

Roman Słowiński

Instytut Automatyki
Politechniki Poznańskiej

Bolesław Soniewicki

Centralny Ośrodek Oświaty
i Postępu w Rolnictwie
Oddział w Poznaniu

ALGORYTM WIELOKRYTERIALNEGO ROZDZIAŁU ZASOBÓW W SIECIOWYM
PLANOWANIU PRZEDSIĘWZIĘĆ ORAZ JEGO IMPLEMENTACJA
MIKROKOMPUTEROWA

Streszczenie. Referat przedstawia implementację mikrokomputerową algorytmu rozdziału zasobów w sieciowym planowaniu przedsięwzięć (cf. [8]) oraz jego uogólnienie na przypadek wielokryterialny (cf. [9]). Rozpatruje się deterministyczny system rozdziału zasobów odnawialnych, nieodnawialnych i podwójnie ograniczonych między czynności niepodzielne.

1. Wprowadzenie

Od szeregu lat w ramach badań operacyjnych intensywnie rozwijane jest wspomaganie podejmowania decyzji dotyczących realizacji przedsięwzięć złożonych ze skończonych zbiorów prac zwanych czynnościami. Czynności te są na ogół powiązane ograniczeniami kolejnościowymi typu relacji częściowego porządku, przedstawianej najczęściej za pomocą grafu. Do realizacji przedsięwzięcia służą określone zasoby dostępne w ograniczonych ilościach. Każda czynność zgłasza zapotrzebowanie na te zasoby, a czas jej wykonywania zależy w ogólności od ich rodzaju i ilości. Celem jest taki rozdział zasobów między czynności w czasie, by spełniając ograniczenia kolejnościowe, wykonać przedsięwzięcie w jak najlepszy sposób z jednego lub wielu punktów widzenia.

Ustalenie terminów wykonywania czynności nazywa się generowaniem harmonogramu. Określony model przedsięwzięcia obejmujący definicję zbioru zasobów, zbioru czynności i zbioru kryteriów oceny harmonogramów nazywa się systemem rozdziału zasobów [1].

W niniejszej pracy rozpatrujemy deterministyczny system rozdziału zasobów odnawialnych, nieodnawialnych i podwójnie ograniczonych między czynności niepodzielne. Dla każdej czynności dopuszcza się wiele możliwych wariantów przydziału zasobów; każdemu wariantowi odpowiada określony czas wykonywania czynności. Zakłada się, że wszystkie dane liczbowe są liczbami całkowitymi. Ograniczenia kolejnościowe między czynnościami dane są w postaci

grafu acyklicznego, spójnego, w konwencji "czynność na łuku". Zaproponowany algorytm oparty jest na idei metody podziału i ograniczeń. Jak wiadomo, w metodzie tej znajomość dobrego rozwiązania początkowego pozwala na znaczne ograniczenie nakładów obliczeniowych, ponieważ szybciej i częściej następuje ucinanie generowanego drzewa rozwiązań. W pracy uzyskujemy dobre rozwiązanie początkowe za pomocą procedur priorytetowych, które w pewnych przypadkach mogą nawet samodzielnie generować rozwiązania zadowalające. Procedury te oparte są na zastosowaniu specjalnych funkcji nadających priorytet czynnościom przedsięwzięcia; przydział zasobów odbywa się w kolejności określonej przez funkcję priorytetową.

Proponowana metoda uwzględnia różne kryteria typu czasowego lub kosztowego. Śnawia się również jej uogólnienie mające na celu generowanie harmonogramów kompromisowych ze względu na oba typy kryteriów brane jednocześnie pod uwagę.

W kolejnych rozdziałach pracy sformułujemy zadanie rozdziału zasobów, przedstawimy priorytetowe algorytmy generujące wstępne rozwiązania heurystyczne dla optymalnego algorytmu podziału i ograniczeń. Podamy również propozycje modyfikacji tego algorytmu w celu przekształcenia go w algorytm heurystyczny, a także propozycję optymalnego i heurystycznego algorytmu optymalizacji wielokryterialnej.

