

Stanisław Kryński, Henryk Potrzebowski
Instytut Badań Systemowych PAN, Warszawa

DUŻEJ SKALI KOMBINATORYCZNY MODEL PLANOWANIA PRODUKCJI I LOSOWY ALGORYTM JEGO ROZWIĄZANIA

Streszczenie. Rozważane jest zagadnienie wyznaczania harmonogramu wieloetapowej realizacji dużej liczby niezależnych zleceń produkcyjnych z kryterium minimalizacji przekroczeń możliwości produkcyjnych. Sformułowano model matematyczny zagadnienia jako całkowitoliczbowe zadanie z warunkami wyboru wariantu. Przedstawiono przybliżoną procedurę złożoną z heurystycznego algorytmu wyznaczania rozwiązania startowego i losowego "termodynamicznego" algorytmu jego ulepszenia. Przytoczono przykłady obliczeniowe z praktyki.

1. Wprowadzenie

Rozpatrywane zagadnienie jest problemem sterowanego rozdziału zasobów pomiędzy operacje o modelach dyskretnych. Sformułowanie zagadnienia zakłada, że dla każdego zlecenia produkcyjnego ustalony jest zbiór dopuszczalnych wariantów realizacji, które charakteryzują się następującymi cechami:

- skończona jest liczba operacji składających się na wykonanie zlecenia,
- ustalona jest długotrwałość i szybkość wykonywania każdej operacji tak, że w każdej jednostce czasu znane są zapotrzebowania zasobowe,
- skończone są wielkości wyprzedzeń międzyoperacyjnych,
- dla danego wariantu realizacji ustalony jest termin zakończenia (rozpoczęcia) realizacji zlecenia.

Przy danych zasobach produkcyjnych (np. siła robocza, środki techniczne zabezpieczenia produkcji, materiały itp.) i ograniczonych terminach zakończenia i rozpoczęcia zleceń należy dokonać wyboru wariantu realizacji każdego zlecenia, przy którym określona funkcja wielkości przekroczeń zasobowych (suma przekroczeń po czasie i zasobach lub suma szczytowych przekroczeń zasobowych) przyjmuje wartość minimalną. Z teoretycznego punktu widzenia zagadnienie to dla dyskretnej osi czasu może być rozpatrywane jako zadanie programowania matematycznego z liniowymi ograniczeniami i warunkami binarności części zmiennych. Liniowa relaksacja tego zagadnienia była przedmiotem rozważań w pracach [1,8,9]. Modelując rzeczywisty seryjny proces produkcyjny, w takim sformułowaniu charakteryzuje się ono dużymi rozmiarami: wieloma tysiącami zmiennych i ograniczeń. Zwykle ma to miejsce również wtedy, gdy zbiór wariantów realizacji zlecenia produkcyjnego ograniczymy do jego możliwych terminów zakończenia przy jednym ustalonym dia-

gramie realizacji. Praktyczne zatem wykorzystanie takiej postaci modelu dla zagadnienia harmonogramowania produkcji jest uzasadnione tylko, gdy dysponujemy wyspecjalizowanym i efektywnym programem lub pakietem programów.

W niniejszej pracy przedstawiamy odmienne podejście do problemu: kombinatoryczną metodę losowego przeglądu zbioru rozwiązań dopuszczalnych. Jej zasadniczy element, tzw. procedurę termodynamiczną, opartą na pomysśle Kirkpatricka i innych [6], efektywnie zastosowano do rozwiązywania kombinatorycznych zagadnień rozmieszczania w pracy Burkarda i Rendla [2] (zob. też [3]). Cechuje się ona stosunkowo niewielkim zapotrzebowaniem na pamięć komputera i łatwością zaprogramowania. Przy jej użyciu w sposób losowy generowany jest ciąg rozwiązań dopuszczalnych, przy czym w odróżnieniu od zwykłej metody Monte Carlo akceptacja kolejno wygenerowanego rozwiązania następuje nie tylko w przypadku poprawy funkcji kryterium ale czasem także w przypadku jej pogorszenia. Ponadto metoda dopuszcza ocenę nie tylko przy pomocy przyjętego wskaźnika jakości, ale również ze względu na inne, nawet nieformalne kryteria oceny. W razie potrzeby bez większych trudności mogą być modyfikowane niektóre parametry modelu. Oznacza to, że metoda może być wykorzystywana w sposób interakcyjny.

