REVISTA DE ANALIZĂ NUMERICĂ ŞI TEORIA APROXIMAŢIEI
Volumul 1, Fascicola 2, 1972, pp. 173–197

ÎMBUNĂTĂŢIREA PROGRAMĂRII ÎN TIMP A FABRICAŢIEI LA PRODUCŢIA ÎN SERIE CU STOCURI TAMPON

de

L. NÉMETI
(Cluj)

1. Introducere

În lucrarea de faţă vor fi prezentate unele consideraţii ce constituie anumite îmbunătăţiri ale rezultatelor obţinute de către autor în lucrările [9], [10] (vezi şi [8]).

Prima observaţie se referă la noţiunea de perioadă a fabricaţiei. Înţelegem prin producţia în serie o fabricaţie periodică, la care se prelucrează un şir de loturi de piese, loturi de altfel identice între ele, iar prelucrarea unui astfel de lot constituie tocmai o perioadă. Să fie dat (întocmit) programul de fabricaţie al unui lot care se reprezintă cel mai bine cu ajutorul unei diagrame Gantt (vezi fig. 1). Întocmirea unui astfel de program este o problemă a programării (sau planificării) în timp a fabricaţiei sau problema de ordonanţare a operaţiilor şi aparţine domeniului planificării operative.

Pe figură, fiecărui loc de muncă (fiecărei maşini de prelucrare) îi este rezervată o axă orizontală (timpul) şi operaţiile ce se execută pe maşina respectivă sînt indicate prin nişte segmente (îngroşate pe axa respectivă).

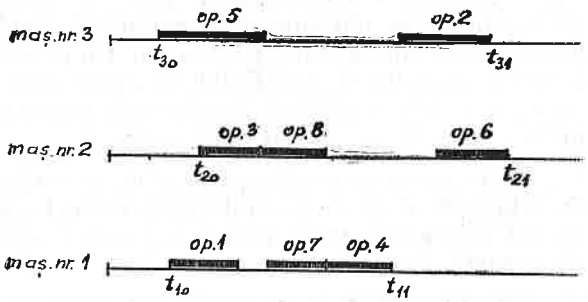

Fig. 1

În acest program, fiecare maşină posedă un interval de timp al funcţionării sale; lungimea acestor intervale este

$$\text{la maşina nr. } 1: d_1 = t_{11} - t_{10}$$

$$\text{,, ,, ,, } 2: d_2 = t_{21} - t_{20}$$

$$\text{,, ,, ,, } 3: d_3 = t_{31} - t_{30}$$

Notăm cu

$$d = \max\,(d_1, d_2, d_3) \text{ sau mai general}$$

$$d = \max_k d_k, \tag{1.1}$$

unde d_k înseamnă durata intervalului de funcţionare a locului de muncă nr. k şi mărimea d indică durata perioadei de fabricaţie, întrucît după decurgerea acestei durate (socotită de la momentul de începere a fiecărei maşini, adică de la t_{10} respectiv t_{20} respectiv t_{30}) procesul de fabricaţie poate fi reînceput, prelucrînd lotul următor.

O altă noţiune este ciclul de fabricaţie. Durata acestuia este:

$$\delta = \max\,(t_{11}, t_{21}, t_{31}) - \min\,(t_{10}, t_{20}, t_{30}),$$

adică durata intervalului în cursul căruia procesul de fabricaţie se desfăşoară. Are loc

$$\delta \geqslant d. \tag{1.2}$$

Dacă durata perioadei se micşorează, acest lucru înseamnă că producţia dintr-un interval dat (de ex. dintr-un an) creşte şi creşte totodată şi realizarea valorică a întreprinderii (de ex. venitul net). Dacă durata ciclului de fabricaţie se micşorează înseamnă că fondurile pentru producţia neterminată sînt imobilizate pentru un timp mai scurt, cheltuielile legate de această imobilizare (dobîndă) descresc ceea ce echivalează de asemenea cu o creştere a venitului. Aşadar, din punct de vedere economic se cere ca ambele durate să fie cît mai scurte.

În legătură cu acestea, va trebui să revizuim definiţia duratei perioadei de fabricaţie dată de formula (1.1) întrucît, după cum vom vedea mai jos, se pot ivi cazuri, unde perioada poate fi micşorată sub mărimea indicată de această formulă.

Să considerăm într-adevăr un program de fabricaţie reprezentat în fig. 2.

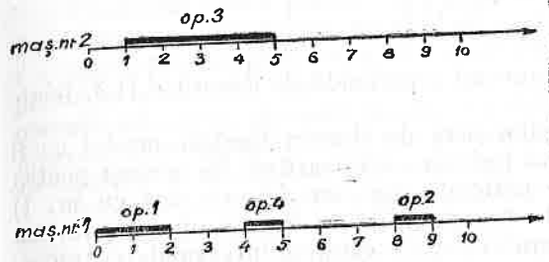

Fig. 2

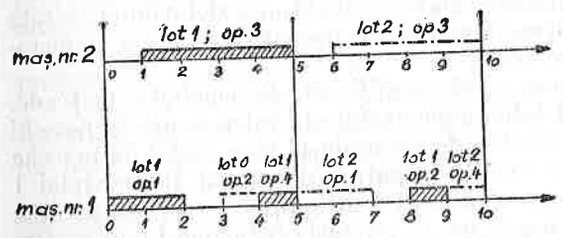

Fig. 3

Maşina nr. 1 funcţionează în intervalele de timp [0, 2) [4, 5) şi [8, 9), iar maşina nr. 2 în [1, 5). Să presupunem că pauzele în funcţionarea maşinii nr. 1 sînt impuse de motive tehnologice. De exemplu, operaţia nr. 4 nu poate fi începută înaintea momentului $t = 4$, fiindcă operaţia premergătoare (care se execută pe o altă maşină pe care nu am indicat-o în diagramă pentru a nu complica reprezentarea) se termină abia în acest moment. Conform definiţiei (1.1), durata perioadei ar fi $d = 9$. Perioada poate fi însă scurtată în modul următor (vezi figura 3).

Operaţia nr. 2 o putem aşeza în intervalul gol [2, 4), bine înţeles nu pentru lotul nr. 1 a cărui prelucrare este reprezentată în fig. 2, ci a lotului premergător, cel cu nr. 0. În figura 3 sînt reprezentate două perioade consecutive şi se vede prelucrarea integrală a lotului nr. 1 şi cîte o parte din prelucrarea lotului nr. 0 şi a lotului nr. 2, durata perioadei fiind de această dată egală cu 5 (în loc de 9 unităţi de timp). După cum se vede, ciclurile de fabricaţie a loturilor consecutive nu sînt disjuncte ci ele se întrepătrund. De fapt, programul din fig. 3 este în esenţă identic cu cel din fig. 2, repartizarea operaţiilor la loturile diferite fiind însă alta. Prin acest artificiu, am reuşit să reducem durata perioadei de la $d = 9$ la $d = 5$, durata ciclului de fabricaţie $\delta = 9$ rămînînd aceeaşi la ambele variante ale programului.