2. Sformułowanie problemu

W niniejszym artykule zastosujemy definicję systemu rozdziału zasobów zaproponowaną w pracach [1, 8, 10]:

$$S = (R, A, \leftarrow, Q)$$

gdzie:

R - skończony zbiór zasobów dostępnych dla realizacji czynności przedsięwzięcia,

A - skończony zbiór czynności,

\leftarrow - relacja częściowego porządku w zbiorze A , przedstawiana najczęściej za pomocą grafu,

Q - skończony zbiór kryteriów oceny wykonania przedsięwzięcia.

W zbiorze R wyróżniamy trzy kategorie zasobów:

1. Zasoby odnawialne, dla których tylko liczba jednostek w każdej chwili wykonywania czynności jest ograniczona, np. siła robocza, maszyny.
2. Zasoby nieodnawialne, dla których tylko zużycie w okresie wykonywania czynności jest ograniczone, np. energia, środki finansowe, woda, surowce.
3. Zasoby podwójnie ograniczone, dla których ograniczenia dotyczą zarówno liczby dostępnych jednostek w każdej chwili realizacji przedsięwzięcia, jak i zużycia w okresie tej realizacji. Najczęściej spotykany zasób

podwójnie ograniczonym są środki finansowe, dla których ograniczona jest nie tylko całkowita suma, lecz także intensywność inwestowania w każdej jednostce czasu realizacji przedsięwzięcia.

Dostępne zasoby mogą być:

- 1) podzielne w sposób dyskretny, np. siła robocza, maszyny, środki finansowe, oraz
- 2) podzielne w sposób ciągły, np. moc, energia.

W niniejszej pracy uwzględnimy tylko zasoby podzielne w sposób dyskretny. Dyskretność zasobów pociąga za sobą dyskretny charakter zapotrzebowań czynności na zasoby.

W systemie są określone wektory dostępności na wszystkie rodzaje i kategorie zasobów:

$$B^O = (B_1^O, \dots, B_p^O)$$

$$B^N = (B_1^N, \dots, B_v^N)$$

$$B^{P1} = (B_1^{P1}, \dots, B_u^{P1})$$

$$B^{P2} = (B_1^{P2}, \dots, B_u^{P2})$$

- gdzie: B^O - ograniczenie dostępności zasobów odnawialnych,
 B^N - ograniczenie dostępności zasobów nieodnawialnych,
 B^{P1} - ograniczenie dostępności zasobów podwójnie ograniczonych w każdej chwili realizacji przedsięwzięcia,
 B^{P2} - ograniczenie zużycia zasobów podwójnie ograniczonych w okresie realizacji przedsięwzięcia.

Dla każdej czynności przedsięwzięcia określone są możliwe warianty przydziału zasobów odnawialnych, nieodnawialnych i podwójnie ograniczonych w każdej jednostce czasu wykonywania. Dla czynności a_k warianty te podane są w postaci następujących wektorów dla poszczególnych kategorii zasobów:

$$R_{k1}^O = (R_{k11}^O, R_{k12}^O, \dots, R_{k1p}^O)$$

$$k = 1, 2, \dots, n$$

$$R_{kw}^O = (R_{kw1}^O, R_{kw2}^O, \dots, R_{kwv}^O)$$

$$R_{k1}^N = (R_{k11}^N, R_{k12}^N, \dots, R_{k1v}^N)$$

$$k = 1, 2, \dots, n$$

$$R_{kw}^N = (R_{kw1}^N, R_{kw2}^N, \dots, R_{k1v}^N)$$

$$R_{k1}^P = (R_{k11}^P, R_{k12}^P, \dots, R_{k1u}^P)$$

$$k = 1, 2, \dots, n$$

$$R_{kw}^P = (R_{kw1}^P, R_{kw2}^P, \dots, R_{kwu}^P)$$

gdzie w jest liczbą wariantów, a v i u , odpowiednio, liczbą rodzajów zasobów nieodnawialnych i podwójnie ograniczonych.

