

Waldemar KACZMARCZYK  
Akademia Górniczo-Hutnicza w Krakowie

## STEROWANIE OPERATYWNE W PRZEPLYWOWYM SYSTEMIE PRODUKCYJNYM<sup>1</sup>

Streszczenie: Dla zakładu produkcji farb opracowany został heurystyczny algorytm harmonogramowania produkcji wykorzystujący schemat obliczeń programowania dynamicznego. Produkty wytwarzane są w porcjach, które mają stałe objętości i uszeregowane są w kampanie. System produkcyjny składa się z wielu stadiów, a w każdym z nich może być wiele równoległych dowolnych maszyn (hybrid flow shop). Dzielenie i mieszanie porcji produktów sprawia, iż ograniczenia kolejnościowe operacji mają złożony charakter.

## PRODUCTION SCHEDULING IN A HYBRID FLOW SHOP

Summary: A heuristic scheduling algorithm for a paint production plant based on an application of dynamic programming method is proposed. The plant products are produced in constant volume batches scheduled in campaigns. The plant consists of several processing stages and each stage has a few different parallel machines (hybrid flow shop). Precedence relations can be complex because the batches can be divided and mixed.

## ABLAUFPLANUNG IN EINEM MEHRSTUFIGEN PRODUKTIONS-SYSTEM

Zusammenfassung: Für ein Farben produzierendes Betrieb ist ein Ablaufplanungsverfahren vorgeschlagen worden, das das Schema der Dynamischen Programmierung verwendet. Die Produkte werden in Chargen hergestellt, die einen festen Volumen haben und in Kampagnen eingeordnet werden. Das Produktionssystem besteht aus mehreren Stufen und in jeder Stufe kann es mehrere parallele verschiedene Maschinen geben (hybrid flow shop). Wegen des Teilens und des Mischens der Chargen nehmen die Reihenfolgerestriktionen komplexe Formen an.

<sup>1</sup>Praca częściowo finansowana z funduszu badań własnych 10.200.10.

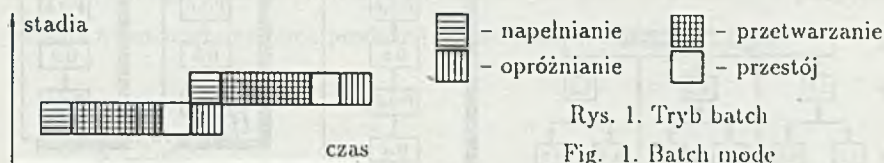
## 1. Wprowadzenie

W rozważanym zakładzie produkowanych jest na przemian na tych samych maszynach wiele różnych płynnych produktów. Produkty te przetwarzane są zawsze w porcjach (tzw. szarżach) o z góry określonej objętości, dzięki czemu można uznać, że rozważany system produkcyjny jest dyskretny. System ten składa się z wielu stadiów, a w każdym z nich znajduje się wiele maszyn wykonujących z różną prędkością takie same operacje. Produkty nie muszą być przetwarzane we wszystkich stadiach, ale ich kolejność jest dla wszystkich produktów zawsze taka sama.

Zadanie sterowania operatywnego takim procesem produkcyjnym jest bardzo podobne do problemu harmonogramowania operacji określanego jako *hybrid (flexible) flow shop*. Problem ten można sobie wyobrazić jako kombinację problemów szeregowania operacji na maszynach równoległych (*parallel shop*) i w przepływowym systemie z tylko jedną maszyną w każdym stadium (*flow shop*).