Zapotrzebowanie na metodę o takich cechach pojawiło się w związku z badaniami, jakie były prowadzone w Zakładzie Programowania Matematycznego IBS PAN nad zbudowaniem komputerowego systemu planowania miesięcznego dla Zakładu Produkcyjno-Montażowego w ZWUT w Warszawie. O wyborze metody ostatecznie zaważył fakt, że tamtejszy ośrodek obliczeniowy nie dysponował odpowiednim, tj. wystarczająco efektywnym pakietem programów rozwiązywania zadań liniowych mieszanych. Zawiodły też wszystkie teoretycznie uzasadnione techniki, takie jak: relaksacja, dekompozycja lub agregacja omówione w [9]. Prezentowane tutaj algorytmy pozwoliły na przezwycięzenie tych trudności.

Różne odmiany metody i generatora wariantów realizacji zleceń były kodowane w fortranie IV na maszynie Cii Honeywell Bull Level 64, pracującej w systemie GCOS. Najlepsze z nich użyto do wyznaczenia harmonogramów produkcji w warunkach rzeczywistych i do opracowania ostatniej wersji komputerowego systemu planowania opisanego w [4]. Wyniki potwierdzają użyteczność proponowanego podejścia.

Opisana metoda może być stosowana do rozwiązywania zadań harmonogramowania dla zakładów o rozmaitych strukturach produkcyjnych i warunkach technologicznych, pracujących w systemie zleceń. Dlatego przedstawiona zostanie w sposób ogólny, bez drobiazgowego wnikania w nieistotne specyficzne szczególności natury technologicznej bądź organizacyjnej.

2. Sformułowanie zagadnienia

Rozpatrujemy okres (horyzont planowania) o długości m dni. Liczba branych pod uwagę zasobów jest r . Dla każdego dnia $l=1, \dots, m$ i dla $l=m+1$ (tj. dla dalszego horyzontu) oraz zasobu $k=1, \dots, r$ dana jest przepustowość (limit) b_{kl} .

Niech J_i będzie zbiorem wariantów realizacji i -tego zlecenia produkcyjnego, $i=1, \dots, n$, a a_{ijkl} - zapotrzebowaniem na k -ty zasób, $k=1, \dots, r$, w l -tym dniu, $l=1, \dots, m, m+1$, związanym z realizacją tego zlecenia j -tym wariantem, $j \in J_i$. Niech x_{ij} będzie zmienną binarną o wartościach:

$$x_{ij} = \begin{cases} 1, & \text{jeżeli } i\text{-te zlecenie realizowane jest } j\text{-tym wariantem} \\ 0, & \text{w przypadku przeciwnym.} \end{cases}$$

Rozpatrujemy zadanie minimalizacji funkcji przekroczeń zasobowych przy warunku, że w przypadku każdego zlecenia wybieramy jeden wariant realizacji, tj. zadanie postaci

$$\mu(Ax-b) \rightarrow \min, \quad x \in X \quad (1)$$

gdzie

$$X = \{x \mid \sum_{j \in J_i} x_{ij} = 1, \quad i=1, \dots, n; \quad x_{ij} = 0 \text{ lub } 1, \quad i=1, \dots, n, \quad j \in J_i\} \quad (2)$$

Funkcja $\mu(\cdot)$ w sformułowaniu (1) w szczególności może być normą L_1 , L_2 lub L_∞ . Praktycznie ograniczono się jednak do jej dwóch następujących postaci:

$$(W1) \quad \mu(Ax-b) = \sum_{k=1}^r \sum_{l=1}^{m+1} (y_{kl})_+$$

$$(W2) \quad = \sum_{k=1}^r \max_{l=1, \dots, m+1} (y_{kl})_+$$

gdzie $y_{kl} = \sum_{i=1}^n \sum_{j \in J_i} a_{ijkl} x_{ij} - b_{kl}$ jest przekroczeniem możliwości zasobowych b_{kl} przy rozwiązaniu x , a $(y_{kl})_+$ oznacza część dodatnią y_{kl} .