Model matematyczny czynności przedstawia jej realizację w funkcji dostarczonych zasobów. Zakłada się przy tym, że czynność jest wykonywana tylko wtedy, gdy dostarczone jej zasoby zgodnie z jednym z wariantów określonych w jej zapotrzebowaniu. Modelem matematycznym jest tutaj czas wykonania danej czynności zależny od wybranego wariantu przydziału zasobów. Dla każdej czynności przedsięwzięcia zdefiniowany jest więc wektor:

$$D_k = (d_{k1}, \dots, d_{kw}), \quad k = 1, 2, \dots, n$$

gdzie d_{kj} ($j=1, \dots, w$) określa czas wykonywania czynności a_k pod warunkiem przydzielenia jej zasobów zgodnie z wariantem j .

W pracy tej zakładamy brak możliwości przerywania wykonywania czynności, co jest charakterystyczne dla przedsięwzięć wykonywanych z udziałem ludzi.

Dla każdej czynności podany jest ponadto:

- dyrektywny termin rozpoczęcia wykonywania α_k ,
- dyrektywny termin zakończenia wykonywania δ_k .

Czynności ze zbioru A mogą być niezależne, czyli jednocześnie gotowe do wykonania lub też zależne według relacji częściowego porządku określonej w zbiorze A . Jak już wspomniano, relację $<$ przedstawia się najczęściej w postaci grafu skierowanego, acyklicznego, w konwencji "czynność na łuku". Dodatkowo może być określona relacja podporządkowania. Dana czynność może być wykonywana równolegle z czynnością, której jest podporządkowana lub też po jej zakończeniu.

Elementami zbioru Q są kryteria optymalności. Mogą to być kryteria (funkcje celu) typu czasowego lub kosztowego. Jednak przed przedstawieniem rozpatrywanych w pracy kryteriów podany definicję harmonogramu.

Rozwiązanie dopuszczalne (harmonogram) definiuje się jako macierz:

$$H = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1n} \\ h_{21} & h_{22} & \dots & h_{2n} \end{bmatrix}$$

gdzie h_{1i} , $i = 1, \dots, n$, oznacza jednostkę czasu, w której należy rozpocząć wykonywanie czynności a_1 , natomiast h_{2i} , $i = 1, \dots, n$, określa numer wariantu przydziału zasobów.

W niniejszej pracy będziemy rozpatrywać następujące kryteria:

1. Całkowity czas wykonywania przedsięwzięcia:

$$T = \max_{i \in A} (t_i^Z)$$

gdzie t_i^Z - termin zakończenia wykonywania czynności a_i

2. Aktualna wartość zysku:

$$AWZ = \max \left(\sum_{t=1}^T \left(\sum_{k \in C_t} R_{kh_{2k}^n}^n - \sum_{k \in C_{t-1}} R_{kh_{2k}^n}^n \right) x_t + B_j^n \right)$$

gdzie: C_t - zbiór czynności wykonywanych (aktywnych) w okresie t ,

x_t - aktualna wartość współczynnika dyskonta, dla okresu t , przy stopie procentowej równej I

$$x_t = \left(\frac{1}{1+I} \right)^{t-1}$$

3. Równomierność wykorzystania zasobów odnawialnych,

Kryterium to stosowane jest w stosunku do poszczególnych rodzajów zasobów odnawialnych. Możliwe są tutaj dwa sformułowania:

- maksymalne zapotrzebowanie na zasób:

$$F = \min \left(\max \left(\sum_{k \in C_t} R_{kh_{2k}^0}^0 \right) \right) \quad 1 \leq t \leq p$$

- suma kwadratów różnic między wartością dostępną a wykorzystaną:

$$F = \min \left(\sum_{t=1}^T \left(B_1^0 - \sum_{k \in C_t} R_{kh_{2k}^0}^0 \right)^2 \right) \quad 1 \leq t \leq p$$