Rozwiązań optymalnych tego typu problemów można szukać wykorzystując programowanie dynamiczne. Jest to jednak sensowne tylko przy niewielkiej liczbie maszyn i operacji, gdyż w przeciwnym razie czas obliczeń będzie niezwykle długi. Więcej danych na ten temat można znaleźć np. w pracy Brah, Hunsucker [1]. W praktyce muszą więc być stosowane metody heurystyczne. Dla problemu *hybrid flow shop* z dowolną liczbą stadiów nie ma zbyt wielu różnych propozycji. Wittrock [8] skonstruował algorytm heurystyczny aproksymujący algorytmy oparte na programowaniu dynamicznym. Sawik [5] przedstawił iteracyjną heurystykę harmonogramowania wstecznego, tzn. od ostatniego stadium i od końca okresu. U Kaczmarczyka [4] i Sawika [6] znaleźć można proste heurystyki oparte na dyspozytorskich regułach harmonogramowania. We wszystkich wyżej wymienionych pracach przyjęto, że maszyny wewnątrz każdego ze stadiów są identyczne. W niniejszej pracy natomiast przedstawiona zostanie metoda rozszerzająca zastosowanie algorytmów wykorzystujących reguły dyspozytorskie na modele z dowolnymi maszynami wewnątrz stadiów i złożonymi ograniczeniami kolejnościowymi operacji.

W literaturze stosowane jest zazwyczaj słownictwo przejęte z przemysłu maszynowego. W tej pracy używane będą natomiast nazwy typowe dla przemysłu chemicznego: w miejsce maszyn pojawią się więc *aparaty*, części zastąpione zostaną przez *szarże*, a za-



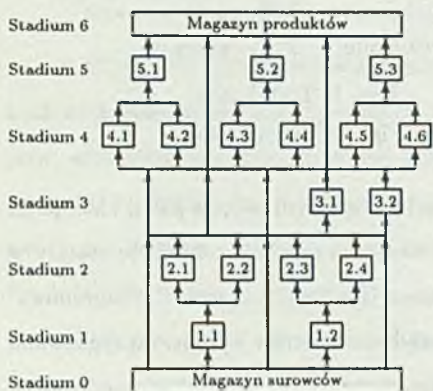
miast o partiach części będzie mowa o **kampaniach**. Przy czym pojęcia partii i kampanii są wprawdzie podobne, ale nie identyczne. Na użytek tego artykułu pominięte zostały w opisie modelu zakładu pewne jego cechy. Nie mają one istotnego znaczenia dla zaproponowanej tu metody harmonogramowania, a komplikowałyby nadmiernie jej opis. Zrezygnowano m.in. z uwzględniania faktu pracy niektórych aparatów w tzw. trybie *conti*, równoczesnego przetwarzania wielu szarż w jednym aparacie, a także czyszczenia aparatów przy zmianie przetwarzanego produktu.

## 2. Opis problemu

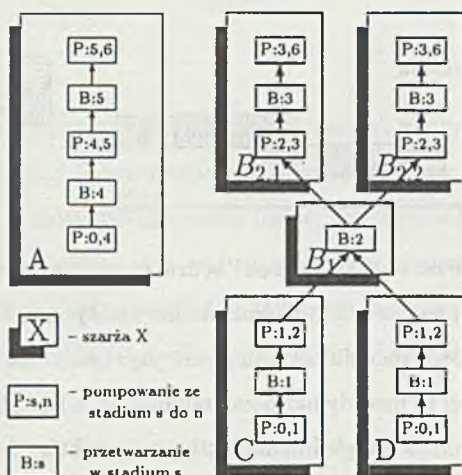
Na rys. 1 przedstawiony został za pomocą diagramu Gantt'a przebieg produkcji wyrobów płynnych w tzw. trybie **batch**, charakterystycznym dla rozważanego zakładu. Jedyną istotną różnicą między takim procesem produkcyjnym a produkcją jednostkową znaną z przemysłu maszynowego jest fakt, iż podczas "transportu" (przepompowywania) szarży między dwoma aparatami oba te aparaty są w tę całą operację zaangażowane.

Schemat systemu produkcyjnego przedstawiony został na rys. 2. Aparaty w niektórych stadiach są w istocie magazynami, ale ponieważ produkty poddawane są w nich różnym zabiegom, traktowane one tu będą tak jak prawdziwe aparaty. Jeśli pewien produkt musi być przetwarzany w danym stadium, to z reguły do wyboru stoi kilka aparatów pracujących z różnymi prędkościami. Te różnice prędkości zależą w niektórych stadiach od przetwarzanego akurat produktu. W niektórych stadiach przetwarzanie pewnych produktów możliwe jest tylko w niektórych ich aparatach.

