

## Egy többcélú ütemezési probléma megoldása (esettanulmány)

### Bevezetés

A cikkben egy speciális többcélú gyártásütemezési feladattal foglalkozunk. Olyan vállalatnál merült fel, ahol a termékeket kis szériában, a vásárlók igénye szerinti összetételben gyártják. A probléma az volt, hogy a változó igényeknek megfelelő gyártás egyes gyártóhelyeken a szűk kapacitás miatt akadozott.

Az ütemezendő műveletek az alkatrészgyártás műveletei; az esetenként szűk kapacitásúnak bizonyuló gyártóhely az alkatrészgyártó gyáregység egyik üzeme. Itt nagyteljesítményű numerikus vezérlésű forgácsoló gépek vannak. Az alkatrészek igény szerinti gyártását úgy kell megoldani, hogy emellett az NC gépeket — amelyeken a gyártási folyamat fontos műveleteit végzik — a lehető legjobban használják ki. Az NC gépekkel történő megmunkálás előtt jelentős előgyártás van, méghozzá más üzemekben, ezért az NC gépek megfelelő kihasználásához valójában több üzem összehangolt működésére lenne szükség. Az NC gépek folyamatos üzemeltetésének ára azonban semmiképpen sem lehet a műveletközi készletek növelésre. Így merült fel olyan számítógépes program elkészítésének az igénye, amely az NC gépek minél jobb kihasználásának előtérbe helyezésével, egyéb szempontokat is figyelembe véve, műszakokra ütemezi az alkatrészgyártás műveleteit.

Minthogy az adott időszak alkatrészeinek gyártási átfutási idejét az állásidők növelik, az elhúzódo gyártás pedig növeli a pufferkészleteket és a befejezetlen állományt, kapacitást köt le: cél a teljes gyártási átfutási idő minimalizálása is.

Az ütemezett alkatrészgyártás fő feladata a szereldék ellátása. A különböző késztermékeket más-más műhelyben szerelik. Fontos cél a szereldék folyamatos működése is. Ezt a célt úgy is meg lehet fogalmazni, hogy az azonos késztermékhez tartozó alkatrészek befejezési időpontjai között a lehető legkisebb legyen az eltérés.

Az NC gépek jó kihasználását segíti az átállítási idők szempontjából jó terméksorrend meghatározása is.

Az NC gépeken elég sokfajta alkatrész megmunkálására van lehetőség. Beállításuk előtt többféle forgácsoló gépen, természetesen jóval alacsonyabb hatékonysággal végezték azt a munkát, amit ma az NC gépeken végeznek. Ez a hagyományos technológia még ma is él, a régebbi típusú gépek működnek. Ezért mód van arra, hogy ha egy-egy tervidőszakban nem lehet a tervben előírt összes alkatrészt NC gépen megmunkálni, bizonyos termékekre előírhatjuk a hagyományos technológiát. Ilyen esetekben megtakarítás érhető el azáltal, hogy a hagyományos technológiát azokra a termékekre írjuk elő, amelyeknél az NC-technológia kevésbé hatékony a hagyományos technológiához képest.

Az NC gépek ütemezésénél figyelembe kell venni bizonyos műszaki kényszerfeltételeket is. Az alkatrészeket meghatározott felfogókészülékkel kell a gépekre rögzíteni. Mindegyik típusú felfogókészülékből csak egy van, ezért az azonos felfogókészüléket igénylő alkatrészek nem gyárthatók párhuzamosan.

Az ütemezésnél időnként figyelembe kell venni egyes alkatrészek sürgősségét és egyes gépek kiemelt szerepét is.

### A feladat megfogalmazása

A probléma matematikai leírásához vezessük be a következő jelöléseket; Legyen  $I_N = 1, 2, \dots, N$  a gyártandó alkatrészek halmaza,  $p_i$  az  $i \in I_N$  alkatrészből gyártandó sorozat gyártási ideje az NC-n; az  $i$  és  $j$  közötti átállítási időt jelölje  $a_{ij}$  ( $i, j \in I_N$ );  $a_{0j}$  legyen a  $j$  sorozathoz tartozó beállítási idő.