Să dezvoltăm această idee într-un mod mai general (ideia schemei ce urmează se datoreşte d-lui B. SUSSMANN, cercetător la SEMA, Paris). Există mai multe maşini (fiecare de alt tip) fiecăreia dîndu-i-se ce operaţii va executa (această repartizare a operaţiilor rezultă din procesul tehnologic adoptat). Pentru fiecare operaţie se cunoaşte durata de execuţie. Există perechi de operaţii (ele aparţin maşinilor diferite) la care se dă o relaţie de succesiune tehnologică: cea de a doua operaţie poate fi începută abia după ce prima operaţie s-a terminat. Pentru fiecare maşină se poate uşor determina durata minimă a executării tuturor operaţiilor ce-i sînt repartizate: această durată T_k (la maşina nr. k) este egală cu suma duratelor de execuţie a operaţiilor repartizate la maşina k. Durata perioadei celei mai scurte posibile este atunci, evident

$$d = \max_k T_k. \tag{1.3}$$

Un program care realizează tocmai o perioadă de durată d (1.3) poate fi întocmit în modul următor:

În diagrama Gantt a maşinilor, axa de timp a fiecărei maşini va fi împărţită în perioade de durata d (originea arbitrară să fie aceeaşi pentru toate axele, vezi fig. 4). Într-o perioadă oarecare (de ex. cea cu nr. 1) vor fi fixate, în mod arbitrar, toate operaţiile — fiecare operaţie la axa maşinii căreia îi aparţine. Singura restricţie este ca intervalele ce reprezintă operaţii aparţinînd aceleiaşi maşini, să fie disjuncte. Lungimea segmentului reprezentînd o operaţie să fie proporţională cu durata de execuţie a acesteia.

Pe figură maşina nr. 1 este cea mai încărcată, ea determină durata d a perioadei. Numerele deasupra segmentelor operaţiilor indică numerotarea, de altfel arbitrară, a acestora.

Acest program se reproduce, fără modificări, la celelalte perioade. Aşa cum s-a mai spus, procesul tehnologic stabileşte între anumite perechi de operaţii o relaţie de succesiune. Fie de exemplu (1,6) o astfel de pereche ordonată. Atunci trasăm o linie de la capătul (drept) al intervalului 1 la începutul (capătul stîng) al intervalului 6 (vezi figura). Fie o altă pereche de acest fel: (6, 8). Aici, în perioada nr. 1, operaţia 8 începe înaintea terminării operaţiei 6. De aceea, linia de la sfîrşitul operaţiei 6 va fi trasată la începutul operaţiei 8 din *perioada nr.* 2, operaţia nr. 8 din perioada nr. 1 aparţinînd lotului nr. 0 ce precedă lotul nr. 1, cel studiat.

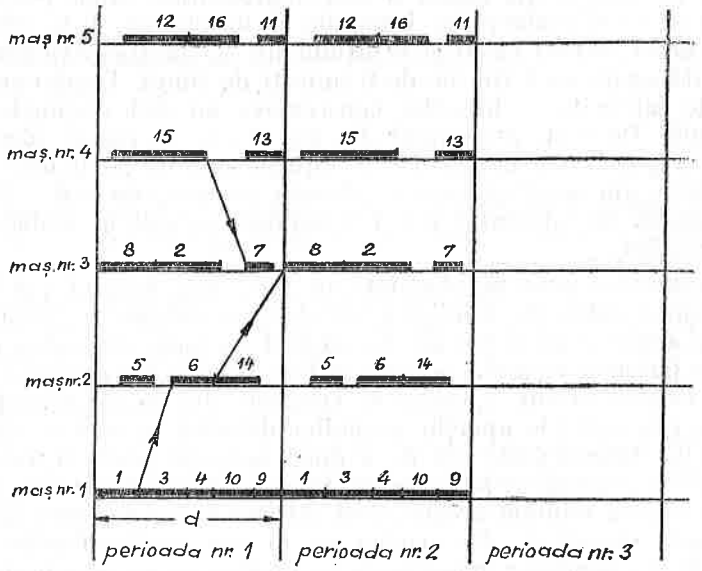

Fig. 4

Procedînd astfel pentru fiecare pereche de operaţii legate prin relaţia de succesiune tehnologică, se stabileşte programul de fabricaţie, cu durata d a perioadei. Ciclul de fabricaţie durează de la perioada 1 pînă la ultima perioadă unde mai apar săgeţile de legătură susmenţionate între două operaţii succesive. Această durată δ a ciclului depinde de ordinea aleasă la aşezarea operaţiilor în intervalul perioadei. Se caută tocmai ordinea optimă de aşezare a operaţiilor care minimizează durata ciclului de fabricaţie.

Această problemă este susceptibilă la diverse generalizări. În primul rînd, o pereche de operaţii ca (6, 8) poate fi pusă în aceeaşi perioadă, adică săgeata de legătură de la op. 6 la op. 8 arată spre stînga şi nu spre dreapta admiţînd că a fost realizat — într-o perioadă de pregătire — un stoc tampon înaintea maşinii nr. 3 care execută operaţia 8. Atunci, cînd urmează să se înceapă cu executarea operaţiei nr. 8, se vor folosi piesele din stocul tampon, urmînd ca acest stoc odată consumat, să fie reîntregit după executarea operaţiei 6 şi să aştepte să fie folosit, în perioada următoare, pentru prelucrarea lotului următor.

În acest fel ciclul de fabricaţie poate fi prescurtat şi el, avînd loc în schimb o creştere a stocurilor tampon care la rîndul lor îmobilizează şi ele un fond, fapt care generează cheltuieli suplimentare.

O altă generalizare constă în posibilitatea de a nu alege tocmai durata cea mai scurtă posibilă a perioadei. O perioadă ceva mai lungă face eventual posibilă o scurtare a duratei ciclului de fabricaţie respectiv o micşorare a cantităţilor necesare la stocuri tampon.

În rezumat, avem trei funcţii de minimizat:

d = durata perioadei de fabricaţie

δ = durata ciclului de fabricaţie

σ = valoarea totală a stocurilor tampon.

În lucrarea prezentă vor fi discutate două modalităţi de compunere a funcţiei economice de optimizat.

În paragrafele 2, 3 şi 4 se va alege de la bun început $d = d_{\min}$, unde $d_{\min}$ este mărimea definită prin relaţia (1.3) şi se cere minimizarea unei combinaţii din δ şi σ. În paragraful 5 se va formula cazul mai general al unei funcţii de optimizat $f(d, \delta, \sigma)$. În fine paragraful 6 tratează situaţia cînd dintr-un tip de maşină de prelucrare există mai multe exemplare, cazul în care repartizarea operaţiilor pe maşini nu mai este determinată în mod univoc de către tehnologia folosită.

2. Modelul nr. 1

Acest model tratează situaţia cînd din fiecare tip de maşină de prelucrare (loc de muncă) există un singur exemplar, iar scurtării duratei perioadei i se dă o prioritate absolută faţă de micşorarea cheltuielilor legate de imobilizarea fondurilor circulante (producţia neterminată).

Pentru simplificarea tratării vom alege mărimea unității de timp egală cu durata perioadei de fabricație, fixată prin relația (1.3). Operațiile de executat (operație: prelucrarea de un anumit fel a unui lot de piese identice pe un anumit tip de mașină de prelucrare; mărimea lotului se consideră a fi dată) sînt numerotate, în mod arbitrar, de la 1 la n. Aceste numere formează mulțimea

$$N = \{1, 2, \ldots, n\} \tag{2.1}$$

ce reprezintă mulțimea operațiilor.

Notăm cu $H = \{(i, j)\} \subset N \times N$ mulțimea perechilor (ordonate) de operații între care există o relație de succesiune tehnologică; relația $(i, j) \in H$ afirmă că operația j este un succesor tehnologic direct al operației i.

Mulțimea H este antisimetrică în sensul că are loc

$$(i, j) \in H \Rightarrow (j, i) \notin H. \tag{2.2}$$

Notăm cu $J = \{(i, j)\} \subset N \times N$ mulțimea perechilor de operații care se execută pe aceeași mașină. Mulțimea J este simetrică în sensul că are loc

$$(i, j) \in J \Leftrightarrow (j, i) \in J. \tag{2.3}$$

Momentul de începere a operației nr. i se notează cu x_i. Dacă o operație i începe la momentul x_i, atunci există o operație i (la alte loturi decît lotul studiat) cu momentul de începere $x_i + k$, unde k este un întreg oarecare. Pentru a preciza sensul lui x_i, se va alege un lot oarecare conținînd toate piesele de prelucrat, iar x_i înseamnă începerea operației nr. i la *lotul ales*.