4. Średnie ważone opóźnienie wykonania czynności:

$$OP = \min \left(\frac{1}{n} \sum_{k=1}^n \omega_k (t_k^z - \delta_k) \right)$$

5. Całkowite opóźnienie wszystkich czynności:

$$LCO = \min \left(\sum_{k=1}^n (t_k^z - \delta_k) \right)$$

6. Ważony przepływ czynności:

$$WP = \min \left(\frac{1}{n} \sum_{k=1}^n \omega_k (t_k^z - \alpha_k) \right)$$

7. Średnie ważone spóźnienie czynności:

$$SWS = \min \left(\frac{1}{n} \sum_{k=1}^n \omega_k \max(0, t_k^z - \delta_k) \right)$$

8. Łączne zużycie zasobu nieodnawialnego:

$$\text{ŁZZN} = \min \left(\sum_{k=1}^n d_{kh_{2k}} R_{kh_{2k}^n}^n \right)$$

9. Łączne zużycie zasobu podwójnie ograniczonego:

$$\sum_{k=1}^n d_{kh} h_{2k} R_{kh}^p h_{2k}^m, \quad 1 \leq m \leq u$$

10. Ważone zużycie, czyli koszt wykonania wszystkich czynności zbioru A:

$$WZ = \min \left(\sum_{k=1}^n \left(\sum_{j=1}^v d_{kh} h_{2k} C_j^n R_{jh}^n h_{2k}^j \right) + \sum_{m=1}^u d_{kh} h_{2k} \left(C_m^p R_{kh}^p h_{2k}^m \right) \right)$$

gdzie C_j^n i C_m^p - wagi reprezentujące koszt jednostkowy zasobów nieodnawialnych i podwójnie ograniczonych.

Dla zapewnienia istnienia rozwiązania dopuszczalnego przyjmuje się, że żadna ze składowych wektorów zapotrzebowań zasobowych czynności nie przekracza dostępnej ilości odpowiedniego rodzaju zasobu w systemie.

Przyjmując, że kryterium optymalności jest czas wykonania zbioru A oraz że zbiór A zawiera czynność a_n będącą następnikiem wszystkich pozostałych czynności przedsięwzięcia, zadanie wyznaczania optymalnego harmonogramu można sformułować w następujący sposób:

$$\text{zminimalizować} \quad T = h_{1n} + d_{nh} h_{2n}$$

przy ograniczeniach:

$$(1) \quad h_{1j} + d_{jh} h_{2j} \leq h_{1i}, \quad j \in S_k, \quad k = 1, \dots, n$$

$$(2) \quad \sum_{k \in C_t} R_{kh}^0 h_{2k}^1 \leq B_1^0, \quad 1 = 1, \dots, p, \quad 1 \leq h_{2k} \leq w, \quad 1 \leq t \leq T$$

$$(3) \quad \sum_{k=1}^n d_{kh} h_{2k} R_{kh}^n h_{2k}^j \leq B_j^n, \quad j = 1, \dots, v$$

$$(4) \quad \sum_{k \in C_t} R_{kh}^p h_{2k}^m \leq B_m^{p1}, \quad m = 1, \dots, u$$

$$(5) \quad \sum_{k=1}^n d_{kh} h_{2k} R_{kh}^p h_{2k}^m \leq B_m^{p2}, \quad m = 1, \dots, u$$

$$(6) \quad t_k^z = h_{1k} + d_{kh} h_{2k} - 1, \quad k = 1, \dots, n$$

gdzie S_k jest zbiorem numerów bezpośrednich poprzedników czynności a_k ($k=1, \dots, n$).

Pierwszy z warunków dotyczy spełnienia ograniczeń kolejnościowych, drugi natomiast spełnienia ograniczeń zasobowych związanych z zasobami odnawialnymi, trzeci z zasobami nieodnawialnymi, a czwarty i piąty z zasobami podwójnie ograniczonymi. Szósty warunek zapewnia ciągłość wykonywania czynności.

3. Heurystyczne algorytmy priorytetowe

Główną cechą tych algorytmów jest zastosowanie funkcji nadawania priorytetów czynnościom przedsięwzięcia. Priorytet ten ustala kolejność przydzielania zasobów do czynności. Algorytm taki zaproponował po raz pierwszy J.E.Kelley jr. [5] i nazwał go równoległym.

