AZ IGÉNY SZERINTI TÖMEGGYÁRTÁS TERMELÉSPROGRAMOZÁSÁNAK HEURISZTIKUS MEGOLDÁSI MÓDSZERE

Kulcsár Gyula
egyetemi tanársegéd
Miskolci Egyetem, Informatikai Intézet, Alkalmazott Informatikai Tanszék

A cikkben az igény szerinti tömeggyártás termelésprogramozási feladatával foglalkozunk. A korszerű gyártó-szerelő rendszerekben jellemzően előforduló alternatív technológiai útvonalak, korlátozottan rendelkezésre álló többfunkciós párhuzamos gépek, sorrendfüggő átállási idők, terméktől függő termelési sebességek valamint határidős munkák együttes figyelembevételével gyakorlati igényekhez alkalmazkodó ütemezési koncepciót mutatunk be. Az erőforrások allokálásának és a munkák ütemezésének modellezésére egy új kiterjesztett rugalmas Flow Shop ütemezési modellt használunk. Ismertetünk egy heurisztikus megoldási módszert, amely lokális keresési technikát valamint indexelt adatmodellen alapuló gyors szimulációs kiértékelést kombinál.

1. BEVEZETÉS

Napjainkban a termelő nagyvállalatok egy része közvetlenül a vevők igényeit próbálja meg kiszolgálni (pl.: háztartási gépek, lámpák gyártása). A versenyképesség növelése érdekében szükséges, hogy minél jobban alkalmazkodjanak a piaci körülmények gyors változásaihoz. Ennek érdekében a gyártási hatékonyságot és a szállítóképességet kell javítani [12], [13]. Az előbbit az erőforrások minél jobb kihasználásával, az utóbbit pedig, előzetes becslésen alapuló raktárra történő gyártással tudják növelni. A vállalatok sikeressége nagymértékben a megrendelők igényeinek magas szintű kielégítésén múlik, ennek egyik fontos feltétele a megrendelésekhez tartozó munkák ütemezésének minél hatékonyabb megoldása [8].

A finomprogramozás a gyártásirányítás (Manufacturing Execution System, MES) szintjén fellépő rövid távú, valósidejű tervezési feladat megoldását követeli meg [2]. Ezen a szinten az ütemezési feladat azt jelenti, hogy ismert és pontosan meghatározott (technológiai, anyag stb.) korlátozások figyelembe vételével, a gyártásra kiadott, ismert belső rendeléseken alapuló munkák elvégzéséhez gyártási erőforrások (gépek, eszközök, stb.) allokálását és munkák indítási időpontjának sorozatát kell megtervezni úgy, hogy a definiált korlátozások teljesüljenek, és a vállalat magasabb szintjén megfogalmazott célok megvalósuljanak.

A diszkrét gyártási folyamatokban a gyártás rendszerint elkülönült gépeken zajlik. Attól függően, hogy az egyes műveletek a gyártó (szerelő) műhely gépein hogyan kerülnek végrehajtásra, megkülönböztethetünk soros illetve nem soros elrendezésű gyártórendszereket [14]. Tömeggyártás és igény szerinti tömeggyártás során a termékek gyártási folyamata az első kategóriába tartozik [7]. A soros elrendezésű gyártási struktúrához kapcsolódó alapmodellben (Flow Shop) adott számú különböző gép a technológiai sorrendnek megfelelően egymás után van elhelyezve. Minden munkadarab ezeken a gépeken halad végig.

Ha megengedett hogy az egyes gépeken a munkák sorrendje eltérő legyen, akkor előzéses, ellenkező esetben előzésmentes modellről van szó. Ezekről a modellekről és megoldási módszereikről számos munka nyújt részletes áttekintést (pl.: [4], [5], [10], [11], [16], [15], [20]).

Az alapmodellnek párhuzamos gépekkel történő kibővítéseként ismert a rugalmas egyutas (Flexible Flow Shop) modell. Ebben a modellben összetett munkahelyek vannak definiálva.