Partea întreagă a lui x_i: $n_i = [x_i]$ precizează perioada în care are loc începerea operației la lotul ales, iar partea fracționară $\xi_i = \{x_i\}$ precizează locul intervalului de timp ce corespunde executării operației (mai bine zis începerii acestei operații) în interiorul perioadei.

Se dă timpul de execuție t_i a operației i care este un număr pozitiv subunitar (cu scara de timp precizată mai sus). Se presupune mai departe că o operație odată începută, ea nu se întrerupe pînă la terminarea ei. Se mai presupune că o mașină lucrează la un moment dat la executarea cel mult a unei operații și că o operație se execută pe o singură mașină.

Se mai face o simplificare. Se face abstracție de timpii de pregătire (de reglare) a locului de muncă în vederea începerii unei operații, precum și de timpul de control interfaze și de transportul lotului de la un loc de muncă la următorul (vezi lucrările [8], [9], [11]). Aceste simplificări sînt de altfel neesențiale și problema poate fi formulată în termeni mai generali, fără ca metodele de rezolvare să sufere vre-o modificare importantă.

Alte necunoscute sînt:

$$s_i, \quad i \in N'$$

ce indică mărimea stocului tampon pregătit pentru executarea operației i, unitatea de măsură fiind mărimea lotului din piesa respectivă. $N' \subset N$ este mulțimea operațiilor ce posedă predecesori tehnologici imediați:

$$N' = \{j | (i, j) \in H\}. \tag{2.4}$$

Mărimile s_i sînt prin definiție nenegative

$$s_i \geqslant 0, \quad i \in N'. \tag{2.5}$$

Se mai notează cu

$$\delta = n_0 + \xi_0 \tag{2.6}$$

durata ciclului de fabricație (n_0: întreg, $0 \leqslant \xi_0 < 2$).

Cu privire la mărimea stocului tampon s_i mai introducem o restricție care pe de o parte va simplifica mult calculele, pe de altă parte aproximează mai bine situația reală.

Pînă acum mărimile s_i au fost supuse numai condițiilor (2.5). Este clar însă că valori $s_i > 1$ nu au nici o motivare tehnico-economică. De asemenea un stoc tampon s_i cu $0 < s_i < 1$ este ineficace. Într-adevăr stocul tampon s_i are menirea de a face posibilă începerea executării operației i în orice moment liber al mașinii corespunzătoare de prelucrare. Dacă însă are loc $s_i < 1$, atunci acest stoc nu va asigura neapărat o derulare nestînjenită a operației, căci s-ar putea întîmpla că la epuizarea stocului operația ar trebui întreruptă piesele din operația precedentă nefiind încă sosite. Pentru necunoscutele s_i vom admite deci numai valorile 0 și 1.

Cu aceste considerații preliminare putem trece la construirea modelului matematic al problemei noastre. Necunoscutele sînt:

$$n_i, \ \xi_i, \quad i \in N$$
$$s_i, \quad i \in N'$$

precum și

$$n_0 \text{ și } \xi_0$$

care să minimizeze funcția economică

$$z = n_0 + \xi_0 + \sum_{i \in N'} \alpha_i s_i \tag{2.7}$$

cu coeficienţii pozitivi α_i daţi. Justificarea economică a acestei forme se va da în paragraful 5.

Necunoscutele sînt supuse la următoarele restricţii:

(2.8 a) $$n_i \in \mathbf{N}, \quad i \in N$$

(2.8 b) $$0 \leqslant \xi_i < 1, \quad i \in N$$

(2.8 c) $$s_i \geqslant 0, \quad i \in N$$

(2.8 d) $$n_0 - n_i \leqslant 0, \quad i \in N$$

(2.8 e) $$\xi_0 - \xi_i \geqslant t_i, \quad i \in N^0$$

Notăm aici cu

(2.9) $N^0 = \{i | n_i = n_0\} \subset N$ mulţimea operaţiilor care încep în ultima perioadă unde lotul considerat mai este supus unei prelucrări.

În afară de acest grup de restricţii, mai avem de formulat două condiţii. Prima se numeşte condiţia de continuitate şi exprimă condiţiile care asigură derularea fără întrerupere a operaţiilor. Este vorba deci de o relaţie între necunoscutele n, ξ, s ale unei perechi de operaţii care se află într-o relaţie de succesiune tehnologică ($(i, j) \in H$).

Modelul trebuie să cuprindă trei situaţii tipice (vezi figura 5).

Situaţia a) este caracterizată prin condiţiile ($(i, j) \in H$)

(2.10) $$\begin{cases} \xi_j - \xi_i \geqslant t_i \\ n_j - n_i \geqslant 0; \end{cases}$$

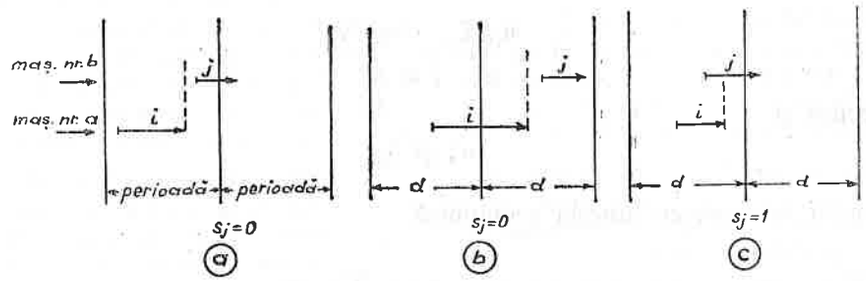

Fig. 5

Situaţia b) este redată prin

(2.11) $$\begin{aligned} \xi_j - \xi_i &\geqslant t_i - 1 \\ n_j - n_i &\geqslant 1; \end{aligned}$$

iar situaţia c) prin

(2.12) $$s_j \geqslant 1.$$

Observăm că din cauza condiţiei (2.7) restricţia (2.8 c) asigură anularea lui s_j la primele două situaţii, de asemenea că $s_j = 1$ rezultă din (2.7) şi (2.12).

Cele trei condiţii le contopim într-una singură ce conţine şi semnul relaţiei logice de disjuncţie (V):

(2.13) $$\xi_j - \xi_i \geqslant t_i,\ n_j - n_i \geqslant 0 \vee \xi_j - \xi_i \geqslant t_i - 1,\ n_j - n_i \geqslant 1 \vee s_j \geqslant 1, \quad (i, j) \in H.$$ [2]

Cea de a doua condiţie se numeşte condiţie de încărcare sau de neinterferenţă şi asigură că intervalele de timp $[\xi_i, \xi_i + t_i)$ ce reprezintă desfăşurarea unei operaţii şi care se referă la operaţii ce se execută pe aceeaşi maşină ($(i, j) \in J$) să fie disjuncte. Relaţiile conţin numai variabilele ξ_i.

Să presupunem că operaţia i o aşezăm înaintea operaţiei j. Atunci se pot ivi două situaţii (fig. 6).

În situaţia a) trebuie să avem

(2.14) $$\xi_j \geqslant \xi_i + t_i$$

În situaţia b), la condiţia (2.14) se mai adaugă restricţia

(2.15) $$\xi_j + t_j - 1 \leqslant \xi_i$$

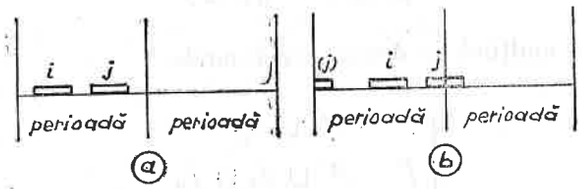

Fig. 6

[2] Virgula ce intervine în interiorul unui termen al disjuncţiei reprezintă, ca de obicei, semnul conjuncţiei logice.

condiţie ce este verificată de altfel şi în situaţia a). Ţinînd seamă şi de posibilitatea aşezării operaţiilor i şi j în ordinea inversă, condiţiile de neinterferenţă vor fi

$$\xi_j - \xi_i \geqslant t_i,\ \xi_i - \xi_j \geqslant t_j - 1 \vee \xi_i - \xi_j \geqslant t_j,\ \xi_j - \xi_i \geqslant t_i - 1 \qquad [i, j] \in J^3. \tag{2.16}$$

Formulele (2.7), (2.8), (2.13) şi (2.16) formează împreună primul model al ordonanţării operaţiilor în cazul producţiei în serie; acest model constituie o problemă de programare liniară cu condiţii logice (disjuncţii).

