EGY ERŐFORRÁS-ALLOKÁLÓ ELJÁRÁS

SZŐKE ZOLTÁN

(Közlésre érkezett: 1974. december 9.)

A CPM hálótervezési módszer egy hálóterv minden tevékenységéhez hozzárendel egy időtartamot, ami a tevékenység megvalósításához szükséges időt. Ez az idő tehát rögzített érték, és úgy értelmezhető, hogy folytonosan (megszakítás nélkül) telik el a tevékenység megkezdése és befejezése között. A tevékenységek ugyanakkor munkafeladatokat jelentenek, amelyeket erőforrásokkal (emberi munka, gép, idő, anyag, pénz stb...) végezhetünk el. Ezek az erőforrások viszont általában korlátozott kapacitással rendelkeznek, ezért előfordulhat, hogy az egyszerre elvégezhető tevékenységek megvalósítására nem jut elegendő erőforrás. Felmerül a kérdés, hogy az egyes tevékenységek mikor és milyen erőforrás-mennyiséggel kerüljenek megvalósításra. Az ilyen problémák megoldására szolgálnak az erőforrás-allokáló módszerek, amelyek az erőforráskorlátokat mind az egész háló, mind az azonos időszakban ütemezett tevékenységek vonatkozásában figyelembe veszik. Ezek közé tartozik RAMPS és a magyar ERALL—2 módszer, amit már több országban sikeresen alkalmaznak.

Az ERALL—2 eljárás elsősorban az építőipar számára készül, így feltételi rendszere és célfüggvénye is elsősorban az építőipari kivitelező és tervező szervek struktúráját és érdekeit tükrözi.

Az eljárás többhálós
többkapacitásos
determinisztikus
diszkrét változójú
párhuzamos módszerű
dinamikus programozási feladat modellje.

Az eljárás változtatható sebességgel dolgozik, a kezelt tevékenységek különböző jellegűek lehetnek és különleges kikötések is tehetők.

Az ütemezési feladat megfogalmazása:

(Legyen P egy feladat irányított, körútmentes hálóterve.)

1. A feladat elvégzéséhez szükséges erőforrásokat jelölje

$$(k = 1, 2, \dots, K)$$

2. Az $(i, j)^{a_k}_{\in P}$ tevékenységhez rendelt, a_k erőforrással kapcsolatos munkamennyiséget jelölje M_{ij}^{ak}.

3. Minden (i, j) tevékenységhez hozzárendelünk egy $\underline{c}_{ij}^{ak}$ illetve $\overline{c}_{ij}^{ak}$ ún. helyi alsó, illetve helyi felső korlátot. Ezek azt az erőforrás-mennyiséget jelentik az erőforrásfajtából, mely a tevékenység végzéséhez (elsősorban technikai ill. technológiai okokból) legalább ill. legfeljebb szükséges. Természetesen

$$0 \leq \underline{c}_{ij}^{ak} \leq \overline{c}_{ij}^{ak}$$

4. Az egész modellre nézve értelmezve van az — egyszerűség kedvéért C^a-val jelölt — C^{ak} globális erőforráskorlát, amely az a_k erőforrásból a teljes terv időszakában rendelkezésre

$$C^a \geq \overline{c}_{ij}^{ak}, \quad (i, j) \in P$$

(Megjegyezzük, hogy C^a változó is lehet.)

5. Jelölje T az éppen ütemezésre kerülő időpediódust, (T = 1, 2, . . . ,), és X_{ij}^{ak} (T) a T időperiódusban az (i, j) tevékenységen az a_k erőforrásból ütemezett mennyiséget, T_j pedig a j esemény tényleges bekövetkezési időpontját (a legelső időperiódust, amelyben már az összes $(i_r, j) \in P$ alakú tevékenység befejeződött).

6. Ezen fogalmak alapján az *ütemezési feladat:*
P-halmaz olyan időbeli elrendezése (ütemezése), hogy tetszőleges, de rögzített időperiódusra nézve a

$$\sum_{(i, j) \in P} x_{ij}^{ak}(T) \leq C^a$$

egyentlőtlenség fennállása mellett és az egyes tevékenységekre előírt különleges kikötések betartásával $T_n \rightarrow \min$ legyen (ahol n a háló befejező eseményének a kódszáma).