Poniżej przedstawiamy algorytm równoległy dla przypadku wielu wariantów przydziału zasobów do czynności przedsięwzięcia, zapisany w pseudo-pascalu. Kryterium oceny harmonogramu jest całkowity czas realizacji przedsięwzięcia. Dla uproszczenia uwzględniono w tej procedurze tylko zasoby odnawialne.

procedure PR;

{Opis zmiennych procedury:

- N - zbiór numerów czynności niewykonanych,
- L - lista czynności uporządkowana według nierosnącej wartości funkcji priorytetowej,
- P - zbiór numerów czynności gotowych do wykonania,
- Q - zbiór numerów czynności aktualnie wykonywanych,
- Calculate - funkcja obliczania,
- F_{pr} - funkcja priorytetowa określona na zbiorze A,
- choice - funkcja przydziału wariantu,
- choice1, choice2 - funkcje wyboru,
- tail - funkcja umieszczająca czynność na końcu listy,
- w - maksymalna liczba zdefiniowanych wariantów przydziału zasobów.

Pozostałe symbole zgodne są z oznaczeniami zdefiniowanymi wcześniej}

begin

{..ustalenie warunków początkowych }

N ← (1, ..., n);

L, P, Q ← (0);

for i ← 1 to n do Calculate ($F_{pr}(i)$);

repeat

 choice1 (max ($F_{pr}(i)$);

$1 < i < n$);

 tail (L) ← (i);

 N ← N - (i)

until N = (0);

T ← 0;

{ generacja harmonogramu }

repeat

 { generacja liczby czynności gotowych do wykonania }

 for i from L do

 begin

 b ← true;

 for j from S_i do

 begin

 if ($j \in L \vee j \in P \vee j \in Q$) then b ← false

 if b then

```

begin
  L ← L - (i);
  P ← P + (i)
end
end;
end;
{ generacja zbioru Ct }
while b do
begin
  choice (h21, i, b);
  if b then
  begin
    h11 ← T;
    P ← P - (i);
    Q ← Q + (i);
    B10, ..., Bp0 ← B10 - Rih2110, ..., Bp0 - Rih21p0
  end
end;
{ zakończenie wykonywania czynności }
choice2 ( min (d));
  i from Q
T ← T + di;
for j from Q do
  if dj = di then
  begin
    Q ← Q - (i);
    B10, ..., Bp0 ← B10 + Rjh2j10, ..., Bp0 + Rjh2jp0
  end
until L, P, Q = (0)
end;

```

Funkcja Calculate (F_{pr}) powoduje obliczenie dla każdej czynności wartości priorytetu, która to wartość determinuje kolejność "pobierania" czynności do przydziału zasobów. Dla pozostałych kryteriów zdefiniowanych w rozdziale 2 przyjęto następujące funkcje:

Kryteria 1,3,8,9,10: Przydział maksymalny priorytet tej czynności, która posiada minimalną wartość najpóźniejszego momentu rozpoczęcia jej wykonywania, wynikającą z analizy drogi krytycznej projektu.

Kryterium 2: Przydział maksymalny priorytet tej czynności, która posiada wariant przydziału wybranego zasobu o minimalnej wartości.

Kryteria 4,7: Przydział maksymalny priorytet tej czynności, dla której wyrażenie $\frac{\delta_k}{\omega_k}$ osiąga wartość minimalną.

Kryterium 5: Przydział maksymalny priorytet tej czynności, która posiada minimalną wartość dyrektywnego terminu zakończenia wykonywania.

Kryterium 6: Przydział maksymalny priorytet tej czynności, dla której wyrażenie $\frac{\delta_k}{\omega_k}$ osiąga wartość minimalną.

Funkcja choice (h_{21}) powoduje przydział wariantu przydziału zasobów do aktualnie rozpatrywanej czynności. Dla poszczególnych kryteriów zastosowano następujące procedury przydziału wariantu:

Kryteria: 1,3,4,5,6,7

```

procedure choice ( $h_{21}, i, b$ );
begin
  b ← false;
  for i from P do
    for w ← 1 to r do
      if  $R_{i w 1}^0, \dots, R_{i w p}^0 \leq B_1^0, \dots, B_p^0$  then
        begin
           $h_{21} \leftarrow w$ ; b ← true
        end
    end
end;

```

end;

Kryterium 2,8,9,10: Przydziel wariant przydziału zasobów minimalnej wartości zapotrzebowania na wybrany zasób.