Pentru rezolvarea unei astfel de probleme se utilizează bine cunoscutul procedeu Branch and Bound (resp. S.E.P. resp. algoritmul lui F. RADÓ; vezi [3], [5], [15], [16]). Credem că nu este cazul să descriem această metodă, menţionăm doar că ea revine la rezolvarea unui şir de programări obişnuite, numite probleme fundamentale ale problemei considerate. În paragraful următor vom arăta forma acestor probleme fundamentale, cum se rezolvă ele şi cum se generează probleme fundamentale noi din alte probleme fundamentale deja rezolvate.

3. Problemele fundamentale

O problemă fundamentală Q constă din condiţiile obişnuite ale modelului considerat (restricţiile (2.7) şi (2.8)) şi din cîte un termen din fiecare disjuncţie a unei submulţimi a mulţimii restricţiilor (2.13) şi (2.16).

Se dau aşadar mulţimile

H_1, H_2, H_3 disjuncte două cîte două şi pentru care are loc

$$H_1 \cup H_2 \cup H_3 = H_0 \subset H; \tag{3.1}$$

J_1 şi J_2, disjuncte, pentru care are loc

$$J_1 \cup J_2 = J_0 \subset J. \tag{3.2}$$

Din aceste mulţimi se deduc următoarele:

$$\begin{cases} H' = H_1 \cup H_2 \\ J' = H' \cup J_0 \cup \bar{J}_0 \\ N_0 = \{j \mid (i, j) \in H_3\}. \end{cases} \tag{3.3}$$

[3] La această formulă, perechile $[i, j]$ sînt neordonate, adică (i, j) şi (j, i) se confundă

Aici mulţimea $\bar{A}$ de perechi de numere este definită prin relaţia

$$(i, j) \in A \Leftrightarrow (j, i) \in \bar{A}. \tag{3.4}$$

Se mai notează

$$a_{ij} = \begin{cases} 0, & (i, j) \in H_1 \\ 1, & (i, j) \in H_2. \end{cases} \tag{3.5}$$

Problema Q are atunci următoarea înfăţişare:

$$\begin{cases} \xi_j - \xi_i \geqslant t_i, & (i, j) \in H_1 \cup J_1 \cup \bar{J}_2 \\ \xi_j - \xi_i \geqslant t_i - 1, & (i, j) \in H_2 \cup \bar{J}_1 \cup J_2 \end{cases} \tag{3.6}$$

$$n_j - n_i \geqslant a_{ij}, \quad (i, j) \in H' \tag{3.7}$$

la care se adaugă restricţiile (2.8 a), (2.8 b), (2. d) şi (2.8 e):

$$\left.\begin{cases} 0 \leqslant \xi_i < 1 \\ n_i \in \mathbf{N} \\ n_0 - n_i \geqslant 0 \end{cases}\right\} i \in N \qquad \begin{cases} \xi_0 - \xi_i \geqslant t_i & i \in N^0 \end{cases} \tag{3.8}$$

şi se cere ca funcţia

$$y = n_0 + \xi_0 + \sum_{i \in N_0} \alpha_i \tag{3.9}$$

să fie minimă.

Pentru rezolvarea problemei Q observăm în primul rînd că restricţiile lui Q se împart în două grupe: prima conţine numai necunoscutele n_i, iar cealaltă numai necunoscutele ξ_i.

Vom avea astfel

$$\left.\begin{cases} n_i \in \mathbf{N} \\ n_0 - n_i \geqslant 0 \end{cases}\right\} i \in N \tag{3.10}$$

$$n_j - n_i \geqslant a_{ij} \quad (i, j) \in H' \tag{3.11}$$

cerînd ca n_0 să fie minim (problema Q_n) şi

$$\begin{cases} 0 \leqslant \xi_i < 1 & i \in N \\ \xi_0 - \xi_i \geqslant t_i & i \in N_0 \end{cases} \tag{3.12}$$

$$\xi_j - \xi_i \geqslant b_{ij} \quad (i, j) \in J' \quad (-1 < b_{ij} < 1) \tag{3.13}$$

cerînd ca ξ_0 să fie minim (problema Q_ξ).

În vederea rezolvării problemei Q_n, se ataşează relaţiilor (3.11) un graf $G(N, H')$ orientat şi cotat, atribuind unui arc $(i, j') \in H'$ lungimea a_{ij} (0 sau 1). Problema Q_n este o problemă de potenţial (vezi de ex. [1], [2] sau sau [17]) pentru care se caută aşa numita soluţie minimă universală (cu condiţia suplimentară $n_i \geqslant 1$). Din cauza lungimii 0 sau 1 a arcelor, această soluţie verifică automat prima restricţie (3.10), iar pentru n_0 obţinem

$$n_0 = \max_{i \in N} n_i. \tag{3.14}$$

Dacă problema Q_n este compatibilă, atunci eventualele circuite din G constau exclusiv din arce de lungime zero, ceea ce face ca valorile n_i aparţinînd unor vîrfuri dintr-un circuit să fie egale între ele. Dacă se formează graful redus $G^*(N^*, H^*)$ al lui G (vezi anexa 1), atunci acest graf va fi fără circuite, a cărui soluţie minimă universală se obţine foarte uşor (vezi anexa 2).

Trecînd la problema Q_ξ, relaţiilor (3.13) li se ataşează de asemenea un graf orientat cotat $\Gamma(N, J')$. Aceste condiţii reprezintă şi ele o problemă de potenţial care este compatibilă dacă şi numai dacă graful Γ nu posedă circuite de lungime strict pozitivă. Dacă această condiţie este verificată, atunci adăugîndu-se la (2.13) condiţiile $\xi_i \geqslant 0$, putem determina soluţia minimă universală a problemei. Metodele de rezolvare, bine cunoscute, sînt iterative. Cîteva îmbunătăţiri ale acestora sînt descrise în anexa 3.

Dacă şi numai dacă soluţia găsită verifică şi condiţiile $\xi_i < 1$, atunci problema Q_ξ este compatibilă şi are loc

$$\xi_0 = \max_{i \in N^0} (\xi_i + t_i). \tag{3.15}$$

Pentru compararea între ele a soluţiilor diferitelor probleme fundamentale se va folosi funcţia y din (3.9).

Dintr-o problemă fundamentală rezolvată Q se generează mai multe probleme fundamentale noi Q'_1, Q'_2 (ev. şi Q'_3), după cum urmează.

La formarea problemei fundamentale Q s-au folosit o parte $H_0 = H_1 \cup H_2 \cup H_3 \subset H$ din disjuncţiile (2.13) şi o parte $J_0 = J_1 \cup J_2 \subset J$ din disjuncţiile (2.16). Soluţia obţinută va verifica şi o parte din $H \setminus H_0$ şi

$J \setminus J_0$. Mulţimile disjuncţiilor neverificate prin soluţie le vom nota cu $K \subset H \setminus H_0$ şi $L \subset J \setminus J_0$. Din mulţimea $K \cup L$ se va alege una din disjuncţiile (2.13) sau (2.16). La restricţiile problemei Q se va adăuga pe rînd cîte un termen din disjuncţia aleasă. Dacă aceasta este de tipul (2.16) adică cu doi termeni, vor lua naştere două probleme fundamentale noi Q'_1 şi Q'_2. Dacă disjuncţia aleasă este de tipul (2.13), vor lua naştere trei probleme: Q'_1, Q'_2 şi Q'_3. Observăm că în acest ultim caz, soluţia problemei Q'_3 este identică cu cea a problemei Q, însă mulţimea N_0 folosită în calculul funcţiei y (3.9) se va mări cu un element nou.