Występujący w modelu $(m+1)$ -szy dodatkowy przedział oznacza dalszy horyzont planowania. Dla tego przedziału arbitralnie ustalono

$$b_{k,m+1} = \max\{0, c_k - \sum_{i=1}^m b_{ki}, \sum_{i=1}^n \min_{j \in J_i} a_{ijk, m+1}\}, \quad (3)$$

gdzie c_k jest niezbędnym sumarycznym zapotrzebowaniem na k -ty zasób zgłaszanym przez wszystkie zlecenia produkcyjne.

W modelu (1) zasób jest pojęciem ogólnym i może oznaczać zdolność produkcyjną komórki produkcyjnej, wydzielonego wewnątrz niej stanowiska lub specjalistycznej linii produkcyjnej. Może oznaczać również liczbę unikalnych przyrządów lub narzędzi lub wielkość sztuczną, obliczoną dla uwzględnienia priorytetu, uzyskania określonej wyгоды w prowadzeniu procesu technologicznego itd.

Formalnie model (1) nie ulega zmianie, jeżeli w miejsce przekroczeń y_{kl} brać $w_{kl} \cdot y_{kl}$, gdzie w_{kl} jest pewną wagą. W [4] przyjmowano dla każdego k

$$w_{kl} = \begin{cases} \phi_k & \text{dla } i=1, \dots, m \\ \phi_k \xi & \text{dla } l=m+1 \end{cases},$$

gdzie ϕ_1, \dots, ϕ_r - wagi przypisywane poszczególnym zasobom, a ξ - parametr decydujący o wielkości produkcji przesuniętej do realizacji w dalszym horyzoncie ($0 < \xi < 1$). Taka modyfikacja istotnie podnosi wartość praktyczną modelu.

Macierz A zwykle charakteryzuje się dużymi rozmiarami i jej pamiętanie w formie rozwiniętej nie zawsze jest możliwe, o ile nie dysponujemy komputerem z dostatecznie dużą pamięcią. Dlatego mają tu zastosowanie specjalne sposoby kodowania. W systemie [4] wykorzystano w tym celu reprezentację diagramu czasowego realizacji zlecenia w postaci listy zadań dziennych. Mianowicie, w przypadku i -tego zlecenia, dla każdego z q_i ($q_i > 1$) zadań składowających się na jego realizację (liczba q_i obejmowała również zadania "sztuczne") określono:

v_q - wielkość wyprzedzenia (w dniach) względem ostatniego zadania, $q=1, \dots, q_i$

s_q - wskaźnik zasobu, $1 < s_q < r$

a_q - wielkość zapotrzebowania na s_q -ty zasób.

Dla ostatniego z wykonywanych zadań dziennych dane były najwcześniejszy \underline{d}_i i najpóźniejszy \bar{d}_i termin wykonania. Natomiast zbiór wariantów (stosowano jeden sposób rozwinięcia diagramu realizacji zlecenia) utożsamiano z możliwymi terminami wykonania zlecenia, tj.

$$J_i = \{ \underline{d}_i, \underline{d}_i+1, \dots, \bar{d}_i \} \quad (5)$$

3. Metoda termodynamiczna

3.1.1. Schemat ogólny metody

Dla rozwiązania (1) przystosowujemy algorytm opisany przez Burkarda i Rendla w [2]. W idei swojej łączy on dwa zasadniczo różne mechanizmy: symulacyjną procedurę modelowania fizycznych zjawisk przebiegających zgodnie z prawem Boltzmann'a oraz algorytm lokalnego przeglądu dla zadań kombinatorycznych.

Niech X będzie skończonym zbiorem i $f: X \rightarrow R$ zadana funkcją. Poszukujemy $x^* \in X$ takiego, że $f(x^*) = \min_{x \in X} f(x)$. Przyjmujemy, że dla każdego $x \in X$ określony jest zbiór $D(x) \subset X$, nazywany otoczeniem punktu x , przy czym $x \notin D(x)$.

W przedstawionym niżej schemacie procedury kolejne polecenia są wykonywane jedno po drugim, chyba że nakazany jest skok w inne miejsce. Występujące w schemacie parametry T, α, β są rzeczywiste, $t > 0$, $0 < \alpha < 1$, $\beta < 1$, a parametry P, I - naturalne.

