i = \langle O_i^1, O_i^2, \dots, O_i^{m_i} \rangle,$$

które należy wykonać w zadanym z góry *porządku technologicznym*, tzn. operacja O_i^k ma być wykonana bezpośrednio po operacji O_i^{k-1} , a przed O_i^{k+1} . Operacja może być: *tokarska*, *frezarska* lub *pomiarowa*. Niech p_i^k będzie czasem wykonywania, n_i^k numerem typu narzędzia a z_i^k zmienną, której wartością jest numer otworu w magazynie, z którego maszyna pobiera narzędzie do wykonywania operacji O_i^k . Dalej, niech s_j będzie numerem typu a c_j *dopuszczalnym* (maksymalnym) *czasem pracy* narzędzia znajdującego się w j -tym otworze. Podobnie, jak w przypadku zadań zakładamy, że:

- a') w każdej chwili maszyna wykonuje co najwyżej jedną operację,
- b') wykonywanie żadnej operacji nie może być przerwane,
- c') każdą operację należy wykonać jednym narzędziem.

Problem sterowania pracą magazynów polega na wyznaczeniu:

- narzędzia do wykonywania każdej operacji $O_i^k \in J_i$, ($J_i \in J$),
- momentów wymiany narzędzi w magazynach,
- zbiorów narzędzi wyjmowanych i wstawianych w trakcie wymiany,
- liczby narzędzi każdego typu potrzebnych do obróbki detali na każdy dzień pracy centrum

tak, aby liczba potrzebnych narzędzi oraz liczba wymian była jak najmniejsza.

Jeśli w czasie pracy maszyny pewne otwory w magazynach są wolne, wówczas umieszczając w nich będziemy dodatkowe narzędzia zwane narzędziami *dublującymi* (awaryjnymi).

Poniżej przedstawimy opis algorytmu sterowania pracą magazynów zakładając, że centrum pracuje w systemie zmianowym.

Stosując algorytm **AHZ** przedstawiony w Rozdziale 4 założymy, że rozwiązaliśmy problem wyjściowy **P** harmonogramowania procesu obróbki (przyjmujemy $p_i = \sum_{j=1}^{n_i} p_i^j$ za czas wykonywania zadania $J_i \in J$), tj. wyznaczyliśmy uporządkowane zbiory zdań na każdy dzień pracy CTF. Ponieważ kolejność wykonywania operacji każdego zadania jest z góry określona (porządek technologiczny) mamy więc, na każdy dzień pracy, wyznaczoną jednoznacznie kolejność wykonywania operacji.

Przypuśćmy, że przystępujemy do rozpatrywania operacji O_i^k , a wszystkie operacje, które mają być przed nią wykonane (w danym dniu), zostały już rozpatrzone. Dla ustalenia uwagi przyjmijmy, że jest to operacja tokarska. Omówimy teraz pewne czynności związane z ustaleniem numeru otworu, z którego maszyna pobiera narzędzie do wykonania tej operacji.

Jeśli w magazynie narzędzi brak jest takiego, którym można wykonać rozpatrywaną operację a są jeszcze wolne otwory, to wykonujemy czynność *wstawiania* narzędzia do magazynu. Natomiast, gdy wszystkie otwory są zajęte, to wykonujemy czynność *wymiany* polegającą na zwolnieniu pewnego otworu (usunięciu znajdującego się w nim narzędzia) i następnie wstawieniu odpowiedniego narzędzia. Omówimy najpierw dokładniej obie te czynności a następnie, podamy metodę wyznaczania momentu (numeru operacji) w którym należy je wykonać.

Wstawianie narzędzia do magazynu.

Jeżeli w magazynie brak jest narzędzia, którym należy wykonać operację O_i^k lub też dopuszczalny czas pracy każdego z narzędzi tego typu, znajdującego się w magazynie, jest mniejszy od czasu wykonywania tej operacji, a są jeszcze wolne otwory, wówczas wstawiamy nowe narzędzie. Czynność ta polega więc na wyznaczeniu wolnego otworu w magazynie i umieszczeniu w nim narzędzia, którym będzie wykonywana rozpatrywana operacja.

Wymiana narzędzi w magazynie.