În încheierea acestui paragraf vom recapitula pe scurt definiţiile diferitelor mulţimi introduse în decursul expunerii

(3.16)

- N : mulţimea operaţiilor (indicilor operaţiilor)
- $H \subset N \times N$: mulţimea perechilor de operaţii între care există o relaţie de succesiune tehnologică.
- $J \subset N \times N$: mulţimea perechilor de operaţii ce se execută pe aceeaşi maşină.
- $N' = \{j \mid (i, j) \in H\} \subset N$: mulţimea operaţiilor ce posedă un predecesor tehnologic.
- $H_0 = H_1 \cup H_2 \cup H_3 \subset H$: mulţimea disjuncţiilor din (2.13) folosite la formarea unei probleme fundamentale Q.
- $J_0 = J_1 \cup J_2 \subset J$: mulţimea disjuncţiilor din (2.16) folosite la formarea unei probleme fundamentale Q.
- $H' = H_1 \cup H_2$
- $J' = H' \cup J_0 \cup \bar{J}_0$
- $K \subset H \setminus H_0$: mulţimea disjuncţiilor (2.13) neverificate de soluţia problemei Q.
- $L \subset J \setminus J_0$: mulţimea disjuncţiilor (2.16) neverificate de soluţia problemei Q.
- $N_0 = \{j \mid (i, j) \in H_3\} \subset N$: operaţiile pentru care are loc $s_i = 1$.
- $N^0 = \{i \mid n_i = n_0\} \subset N$: operaţiile ce încep în ultima perioadă în care lotul considerat mai este supus unei prelucrări.

4. Simplificări şi particularizări

În cele ce urmează vom arăta cîteva particularizări ale modelului prezentat în paragraful 2. Aceste modele particularizate se rezolvă mai uşor, soluţiile lor dau însă valori y superioare celei y minime, obţinută prin rezolvarea modelului din paragraful 2. Este vorba deci de soluţii aproximative ale problemei date care pot fi însă practic foarte utile.

A) Prima aproximaţie se obţine prin condiţia suplimentară

$$s_i = 0, \quad i \in N' \tag{4.1}$$

adică renunţăm la stocurile tampon. Prin această măsură, termenul al treilea din restricţiile (2.13) se elimină, prin urmare numărul problemelor fundamentale de rezolvat se va micşora simţitor. Funcţia economică minimizată

$$z_A = n_0 + \xi_0 \tag{4.2}$$

va avea în schimb o valoare nu mai mică decît valoarea minimă a funcţiei z din (2.7) respectiv a lui y din (3.9).

B) O simplificare şi mai mare obţinem prin restricţia

$$n_0 = n_i = 1, \quad i \in N. \tag{4.3}$$

Condiţiile (2.13) se înlocuiesc de data aceasta cu

$$\xi_j - \xi_i \geqslant t_i \vee s_i = 1, \quad (i, j) \in H \tag{4.4}$$

care nu numai că reduce numărul problemelor fundamentale de rezolvat, dar şi problemele fundamentale se rezolvă mai repede, întrucît lipsesc din ele necunoscutele n_i. O problemă fundamentală va avea forma

$$\begin{cases} \left.\begin{array}{l} \xi_i \leqslant 1 - t_i \\ \xi_j \geqslant 0 \end{array}\right\} i \in N \\ \xi_j - \xi_i \geqslant b_{ij}, \quad (i, j) \in J' \end{cases} \tag{4.5}$$

şi se cere soluţia minimă universală a acestei probleme.

Dacă ea există, obţinem

$$\xi_0 = \max_{i \in N} (\xi_i + t_i) \tag{4.6}$$

şi mărimea de comparare a diferitelor probleme fundamentale va fi

$$z_B = \xi_0 + \sum_{i \in N^0} \alpha_i. \tag{4.7}$$

Menţionăm că în cazul B) mulţimea H_2 din cea de a cincea formulă (3.16) este vidă.

5. Modelul nr. 2

În acest model mai general, nu mai fixăm apriori durata perioadei la cea minimă posibilă, pe considerentul că nu este exclus cazul cînd la o perioadă ceva mai lungă, economiile obţinute prin scurtarea duratei de imobilizare a unor fonduri legate de fabricaţie neterminată să contrabalanseze diminuarea venitului net datorită prelungirii perioadei de fabricaţie, adică datorită micşorării cantităţii producţiei. Funcţia economică de optimizat este deci funcţie de d, δ şi σ. Să vedem forma acestei funcţii.

Vom introduce următoarele notaţii:

T: o durată fixă (de ex. anul)

d: durata perioadei de fabricaţie

δ: durata ciclului de fabricaţie

v: preţul de vînzare a unui lot de produse

u': cheltuielile directe (variabile) pentru un lot (salarii direct productive şi materii prime)

u'': cheltuielile indirecte (constante): regie — pe unitatea de timp.

$\sigma = \sum_i \gamma_i s_i$: valoarea totală a stocurilor tampon ($\gamma_i > 0$)

β: rata dobînzii pentru fondurile imobilizate

λ: coeficientul duratei medii de imobilizare ($0 < \lambda < 1$).

Această ultimă noţiune cere o explicaţie mai detaliată. Cheltuielile directe pe o perioadă (u') nu se cheltuiesc deodată ci repartizate pe durata δ a ciclului de fabricaţie. Durata *medie* de imobilizare a acestor fonduri pînă la recuperarea lor (sfîrşitul ciclului de fabricaţie resp. vînzare) o vom considera ca o parte, o fracţie din durata ciclului: $\lambda\delta$. Prin urmare dobînda pentru imobilizarea acestui fond va fi: $\beta\lambda\delta u'$.

În acest fel obţinem diferite sume parţiale din care se compune venitul net (pe an) al intreprinderii.

$\frac{T}{d}$: numărul perioadelor pe an;

$v\frac{T}{d}$: suma încasărilor pe an;

$u'\frac{T}{d}$: cheltuielile directe pe an;

$\beta\lambda\delta u'$: dobînzile cheltuelilor directe pe o perioadă;

$\beta\lambda\delta u'\frac{T}{d}$: idem pe an;

$u''T$: cheltuielile indirecte pe an;

$\beta\lambda d^2u''$: dobînzile cheltuielilor indirecte pe o perioadă;

$\beta\lambda du''T$: idem pe an;

$\beta\sigma d$: dobînzile pentru fondurile imobilizate prin stocurile tampon, pe o perioadă;

$\beta\sigma T$: idem pe an.

Venitul net anual se calculează din aceste mărimi fiind egal cu

$$T\left[\frac{v}{d}-\frac{u'}{d}-\frac{\beta\lambda\delta u'}{d}-u''-\beta\lambda du''-\beta\sigma\right].$$

Funcţia economică, de *maximizat* va fi atunci

$$w=\frac{A-B\delta}{d}-\left(d+\sum_{i\in N'}\alpha_i s_i\right) \tag{5.1}$$

cu coeficienţii daţi

$$\begin{cases} A=\dfrac{v-u'}{\beta\lambda u''} \\ B=\dfrac{u'}{u''} \\ \alpha_i=\dfrac{\gamma_i}{\lambda u''}. \end{cases} \tag{5.2}$$

Formula (2.7) rezultă de aici imediat cu $d=1$ şi $\delta=n_0+\xi_0$.