4. Algorytm oparty na metodzie podziału i ograniczeń

Algorytmy priorytetowe dostarczają dobrych rozwiązań początkowych dla algorytmu opartego na metodzie podziału i ograniczeń, który umożliwia wyznaczenie rozwiązania optymalnego. Polega ona na ograniczonym przeglądzie przestrzeni rozwiązań mającej strukturę drzewa. Algorytm optymalny oparty na tej metodzie zaproponowano w pracy [8]. Algorytm ten można zastosować dla różnych funkcji celu. Umożliwia on decydentowi udział w procesie poszukiwania rozwiązania kompromisowego w trybie dialogowym.

W praktyce ze względu na długi czas obliczeń algorytmu optymalnego ogranicza się liczbę przeglądanych rozwiązań, wobec czego brak jest gwarancji, że otrzymane rozwiązanie jest optymalne. Taki algorytm jest więc algorytmem heurystycznym umożliwiającym poprawienie rozwiązań początkowych dostarczonych przez algorytmy priorytetowe.

5. Implementacja algorytmu

Powyższy algorytm optymalny i heurystyczny zaimplementowano na mikrokomputer IBM XT/AT.

Programy napisane zostały w języku TURBO-PASCAL z wykorzystaniem pakietu graficznego Turbo Graphix Toolbox firmy Borland. Dla każdej funkcji celu napisano osobny program. Programy wywoływane są z programu głównego za pomocą funkcji chain.

W głównym menu programu znajdują się następujące opcje:

- czytanie danych do pliku,

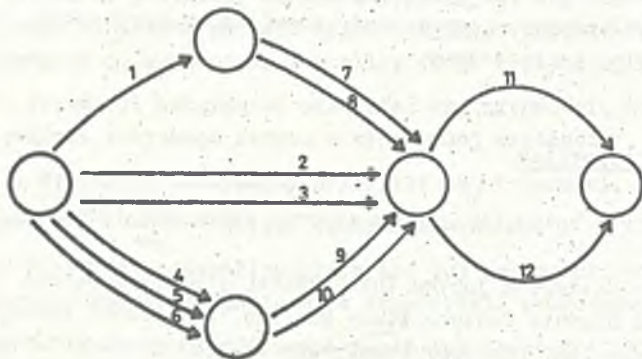
- algorytmy priorytetowe,
- algorytm podziału i ograniczeń,
- modyfikacja danych.

W dwóch środkowych opcjach kolejne menu umożliwia wybranie odpowiedniego algorytmu jednokryterialnego.

W przypadku uruchomienia algorytmu opartego na metodzie podziału i ograniczeń należy liczyć się z dłuższym czasem obliczeń. W każdym przypadku istnieje jednak możliwość przerwania obliczeń i przeglądania aktualnie najlepszego ze znalezionych rozwiązań. W przypadku uznania tego rozwiązania za zadowalające algorytm zatrzymuje się, a w przeciwnym razie obliczenia są kontynuowane.

6. Przykład działania programu

W celu zilustrowania działania programu rozważmy następujące przedsięwzięcie, będące elementem zespołu prac wykonywanych w jednej ze spółdzielni rolniczych w dziale produkcji roślinnej. Przedsięwzięcie to obejmuje 12 czynności. Graf ograniczeń kolejnościowych przedstawiono na rys. 1, natomiast parametry każdej czynności zamieszczono w tablicy 1. Program ten rozwiązujemy dla kryterium czasu realizacji przedsięwzięcia.