Minden egyes munkahelyen adott számú egymással teljesen egyenértékű gép található. A gépek száma munkahelyenként eltérő lehet. A munkadarabokat minden munkahelyen, annak egy kiválasztott gépén kell megmunkálni. Ebben a modellben tehát megjelenik a gépválasztás feladata is, ugyanakkor továbbra is fontos szerepet játszik a munkák gépenkénti sorrendjének meghatározása.

A szakirodalomban számos cikk különböző mélységekben foglalkozik a rugalmas Flow Shop ütemezési modellel (pl.: [1], [3], [5], [9], [17], [18], [19]). A témakörhöz tartozó cikkek többsége célfüggvényként a rendeléscsoport legkésőbbi befejezési idejét (makespan) használja. Sokkal kevesebb eredmény ismert határidős gyártás (make to order) esetén a késésekkel kapcsolatos célfüggvények minimalizálására.

A jelenleg ismert többgépes modellek nem tudják figyelembe venni egyszerre az igény szerinti tömeggyártás jellegzetességeit: több művelet együttes végrehajtására képes gépeket, technológiai útvonal alternatívákat, gépek változó rendelkezésre állási időintervallumait, eltérő termelési sebességeket és sorrendfüggő átállási időket, így szükség van ezeknek a modelleknek a kiterjesztésére, továbbfejlesztésére.

2. A VIZSGÁLT GYÁRTÁSI FOLYAMAT

Az igény szerinti tömeggyártást folytató vállalatok különböző típusú termékeket állítanak elő. Jellemzően, adott egy rendelés állomány, amelyet a külső rendelések és az előrejelzések figyelembe vételével a vállalat termeléstervezés szintjén definiálnak. Minden egyes rendelés meghatározott típusú, adott darabszámú egyforma termék, adott határidőre történő legyártását igényli. A műhelyszintű irányításban fontos szerepet játszik a logisztikai egység, amely előre megadott darabszámú, azonos termék együttesét jelenti.

Az általunk vizsgált gyártási struktúra gyártó-szerelő jellegű, automatizált gépcsoportokból (gépekből és/vagy gépsorokból) áll. A gépek közötti anyagmozgatás eszköze a paletta. Egy rendelés paletták halmazának is tekinthető, ahol a paletták számát a rendelt mennyiség és a paletták termékfüggő mérete együttesen határozza meg.

A termékek előállításához legfeljebb négy, vagy annál kevesebb műveletet kell kötött sorrendben végrehajtani. A műveletek operációk sorozatából tevődhetnek össze. Mivel a műveletek nem megszakíthatók, ezért ezek tekinthetők az ütemezés során a legkisebb allokációs egységeknek. Az egyes műveletek elvégzéséhez – a megfelelő gépeken túlmenően – meghatározott anyagoknak, komponenseknek is rendelkezésre kell állni adott mennyiségben. A gyártás rugalmas jellegéből következik, hogy egy adott típusú termék különféle anyagok, komponensek, gépek és útvonalak használatával is előállítható.

Az rendszerben szereplő gépek – amelyek valójában különböző gyártási képességekkel rendelkező gépsorok – a terméktől függő gyártási sebességekkel, a munkák sorrendjétől függő átállási időkkel, rendelkezésre állási időintervallumokkal és egy adott termékcsoportra érvényes műveletsorozattal jellemezhetők.

A gyártás során egy adott gép egyszerre csak egy munkán dolgozhat, és egy adott munka egyszerre csak egy géphez lehet hozzárendelve. A gépek közötti tároló helyek mérete nem korlátozott.

Az ütemterv elkészítésekor azt is figyelembe kell venni, hogy az ütemezési időhorizonton a műhely bizonyos gépei már korábbi, még be nem fejezett, nem módosítható feladatokkal terheltek, így az utolsó megerősített és érvényben lévő ütemterv következményei hatással vannak az új ütemtervre.