Az ütemezés alapelve a következő:

Legyen Q(T) a T időperiódusban az ütemezhető tevékenységek halmaza, Q'(T) a T időperiódusban már beütemezett tevékenységek halmaza, és Z^a(T) az a_k erőforrásból a T időperiódusban még rendelkezésre álló mennyiség.

$$Z^a(T) = C^a - \sum_{(i, j) \in Q'(T)} x_{ij}^{ak}(T). \tag{1}$$

Végül legyen M_{ij}^{ak} (T) a T időperiódusban az (i, j) tevékenységen még hátralevő munkamennyiség.

(i, j) $\in$ Q(T), azaz az (i, j) a T időperiódusban ütemezhető, ha

$$\left.\begin{aligned} \sum_{(h, i) \in P} \sum_{k=1}^{k} M_{hi}^{ak}(T) &= 0 \\ \sum_{k=1}^{k} M_{ij}^{ak}(T) &> 0 \end{aligned}\right\} \tag{2}$$

Az eljárás kiszámítja a tevékenységnek rohamtartamait:

$$d_{ij} = \max_k \frac{M_{ij}^{ak}}{c_{ij}^{ak}}$$

és a rohamtartamokkal elvégzi a CPM-időtervezést.

Ezután kezdődik a tulajdonképpeni ütemezés. Az algoritmus az időpontban ütemezhető tevékenységeket sorba rendezi maximális időtartalékuk növekvő sorrendje szerint, és ebben a sorrendben ütemezi őket, míg a rendelkezésre álló erőforrások ki nem merülnek.
A tevékenységekre ütemezett mennyiségek általában:

$$x_{ij}^{ak}(T) = \min\{\bar{c}_{ij}^{ak},\ Z^a(T),\ M_{ij}^{ak}(T)\} \quad k = 1, 2, \ldots K$$

Megjegyzés:

Az „általában" kifejezés itt arra utal, hogy ha az egyes tevékenységeken levő különleges kikötések és a korlátozó feltételek megkívánják, akkor az $X_{ij}^{ak}(T)$ ütemezésre kerülő erőforrás-mennyiség ennek megfelelően módosul. Ezután T helyébe T + 1-et írunk, és az ütemezést ismét elvégezzük. Az eljárást addig folytatjuk, amíg minden tevékenység ütemezésre kerül.

Egy új erőforrás-allokáló algoritmus

A továbbiakban az 1—6. pontokban leírt ütemezési feladatnak egy új megoldását adjuk. Az eljárás heurisztikus, optimális voltát nem is igyekszünk bizonyítani. Modellünkben minden tevékenységhez több, különböző erőforrásokkal végezhető munkamennyiség rendelhető, amelyeket egymástól függetlenül ütemezhetünk.
Az eljárás:
többkapacitásos,
párhuzamos rendszerű (az ütemezhető tevékenységeket és a tevékenységeken az erőforrásokat egyszerre kezeli);
változtatható sebességgel dolgozik, és bármelyik erőforrással tetszőleges tevékenységen végzett munka bármikor, tetszőleges időre megszakítható, továbbá
diszkrét változójú és
dinamikus.
Az algoritmus a korlátozó feltételek betartása mellett a befejezési határidő minimalizálására törekszik „kritikus út kiegyenlítéssel" (azaz a maximális tartalékidők kihasználásával). Egy ütemezés ugyanis — adott kapacitásszintek mellett — akkor biztosan optimális, ha az ütemezés után minden út kritikus.

A gyakorlatban egy tevékenység megszakítása általában költséggel jár. Bár az algoritmus költségtervezéssel nem foglalkozik, mégis figyelembe veszi ezt a tényezőt, ugyanis megszakítást csak akkor alkalmaz, ha — az ütemezési időperiódusban a minimális erőforrás-szükséglet nagyobb, mint a C^a globális erőforráskorlát.

Az algoritmus ismertetése.