Legyen továbbá  $e_i$  az  $i$  alkatrész sorozatának NC előtti átfutási ideje,  $u_i$  pedig az NC utáni átfutási ideje. A  $k$ -adik gépen felhasználható gépórák számát jelöljük  $T_k$ -val. ( $k = 1, 2, \dots, K$ ;  $K$  az NC gépek száma.) Legyen végül  $J$  a késztermékek száma, és a  $j$  késztermékhez tartozó alkatrészek

halmaza  $P_j$ , azaz  $I_N = \bigcup_{j=1}^J P_j$ .

Feladatunk: határozzuk meg

- az  $i$  alkatrész gyártásának kezdési időpontját  $E_i$ -t,
- az  $i$  alkatrész gyártásának befejezési időpontját  $U_i$ -t,
- az  $i$  alkatrész NC megmunkálásának kezdési időpontját  $t_{ie}$ -t,
- az  $i$  alkatrész NC megmunkálásának befejezési időpontját  $t_{ib}$ -t.

Az ütemezési időszak hossza (napokban) az összes NC gépre azonos, de a meghibásodások, új gyártási programok kipróbálására fordított idők miatt a statisztikák alapján az egyes gépeknél általában más-más hasznos időalappal számolunk. A meghatározandó időadatokat órákban számoljuk, de a gyakorlati megvalósítás lehetőségeit figyelembe véve, a kezdési és befejezési időpontokat végül nem órákban, hanem munkanapokban adjuk meg, megjelölve a műszakot is.

A bevezetett jelölések segítségével a célokat a következőképpen írhatjuk le:

1. A teljes gyártási átfutási idő minimalizálása;

$$\min_{i, j \in I_N} | \max_{i \in I_N} U_i - \min_{j \in I_N} E_j |$$

2. Az azonos késztermékhez tartozó alkatrészek együtt-tartása:

$$\min_{j=1, 2, \dots, J} | \max_{i_1, i_2 \in P_j} (U_{i_1} - U_{i_2}) |$$

3. Az átállítási idők minimalizálása:

$$\min_{L(k)} \sum_{k=1}^K \sum_{k[i], k[i+1] \in L(k)} a_{k[i], k[i+1]}$$

ahol

$$L(k) = \{k[1], k[2], \dots, k[n]\}$$

a  $k$  géphez rendelt alkatrészek megmunkálási sorrendje. Az összes lehetséges permutáció közül csak azok a  $L(k)$  sorrendek fogadhatók el, amelyek a gépek terhelését a megadott időkorlátokon belül hagyják, azaz teljesül, hogy:

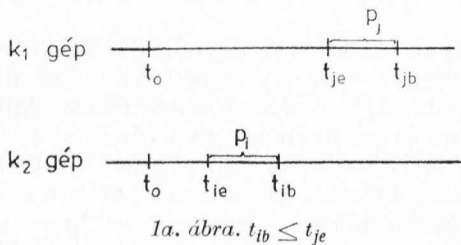
$$F_k = \sum_{k[i] \in L(k)} p_{t[i]} + \sum_{k[i], k[i+1] \in L(k)} a_{k[i], k[i+1]} \leq T_k.$$

A bevezetésben említett további célok is megfogalmazhatók matematikailag, azonban lényegesen egyszerűbbek az eddigieknél. Például az, hogy műszaki kényszerfeltételek miatt bizonyos alkatrészek nem gyárthatók párhuzamosan, az  $I_N$  halmaz egy felosztását jelenti:

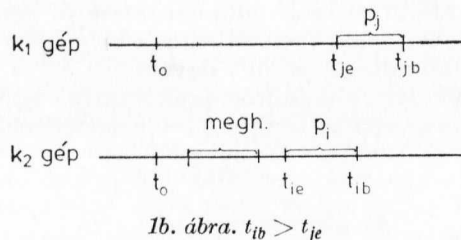
$$I_N = \bigcup_{m=1}^M S_m.$$

Ez a szempont a 3. cél elérését segíti. Azok között a munkadarabok között, amelyekhez azonos felfogókészülékkel használnak, az átállítási idő igen kicsi, tehát, ha ezeket az alkatrészeket az NC gépeken egymás után gyártják, az átállítási idők csökkentését segítik. A párhuzamosságot kizáró feltételt nem is érdemes másképpen megfogni, mint úgy, hogy az ilyen alkatrészeket egymás után soroljuk. Ennek az az oka, hogy a gépek párhuzamosan működnek, és ezért valamelyik meghibásodása előre nem tervezett párhuzamos gyártást okozhat.