Na rys. 2 strzałki pokazują możliwe drogi i kierunki transportu produktów. Jak widać, nie zawsze możliwy jest transport porcji produktu pomiędzy dwoma dowolnymi aparatami dwóch sąsiednich stadiów. Zakłada się, że dopuszczalność przetwarzania produktów przez poszczególne aparaty i możliwości transportu między aparatami są sko-



Rys. 2. System produkcyjny  
Fig. 2. Production system



Rys. 3. Receptury produktów  
Fig. 3. Recipes for products

relowane. Jeśli na przykład pewien produkt nie może być przetwarzany ani w aparacie 2.1, ani w aparacie 2.2 (rys. 2), to nie może on być również przetwarzany w aparacie 1.1, gdyż nie można byłoby go nigdzie dalej z tego aparatu przepompować. Ponadto przyjmuje się, że wydajność środków transportu nie stanowi żadnego ograniczenia i dlatego nie będzie ona tu dalej uwzględniana.

Na rys. 3 przedstawione są receptury przykładowych produktów: *A* i *B*. Produkt *A* nie ma żadnych półproduktów, dzięki czemu ograniczenia kolejnościowe jego operacji mają prosty (liniowy) charakter. Natomiast produkt *B* ma dwa półprodukty: *C* i *D*. Po ich zmieszaniu powstaje szarża  $B_1$  o podwójnej niejako objętości. Do dalszego przetwarzania musi ona zostać podzielona na dwie mniejsze szarże:  $B_{2,1}$  i  $B_{2,2}$ . Ograniczenia kolejnościowe operacji, które trzeba wykonać przy realizacji receptury jednej "podwójnej" szarży produktu *B*, mają więc charakter złożony (nieliniowy).

W ramach programu produkcyjnego należy wykonać zazwyczaj wiele szarż każdego produktu. Wszystkie szarże danego produktu muszą zostać wykonane w ramach jednej kampanii. Oznacza to, że jeśli na danym aparacie ma być przetwarzanych wiele szarż pewnego produktu, to muszą one być przetwarzane na tym aparacie po kolei, bez przerw na przetwarzanie innych produktów. Szarże z jednej kampanii mogą być jednak przetwarzane w wybranych stadiach równolegle na kilku aparatach. Określenie w każdym stadium liczby

aparatów zaangażowanych w każdą z kampanii jest jednym z istotnych podproblemów zadania harmonogramowania produkcji w rozważanym zakładzie.

### 3. Opis algorytmu

Jak już wcześniej wspomniano, problem flow shop może być postrzegany jako specyficzny przypadek problemu hybrid flow shop. Opis proponowanego w niniejszej pracy algorytmu wygodnie jest rozpocząć od przedstawienia metody rozwiązywania problemu flow shop znanej z pracy Gupty [2]. W każdej iteracji tej heurystyki harmonogram częściowy, powstały w poprzednich iteracjach, "wydłużony" zostaje przez dodanie na jego końcu planu wykonania jednej z części, które nie zostały w nim jeszcze uwzględnione. Do dołączenia wybierana jest każdorazowo ta część, dla której przestoje powstające między jej operacjami a dotychczasowym harmonogramem są najmniejsze.

W problemie flow shop zakłada się, że w każdym stadium jest tylko jedna maszyna. Chcąc rozszerzyć zastosowanie heurystyki Gupty na problem hybrid flow shop, określić trzeba sposób i kryteria wyboru maszyn we wszystkich stadiach. Kaczmarczyk [4] i Sawik [6] przedstawili dla modelu z identycznymi równoległymi maszynami algorytmy, w których wybierana jest każdorazowo ta najwcześniej dostępna.