În afara funcţiei economice, restricţiile problemei suferă şi ele o uşoară modificare. Pentru momentele de începere a operaţiillor vom scrie:

$$x_i=d(n_i+\xi_i) \tag{5.3}$$

şi în locul formulei (2.6)

$$\delta=d(n_0+\xi_0) \tag{5.4}$$

Condiţiile (2.8) devin

$$\begin{cases} n_i\in \mathbf{N}. \\ 0\geqslant \xi_i<1 \end{cases} \quad i\in N$$
$$s_i\geqslant 0, \quad i\in N'$$
$$n_0-n_i\geqslant 0, \quad i\in N \tag{5.5}$$
$$\xi_0-\xi_i\geqslant\frac{t_i}{d}, \quad i\in N^\circ.$$

Condiţia (2.13) se transformă în

$$\xi_j-\xi_i\geqslant\frac{\lceil t_i\rceil}{d},\ n_j-n_i\geqslant 0\vee\xi_j-\xi_i\geqslant\frac{t_i}{d}-1,\ n_j-n_i\geqslant 1\vee s_j\geqslant 1, \quad (i,j)\in H \tag{5.6}$$

şi condiţiile (2.16) se modifică în

$$\xi_j-\xi_i\geqslant\frac{t_i}{d},\ \xi_i-\xi_j\geqslant\frac{t_j}{d}-1\vee\xi_i-\xi_j\geqslant\frac{t_i}{d},\ \xi_j-\xi_i\geqslant\frac{t_i}{d}-1, \quad [i,j]\in J \tag{5.7}$$

Funcţia economică (5.1) se transcrie în

$$w=\frac{A}{d}-\left(n_0+\xi_0+d+\sum_{i\in N'}\alpha_i s_i\right):\ \text{maxim!} \tag{5.8}$$

Problemele fundamentale, rezultate din acest sistem de restricţii, nu mai sînt probleme de potenţial, ba nici nu mai sînt liniare. Pentru rezolvarea lor, ele pot fi considerate ca probleme de programare (de potenţial) parametrice, cu parametrul d. Atunci urmează să se rezolve un şir de astfel de probleme, cu diferite valori pentru d, soluţia definitivă aproximîndu-se prin acest şir de probleme cu orice precizie dorită.

În acest scop este util să se ştie mai mult despre alura funcţiei: $w_{\max}(d)$. O remarcă în acest sens este

$$\lim_{d\to\infty} w_{\max}=-\infty. \tag{5.9}$$

Pentru determinarea valorii d_{opt} ce maximizează funcţia w_{max}, metoda expusă în lucrarea [14] poate fi utilă.

6. Mai multe maşini de acelaşi tip

Metoda ce urmează a fi prezentată este aproximativă, însă poate furniza rezultate apropiate de cea optimă. Ideea de bază constă în aceea că operaţiile destinate unui anumit tip de maşini de prelucrare — repartizare determinată prin procesul tehnologic — vor fi repartizate pe exemplarele tipului de maşină înaintea începerii întocmirii propriu zise a programului. Această repartizare constituie aşadar primul pas la întocmirea programului operativ. Repartizarea se face în modul următor:

Pe un anumit tip de maşini de prelucrare din care există m exemplare, urmează să fie executate ν operaţii ale căror timpi de execuţie formează mulţimea $T = \{t_1, t_2, \ldots, t_\nu\}$. Se caută o partiţie în m mulţimi a lui

$$T = \bigcup_{k=1}^{m} T_k \tag{6.1}$$

care să *minimizeze* mărimea

$$S^* = \max_{k=\overline{1,m}} S_k \tag{6.2}$$

unde s-a notat cu

$$S_k = \sum_{t_i \in T_k} t_i. \tag{6.3}$$

O astfel de repartizare a operaţiilor între exemplarele diferitelor tipuri de maşini va minimiza cu siguranţa durata d a perioadei de fabricaţie, însă nu asigură optimizarea funcţiei economice z din formula (2.7) respectiv a lui w din (5.2). Obţinem o soluţie aproximativă care poate fi însă destul de aproape de soluţia corectă, optimă.

Pentru construcţia partiţiei dăm în cele ce urmează o metodă care este şi ea una aproximativă, euristică. Este însă posibil ca acest algoritm să se preteze unor îmbunătăţiri respectiv completări prin care totuşi să se ajungă în fiecare caz la soluţia optimă căutată.

Se dă şirul de numere $T = (t_1, t_2, \ldots, t_\nu)$ ordonat necrescător. Se poate presupune că t_i sînt numere naturale. Notăm cu

$$S = \sum_{i=1}^{k} t_i \tag{6.4}$$

şi

$$M = \frac{S}{m}. \tag{6.5}$$

În vederea construirii submulţimilor $T_k \subset T$ se vor separa întîi elementele t_i pentru care are loc

$$t_i \geqslant M.$$

Fiecare din aceste elemente formează el singur cîte o mulţime M_k. În cele ce urmează notăm cu $[x]'$ cel mai mic număr întreg care nu este mai mic decît numărul x. Vom alege numărul

$$P = \max\ ([t_1]',\ [M]'). \tag{6.6}$$

O submulţime T se formează în continuare din cel mai mare număr $t_i \in T$ încă nefolosit; la acesta se adaugă următorul cel mai mare număr $t_j \in T$ (de asemenea încă nefolosit) pentru care are loc

$$t_i + t_j \leqslant P.$$

Se procedează în acest fel mai departe, respectînd totdeauna condiţia

$$S_k = \sum_{t_i \in T_k} t_i \leqslant P. \tag{6.7}$$

Pe urmă se trece la întocmirea submulţimii T_{k+1}. Dacă s-au construit toate submulţimile T_k cu $S_k \leqslant P$ şi încă au mai rămas elemente t_i nefolosite, urmează o procedură de completare. Anume cel mai mare din elementele rămase nefolosite se adaugă la submulţimea T_k avînd cea mai mică sumă S_k. Acest procedeu se continuă pînă ce toate elementele din T au fost repartizate.

Se dau în continuare cîteva exemple ilustrative.

Exemplul nr. 1.

$T = (10, 10, 10, 8, 8, 8, 7, 6, 6, 4, 3, 1, 1)$ $\nu = 13$ $S = 82$ $m = 3$

$M = 27{,}33$ $P = 28$

$T_1 = \{10, 10, 8\}$ $S_1 = 28$

$T_2 = \{10, 8, 8, 1, 1\}$ $S_2 = 28$

$T_3 = \{7, 6, 6, 4, 3\}$ $S_3 = 26$, soluţie optimă

Exemplul nr. 2.

Acelaşi T, ν şi S; $m = 6$ $M = 13, 67$ $P = 14$

$T_1 = \{10, 4\}$ $S_1 = 14$

$T_2 = \{10, 3, 1\}$ $S_2 = 14$

$T_3 = \{10, 1\}$ $S_3 = 11$

$T_4 = \{8, 6\}$ $S_4 = 14$

$T_5 = \{8, 6\}$ $S_5 = 14$

$T_5 = \{8\}$ $S_6 = 8$

A rămas nefolosit elementul 7. Procedura de completare îl adaugă la T_6 şi se obţine $T'_6 = \{8, 7\}$ $S'_6 = 15 = S^*$. Soluţie optimă.

Exemplul nr. 3.

$T = (11, 10, 9, 7, 5, 4, 3, 2)$ $\nu = 8$ $S = 51$

$m = 3$ $M = P = 17$

$T_1 = \{11, 5\}$ $S_1 = 16$

$T_2 = \{10, 7\}$ $S_2 = 17$

$T_3 = \{9, 4, 3; 2\}$ $\begin{cases} S_3 = 16 \\ S'_3 = 18, \end{cases}$

nu este soluţie optimă. O astfel de repartiţie optimă este

$T_1 = \{11, 4, 2\}$

$T_2 = \{10, 7\}$ $S_1 = S_2 = S_3 = S^* = 17.$

$T_3 = \{9, 5, 3\}$

O b s e r v a ţ i e. Pentru ordonanţarea operaţiilor în cazul producţiei de unicate sau serie mică, există modele şi metode de rezolvare care se bazează în majoritatea cazurilor tot pe proceduri de tip Branch and Bound. Problemele nu sînt prea dificile în cazul cînd fiecare maşină de prelucrare este de alt tip (vezi [6], [18]), devin însă foarte complexe, cînd există mai multe exemplare din acelaşi tip de maşină ([4], [7], [8], [12], [13]).