Algorytm I

START: wyznac rozwiązanie startowe $x_0 \in X$, podstaw $x := \bar{x} := x_0$, $\bar{f} := f(x)$
ustal początkowe wartości T, P, I, α, β , podstaw
 $l := 1$, $ch := false$, $i := 1$

PRÓBA: z rozkładem równomiernym wybierz losowo rozwiązanie $y \in D(x)$

TEST1: jeżeli $f(y) < \bar{f}$ to AKCEPTACJA

TEST2: podstaw $\Delta := f(y) - \bar{f}$. Z rozkładem równomiernym wylosuj liczbę $\delta \in (0, 1)$.
Jeżeli $\delta > \exp(-\Delta/T)$ to POWRÓT

AKCEPTACJA: podstaw $x := y$, $ch := true$
Jeżeli $f(x) < \bar{f}$ to $\bar{x} := x$, $\bar{f} := f(x)$

POWROT: jeżeli $l = P$ to TEST STOPU, w przypadku przeciwnym
 $l := l + 1$, skocz do PRÓBA

TEST STOPU: jeżeli $ch = false$ lub $i = I$ to STOP (\bar{x} - rozwiązanie zadania)

ZMIANA: $T := \alpha T$, $P := E(\beta P)$, $ch := false$, $l := 1$, skocz do PRÓBA.

Algorytm, jak łatwo zauważyć, w sposób losowy dokonuje przeglądu elementów zbioru X przechodząc od ostatnio zaakceptowanego rozwiązania x do rozwiązania y leżącego w jego otoczeniu. Jeżeli w nowo wygenerowanym punkcie y wartość funkcji $f(y)$ jest niższa, to jest on akceptowany. W przypadku przeciwnym akceptacja dokonywana jest z prawdopodobieństwem zależnym od wzrostu wartości funkcji f i od aktualnej wartości parametru T (zwanego temperaturą). Próbkowanie zbioru rozwiązań realizowane jest seriami. W czasie realizacji serii prób wartość T jest stała. Jeżeli w wyniku realizacji pełnej serii prób żadne rozwiązanie nie zostanie zaakceptowane, to algorytm skończy przegląd. Przed rozpoczęciem nowej serii prób wartość T jest zmniejszana ($T := \alpha T$), natomiast liczba wykonywanych prób wzrasta ($P := E(\beta P)$). Parametr I ogranicza liczbę serii. W wyniku działania algorytmu otrzymujemy \bar{x} z ciągu wszystkich wygenerowanych rozwiązań, dla którego f przyjęło wartość minimalną.

Nie istnieją uniwersalne reguły zadawania startowych wartości parametrów T, P, I, α, β . Powinny być one jednak uzależnione od rozmiaru zadania i mocy zbiorów $D(x)$, a także rozpiętości zbioru wartości f na prostej R .

3.2. Zastosowanie metody

Przyjmując, że X określone jest za pomocą (2), zastosowanie algorytmu T do zadania harmonogramowania (1) poza określeniem sposobu wyznaczania roz-

wiązania startowego (omówimy to w punkcie następnym) sprecyzowania wymaga zbiór rozwiązań sąsiednich $D(x)$ dla rozwiązania x oraz sposób prowadzenia obliczeń.

Zbiór $D(x)$ występujący w kroku PRÓBA (patrz algorytm T) definiujemy jako

$$D(x) = \{y < x \mid \exists j \in \{1, \dots, n\} \exists j, h \in J_1 (j \neq h \wedge x_{1j} = y_{1h} = 1 \wedge \forall p \neq i \forall q \in J_p (x_{pq} = y_{pq}))\}$$

Oznacza to, że wygenerowanie y sąsiedniego z x wymaga wylosowania $i \in \{1, \dots, n\}$ oraz $h \in J_1$ takich, że $x_{1h} = 0$. Rozwiązanie y różni się od x wartościami dokładnie dwóch współrzędnych $y_{1h} = 1 = 1 - x_{1h}$ oraz $y_{1j} = 0 = 1 - x_{1j}$.