Poniższy algorytm przeznaczony jest dla problemu z dowolnymi maszynami w stadiach i ze złożonymi ograniczeniami kolejnościowymi operacji. Wykorzystuje on schemat obliczeń znany z programowania dynamicznego do przeglądu zbioru możliwych kombinacji aparatów i dokonuje wyboru jednej z nich, uwzględniając przy tym różne kryteria. Poniżej przedstawiony jest ogólny schemat proponowanego algorytmu, zasadniczo zbieżny z przedstawionym wyżej opisem heurystyki Gupty [2]. "Części" zostały w nim zastąpione jedynie przez "kampanie".

- 
- $\sigma$  – aktualny harmonogram częściowy,
- $\sigma\sigma_p$  – harmonogram  $\sigma$  rozszerzony o dołączony na jego końcu harmonogram  $\sigma_p$  kampanii produktu  $p$ ,
- $P$  – zbiór wszystkich (typów) produktów,
- $\bar{P}$  – zbiór produktów nie włączonych jeszcze do harmonogramu  $\sigma$ .
-

## Ogólny schemat algorytmu harmonogramowania

**Krok 1:** Przyjmij  $\sigma = \emptyset, \overline{P} = P$ .

**Krok 2:** [Wybór aparatów dla kampanii] Określ najkorzystniejszy ("optymalny") harmonogram  $\sigma_p^\circ$  dla kampanii każdego produktu  $p \in \overline{P}$  na końcu harmonogramu  $\sigma$ :  $\sigma_p^\circ = \arg \text{opt}_{\sigma_p}(\sigma \sigma_p)$ .

**Krok 3:** [Wybór kampanii] Znajdź produkt  $p^\circ$ , którego kampania najlepiej daje się dołączyć do harmonogramu  $\sigma$ :  $p^\circ = \arg \text{opt}_{p \in \overline{P}}(\sigma \sigma_p^\circ)$ .

**Krok 4:** Dołącz harmonogram kampanii wybranego produktu  $p^\circ$  do harmonogramu  $\sigma$ :  $\sigma = \sigma \sigma_{p^\circ}^\circ, \overline{P} = \overline{P} \setminus \{p^\circ\}$ .

**Krok 5:** Jeśli  $\overline{P} = \emptyset$  to STOP, jeśli nie, to wróć do kroku 2.

Krok 2. i 3. algorytmu zostaną jeszcze szczegółowiej omówione w następnych podrozdziałach. Tu konieczne jest jedynie pewne wyjaśnienie odnośnie kroku 2. Określenie harmonogramu dla kampanii jest jednoznaczne z określeniem harmonogramów kolejno dla wszystkich szarż tej kampanii. A określenie harmonogramu dla jednej szarży sprowadza się do wyboru aparatów, w których będzie ona przetwarzana. W rozdziale 3.1 opisany jest właśnie wybór aparatów dla pojedynczej szarży. W rozdziale 3.2 natomiast przedstawiony został sposób określania liczby aparatów w każdym ze stadiów zaangażowanych w daną kampanię. W rozdziale 3.3 opisany jest sposób wyboru produktu (krok 3.), którego kampania dołączona zostanie do harmonogramu częściowego w aktualnej iteracji algorytmu. Wszystkie wymienione w następnych podrozdziałach kryteria (priorytety) opisane są z braku miejsca jedynie werbalnie. Nie powinno to jednak stanowić poważnego utrudnienia dla czytelnika, gdyż są to wszystko kryteria znane z wielu innych prac.

### 3.1. Wybór aparatów dla jednej szarży

W proponowanym algorytmie ogólne zadanie harmonogramowania rozbite zostało na dwa podproblemy. Pierwszym z nich jest określenie najkorzystniejszego harmonogramu dla jednej całej kampanii na końcu aktualnego harmonogramu częściowego. Określając harmonogram dla pojedynczej szarży z takiej kampanii, należy brać pod uwagę zarówno kryteria optymalizacji lokalnej, jak i pewne inne kryteria, gwarantujące, że suma takich lokalnie dobrych rozwiązań cząstkowych da dobre rozwiązanie całkowite. Przyjęto więc,