Założmy, że wśród narzędzi znajdujących się w magazynie nie ma takiego, którym można wykonać operację O_l^k i wszystkie otwory są zajęte. Czynność wymiany polega na wyznaczeniu numeru pewnego otworu i usunięciu znajdującego się w nim narzędzia a następnie, wykonaniu czynności wstawiania. W pierwszej kolejności usuwać będziemy narzędzie, które nie będzie już używane do wykonywania pozostałych na dany dzień operacji. Jeśli nie ma takiego, wówczas usuwamy narzędzie należące do najliczniejszego typu i mające najmniejszy dopuszczalny czas pracy.

Jeśli z operacją O_l^k jest związana ostatnia, w danym dniu wymiana narzędzi, to wykonujemy dodatkową czynność *dublowania* narzędzi.

Dublowanie narzędzi w magazynie.

Celem dublowania jest wyznaczenie dodatkowego zbioru narzędzi, które będą umieszczone w magazynie w trakcie wykonywania ostatniej, w danym dniu, czynności wymiany. Dublujemy narzędzia, które są najczęściej używane i ulegające awariom.

Powróćmy obecnie do problemu wyznaczania numeru otworu w magazynie, z którego należy pobrać narzędzie do wykonywania operacji.

Jeśli w magazynie jest wiele narzędzi typu n_l^k , którymi można wykonać operację O_l^k , wówczas wybieramy spośród nich takie, dla którego różnica dopuszczalnego czasu pracy i czasu wykonania operacji jest nieujemna i minimalna, tj. wybieramy narzędzie z otworu q dla którego:

$$c_q - p_l^k = \min\{j : j \text{ - jest numerem otworu oraz } c_j - p_l^k \geq 0 \text{ i } s_j = n_l^k\}.$$

Następnie, zmiennej z_l^k nadajemy wartość q (numer otworu, w którym znajduje się wybrane narzędzie) oraz modyfikujemy dopuszczalny czas pracy przydzielonego narzędzia c_q przyjmując $c_q - p_l^k$. Opisana metoda minimalizacji liczby narzędzi, ustalonego typu, potrzebnych do wykonania wszystkich (tego typu) operacji sprowadza się do przedstawionego w Rozdziale 2 problemu pakowania **BP**. Wielkości elementów są równe czasom wykonywania operacji, pudełka mają różne rozmiary i równe żywotności poszczególnych narzędzi. Konstruując algorytm, dla tego problemu, będziemy stosowali strategię *najlepszego dopasowania* (ang. best-fit), z której korzystaliśmy przy wyznaczaniu rozwiązania przybliżonego dla problemu **P**. Dokładny jej opis zamieszczony jest w pracy [2], a oszacowanie uzyskanego rozwiązania daje poniższe twierdzenie.

TWIERDZENIE 3. *Jeżeli L_{opt} jest rozwiązaniem optymalnym, a L_{bf} rozwiązaniem otrzymanym w wyniku stosowania strategii najlepszego dopasowania dla problemu pakowania, to*

$$L_{bf} \leq \frac{17}{10} L_{opt} + 2.$$

Dowód powyższego twierdzenia zamieszczony jest w pracy [2]. ■

Podobnie, jak wyżej opisaliśmy, postępujemy gdy rozpatrywana operacja O_i^k jest operacją frezarską lub pomiarową.

6. Algorytm sterowania magazynami maszyny. W rozdziale tym przedstawimy algorytm wyznaczania numeru otworu, w odpowiednim magazynie, z którego pobierane jest narzędzie do wykonania operacji. Ponadto, wyznaczane są zbiory narzędzi usuwanych i wstawianych oraz numer operacji w czasie trwania której należy dokonać wymiany lub dublowania, a także ilość narzędzi każdego typu potrzebnych do obróbki wszystkich detali.

Zakładamy, że centrum pracuje w systemie zmianowym. W wyniku działania algorytmu **AHZ** wyznaczone zostały uporządkowane zbiory zadań H_i , ($i = 1, 2, \dots, l_{max}$) do wykonania w poszczególnych dniach, gdzie l_{max} jest numerem ostatniego dnia pracy centrum. Ponadto, niech d będzie liczbą otworów w każdym magazynie.