Sposób prowadzenia obliczeń uzależniony jest od rozmiaru zadania i ogólnie biorąc wygodnie jest go realizować w sposób interakcyjny. Po każdym przebiegu użytkownik dokonuje oceny dotychczasowego procesu obliczeniowego i otrzymanych rozwiązań oraz decyduje o sposobie prowadzenia dalszych obliczeń. Jeżeli decyduje się przedłużyć proces przeglądu rozwiązań, to ma następujące możliwości:

- przedłużyć proces bez zmiany jakichkolwiek parametrów
- przedłużyć proces przy zmianie co najmniej jednego parametru algorytmu T, P, I, α, β
- zmienić wagi w_{k1} zgodnie z (4)
- ograniczyć wybór terminu realizacji zlecenia do mniejszego przedziału
- wskaźnik oceny $W1$ zamienić na $W2$ lub odwrotnie
- zmienić listę zasobów
- dokonać zmiany diagramu realizacji zlecenia.

Wiele z tych możliwości realizuje system harmonogramowania produkcji opisany w [4], gdzie zbiór wariantów realizacji zleceń określano zgodnie z (5).

3.3. Wyznaczanie rozwiązania startowego

Rozwiązanie startowe najprościej jest wygenerować losowo, ale znacznie lepiej jest użyć do tego celu metody programowania liniowego lub algorytmu zachłannego. Dobrze rozwiązanie startowe obniża nakład obliczeń.

Za użyciem metody programowania liniowego przemawia szczególna własność modelu (1). Polega ona na tym, że w przypadku wskaźników $W1, W2$ rozwiązanie ciągle zrelaksowanego zadania (1) (tj. warunek binarności x_{1j} zastąpiony nierównością $x_{1j} < 1$) w części złożonej ze zmiennych x_{1j} jest "prawie" całkowitoliczbowe. Zwrócono na to uwagę w [9]. Otrzymane zadanie liniowe ma $(m+1)r+n$ ograniczeń oraz $\sum |J_1| + 2(m+1) \cdot r$ zmiennych dla kryterium $W1$ i $\sum |J_1| + (m+2)r$ zmiennych dla kryterium $W2$. Praktycznie jest to duże zadanie, a jego rozwiązanie wymaga użycia wyspecjalizowanego pakietu programowania

liniowego, nie zawsze dostępnego w zakładowym ośrodku obliczeniowym. Jeżeli odpowiedni pakiet istnieje, zwykle wystarcza w formie rozwiązania startowego wziąć dowolne zaokrąglenie uzyskanego rozwiązania ciągłego. W szczególności może się okazać, że rozwiązanie ułamkowe też jest akceptowalne.

Inna możliwość wyznaczenia rozwiązania startowego sprowadza się do użycia odpowiednio skonstruowanego algorytmu zachłannego. Omówimy tutaj algorytm przedstawiony w pracy [5].

Niech $x = (x_{ij})$ będzie rozwiązaniem, dla którego warunek wyboru $\sum_{j \in J_i} x_{ij} = 1$ spełniony jest jedynie dla pierwszych p , $p < n$ zleceń. Dla $i > p$, $j \in J_i$ jest $x_{ij} = 0$. Za pomocą a_{kl} oznaczmy zapotrzebowanie na k -ty zasób w l -tym dniu odpowiadający takiemu x .

Algorytm H

Podstaw $a_{kl} := 0$ dla każdego k, l oraz $x_{ij} := 0$ dla każdego i, j .

Dla $i = 1, 2, \dots, n$ wykonaj:

WYBÓR: znajdź $\rho_i = \arg \min_{j \in J_i} \{z_{ij}\}$ gdzie

$$z_{ij} = \sum_{k, l | a_{ijkl} \neq 0} (a_{kl} + a_{ijkl} + (a_{kl} + a_{ijkl} - b_{kl})_+)$$

AKCEPTACJA: $x_{i\rho_i} := 1$, $a_{kl} := a_{kl} + a_{i\rho_i} k l$ dla każdego k, l .

Algorytm daje rozwiązanie w $O(\sum |J_i|)$ krokach. Jak podano w pracy [4], dawał on średnio o 50% lepsze rozwiązanie niż algorytm losowy.

4. Przykłady obliczeniowe

Dla ilustracji omawianego podejścia przytoczymy tutaj niektóre wyniki testowe dla danych rzeczywistych otrzymane za pomocą eksperymentalnego systemu harmonogramowania produkcji, opracowanego dla wspomnianego we wprowadzeniu zakładu PM w ZWUT. Wyniki te w nieco innym ujęciu prezentowane były w [5].