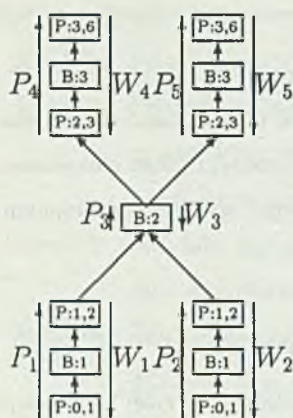
iż należy wybierać takie aparaty, aby:

- S1. przetwarzanie szarży zostało jak najwcześniej zakończone,
- S2. powstały jak najmniejsze dodatkowe przestoje aparatów (suma czasu oczekiwania aparatu na szarżę, jeśli był on wolny przed przybyciem szarży, i czasu blokowania aparatu przez szarżę w oczekiwaniu na zwolnienie jakiegoś aparatu w następnym stadium),
- S3. czasy wykonywania operacji były jak najkrótsze,
- S4. wybrane aparaty miały jak najmniejsze pozostałe obciążenie.

Ostatnie trzy kryteria odnoszą się tylko do poszczególnych stadiów, a nie całej kombinacji aparatów. Chcąc uzyskać ogólną ocenę jakiejś wybranej kombinacji, należy najpierw obliczyć dla tych kryteriów wartości średnie ze wszystkich stadiów, a następnie wyznaczyć ważoną sumę wszystkich czterech kryteriów. Wagi muszą zostać wyznaczone na drodze eksperymentów obliczeniowych.

Liczba wszystkich możliwych kombinacji aparatów może być bardzo duża. Dlatego też zadanie wyboru aparatów zostało potraktowane jak wieloetapowy problem decyzyjny i rozwiązane za pomocą programowania dynamicznego. Z każdym etapem tego problemu związany jest wybór aparatu w jednym stadium. Przeszukiwanie zbioru rozwiązań rozpoczyna się nie od ostatniego stadium (etapu), jak to jest zazwyczaj w programowaniu dynamicznym przyjmowane, lecz od pierwszego. Tylko w ten sposób możliwe jest określenie kształtu wcześniejszej części harmonogramu i wyznaczenie wartości wszystkich kryteriów, a dokładnie rzecz biorąc, ich średnich ze wszystkich wcześniejszych stadiów. W trakcie przeszukiwania zbioru rozwiązań pamiętać trzeba o ograniczonych możliwościach transportu między stadiami. Ostateczny wybór aparatów odbywa się, oczywiście, w odwrotnej kolejności, tzn. od ostatniego do pierwszego stadium. Należy zaznaczyć, iż stosowane tu kryteria wyboru nie spełniają tzw. warunku Markowa, przez co programowanie dynamiczne nie może dać rozwiązań optymalnych tego podproblemu. Nie ma to jednak większego znaczenia, gdyż i tak cały przedstawiony tu algorytm jest jedynie heurystyką.

Dotychczas przedstawiona metoda wyboru aparatów może być zastosowana jedynie do produktów o liniowych ograniczeniach kolejnościowych operacji. Na rys. 4 przedsta-



Rys. 4. Programowanie dynamiczne dla produktów ze złożonymi ograniczeniami kolejnościowymi operacji

Fig. 4. Dynamic programming for products with complex precedence relation

Cztery dopuszczalne kolejności obliczeń:

$$1. P_1 W_1 P_2 W_2 P_3 P_4 W_4 W_3 P_5 W_5$$

$$2. P_2 W_2 P_1 W_1 P_3 P_4 W_4 W_3 P_5 W_5$$

$$3. P_1 W_1 P_2 W_2 P_3 P_5 W_5 W_3 P_4 W_4$$

$$4. P_2 W_2 P_1 W_1 P_3 P_5 W_5 W_3 P_4 W_4$$

$P_i$  - przeszukiwanie zbioru rozwiązań odcinka  $i$ ,

$W_i$  - ostateczny wybór aparatów na odcinku  $i$ .

wiono schematycznie sposób wyboru kombinacji aparatów dla szarż o złożonych ograniczeniach kolejnościowych. Cała receptura została podzielona na liniowe części (odcinki). Dla każdej z nich programowanie dynamiczne musi być przeprowadzone osobno. Ich kolejność nie jest obojętna, gdyż najpierw muszą zostać przeliczone te zawierające operacje wcześniejsze. Jedyne kolejność odcinków równoległych jest dowolna. Przyjęto ponadto zasadę, że odcinki równoległe przeliczane będą w kolejności od tych z największą sumą czasów wykonywania operacji do tych z najmniejszą. Wybór aparatów dla wcześniej przeliczanych części receptury ogranicza możliwość wyboru dla następnych, gdyż wybrany jest już w tym momencie aparat, w którym odbywać się będzie mieszanie bądź dzielenie szarży.