Metoda de repartizare prealabilă a operaţiilor pe diferitele exemplare dintr-un tip de maşină reduce problema ordonanţării studiate la cazul mai simplu cînd din fiecare tip de maşină există un singur exemplar. Este o metodă de rezolvare cu aproximaţie, dar poate fi foarte utilă.

A n e x a nr. 1

Un graf orientat se numeşte *tare conex*, dacă de la orice vîrf al său duce un drum spre orice alt vîrf din graf. O componentă tare conexă a unui graf este un subgraf maximal tare conex al grafului considerat.

Următoarele proprietăţi sînt evidente:

— orice vîrf dintr-un graf tare conex face parte din cel puţin un circuit;

— dacă un vîrf dintr-un circuit aparţine unei componente tare conexe, atunci toate vîrfurile circuitului aparţin acestei componente;

— orice graf conţine cel puţin o componentă tare conexă;

— vîrfurile componentelor tare conexe ale unui graf constituie o partiţie pentru mulţimea vîrfurilor grafului.

Din graful G se poate deduce un alt graf $G'(N', H')$ numit graful redus al lui G, cu următoarele proprietăţi:

— între vîrfurile lui G' şi între componentele tare conexe ale lui G există o corespondenţă biunivocă: fiecărui vîrf $i \in N'$ îi corespunde o componentă tare conexă $G_i(N_i, H_i)$ din G şi invers;

— fie două vîrfuri $i, j \in N'$ din G'; în graful redus G' există un arc $(i, j) \in H'$ dacă şi numai dacă în componentele tare conexe G_i respectiv G_j — ce corespund vîrfului i respectiv j — există un vîrf $\alpha \in N_i$ respectiv un vîrf $\beta \in N_j$ în aşa fel încît să aibă loc $(\alpha, \beta) \in H$.

Este evident că graful redus este fără circuite.

Indicăm un algoritm foarte simplu ([1]) pentru determinarea tuturor vîrfurilor unei componente tare conexe, ce conţine un vîrf $i \in H$ dat.

— vîrful i se marchează $\pm$;

— orice succesor al unui vîrf marcat cu $+$ se marchează $+$;

— orice predecesor al unui vîrf marcat cu $-$ se marchează $-$;

Mulţimea vîrfurilor marcate cu $\pm$ constituie componenta căutată. Acest algoritm permite găsirea tuturor componentelor tare conexe ale unui graf, deci şi construirea grafului redus.

A n e x a 2

Vîrfurile unui graf fără circuite pot fi aranjate (clasificate) pe nivele. La nivelul nr. 1 se găsesc vîrfurile fără predecesor (acest nivel nu poate fi vid la un graf fără circuite). La nivelul $k > 1$ se găsesc vîrfurile avînd toţi predecesori la nivele inferioare cel puţin un predecesor aflîndu-se la nivelul $k - 1$.

Să considerăm problema de potenţial în forma simplă care intervine în lucrarea de faţă: necunoscutele x_i, $i \in N = \{1, 2, \ldots, n\}$ verifică restricţiile

$$(A\ 2.1) \quad \begin{cases} x_i \geqslant 0, & i \in N \\ x_j - x_i \geqslant a_{ij} & (i, j) \in H \subset N \times N \quad (a_{ij} \in \mathbf{R}). \end{cases}$$

Să presupunem că graful $G(N, H)$, ataşat problemei nu are circuite şi vîrfurile sale au fost aranjate pe nivele. Nivelul nr. k conţine ca vîrfuri elementele submulţimii N_k, unde $N_1, N_2, \ldots$ formează o partiţie a lui N.

Soluţia minimă universală se obţine prin determinarea pe rînd, în ordinea nivelelor, a necunoscutelor.

Pentru necunoscutele ataşate vîrfurilor din nivelul 1 se pune:

$$x_i = 0, \quad i \in N_1. \tag{A. 2.2}$$

La determinarea necunoscutelor ataşate vîrfurilor de pe nivelul k, necunoscutele aparţinînd nivelelor inferioare au fost deja determinate. Are loc:

$$x_i = \max_{r \in P(i)} (x_r + a_{ri}) \quad i \in N_k \quad (k > 1), \tag{A 2.3}$$

unde mulţimea predecesorilor vîrfului i s-a notat cu $P(i)$.

În cele ce urmează se indică un algoritm simplu pentru determinarea partiţiei în nivele a vîrfurilor unui graf fără circuite; credem că acest algoritm (cel puţin în această formă) nu a fost încă publicat.

— Toate vîrfurile fără predecesor se marchează cu 1;

— Orice alt vîrf va fi marcat, dacă toţi predecesorii săi au fost marcaţi; dacă marca cea mai mare a predecesorilor este k, atunci vîrful în cauză se marchează cu $k + 1$.

Anexa nr. 3

Determinarea soluţiei minime universale a unei probleme (A 2.1), dacă graful ataşat posedă circuite, se face cu metode iterative. Iată un astfel de procedeu:

Ca primă iteraţie se porneşte cu

$$x_i^{(1)} = 0, \quad i \in N. \tag{A 3.1}$$

Să presupunem că s-a obţinut cea de a r — a iteraţie. Iteraţia a $r + 1$ — a va fi atunci

$$x_i^{(r+1)} = \max_{s \in P_0(i)} (x_s^{(r)} + a_{si}), \quad i \in N \tag{A 3.2}$$

unde $P_0(i) = P(i) \cup \{i\}$ şi s-a pus $a_{ii} = 0$, $i \in N$.

Dacă problema (A 2.1) este compatibilă (graful ataşat problemei nu admite circuite de lungime strict pozitivă) atunci după cel mult n interaţii ($r \leqslant n$) va avea loc

$$x_i^{(r-1)} = x_i^{(r)}, \quad i \in N. \tag{A 3.3}$$

În cazul contrar, problema este incompatibilă. Există mai multe posibilităţi de a accelera procedeul iterativ descris aici. Vom arăta cîteva din astfel de metode.

a) Dacă cunoaştem un vector ξ_i, $i \in N$ ce verifică numai o parte din cel de al doilea grup din condiţiile (A 2.1) atunci ca soluţie de pornire putem folosi în locul mărimilor $x_i^{(1)} = 0$ din (A 3.1) şi mărimile

$$x_{(i)}^{(1)} = \xi_i, \quad i \in N \tag{A 4.3}$$

ceea ce accelerează în general calculele.

b) Să presupunem că la iteraţia $r + 1$ — a, s-au calculat deja primele $i - 1$ componente: $x_1^{(r+1)}, x_2^{(r+1)}, \ldots, x_{i-1}^{(r+1)}$. Atunci, dacă pentru determinarea celei de a i — a componente, intervin şi componente cu nr. de ordine mai mic, valorile acestora vor fi folosite din iteraţia $r + 1$. Notînd cu $P_-(i)$ mulţimea predecesorilor vîrfului i din graful G ataşat sistemului (A 2.1), predecesori al căror indice j este mai mic decît i, iar cu $P_+(i)$ restul predecesorilor vîrfului i inclusiv pe acesta, adică

$$\begin{cases} P_-(i) = \{j \mid (j, i) \in H\} \cap \{1, 2, \ldots, i - 1\} \\ P_+(i) = P_0(i) - P_-(i) \end{cases} \tag{A 3.5}$$

vom folosi sistemul de valori

$$x_i^{(r+1)} = \max \left[\max_{s \in P_-(i)} (x_s^{(r+1)} + a_{si}), \max_{s \in P_+(i)} (x_s^{(r)} + a_{si})\right], \; i \in N. \tag{A 3.6}$$

Măsură în care această metodă reduce volumul calculelor depinde de ordinea în care componentele x_i vor fi calculate. O indicaţie utilă în această privinţă ne poate da cea de a treia metodă.

c) În primul rînd se va arăta o metodă cu ajutorul căreia un graf poate fi descompus în două grafe parţiale $G_1(N, H_1)$ şi $G_2(N, H_2)$ unde H_1 şi H_2 constituie o partiţie a arcelor din H şi unde grafele G_1 şi G_2 sînt fără circuite.