3. KITERJESZTETT ÜTEMEZÉSI MODELL

Az ütemezési feladatok formális leírásának eszközeként a szakirodalomban az α|β|γ formalizmus használata a legismertebb [1]. A szimbólumok jelentése a következő:

- α: erőforrás környezetet leíró paraméter lista,
- β: korlátozásokat, végrehajtási jellemzőket leíró paraméter lista,
- γ: célfüggvényeket kitűző paraméter lista.

A listákban szereplő paraméterekre számos javaslat van az irodalomban. Mivel a vázolt ütemezési feladat nem sorolható be egyik ismert ütemezési feladatosztályba sem, így egy új kiterjesztett (Extended Flexible Flow Shop, EFFS) modellt vezettünk be a feladat formális leírására [6]:

$$M, Q, F, Set_{i,j,m}, Cal, 4 \left| R_i, D_i \right| f \qquad (1)$$

Az egyes szimbólumok jelentése a következő:

$\alpha = [\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6]$

α_1: Az erőforrás jellege. *M*: többcélú párhuzamos gépek megengedettek.

α_2: Az alternatív gépek jellege. *Q*: eltérő sebességű (intenzitású) gépek.

α_3: A műveleti sorrend előírása. *F*: a műveletek sorrendje kötött.

α_4: A gépek beállításra vonatkozó előírás. $Set_{i,j,m}$: terméksorrendtől és géptől függő időadatok.

α_5: Erőforrásokra vonatkozó speciális előírások. *Cal:* Gépekre előírt rendelkezésre állási időintervallumok. A gépek csak ezeken belül működhetnek.

α_6: A munkahelyek maximális száma. *4*: a műveletek száma négy.

$\beta = [\beta_1, \beta_2]$.

β_1: A munkák indíthatóságára vonatkozó előírás. R_i: munkánként előírt legkorábbi indítási időpont.

β_2: A munkák befejezésére vonatkozó előírás. D_i: munkánként előírt legkésőbbi befejezési időpont.

γ: A kijelölt célfüggvény, amelynek a minimumát keressük.

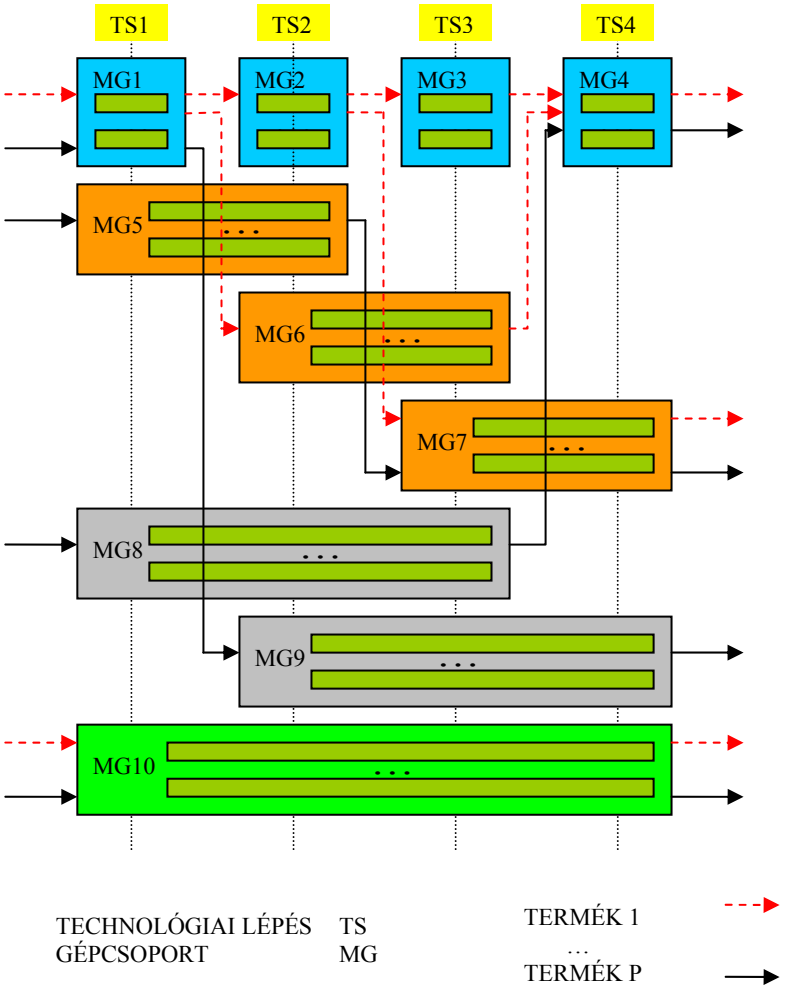
4. AZ ÜTEMEZÉSI FELADAT HEURISZTIKUS MEGOLDÁSA