Luty 85

Liczba zleceń	234
Liczba dni roboczych	21
Liczba podstawowych komórek produkcyjnych	7
Liczba zasobów	40

przebieg	a	b	kryterium	ξ	start	koniec	poprawa %	CPU min
H			W1	0.5		26746.5		0.754
1	4820	1013	W1	0.5	26746.5	13847.8	48.2	6.531
2	1000	134	W1	0.5	13847.8	13730.6	0.85	1.864
3	3000	805	W2	-	3658.8	3596.8	1.7	7.42
4	3000	418	W1	0.4	13982.3	12973.0	1.84	4.189
5	3000	323	W1	0.3	12405.8	12177.1	1.84	4.206

Marzec 85

Liczba zleceń	214
Liczba dni roboczych	23
Liczba podstawowych komórek produkcyjnych	7
Liczba zasobów	24

Przebieg	a	b	kryterium	ξ	start	koniec	poprawa %	CPU min
H			W1	0.5		19508.4		1,096
1	19658	1549	W1	0.4	19134.8	13246.0	30.78	18.684
2	4850	445	W1	0.5	14541.6	14459.7	0.56	5.351
3	2600	795	W2	-	2185.1	2156.6	1.3	4.774

W tablicach H oznacza użycie algorytmu H prezentowanego w punkcie 3.3, a - liczbę wygenerowanych rozwiązań (w kroku PROBA algorytmu T), b - liczbę zaakceptowanych rozwiązań (w kroku AKCEPTACJA algorytmu T).

W obliczeniach kryterium W2 stosowano już po użyciu kryterium W1. Modyfikowano przy tym dopuszczalne terminy realizacji zleceń, tj skracano je w taki sposób, że nie był już potrzebny parametr ξ występujący w (4).

Początkową wartość parametru T ("temperature") uzależniono od wartości W_0 wskaźnika jakości dla rozwiązania startowego x_0 oraz od wartości pewnego parametru C. Była ona wyznaczana zgodnie ze wzorem $T = -C \cdot W_0 / \exp(0.5)$. Dzięki temu przyrost wartości wskaźnika $\Delta = C \cdot W_0$ akceptowany był w pierwszej iteracji z prawdopodobieństwem 0,5. W następnych iteracjach T było zmniejszane w wyniku mnożenia przez ustalony współczynnik $\gamma \in (0,1)$; najczęściej przyjmowano $\gamma = 0.7$. Po osiągnięciu przez T odpowiednio małej wartości ($T < 10^{-4}$) praktycznie wszystkie pogarszające wskaźnik rozwiązania były odrzucane. W ten sposób procedura stawała się procedurą losowego lokalnego przeglądu Monte Carlo. W programach przewidziano też możliwość zadawania z góry momentu przełączania na strategię metody Monte-Carlo, niezależnie od wartości T.

5. Zakończenie

Procedura termodynamiczna może być stosowana do przybliżonego rozwiązywania rozmaitych zadań kombinatorycznych, dla których nie istnieją efektywne, tj. wielomianowe algorytmy, w przypadku gdy zachodzą:

- a) łatwo jest wyznaczyć element $D(x)$ dla dowolnego $x \in X$,
- b) łatwo obliczyć wartość $f(y)$ dla $y \in D(x)$, o ile znane jest $f(x)$.

Obydwa warunki są spełnione w przypadku omawianego modelu (1). Wygenerowanie y sąsiedniego z x wymaga wylosowania wskaźnika zlecenia i nowego wariantu jego realizacji. Nakład zaś obliczeń na wyznaczenie przyrostu $f(y)$ względem $f(x)$ jest rzędu liczby niezerowych elementów w kolumnie macierzy $A = (a_{ijkl})$.

W przedstawionym sformułowaniu zagadnienia harmonogramowania produkcji odnajdujemy elementy wspólne z podejściem do tych zagadnień reprezentowanym w wielu firmowych systemach planowania produkcji. Przykładem są tu systemy planowania MMS (od Materials Management System) i MRP (od Material Requirement Planning), znane np. z [7]. Ścisłej biorąc, sformułowanie (1) w znacznym stopniu uwzględnia strukturę danych, na których te systemy pracują. Jedną z możliwości wykorzystania modelu jest zatem możliwość jego dołączenia do firmowego pakietu planowania w postaci modułu optymalizacji.