$Q^{a_k}(T)$ jelentse a T időperiódusban az a_k erőforrással ütemezhető tevékenységek halmazát.
$(i, j) \in Q^{a_k}(T)$, ha (i, j) eleget tesz (2)-nek és $M_{ij}^{a_k}(T) > 0$. $A = A(T)$ jelentse a T időperiódusban ütemezhető erőforrások halmazát.

$$a_k \in A, \text{ ha } Q^{a_k}(T) \neq \emptyset \text{ (azaz ha } \exists (i, j) \in Q^{a_k}(T)\text{)}.$$

Az ütemezés kezdetén T a kívánt kezdési idővel egyenlő. Legyen yij(T) az (i, j) tevékenység elvégzéséhez a periódusban még rendelkezésre álló idő:

$$yij(T) = tse\,(i, j) - T$$

yij(T) az a legnagyobb idő, amely alatt az (i, j) tevékenység elvégezhető anélkül, hogy az egész háló határidő-elcsúszást szenvedne.
Kiszámítjuk az egyes tevékenységekhez tartozó munkamennyiségek elvégzésének rohamtartamát, a $d_{ij}^{a_k}$-t:

$$d_{ij}^{a_k} = \frac{M_{ij}^{a_k}}{\overline{c}_{ij}^{a_k}}$$

(Ha $\frac{M_{ij}^{a_k}}{\overline{c}_{ij}^{a_k}}$ nem egész, akkor felkerekítjük.)

Az $M_{ij}^{a_k}(T)$ aktuális munkamennyiséggel képzett rohamtartamokat, aktuális rohamtartamnak nevezzük, és $d_{ij}^{a_k}(T)$-vel jelöljük. A továbbiakban — ha mást nem mondunk — egy osztás nem egész eredményét mindig felfelé kerekítjük.
Egy (i, j) tevékenység rohamtartama: $dij = \max\limits_{k} \{d_{ij}^{a_k}\}$
A dij rohamtartamokkal elvégezzük a CPM/TIME számítást. Ezután kezdődik a tulajdonképpeni ütemezés, melynek algoritmusa a következő:

1. Vegyük az (o, h) alakú tevékenységeket. Határozzuk meg az A és a $Q^{a_k}(T)$ halmazokat. Menjünk a 2-hoz.
2. A_1 legyen azon a_k erőforrások halmaza, amelyekre

$$\sum_{(i, j) \in Q^{a_k}(T)} \min \left\{ \overline{c}_{ij}^{a_k}, \max \{ M_{ij}^{a_k}(T), \underline{c}_{ij}^{a_k} \} \right\} \leq C^{a} \qquad (3)$$
$$a_k \in A$$

teljesül. $A_1 \subset A$.

Legyen $A_2 = A - A_1$
Ha $A_2 \neq \emptyset$, akkor menjünk 3-hoz, egyébként menjünk 7-hez.

3. Számítsuk ki yij(T)-t.

$$\text{Legyen } \underline{c}'^{a_k}_{ij}(T) = \max \left\{ \frac{M_{ij}^{a_k}(T)}{yij(T)}, \underline{c}_{ij}^{a_k} \right\} \begin{matrix} (i, j) \in Q^{a_k}(T) \\ a_k \in A_2 \end{matrix}$$

Azon a_k erőforrások halmazát, amelyekre

$$\sum_{(i,j) \in Q^{a_k}(T)} \underline{c}'^{a_k}_{ij}(T) \leq C^a \qquad a_k \in A_2$$

teljesül, jelölje A_3. Legyen $A_4 = A_2 - A_3$. Menjünk 4-hez.

4. Legyen

$$Z^a(T) = C^a - \sum_{(i,j) \in Q^{a_k}(T)} \underline{c}'^{a_k}_{ij}(T), \quad a_k \in A_3$$

$Z^a(T)$ a T időperiódusban még ki nem használt kapacitást jelenti. Ezt kell szétosztani még az ütemezésre kerülő tevékenységek között.
Ezt az elosztást a következő algoritmus végzi el:

a) Vegyük az $(i, j) \in Q^{a_k}(T)$ $a_k \in A_3$) tevékenységeket, rendezzük sorba őket yij(T) értékeik csökkenő nagysága szerint. Vegyük a sorrendben az első tevékenységet, és menjünk b)-hez.