### 3.2. Wybór aparatów dla kampanii

Określenie harmonogramu dla całej kampanii polega na wyznaczeniu harmonogramów kolejno dla poszczególnych szarż tej kampanii. Należy przy tym zadbać o to, by liczba aparatów zaangażowanych w daną kampanię w każdym ze stadiów nie była ani za duża, ani za mała. W niniejszym algorytmie zakłada się, iż dla każdego produktu w jednym ze stadiów maksymalna liczba aparatów określona zostanie przez eksperta z góry. Wybierając w tym stadium aparaty dla kolejnych szarż danej kampanii trzeba zważać, by nie przekroczyć tej liczby. W pozostałych stadiach nie ma żadnych sztywnych ogra-



niczeń, a sensowny wybór aparatów zapewniany jest poprzez odpowiednie wyważenie dwóch kryteriów:

S1. dążąc do jak najwcześniejszego zakończenia przetwarzania kolejnych szarż, preferowane są te najwcześniejsze aparaty,

S2. minimalizując przestoje, preferuje się te aparaty, które akurat od niedawna są wolne.

W efekcie w stadiach o długich czasach wykonywania operacji kryterium S1 sprawi, iż wykorzystanych zostanie wiele aparatów. W stadiach, w których czasy operacji są krótkie, kryterium S2 wymusi ograniczenie liczby zaangażowanych aparatów. Stadium, w którym maksymalna liczba aparatów została sztywno określona, wymusza zatem pewną określoną "szerokość" kampanii w innych stadiach, umożliwiając przy tym elastyczne dopasowywanie harmonogramów kampanii do aktualnego harmonogramu częściowego.

### 3.3. Wybór kampanii

Po określeniu "optymalnych" harmonogramów dla każdej z pozostałych do zaplanowania kampanii należy wybrać jedną z nich i dołączyć ją do aktualnego harmonogramu częściowego. Za najlepszą uznać należy tę, która:

K1. obciąża te najwcześniejsze aparaty,

K2. przyczynia się do powstania najmniejszych dodatkowych przestojów aparatów,

K3. obciąża aparaty o największym pozostałym obciążeniu,

K4. obciąża akurat te aparaty, które najszybciej ją przetwarzają,

K5. ma długi łączny czas przetwarzania (bez wliczania przestojów),

K6. stadia końcowe obciąża bardziej niż początkowe (podobnie jak w znanym algorytmie Johnsona dla problemu flow shop).

Całkowita ocena kampanii obliczana jest jako ważona suma wszystkich sześciu kryteriów. Wagi poszczególnych kryteriów mogą zostać wyznaczone eksperymentalnie.

## 4. Omówienie wyników

Za pomocą przedstawionego w tej pracy algorytmu obliczonych zostało dla rzeczywistych danych kilka harmonogramów. Trzy z nich opisane zostały w tablicy 1. Nie mogą one być, niestety, przyjęte za podstawę do obiektywnej oceny efektywności algorytmu.

Tablica 1

Lp.	Liczba			t <sup>†</sup> [min.]	e <sup>‡</sup> [%]
	kampanii	szarż	operacji		
1	36	155	10307	23	79
2	36	178	12034	28	81
3	36	210	16419	34	73

## Wyniki obliczeń

† czas obliczeń na IBM 386.

‡ stopień wykorzystania aparatów w najbardziej obciążonym stadium.