Najpierw przedstawimy procedurę **NR_OTWORU** wyznaczającą, zgodnie z zasadą podaną w poprzednim rozdziale, numer otworu z którego pobierane jest narzędzie do wykonywania ustalonej operacji, a następnie algorytm sterowania magazynami. Dla uproszczenia zapisu będziemy korzystali z elementów języka Pascal.

procedure NR_OTWORU (O_v^j);

{ Wyznaczanie narzędzia (nr otworu w magazynie) do wykonywania operacji O_v^j . Typ operacji (tokarska, frezarska, pomiarowa) określa magazyn, z którego należy pobrać odpowiednie narzędzie. }

begin

min := maxint; $z_v^j := 0$;

repeat

for $i := 1$ **to** d **do**

if ($s_i = n_v^j$) **and** ($c_i \geq p_v^j$) **then**

if min > $c_i - p_v^j$ **then**

begin

{ Wybieramy narzędzie z i - tego otworu }

min := $c_i - p_v^j$; $z_v^j := i$

end;

if min = maxint **then** { Brak odpowiedniego narzędzia }

if (są wolne otwory) **then** wstaw narzędzie **else** wymiana

until $z_v^j \neq 0$

end NR_OTWORU;

Opis czynności wstawiania i wymiany narzędzi w magazynie zamieściliśmy w poprzednim rozdziale.

Obecnie przedstawimy algorytm sterowania magazynami narzędzi podczas procesu obróbki detali.

```

procedure AMG;
{Przydzielanie narzędzi do wykonywania operacji}
begin
  for  $i:=1$  to  $l_{max}$  do                                {Kolejny dzień pracy}
  for each  $J_l \in H_i$  do                                {Zadanie}
  for each  $O_l^k \in J_l$  do                                {Operacja}
  case  $O_l^k$  of                                           { $O_l^k$  określa typ operacji}
    TOKARSKA: NR_OTWORU ( $O_l^k$ );
    FREZARSKA: NR_OTWORU ( $O_l^k$ );
    POMIAROWA: NR_OTWORU ( $O_l^k$ )
  end AMG;

```

Przedstawiony wyżej algorytm wyznacza narzędzia (numery otworów), do wykonywanych przez maszynę operacji oraz określa pewne przybliżone rozwiązanie dla problemu minimalizacji liczby użytych narzędzi. Jego złożoność obliczeniowa jest równa $O(d * lop)$, gdzie lop jest liczbą wszystkich operacji, a d liczbą otworów w magazynie.

Dla jasności opisu przyjęliśmy, że w trakcie czynności wstawiania operator może wstawić lub usunąć tylko jedno narzędzie. Poniżej, opiszemy metodę umożliwiającą usuwanie i wstawianie jednocześnie wielu narzędzi.

Przed rozpoczęciem wykonywania pierwszej, w danym dniu, operacji operator umieszcza w otworach magazynów narzędzia, które powinny być wstawione aż do momentu pierwszej wymiany. W trakcie tej wymiany usuwamy wszystkie niepotrzebne już narzędzia oraz we wszystkich wolnych otworach umieszczamy narzędzia, które mają tam być wstawione w trakcie następnych wymian. Tak więc, po pierwszej wymianie, wszystkie otwory magazynu są zajęte i każde z narzędzi będzie używane. Jeśli zajdzie konieczność dokonania następnej wymiany, wówczas postępujemy tak jak wyżej opisaliśmy. Podczas ostatniej w danym dniu wymiany, jeśli pozostaną wolne otwory, wykonujemy dublowanie narzędzi. Wymian dokonujemy w czasie trwania operacji poprzedzającej tę, do której wykonywania brak jest narzędzia w magazynie. Modyfikacja algorytmu AMG pozwalająca na ustalenie zbiorów usuwanych i wstawianych narzędzi, a także liczby potrzebnych narzędzi, jest prosta i nie wpływa na złożoność obliczeniową algorytmu. Dokładny opis omawianych w tym rozdziale algorytmów zamieszczony jest w pracy [1].

Czynności wstawiania, wymiany i dublowania narzędzi mogą być zrealizowane przez algorytmy, których złożoność obliczeniowa jest stała, tj. $O(1)$ i zależy tylko od liczby otworów w magazynie.

Algorytm **AMG** można stosować, bez dodatkowych modyfikacji, także w przypadku, gdy centrum pracuje w systemie ciągłym.

W dalszej części tego rozdziału zilustrujemy działanie algorytmu **AMG** na przykładzie zadania produkcyjnego przedstawionego w Rozdziale 4, wprowadzając dodatkowe dane.