előírt gyártási terv:



megvalósulás:



Ha a problémát meg lehetne oldani többcélú matematikai programozási feladatként exakt módszerekkel, akkor részletes vizsgálati eredményeket közölhetnénk a célfüggvények közti kapcsolatokról. Annak, hogy a heurisztikus megoldás mellett döntöttünk, egyik oka éppen az, hogy a különböző célok közt az összefüggések mérése bonyolult és ugyanakkor megbízhatatlan. Megállapítható például, hogy az átállási idők minimalizálását általában elősegíti

az, hogy bizonyos alkatrészeket egymás után kell gyártani, de az is tény, hogy ez a követelmény nem engedi meg az átállási idők szempontjából legjobb sorrend kiválasztását. A teljes gyártási átfutási idő és az átállítási idők minimalizálása között még nehezebben mérhető az összefüggés.

Igen lényegesen befolyásolta az algoritmust az a tényező, hogy különböző gyártási időszakokban más-más súllyal szerepelnek a különböző célok. Azt állapítottuk meg, hogy a probléma megoldásához a többcélú megközelítés helyett jobb, ha a célokat rangsoroljuk és időszakonként a megrendelő igényeinek figyelembevételével határozzuk meg (paraméterek segítségével) a célok prioritását.

### Megjegyzések

1. Ismert ütemezési problémának feleltethető meg az első cél, ha az elő és utógyártást úgy képzeljük el, mintha egy-egy ideális gépen folyna a gyártás és az NC gépeket is egyetlen gépnek képzeljük. Ilyen feltételezések mellett azzal a három gépes flow-shop problémával állunk szemben, amelynek célja a teljes átfutási idő minimalizálása. Minthogy teljesül, hogy  $\min e_i > \max u_i$  (sőt  $\min u_i > \max p_i$  is), ezért a problémát a két gépes Johnson-elv általánosítása [1] alapján lehetne megoldani, ha ez lenne az ütemezés egyetlen célja.

2. Az ütemezés harmadik célfüggvényével kifejezett feladat az irodalomban az „utazó ügynök probléma” megfogalmazásban szerepel általában. Az ütemezési feladatot a következőképp fogalmazhatjuk át utazó ügynök problémává:

Van  $N$  város (az alkatrészek) és  $K$  ügynök (az NC gépek). Minden városba el kell, hogy jusson egy és csak egy ügynök (minden alkatrészt valamelyik gépen gyártani kell), aki a  $j$  városban  $p_j$  ideig tartózkodik. Az  $i$  és  $j$  városok között az utazási idő  $a_{ij}$  és a  $k$  ügynök legfeljebb  $T_k$  ideig lehet úton. Az utazó ügynök problémának nagy irodalma van [1], [2], többféle közelítő eljárás is született a probléma megoldására. Kiemelten tehát erre a problémára is lehetne már kidolgozott megoldási módszert találni, igaz, hogy az exakt megoldások igen lassúak lennének.

A mi esetünkben ennek a részfeladatnak az exakt megoldásával azért sem érdemes kiemelten foglalkozni, mert a  $p_i \gg a_{ij}$  (minden  $i, j$ -re) jellemzi az adatainkat, azaz az átállítási idők minimalizálásától nem várható túl nagy nyereség. E cél érdekében tehát nem érdemes a megoldási algoritmust nagyon bonyolulttá tenni, a számítógépes időt növelni.

A több célfüggvény miatt, a számítógépes tervezés rugalmassága érdekében, heurisztikus algoritmust készítettünk a probléma megoldására.

### Az algoritmus ismertetése

Az algoritmus a következő fontosabb lépésekből áll:

1. Az  $A = \{a_{ij}\}$  átállítási mátrix minden sorában megkeresi a legkisebb nem nulla és a legnagyobb értéket. Legyenek ezek  $x_i$  illetve  $y_i$  ( $i \in I_N$ ).