Să facem abstracţie pentru moment de orientarea arcelor din graful G, deci le considerăm ca muchii. Le dăm o nouă orientare printr-un procedeu pe care-l vom explica mai jos, obţinînd graful $G'(N, H')$, fără circuite. În acest scop, alegem un vîrf oarecare i şi dăm muchiilor incidente orientarea

spre exterior, după care marcăm vîrful i. Urmînd orientarea acestor arce, ajungem la alte vîrfuri; aici de asemenea se va da muchiilor incidente ce nu sînt încă orientate, orientarea spre exterior şi se va marca vîrful respectiv. Se vede că prin acest procedeu ajungem la vîrfuri ce pot poseda muchii incidente încă neorientate sau arce orientate spre interior, nicidecum arce incidente spre exterior.

Demonstrăm afirmaţia că din vîrful i marcat nu putem ajunge la vîrful j care posedă în această fază un arc incident spre exterior.

Să presupunem că j nu este marcat. Atunci el nu poate avea arce incidente spre exterior, căci astfel de orientări se fixează după o marcare.

Să presupunem că j a fost anterior marcat (înaintea marcării lui i); cu ocazia marcării lui j muchia $[i, j]$ a obţinut orientarea (j, i). Prin urmare după marcarea lui i, nu mai ajungem la j, întrucît arcul ce leagă aceste două vîrfuri posedă orientarea (j, i).

În consecinţă, prin procedeul descris, nu vom închide niciodată vreun circuit.

Schimbînd orientarea arcelor din G', obţinem graful $G''(N, H'')$ care de asemenea nu are circuite.

Grafele căutate $G_1(N, H_1)$ şi $G_2(N, H_2)$ se obţin atunci prin

$$(A\ 3.7) \qquad H_1 = H \cap H' \quad \text{şi} \quad H_2 = H \cap H''.$$

Sistemul S de inegalităţi din $(A\ 2.1)$ se descompune şi el în două subsisteme: S_1 cu $(i, j) \in H_1$ şi S_2 cu $(i, j) \in H_2$. Procedeul de rezolvare este următorul:

Determinăm soluţia minimă universală a sistemului S_1 (formulele $(A\ 2.2)$ şi $(A\ 2.3)$). Notăm această soluţie cu $z_i^{(1)}$, $i \in N$. Pe urmă determinăm soluţia minimă universală a sistemului S_2 la care s-au adăugat restricţiile

$$(A\ 3.8) \qquad x_i \geqslant z_i^{(1)}, \quad i \in N.$$

Soluţia se obţine cu ajutorul unor formule foarte asemănătoare cu $(A\ 2.2)$ şi $(A\ 2.3)$:

$$(A\ 3.9) \qquad \begin{cases} x_i = z_i^{(1)}, & i \in N_1 \\ x_i = \max\,[z_i^{(1)}, \max\limits_{s \in P(i)} (x_s + a_{si})], & i \in N_k,\ k = 2, 3, \ldots \end{cases}$$

obţinînd în acest fel soluţia $z_i^{(2)}$, $i \in N$.

Prin rezolvarea alternată a sistemelor S_1 şi S_2 folosind totodată şi soluţiile din iteraţia premergătoare (în sensul formulei $(A.\ 3.9)$) obţinem în cel mult n paşi soluţia problemei, dacă ea este compatibilă.

VERBESSERUNGEN DER ABLAUFPLANUNG IN DER SERIENPRODUKTION MIT PUFFERBESTÄNDEN

ZUSAMMENFASSUNG

In vorliegender Arbeit wird der Begriff der Herstellungsperiode einer tieferen Analyse unterzogen. Es zeigt sich dass dieser Begriff verbessert werden kann. Die neue Definition zeigt auch neue Möglichkeiten der Ablaufplanung, mit erhöhter wirtschaftlicher Wirksamkeit.

Es werden mehrere mathematische Modelle dieser Problemstellung aufgestellt, wie auch die entsprechenden Lösungsmethoden.

BIBLIOGRAFIE

[1] C. Berge, *Théorie des graphes et ses applications.* Paris, (1958).

[2] C. Berge, A. Ghouila-Houri, *Programmes, jeux et réseaux de transport.* Paris, (1962).

[3] P. Bertier, B. Roy, *Une procédure de résolution pour une classe de problèmes pouvant avoir un caractère combinatoire.* Bull. du Centre Intern. de calcul, Rome, **4**, *1*, 19–28 (1965).

[4] R. Descamps, P. Chevignon, *Algorithmes d'optimisation pour une classe générale de problèmes d'ordonnancement avec limitation des ressources.* Metra **V**, *4*, 603–618 (1966).

[5] E. Ignall, L. Schrage, *Application of the Branch and Bound Technique to some Flow-Shop Scheduling Problems.* JORSA **13**, 3, 400–412 (1965).

[6] L. Németi, *Das Reihenfolgeproblem in der Fertigungsprogrammierung und Linearplanung mit logischen Bedingungen.* Mathematica (Cluj), *6*, 1, 87–99 (1964).

[7] — *Über das Reihenfolgeproblem in der Fertigung im Falle mehrerer Maschinen derselben Art.* Rev. Roum. Math. Pures et Appl. **13**, 835–840 (1968).

[8] — *Sur la programmation temporelle de la fabrication.* Metra, **VII**, 327–338 (1968).

[9] — *Das Reihenfolgeproblem in der Fertigung mit Pufferbeständen.* Rev. Roum. Math. Pures et Appl. **13**, 1009–1016 (1968).

[10] — *Sur le problème de l'ordonnancement dans la fabrication en série.* Rev. Fr. Inf. Rech. Op. **2**, 47–60 (1968).

[11] — *Über die Durchführungsdauer eines Arbeitsganges in der Ablaufplanung.* Bull. Math. Soc. Sc. Math. R.S. Roum. **12**, *3*, 81–86 (1968).

[12] — *On the Scheduling Problem in the Case of Several Machines of the Same Type.* Mathematica (Cluj), **13**, *2*, 251–261 (1971).

[13] L. Németi, F. Radó, *Sur la programmation temporelle de la fabrication.* Mathematica (Cluj), **8**, 1, 109–111, (1966).

[14] F. Radó, *Le calcul approximatif des extrêmes d'une fonction.* Mathematica (Cluj) **3**, *1*, 171–177 (1961).

[15] — *Programare liniară cu condiţii logice.* Comun. Acad. R.P.R. **13**, 1039–1042 (1963).

[16] — *Un algorithme pour résoudre certains problèmes de programmation mathématique.* Mathematica (Cluj), **6**, *1*, 105–116 (1964).

[17] B. Roy, *Cheminement et connexité dans les graphes. Application aux problèmes d'ordonnancement.* Metra, Série speciale no. **1**, 1962.

[18] B. Roy, B. Sussmann, *Les problèmes d'ordonnancement avec contraintes disjonctives.* SEMA, Direction scientifique, Rapport de recherche no. **9**, 1964.

Primit la 18. V. 1972

Academia Republicii Socialiste România
Filiala din Cluj
Institutul de calcul