Az ismertetett ütemezési feladat kombinatorikus tulajdonságai miatt az NP teljes feladatosztályba tartozik. Ezért az elméleti globális optimum keresése helyett, olyan megoldási módszerekkel foglalkozunk, amelyek nagy méretű feladatok (pl. 600 megrendelés, 3000 munka, 160 gép) esetén is elfogadható időn belül megvalósítható ütemtervet állítanak elő, úgy hogy a választott célfüggvény értéke minél közelebb legyen az optimumhoz.

4.1 Megoldási koncepció

A lehető legtágabb értelmezés szerint minden fizikai palettát önálló absztrakt munkának tekintünk. A feladat megoldása során ilyen munkák kerülnek ütemezésre. Nem használunk előre rögzített gyártási sorozatnagyságot. A gyártási sorozatnagyságot az azonos beállítással készülő munkák száma adja meg. A rendelési sorozatnagyság ennél nagyobb vagy kisebb is lehet. Az ütemezés során az egyes gépeken kialakuló gyártási sorozatnagyság az egy munkaköteg, amely két egymást követő beállítási időintervallum közötti azonos termékre vonatkozó rendelt munkák sorozata. A köteg nagysága tehát dinamikusan változhat, megengedve a rendelési sorozat bontását és egyesítését.

A modellben feltételezzük, hogy az ütemezés előtt minden munkához hozzárendelhető egy korlátozás, legkorábbi kezdési időpont, amely a kezdő technológiai lépés indíthatóságának időbeli korlátozására vezeti vissza a gyártási-szerelési komponensek rendelkezésre állását.