b) $\underline{c}'^{a_k}_{ij}$-t egészítsük ki a megengedett legnagyobb mértékkel:

$$c'^{a_k}_{ij} = \min \{ \min \{ \bar{c}^{a_k}_{ij}, M^{a_k}_{ij} \} - \underline{c}'^{a_k}_{ij}, Z^a(T) \}$$

Az így kapott érték lesz az $x^{a_k}_{ij}(T)$ ütemezendő mennyiség:

$$x^{a_k}_{ij}(T) = \underline{c}'^{a_k}_{ij}(T) + c'^{a_k}_{ij}(T)$$

c) -t csökkentsük a kiegészítés mértékével. Ha az így kapott $Z^a(T) \neq 0$, akkor a sorban még hátralevő (i, j) tevékenységekre:

$$x^{a_k}_{ij}(T) = \underline{c}'^{a_k}_{ij}(T)$$

Menjünk e)-hez. Ha pedig $Z^a(T) \neq 0$, akkor menjünk d)-hez.

d) Ha már minden $\underline{c}'^{a_k}_{ij}$-t kiegészítettünk, akkor menjünk e)-hez, egyébként vegyük a sorrendben következőt, és menjünk b)-hez.

e) Ha $A_4 \neq \emptyset$, menjünk 5-höz, egyébként 7-hez.

5. Rakjuk sorba az (i, j) tevékenységeket az a_k erőforrásra vonatkozó maximális időtartalékuk növekvő sorrendjében.

$(a_k \in A_4)$

Egy tevékenységnek a T időperiódusban az a_k erőforrásra vonatkozó maximális időtartaléka:

$$pm(i, j) = tse(i, j) - tfa(i, j) - d^{a_k}_{ij}(T) \qquad (4)$$

Az ütemezés pillanatában tfa i, j) = T; T minden tevékenység maximális időtartalékában szerepel, így az összehasonlításban elhagyható. Elegendő tehát a következő időtartamot kiszámítani (4) helyett:

$$pm'(i, j) = tse(i, j) - d^{a_k}_{ij}(T) \qquad (5)$$

Ha két tevékenységre pm'(i, j) megegyezik, akkor azt vegyük előbb, amelyiknek $d^{a_k}_{ij}(T)$ rohamtartama kisebb. Menjünk 6-hoz.

6. Vegyük minden $a_k \in A_4$ erőforrásra az 5-ben megállapított sorrendben a c'^{ak}_{ij} -k összegét mindaddig, míg

$$\sum_{(i,j) \in Q^{ak}(T)} c'^{ak}_{ij}(T) \leqq C^a \qquad (6)$$

teljesül, de ha még egyet hozzávennénk, akkor (6) már nem teljesülne. A (6)-ban szereplő tevékenységek halmazát jelölje Q(T). Legyen

$$t^{ak} = \min_{(i,j) \in Q_6^{ak}(T)} \{ yij(T) \}$$

$$y^{ak} = | \min \{ (yij(T) - t^{ak} - d^{ak}_{ij}(T)) \mid yij(T) - t^{ak} - d^{ak}_{ij}(T) < 0,$$
$$(i,j) \in (Q^{ak}(T) - Q_6^{ak}(T)) \} |$$

$$y = \max_k \{ y^{ak} \}$$

Módosítsuk a háló befejezési időpontját, s ezzel együtt a még be nem következett j események legkésőbbi bekövetkezési ts(j) időpontját az y időtartammal.

Legyen most

$$Z^a(T) = C^a - \sum_{(i,j) \in Q_6^{ak}(T)} c^{ak}_{ij}(T)$$

Végezzük el az 5-ben leírt kiegészítési algoritmust azzal a módosítással, hogy az e) pont helyébe a következőt írjuk: e) Menjünk 7-hez.