2. Ha  $\sum_{i \in I_N} (p_i + y_i) > \sum_{k=1}^K T_k$ , akkor elvégzi az alkatrészek gazdaságosság szerinti csoportosítását. Azok közül a termékek közül irányít az algoritmus a

hagyományos technológiára, amelyeken az NC megmunkálás a legkisebb hasznót eredményezi a hagyományos technológiához képest. (A gazdaságosság szerinti csoportosítás a kétféle technológia időadatai alapján végezhető.)

3. Kijelöli a technológiai előírások szerint egymás után gyártandó alkatrészek  $S_m$  halmazait. Az NC gépekre ütemezés szempontjából az  $S_m$  halmazok elemeit összevonja egy ideális alkatrésszé. Egy ilyen ideális alkatrész gyártási ideje az összegzett gyártási idő, azaz  $q_m = \sum_{j \in S_m} p_j$ , mivel az ilyen alkatrészek közötti átállítási idő 0-nak tekinthető. Az  $S_m$  halmazok különböző elemei és a többi  $k \notin S_m$  alkatrész közti átállítási idők megegyeznek, ezért az  $S_m$  halmazokon belüli sorrend a többi ütemezési szempontnak megfelelően alakítható ki. Az egyéb szempontok prioritása a futtatások során paraméterrel adható meg. Mivel az  $S_m$  halmazok elemszáma kicsi, ( $\leq 6$ ), a sorrend az összes lehetséges változat közül a legjobb lehetőség kiválasztásával határozható meg.

4. Az összevont alkatrészek ( $S_m$  halmazok) ütemezésénél a gépek közel azonos terhelését, valamint a teljes átfutási idő minimalizálását tekinti célnak az algoritmus.

A gépek közel azonos terhelésének problémája a következő feladatként fogalmazható meg:

Adott  $T_k$  ( $y = 1, 2, \dots, K$ ) hosszúságú szakaszokat adott  $q_m$  ( $m = 1, 2, \dots, M$ ) hosszú diszjunkt szakaszokkal kell lefedni úgy, hogy a le nem fedett szakaszok hossza ne különbözzön lényegesen egymástól, azaz  $\max_{h,l} |d_h - d_l|$  minimális legyen, ahol

$$d_k = T_k - \sum_{S_m \subset L'(k)} q_m,$$

és  $L'(k)$  jelöli azt az indexhalmazt, amelybe tartozó  $q_m$  hosszúságú szakaszokkal fedjük le a  $T_k$  szakaszt.

Az  $S_m$  halmazok kijelölése ( $p_j$  hosszú szakaszok helyett  $q_m$  hosszú szakaszokkal történő lefedés) a felhasználandó szakaszok számát legalább a harmadára csökkenti, ezért a lehetséges lefedések száma — az összevonás nélküli lehetőségekhez képest — lényegesen csökken.

Maga a lefedési algoritmus a következőképpen működik: A  $q_m$  értékeket nagyság szerint csökkenő sorrendbe rendezi. Az  $L'(k)$  indexhalmazokat úgy alakítja ki, hogy a leghosszabb  $q_m$  szakaszt hozzárendeli a  $T_1$  szakaszhhoz, az ezután következőkkel mindig azt a legalacsonyabb indexű  $T_k$  szakaszt fedi le, amelyből még csak a legrövidebb rész van lefedve a már kiválasztott  $q_m$  hosszúságú szakaszokkal. Ha az így kapott  $L'(k)$  indexhalmazokkal meghatározott lefedés nem elfogadható, akkor a leghosszabbaktól kezdve párcenként felcseréli a  $q_m$  szakaszokat a rendezett állományban, figyelembe véve az előző lefedés tapasztalatait, addig, ameddig nem kap egy elfogadható lefedést, illetve ameddig az összes lehetőséget végig nem próbálta. A tapasztalat szerint már legkésőbb a harmadik próbálkozásra elfogadható lefedést kapunk.

5. Az  $L'(k)$  halmazok kijelölése még nem jelent sorrendet, csak hozzárendelést. A sorrend az átállítási mátrix alapján határozható meg, ahol ez lényegesen nevezhető időmegtakarítást eredményez. Az átállítási mátrix minden sorára megnézi, hogy az  $y_i > 2x_i$  egyenlőtlenség teljesül-e. Ha egy  $i$ -re teljesül ez az egyenlőtlenség és az  $i$ , valamint  $x_i$ -t realizáló alkatrész ugyanannak az  $L'(k)$  indexhalmaznak eleme, a gyártási idejüknek megfelelő hosszúságú szakaszok egymás után kerülnek, ha még nincs rájuk sorrendi előírás. Ha a már

meghatározott sorrendet kellene megváltoztatni, megnézi, hogy érdemes-e végrehajtani a cserét, azaz a csere hatására jobban csökken-e az átállítási idők összege, mint amennyit ront a csere a  $T_k$  hosszúságú szakaszok lefedésén. (A  $\max_k |d_k - T_k|$  növekszik-e?)