Rys.1. Graf ograniczeń kolejnościowych

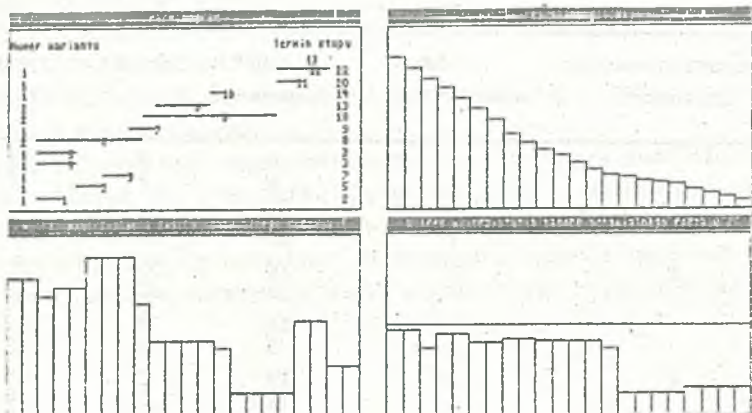
Fig.1. Graph of sequential constraints

Tabela 1

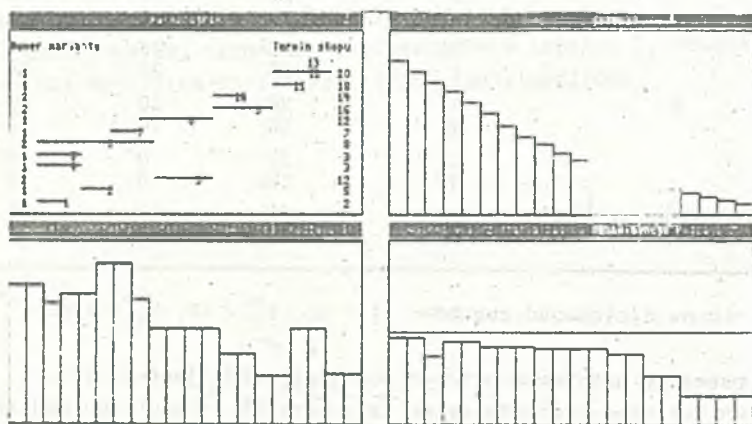
Nr czyn.	Podporządkowanie czynności	Czas wykonywania	Zapotrzebowanie na zasoby pracownicy traktory pieniądze		
1.		2	8	8	10
		4	4	4	6
2.		2	26	26	30
		4	13	13	17
3.	2	2	40	22	40
		4	20	10	23
4.		3	10	20	25
		6	5	10	15
5.	4	3	12	0	3
		6	6	0	2
6.	4	4	66	23	50
		8	33	12	30
7.		2	40	24	41
		4	20	12	25
8.	7	5	24	24	26
		10	12	12	15
9.		2	46	46	45
		4	23	23	26
10.	9	1	20	20	18
		2	10	10	11
11.		2	20	0	4
		4	10	0	2
12.	11	4	24	14	20
		8	12	7	12

Ograniczenie na dostępność zasobów: $B_1^0 = 80$, $B_2^0 = 40$, $B_1^n = 1000$

Wynik zastosowania procedury priorytetowej pokazany jest na rys. 2. Harmonogram ten był następnie ulepszany za pomocą algorytmu typu podziału i ograniczeń. Po pięciu minutach otrzymano harmonogram pokazany na rys. 3.



Rys.2. Harmonogram otrzymany za pomocą procedury priorytetowej
 Fig.2. Scheduling program obtained by priority subroutine



Rys.3. Ulepszony harmonogram otrzymany za pomocą algorytmu
 typu podziału i ograniczeń

Fig.3. Improved scheduling program obtained by bound and branches method

7. Uogólnienie na przypadek wielokryterialny

Algorytm typu podziału i ograniczeń uogólniono w [9] na przypadek wielokryterialny. Program algorytmu wielokryterialnego wykorzystuje procedury opracowane dla przypadku jednokryterialnego, włącznie z procedurami priorytetowymi.

Koncepcja algorytmu wielokryterialnego polega na zastosowaniu procedury dialogowej składającej się z trzech kroków powtarzanych w sposób iteracyjny.