Zakładamy, że zadania ze zbioru J są ciągami ponumerowanych operacji: $J_1 = (1, 2, 3)$, $J_2 = (1, 2, 4, 3)$, $J_3 = (4, 3, 1, 2)$, $J_4 = (2, 1)$, $J_5 = (1, 2, 3)$, $J_6 = (1)$, $J_7 = (2, 1)$, $J_8 = (1)$, $J_9 = (1)$, $J_{10} = (3, 1, 2)$. W Tabeli 2 podane są szczegółowe dane dotyczące operacji. Jeśli operacja o numerze i należy do zadania J_j , (tzn. $O_j^i \in J_j$), to na przecięciu i -tego wiersza oraz j -tej kolumny tabeli umieszczona jest trójka $(\alpha_j^i, p_j^i, n_j^i)$, gdzie $\alpha_j^i \in \{T, F, P\}$ określa typ operacji (T- tokarska, F- frezarska, P- pomiarowa), p_j^i jest czasem wykonania, a n_j^i numerem typu narzędzia (w odpowiednim magazynie), którym należy wykonać operację O_j^i . Ponieważ dwie ostatnie pozycje w każdej trójce są liczbami jednocyfrowymi, więc dla jasności zapisu opuszczono przecinki między kolejnymi elementami.

TABELA 2

nr oper.	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}
1	T12	P22	T11	F33	P32	T21	T23	T32	T22	F23
2	F13	F13	T12	T23	T34		F22			P32
3	P11	T14	F32		T32					T23
4		F21	P31							

Tabela 3 zawiera dane dotyczące narzędzi.

TABELA 3
Żywotności narzędzi

narz. tokarskie		narz. frezarskie		narz. pomiarowe	
nr typu	żyw.	nr typu	żyw.	nr typu	żyw.
1	2	1	2	1	3
2	3	2	3	2	3
3	2	3	3		
4	3				

Wszystkie czasy podajemy w godzinach. Przyjeliśmy ponadto, że magazyny narzędzi tokarskich i frezarskich mają po trzy otwory, a magazyn narzędzi pomiarowych- dwa.

Stosując algorytm **AHZ** wyznaczone zostały, na każdy dzień pracy, uporządkowane zbiory zadań do wykonania $H_1 = (10)$, $H_2 = (1, 7, 9)$, $H_3 = (5)$,

$H_4 = (2, 3)$, $H_5 = (8)$, $H_6 = (6)$, $H_7 = (4)$. Wyniki obliczeń algorytmu AMG przedstawione są w Tabelach 4 i 5. W Tabeli 4, na przecięciu i -tego wiersza oraz j -tej kolumny umieszczony jest numer otworu w magazynie, z którego maszyna pobiera narzędzie do wykonania operacji O_j^i , tj. wartość zmiennej z_j^i .

TABELA 4

nr oper.	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}
1	3	2	2	3	2	3	2	3	3	3
2	3	3	1	3	3		2			2
3	2	3	1		2					3
4		2	1							

TABELA 5

otwór $\mapsto$ dzień	narz. tok.			narz. frez.			narz. pom.	
	1	2	3	1	2	3	1	2
1		3*	3		3*	3	2*	2
2		2*	3	2	3*	2	3	1
3		2*	2	4			2	2
4		2	1	4	2	1	3	2*
5			2*	2				
6			2*	1				
7			3*	3		3*	3	

Z kolei, Tabela 5 zawiera numery typów narzędzi, które należy umieścić w otworach magazynu, w każdym dniu pracy maszyny, przed rozpoczęciem wykonywania pierwszego zadania. Narzędzia te wystarczają do wykonanie wszystkich operacji przypadających na ten dzień pracy i dlatego ani razu nie dokonano wymiany. Gwiazdką oznaczone są narzędzia dublujące, wykorzystywane w przypadku awarii, wstawiane na początku zmiany.

Dla ciągłej pracy centrum wyniki zamieszczone są w Tabeli 6, 7 i 8. Stosując algorytm AHC, zamieszczony w Rozdziale 4, ustalona została kolejność wykonywania zadań $H = (10, 5, 1, 7, 2, 3, 6, 9, 4, 8)$. Wyznaczone algorytmem AMG numery otworów, z których maszyna pobiera narzędzia do obróbki, zamieszczone są w Tabeli 6.

Tabela 7 zawiera numery typów narzędzi wstawianych oraz usuwanych do/z magazynu w trakcie wymiany. W kolumnie oznaczonej literami NWP

TABELA 6

nr	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}
oper.										
1	1	2	3	3	1	1	2	2	1	3
2	3	1	2	3	2		2			2
3	1	3	2		1					3
4		1	2							

TABELA 7

nr otworu	NWP	wymiana 1 usuń wstaw		wymiana 2 usuń wstaw		wymiana 3 usun wstaw	
wymiany w magazynie narzędzi tokarskich							
1	2	2	2	2	1	1	2
2	4	4	3	3	2	2	2
3	3	3	4	4	1	1	3
wymiany w magazynie narzędzi frezarskich							
1	3	3	1				
2	2	2	2				
3	3	3	3				
wymiany w magazynie narzędzi pomiarowych							
1	2	2	1	1	1		
2	2	2	2	2	1		

podajemy numery typów narzędzi wstawianych do magazynu przed rozpoczęciem wykonywania pierwszego zadania.