Ha az  $i$  és az  $i$  után a legkisebb átállítási idővel gyártható alkatrész nem tartozik azonos  $L'(k)$  halmazhoz, és egymás után sorolásukkal lényeges idő megtakarítás érhető el, akkor megpróbálja cseréssel azonos  $L'(k)$  halmazhoz rendelni őket.

Azokat az elemeket, amelyekre az átállítási idők alapján még nincs kijelölve sorrend, az egyéb paraméterekkel meghatározott prioritási sorba rendezett ütemezési célok alapján sorolja.

### A számítógépes megvalósítás tapasztalatai

Az ismertetett algoritmus alapján készített programrendszer batch üzemmódban működik, két havonta üzemeltetjük. Egy két hónapos periódus alatt 60–90 féle terméket kell megmunkálni az NC gépeken. A múlt évben egy-egy tervezésnél 3, vagy 4 NC gép folyamatos működésére lehetett számítani. Egy-egy  $S_m$  halmaz átlagos elemszáma 3, maximális elemszáma 6 volt.

A rendszer rugalmasságát a már ismertetett paraméterezési lehetőségen kívül (a célok prioritása változtatható) más paraméterek megadásának lehetősége is segíti. Meg lehet adni különleges prioritásokat is: kiemelt fontosságú alkatrészeket, gépeket minden tervidőszakban meg lehet jelölni. Ugyancsak a rugalmasságot növeli, hogy az ütemezés részletes kiírása előtt a rendszer egy vonalas tervet (Gantt diagramot) rajzol az NC gépek terheléséről valamint az alkatrészek teljes gyártási tervéről. Ezek a tervek könnyen, gyorsan áttekinthetők, az esetleges változtatások szükségessége leolvasható róluk. A programok ICL System-4 típusú gépen futottak, a teljes futási időnek, amely 20–25 perc, az algoritmus működése csak kis része; 5–10 perc. A paraméterek módosítása után csak az algoritmust kell újra futtatni, ez nem igényel sok gép-ídt. Csak a jóváhagyott vonalas terveknek megfelelő részletes ütemezéseket íratjuk ki. Az eddigi futtatások során legfeljebb két változatban kellett futtatni az algoritmust.

A külső és belső körülmények egyaránt nagyfokú gyártási rugalmasságot követelnek az üzemektől. Éppen ezért nagy fontossága van a gyors döntéselőkészítésnek. Elkezdtük az algoritmus kispépes, interaktív változatának kidolgozását.

(Beérkezett: 1982. március 18-án.)

### IRODALOM

1. CONWAY—MAXWELL—MILLER: *Theory of Scheduling* 1967. Addison — Wesley Publishing Company
2. HAMMER—JOHNSON—KORTE: *Discrete Optimization* 1979. North-Holland Publishing Company

## A MULTI-OBJECTIVE SCHEDULING PROBLEM

In this paper a model and a heuristic algorithm are given for a multi-objective dynamic scheduling problem. The constraints found in this problem are:

- there are limits on the capacity of equipments to manufacture certain parts,
- there are technological restrictions on the sequence in which tasks can be performed on these machines.

The scheduling algorithm is expected to minimize the processing time and the sequence-dependent setup times and to enable the smooth running of the assembly shops where the parts go.

## МНОГОЦЕЛОВАЯ ПРОБЛЕМА ПЛАНИРОВАНИЯ ПРОИЗВОДСТВА

В заметке описаны модель и гевристический алгоритм разработанные для решения многоцелевой динамической проблемы планирования производства деталей.

Самыми важными требованиями планирования являются:

- минимизация пролётного времени производства деталей,
- минимизация времени на переналадку,
- способствовать непрерывной действительности цехов монтировании работающих из деталей, производство которых было планировано алгоритмом.