7. Legyen

$$d'^{ak}_{ij}(T) = \left[\frac{M^{ak}_{ij}(T)}{x^{ak}_{ij}(T)} \right], \quad \begin{array}{l} (i,j) \in Q^{ak}(T), \ a_k \in A - A_4 \\ (i,j) \in Q_6^{ak}(T), \ a_k \in A_4 \end{array}$$

ami azt az időtartamot jelöli, amely alatt az (i, j) tevékenység $x^{ak}_{ij}(T)$ kapacitással folytonosan üzemelhető.
([ ] egész részt jelent.)
Legyen most

$$d^{ak} = \min_{(i,j)} \{ d'^{ak}_{ij} \}, \ (i,j) \in Q^{ak}(T), \ a_k \in (A - A_4) \cup A_5$$

$$d = \min_k \{ d^{ak} \}, (i,j) \in Q_6^{ak}(T), \ a_k \in A_4$$

Az első ütemezési periódusban $A_5 := \emptyset$.
Legyen

$$A_6 \overset{def}{=} \{ a_k \mid d^{ak} = d \},$$

$Q_7^{ak}(T)$ azon (i, j) tevékenységek halmaza,

$$\text{amelyre } d = d^{ak}_{ij}(T), \ (a_k \in A_6).$$

A $(T, T + d)$ időintervallumban $x^{ak}_{ij}(T)$ kapacitással ütemezünk be minden $(i, j) \in Q^{ak}(T)$, $a_k \in (A - A_4) \cup A_5$ és $(i, j) \in Q_6^{ak}(T)$, $a_k \in A_4$ tevékenységet.

Ezeken a tevékenységeken a $T + d$ időpontban az aktuális munkamennyiség:

$$M_{ij}^{a_k}(T + d) = M_{ij}^{a_k}(T) - d \cdot x_{ij}^{a_k}(T)$$

Az $(i, j) \in Q^{a_k}(T) - Q_6^{a_k}(T)$, $a_k \in A_4$ tevékenységekre:

$$M_{ij}^{a_k}(T + d) = M_{ij}^{a_k}(T)$$

Legyen $$Q \overset{\text{def}}{=} \{ (i, j) \mid (i, j) \in \bigcup_{a_k \in A_6} Q_7^{a_k}(T),\ M_{ij}^{a_k}(T + d) = 0 \}$$

$$A_7 \overset{\text{def}}{=} \{ a_k \mid Q^{a_k}(T + d) \neq \emptyset \}$$

$$Q_o^{a_k}(T + d) := Q^{a_k}(T),\ a_k \in A \cup A_5 - A_6$$

$$Q_o^{a_k}(T + d) := Q^{a_k}(T) - Q^{a_k}(T) \cap Q,\ a_k \in A_6$$

$$d'^{a_k}_{ij}(T + d) := d'^{a_k}_{ij}(T) - d,\ a_k \in (A - A_4) \cup A_5 - A_6$$

$$A_5 := [(A - A_4) \cup A_5 - A_6$$

Ha $Q = \emptyset$, akkor menjünk 8-hoz, egyébként 9-hez.

8. Legyen $Q^{a_k}(T + d) = Q_o^{a_k}(T + d)$, $a_k \in A_4 \cup A_5$, és $A = A_4 \cup A_6$. Menjünk 13-hoz.
9. Legyen Q_l azon tevékenységek halmaza, amelyeken a $(T, T + d)$ intervallumban minden munka befejeződött, azaz a tevékenységet elvégeztük.

$$Q_l \overset{\text{def}}{=} \{ (i, j) \mid M_{ij}^{a_k}(T + d) = 0,\ (i, j) \in Q,\ a_k \in A \cup A_5 \}$$

Ha $Q_l \equiv \emptyset$, akkor menjünk 8-hoz, egyébként 10-hez.

10. Legyen E azon j eseményeknek a halmaza, amelyek a $T + d$ időpontban ténylegesen bekövetkezenek, azaz a $T + d$ időpontban minden j-ben végződő tevékenység befejeződik.

$$E \overset{\text{def}}{=} \{ j \mid T_j = T + d \}$$

Ha $E = \emptyset$, akkor menjünk 8-hoz, egyébként 11-hez.