Numer zadania i numer operacji, w trakcie wykonywania której należy dokonać wymiany narzędzi, podajemy w Tabeli 8.

TABELA 8

numer wymiany	numer zadania	numer operacji
magazyn narzędzi tokarskich		
1	5	3
2	3	3
3	6	1
magazyn narzędzi frezarskich		
1	2	2
magazyn narzędzi pomiarowych		
1	1	2
2	2	3

7. Eksperymenty obliczeniowe. Algorytmy harmonogramowania procesu obróbki detali **AHZ** i **AHC**, przedstawione w Rozdziale 4, zostały zaprogramowane w języku Turbo Pascal i przetestowane na przykładach wziętych z praktyki. Obliczony, dla tych przykładów, współczynnik wykorzystania maszyny K był dwukrotnie większy niż dla harmonogramów układanych ręcznie. Algorytmy te były także testowane na wielu losowo wygenerowanych przykładach. Przy ustalonej liczbie zadań n , każdy przykład jest ciągiem danych opisujących kolejne zadanie. Dla zadania $J_i \in J$, wszystkie parametry są realizacją zmiennej losowej o rozkładzie jednostajnym na odcinku: $[10,30]$ – dla czasu załadunku z_i ; $[5,20]$ – dla czasu wyładunku w_i ; $[30,240]$ – dla czasu wykonania p_i ; $[0,1440]$ – dla wagi łuków $f_{ij}, ((J_i, J_j) \in RT)$ oraz $[0,4000]$ – dla najwcześniejszego terminu rozpoczęcia załadunku r_i . Przedziały z których losowano wartości zmiennych losowych wyznaczono tak, aby wartości rzeczywistych danych, dla których prowadziliśmy obliczenia, należały do tych przedziałów. Przyjęto także, czas rozruchu maszyny $Sm = 15$ oraz czas zatrzymania $Zm = 10$. Każdy przykład był liczony dla jedno i dwu zmianowej (dz równe odpowiednio 480 lub 960, tj. 8 i 16 godz.) oraz ciągłej pracy centrum. Oprócz współczynnika wykorzystania maszyny K liczono także wartość ilorazu:

$$K_{OS} = \frac{F_{LB}}{F_{AH}},$$

gdzie F_{AH} jest terminem zakończenia obróbki przez automat wszystkich detali ze zbioru J wyznaczonym odpowiednio przez algorytm **AHZ** lub **AHC**, natomiast

$$F_{LB} = \begin{cases} s + (k-1)(mi + Zm) + mi, & \text{dla pracy zmianowej,} \\ s + Sm, & \text{dla pracy ciągłej,} \end{cases}$$

gdzie

$$s = \sum_{i=1}^n p_i, \quad k = \min \left\{ j : s \leq \sum_{i=1}^j dz_i \right\} \quad \text{i} \quad mi = \min \{ \min \{ z_i : J_i \in J \}, Sm \}.$$

Łatwo wykazać, że wartość wyrażenia F_{LB} jest dolnym ograniczeniem czasu zakończenia wykonywania zadań ze zbioru J .

Wyniki obliczeń przedstawione są w Tabelach 9, 10 i 11. W Tabeli 9 zamieszczone są wyniki dotyczące przykładów, dla których najwcześniejsze terminy rozpoczęcia załadunku zadań są zerami oraz zbiór łuków grafu pierwszeństwa G jest pusty.

Liczba zmian równa 3 oznacza pracę ciągłą.