Szczegółowa analiza tych harmonogramów pozwoliła bowiem stwierdzić, że rozważany zakład ma bardzo niekorzystnie rozłożony potencjał produkcyjny, przez co niemożliwe jest w pełni efektywne wykorzystanie znajdujących się w nim aparatów. Subiektywna ocena harmonogramów pozwala jednak stwierdzić, iż harmonogramowanie kampanii (rozdz. 3.2) przebiega prawidłowo, a całe harmonogramy trudno byłoby ulepszyć manualnie. Przy horyzoncie planowania wynoszącym cztery tygodnie, czas obliczeń rzędu 30 min nie stanowi żadnego poważnego ograniczenia użyteczności algorytmu.

## Podsumowanie

W niniejszej pracy przedstawiony został problem sterowania operatywnego w dyskretnym przepływowym systemie produkcyjnym, w którym w wielu stadiach wytwarzanych jest na przemian na tych samych maszynach wiele różnych płynnych produktów. Od typowych problemów hybrid flow shop zagadnienie to różni się przede wszystkim występowaniem dowolnych równoległych maszyn w stadiach i złożonymi ograniczeniami kolejnościowymi operacji. Zaproponowany został również heurystyczny iteracyjny algorytm harmonogramowania produkcji w tym zakładzie. W każdej iteracji tego algorytmu harmonogram częściowy wydłużony zostaje o harmonogram kampanii jednego produktu. Do określania najkorzystniejszego harmonogramu pojedynczych szarż (porcji produktu) wykorzystano programowanie dynamiczne. Przedstawiono również charakterystykę kilku przykładowych harmonogramów obliczonych dla rzeczywistych danych.

## LITERATURA

- [1] Brah S.A., Hunsucker J.L.: Branch and bound algorithm for the flow shop with multiple processors. *European Journal of Operational Research*, vol. 51, 1991, pp. 88-99.

- [2] Gupta J.N.D.: Heuristic Algorithms for Multistage Flowshop Scheduling Problem. *AIIE Transactions*, vol. 4(1), 1972, pp. 11-18.
- [3] Johnson S.M.: Optimal two- and three-stage production schedules with setup-times included. *Naval Research Logistic Quarterly*, vol. 1, 1954, pp. 61-68.
- [4] Kaczmarczyk W.: Algorytmy sterowania operatywnego w przeplywowych systemach produkcyjnych. Praca magisterska, Akademia Górniczo-Hutnicza w Krakowie, 1990.
- [5] Sawik T.: Multilevel scheduling of multistage production with limited in-process inventory. *Journal of Operations Research Society*, vol. 38, 1987, pp. 651-664.
- [6] Sawik T.: A scheduling algorithm for flexible flow lines with limited intermediate buffers. *Applied Stochastic and Data Analysis*, vol. 9, 1993, pp. 127-138.
- [7] Sriskandarajah C., Sethi S.P.: Scheduling algorithms for flexible flowshops: Worst and average case performance. *European Journal of Operations Research*, vol. 43, 1989, pp. 143-160.
- [8] Wittrock R.J.: Scheduling Parallel Machines with Major and Minor Setup Times. *International Journal of Flexible Manufacturing Systems*, vol. 2, 1990, pp. 329-341.

Recenzent: Prof.dr hab.inż. Mirosław Zaborowski

Wplynęło do Redakcji do 30.04.1994 r.

### Abstract

This paper presents a new heuristic algorithm for a scheduling in a paint production plant. The products are produced in constant volume batches scheduled in campaigns. The plant consists of several of processing stages and each stage has a few different parallel machines. Each batch must be processed by at most one machine at each stage, but some products may skip some stages. Precedence relations can be complex because the batches can be divided or mixed. Similar scheduling problems are called 'hybrid flow shop'. The problem represents a combination of traditional flow shop in which there is only one machine at each stage, and a different parallel machine shop in which there is only one stage with several parallel machines.

The algorithm proposed in this paper is a campaign-by-campaign heuristics, where in every iteration a complete processing schedule is determined because a campaign consists of a lot of batches. The decisions in every iteration are made using a local optimization procedure based on priority rules and an application of dynamic programming. A few computational results are reported.