11. Ha $n \in E$, akkor menjünk 14-hez, egyébként 12-höz.
12. Vegyük a (j, h) alakú tevékenységet $(j \in E)$
Legyen $(j, h) \in Q_8^{a_k}(T + d)$, ha $M_{jh}^{a_k} > 0$, $k = 1, 2, \ldots, K$

$$A_8 \overset{\text{def}}{=} \{ a_k \mid Q_8^{a_k}(T + d) \neq \emptyset \}$$

$$M_{ij}^{a_k}(T + d) := M_{ij}^{a_k},\ (i, j) \in Q_8^{a_k}(T),\ A_k \in A_8$$

$$Q^{a_k}(T + d) := Q_o^{a_k}(T + d),\ a_k \in A_5$$

$$Q^{a_k}(T + d) := Q_o^{a_k}(T + d),\ \bigcup_{a_k \in A_6 - A_7} Q_8^{a_k}(T + d)$$

$$Q^{a_k}(T + d) := Q_8^{a_k}(T + d),\ a_k \in A_8 - (A_6 - A_7) \cap A_8$$

$$A := (A_8 - (A_6 - A_7) \cap A_8)$$

13. Jelentse most a T a T + d értéket és menjünk vissza 2-höz.
14. Vége az eljárásnak.

Ez az algoritmus csupán alapforma. A tevékenységek és a tevékenységeken ütemezhető erőforrások között nem tesz különbséget. Azonban már így is sok különböző gyakorlati feladat modellje lehet.

IRODALOM

[1] James E. Kelley: Critical-path planning and scheduling: matematical basis, Operations Research, 9 (1961): 296—320.
[2] Claude Berge: The theory of grapha and its applications, London, 1962.
[3] A. Charnes and W. M. Raike: One-pass algorithmus for some generalised problema. Operations Research, 1966, No. 5. pp. 914—924.
[4] Dr. Fügedi Tamás: Bevezetés a hálótervezési ismeretekbe. Felsőoktatási jegyzetellátó, Budapest, 1969.
[5] Dr. Danyi Dezső: Hálótervezési módszerek. Országos Ügyvitelgépesítési Felügyelet, Budapest, 1968.
[6] Kádár Iván—Német Lóránt—Szabó István: Kritikus út módszerek és algoritmusaik.
[7] Abramov, Sz. A.—Marincsev, M. I.—Poljakov, P. D.: Hálódiagramos tervezési módszerek. Elektronikus számítógépek alkalmazása a fejlesztési feladatok megoldásában. Budapest, 1966.
[8] Dr. Németh Lóránt: Hálótechnikai termelésprogramozó és erőforrás-allokáló eljárás (ERALL—1) ÉVM. Sz. Ü. O. V., 1965.
[9] Dr. Németh Lóránt: Az ERALL—2 eljárás. (Magyar Építőipar, 1966. okt.)
[10] Útmutató (I—II.) a hálótechnikai módszerek alkalmazásához. Sz. Ü. G. V. kiadványa.
[11] Szabó Zoltán: CPM/TIME-algoritmusok korlátozott kapacitások esetén. Szigma, 1969. II. évf. 3. sz. 182—188.
[12] Dr. Cotel Kornél: A hálós tervezési módszerek és az átbocsátóképesség. Számvitel- és Ügyviteltechnika, 1966. 6. sz. 255—261.
[13] Papp Ottó: Hálótervezési módszerek alkalmazása a műszaki fejlesztésben. Mérnöki Továbbképző Intézet, Budapest, 1966.
[14] Dr. Kocsis Ferenc: Nagyipari építkezések szervezése hálódiagramos módszerekkel. Mérnöki Továbbképző Intézet, Budapest, 1968.
[15] R. G. Busacker, T. L. Saaty: Véges gráfok és hálózatok. Budapest, 1969.
[16] Szabó István: Az ERALL—2 matematikai leírása. Információ és Elektronika, 1967. 2. sz.

EINE PROCEDURE VON DEM KRAFTQUELLEVERTEILUNG

Z. Szőke

Es sei P ein lenkene, rundweglose Netzplan einer Aufgabe.
Die Tätigkeiten bedeuten Arbeitsmengen, die man mit den Kraftquellen (mit beschränkenen Kapazität) erledigen kann.
Die Arbeit gibt eine heuristische Methore dem Terminplan dieser Kraftquelle.
Die Methode verwaltet in gleicher Zeit die Tätigkeiten und die Kraftquellen.
Das Hauptziel der Methode, um die Beendigung der Netzplan mit der Beachtung die Schranke der Kraftquellen zu minimalisieren.