Tabela 10 zawiera wyniki obliczeniowe przykładów, dla których liczba łuków grafu pierwszeństwa jest równa $[n/2]$ natomiast, wagi łuków oraz najwcześniejsze terminy rozpoczęcia wykonywania zadań są zerami. Pary wierzchołków będące początkami i końcami łuku są także generowane lo-

TABELA 9

liczba zadań	liczba zmian	Wsp. wyk. masz. K			Oszacowanie K_{OS}		
		min.	mediana	max.	min.	mediana	max.
10	1	.87	.90	.92	.93	.95	.97
	2	.93	.94	.95	.96	.97	.98
	3	.98	.98	.98	.99	.99	.99
30	1	.90	.90	.90	.95	.95	.96
	2	.95	.95	.96	.97	.98	.99
	3	.99	.99	.99	.99	.99	.99
60	1	.91	.91	.93	.96	.96	.96
	2	.95	.96	.96	.98	.98	.99
	3	.99	.99	.99	.99	.99	.99

TABELA 10

liczba zadań	liczba zmian	Wsp. wyk. masz. K			Oszacowanie K_{OS}		
		min.	mediana	max.	min.	mediana	max.
10	1	.82	.89	.91	.88	.93	.96
	2	.91	.93	.96	.94	.96	.99
	3	.98	.98	.98	.99	.99	.99
30	1	.87	.88	.89	.92	.94	.94
	2	.93	.94	.96	.95	.97	.98
	3	.99	.99	.99	.99	.99	.99
60	1	.89	.92	.94	.92	.93	.96
	2	.97	.97	.98	.99	.99	.99
	3	.99	.99	.99	.99	.99	.99

TABELA 11

liczba zadań	liczba zmian	Wsp. wyk. masz. K			Oszacowanie K_{OS}		
		min.	mediana	max.	min.	mediana	max.
10	1	.77	.85	.87	.88	.90	.92
	2	.49	.54	.68	.52	.51	.71
	3	.39	.54	.64	.39	.54	.64
30	1	.84	.87	.89	.89	.92	.94
	2	.91	.94	.95	.94	.96	.97
	3	.99	.99	.99	.99	.99	.99
60	1	.88	.90	.91	.93	.95	.96
	2	.94	.95	.95	.96	.98	.98
	3	.99	.99	.99	.99	.99	.99

sowo. Jeżeli w trakcie tworzenia grafu pierwszeństwa pojawi się cykl, wówczas generowanie przykładu rozpoczynamy od początku.

Wyniki zamieszczone w Tabeli 11 dotyczą przykładów, generowanych tak samo jak te z Tabeli 10 z tą różnicą, że modyfikowano je generując losowo wagę łuku f_{ij} , $((J_i, J_j) \in E)$ oraz najwcześniejszy termin rozpoczęcia ładowania detalu. Liczba zadań, dla których najwcześniejszy termin rozpoczęcia $r_i \neq 0$, jest równa $\lceil n/4 \rceil$.

Podobnie jak wyżej, generując losowo przykłady, testowaliśmy algorytm AMG sterowania magazynami maszyny. Nie przedstawiamy otrzymanych wyników, bowiem są one podobne do tych, które zamieściliśmy w Tabeli 8, 9 i 10. Czas obliczeń, każdego przykładu, był nie większy niż 1 sekunda.

Uzyskane wyniki w pełni potwierdzają słuszność wybranej metody konstrukcji algorytmów rozwiązania problemu optymalnej pracy centrum.

LITERATURA

- [1] A. Adrabiński, A. Danes, P. Grabiec, J. Grabowski, J. Kucharczyk, M. M. Sysło, S. Sywak, M. Wodecki, *Oprogramowanie dla centrum obróbczego tokarsko-frezarskiego typu KCT według zadania technicznego dostarczonego przez RAFAMET w Kuźni Raciborskiej*, Wrocław, 1987.
- [2] M.R. Garey, D. S. Johnson, *Computers and intractability. A guide to the theory of NP-completeness*, W.H. Freeman and Co., San Francisco, 1979.
- [3] R. J. Jackson, *Scheduling a production line to minimize maximum tardiness*. Research Raport 43, Management Sci. Res. Proj., University of California, Los Angeles, 1955.
- [4] D. S. Johnson, A. Demers, J. D. Ullman, M. R. Garey, R. L. Graham, *Worst-case performance bound for simple one-dimensional packing algorithms*, SIAM, J.Comput., 3, 4 (1974).

Abstract

A sequencing problem for optimal job processing by the turning-lathe.

The paper is dealing with a computer-aided system for a turning-lathe machine. It is divided into two parts. The first one considers the sequencing problem for a job processing by the machine. It describes the problem and then presents a mathematical model and an algorithm for the minimalization of a processing time. We also prove that the problem is NP-hard. The second part, on the other side, considers the problem of the tool replacement by the turning-lathe. An algorithm for finding an optimal sequence of tool replacement intervals in the tool-store of the machine is presented. Extensive computational experiments for problems with up to 60 jobs show a high efficiency of the system.