W kroku pierwszym (przygotowawczym) wyznacza się harmonogramy optymalne lub suboptymalne ze względu na pojedyncze kryteria. W wyniku tego kroku uzyskuje się harmonogram idealny, który w ogólności jest niedopuszczalny. W kroku drugim (obliczeniowym) poszukuje się harmonogramu dopuszczalnego leżącego w minimalnej odległości od harmonogramu idealnego w sensie powiększonej normy Czebyszewa (ważona kombinacja norm L_1 i L_∞). W kroku trzecim decyzyjnym przedstawia się ten harmonogram decydentowi do oceny i jeśli ten uzna go za najbardziej kompromisowy, to algorytm zatrzymuje się. W przeciwnym razie decydent zapytywany jest, na jakim kryterium gotów jest stracić, i o ile, by zyskać na innych. Informacja ta jest podstawą modyfikacji zbioru rozwiązań dopuszczalnych dla następnego kroku obliczeniowego itd.

LITERATURA

- [1] Błażewicz J., Cellary W., Słowiński R., Węglarz J.,: Scheduling under resource constraints - deterministic models, J.C. Boltzer, Basel 1986.
- [2] Cooper D.F.: Heuristic for scheduling resource - constrained project: an experimental investigation, Man. Sci. 22/1976, 1186-1194.
- [3] Elmaghraby S.E., Activity Networks: Project Planning and Control by Networks Models, Wiley, New York 1977.
- [4] Garey M.R., Johnson D.S., Computers and Intractability: A Guide to Theory of NP-Completeness, Freeman W.H., San Francisco 1979.
- [5] Kassyk-Rokicka H.: Mierniki statystyczne, PWE, Warszawa 1986.
- [6] Kelley J.E. jr, The Critical-Path Method: Resource Planning and Scheduling, in: Industrial Scheduling, eds. Muth J.F. and Thompson G.L., Prentice Hall, Engl. Cliffs, New Jersey 1963.
- [7] Luszniwicz A.: Metody wnioskowania statystycznego, PWE, Warszawa 1986.
- [8] Patterson J., Słowiński R., Talbot B., Węglarz J., An Algorithm for a General Class of Precedence and Resource Constrained Scheduling Problems, to appear in EJOR.
- [9] Słowiński R., Węglarz J., An Interactive Algorithm for Multiobjective Precedence and Resource Constrained Scheduling Problems, (w)H.Vriethoff, J.Visser, H.K.Boerma (eds.), Proc. 8th INTERNET Congress, Elsevier North-Holland, Amsterdam 1985, 866-873.
- [10] Słowiński R., Algorytmy sterowania rozdziałem zasobów różnych kategorii w kompleksie operacji, WPP, Poznań 1980.
- [11] Sóniewicki B.: Rozdział zasobów w sieciowym planowaniu przedsięwzięć, praca doktorska niepublikowana, Poznań 1985.
- [12] Węglarz J.: Sterowanie w systemach typu kompleks operacji, PWN, Warszawa-Poznań 1981.

Recenzent: Doc.dr hab.inż.H.Zaborowski

Wpłynęło do Redakcji do 1988-04-30.

МНОГОКРИТЕРИАЛЬНЫЙ АЛГОРИТМ РАЗДЕЛА РЕСУРСОВ В СЕТЕВОМ ПЛАНИРОВАНИИ И ЕГО МИКРОКОМПЬЮТЕРНАЯ ИМПЛЕМЕНТАЦИЯ

Резюме

В статье даётся микрокомпьютерная имплементация алгоритма раздела ресурсов в сетевом планировании а также его обобщение на многокритериальный случай. В статье рассматривается детерминированная система раздела ресурсов обновляемых, необновляемых и вдвойне ограниченных между неделимыми действиями.

A MULTICRITERIA RESOURCE-CONSTRAINED PROJECT SCHEDULING
ALGORITHM AND ITS MICROCOMPUTER IMPLEMENTATION.

Summary

The paper presents a microcomputer implementation of a resource-constrained project scheduling algorithm (cf. [8]) and its generalization to the multicriteria case (cf. [9]). A deterministic system of renewable, nonrenewable and doubly-constrained resource allocation is considered.