LITERATURA

- [1] Baran-Jarosz B., Kryński S., Libura M., Potrzebowski H., Sikorski J., Walukiewicz S.: System wyznaczania optymalnego harmonogramu głównego produkcji sprzętu NC dla Zakładu PM. Cz. II. Modele matematyczne. ZPM-29-U/84. Raport ZPM IBS PAN, grudzień 1984.
- [2] Burkard R.E., Rendl F.: A Thermodynamically Motivated Simulation Procedure for Combinatorial Optimization Problems. European Journal of Oper. Res. 17, 1984.
- [3] Dudziński K., Libura M., Majchrzak J., Potrzebowski H., Sikorski J., Szkatuła K., Waluk B.: Optymalizacja rozmieszczenia elementów w zespołach central telefonicznych. Archiwum Aut. i Telemek. XXX, 1985, z. 4, 375-386.
- [4] Grygiel G., Kryński S.L., Potrzebowski H., Waluk B., Walukiewicz S., Wojtych E.: System wyznaczania optymalnego harmonogramu głównego produkcji sprzętu NC dla Zakładu PM na emc Cii Honeywell Bull Series 60 (Level 64). ZPM-1-U/45/84. Raport ZPM IBS PAN, marzec 1985.
- [5] Grygiel G., Kryński S.L., Libura M., Potrzebowski H., Sikorski J., Waluk B., Wojtych E.: Algorytm wyznaczania optymalnego harmonogramu produkcji i jego zastosowanie w sterowaniu produkcją w TELCOM-ZWUT. Materiały konf. "INFOPROD'85", Bydgoszcz - Ciechocinek, 1985.
- [6] Kirkpatrick S., Gelatt C.D., Jr., Vecchi M.P.: Optimization by Simulated Annealing, Science 220, 1983.

- [7] Materials Management System, Functional Description. 00A37332 REV1, Cii Honeywell Bull Level 62, DPS4, 64DPS and DPS7 Industrial Management System (Transaction Driven). July 1981.
- [8] Potrzebowski H.: Deterministyczne problemy harmonogramowania produkcji przy ograniczeniach zasobowych. ZPM-25-49/84. Raport ZPM IBS PAN, grudzień 1984.
- [9] Wojtych E.: Relaksacja i dekompozycja w rozwiązywaniu mieszanego całkowitoliczbowego zadania harmonogramowania produkcji. ZPM-28-49/85. Raport ZPM IBS PAN, 1985.

Recenzent: Doc.dr h.inż. Tadeusz Sawik

Spłynęło do Redakcji do 1986.04.30

КРУПНАЯ КОМБИНАТОРНАЯ ЗАДАЧА ПЛАНИРОВАНИЯ ПРОИЗВОДСТВА И ВЕРОЯТНОСТНЫЙ АЛГОРИТМ ЕЕ РЕШЕНИЯ

Резюме

В статье рассматривается задача построения расписания для многостадийного производственного процесса выполнения большого числа требований с критерием минимизации превышений производственных ресурсов. В качестве математической модели принимается целочисленная задача с условиями выбора варианта выполнения требования. Для решения задачи разработан эвристический алгоритм нахождения исходного решения и "термодинамический" вероятностный алгоритм его улучшения. Указаны прикладные примеры вычислений.

A LARGE SCALE COMBINATORIAL PRODUCTION PLANNING PROBLEM AND A SIMULATION PROCEDURE FOR SOLVING IT

Summary

A combinatorial multicommodity multistage production planning problem formulated as a multiple-choice integer programming problem, is considered in the paper. A method of its solving is proposed. It consists of two phases: (1) a heuristic greedy - like procedure for finding an initial solution and (2) a thermodynamically motivated simulation procedure for improving this solution. The method is easily coded and implementable on a computer. It is capable of modification of parameters during the execution, according to the quality of intermediate solutions. This feature enable the method to be used interactively.

Computational results obtained for data taken from practice are also presented.