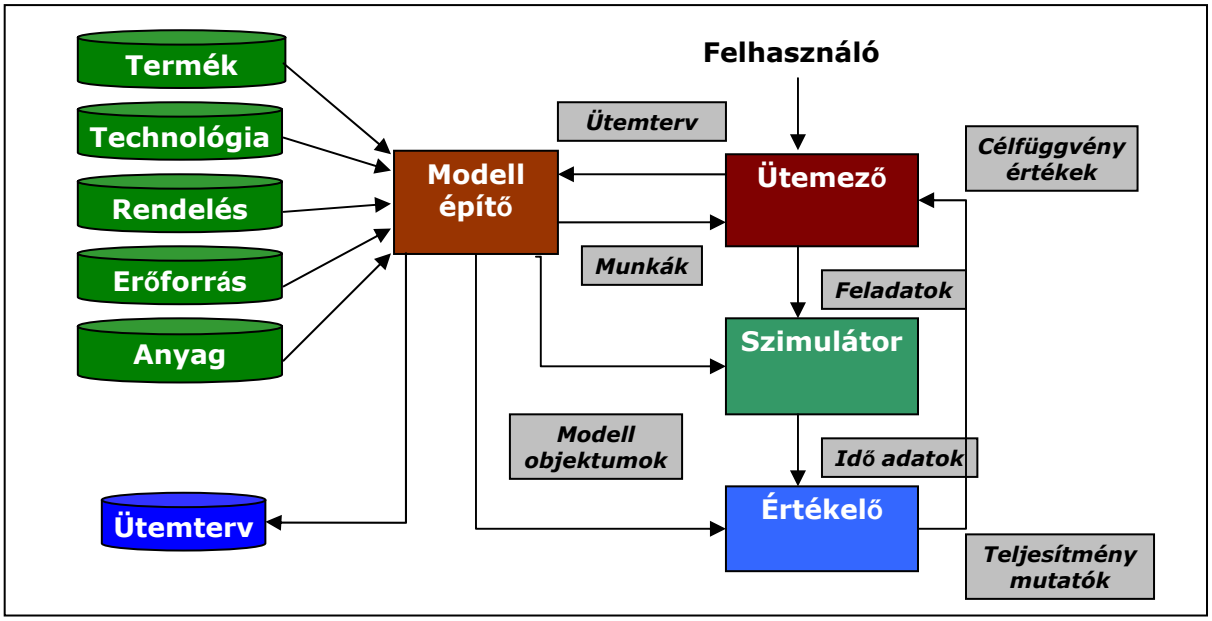
1. ábra: Kiterjesztett rugalmas Flow Shop modell

A gépek által végrehajtható műveleteket technológiai lépésként értelmezzük. Bevezetjük a végrehajtási lépés fogalmát, amely alatt a technológiai lépések olyan sorozatát kell érteni, amely gépváltás nélkül elvégezhető. Egy multifunkcionális gépsorhoz tartozó végrehajtási lépés több technológiai lépést tartalmaz, míg egy hagyományos géphez tartozó csak egyet. Az ütemezés alapegységének nem a technológiai lépést, hanem a végrehajtási lépést tekintjük.

A végrehajtási útvonal fogalmát a végrehajtási lépések olyan sorozataként definiáljuk, amelyben az előforduló technológiai lépések csak egyszeresen vannak lefedve, azaz a végrehajtási lépések közös része üres halmaz.

A végrehajtási lépések típusai alapján gépcsoportokat (géptípus osztályokat) definiálunk (1. ábra). A gépcsoportokon belül különböző darabszámú és gyártási képességű gépek lehetnek. Mivel a gépek nem minden terméktípust képesek kezelni, így a gépcsoportok terméktípusonként (vagy terméktípus csoportonként) eltérőek.

Ezeket a modell tulajdonságokat figyelembe véve a termelési finomprogramot a 2. ábrán vázolt módon oldjuk meg.



2. ábra: A kifejlesztett ütemező rendszer működése

A termelésinformatikai rendszerek adatbázisában tárolt adatok felhasználásával a modellépítő definiálja a rendszerben lévő objektumokat. Kiemelt fontosságú feladata, hogy a rendeléseket (*order*) felbontja munkákra (*job*) és definiálja a korlátozásokat.

Az ütemező az ütemterv elkészítésekor minden egyes munkához hozzárendel egy megfelelő végrehajtási útvonalat, továbbá hozzárendel egy megfelelő gépet a kiválasztott útvonal minden egyes végrehajtási lépésének megfelelő gépcsoportból, valamint meghatározza minden hozzárendelt gépen a végrehajtási sorban elfoglalt pozícióját. Ezáltal a munkákat feladatokra (*task*) botja fel. Egy feladat az egy adott munka adott végrehajtási lépésének adott gépen történő megvalósítását jelenti.

A feladatok és munkák időadatainak számítására gyors szimulációs eljárást használunk. A szimuláció figyelembe veszi az egyes gépek rendelkezésre állási időintervallumait, az egyes gépeken az adott munkák sorrendje által meghatározott átállítási időket, a munka-gép összerendelések alapján számítható megmunkálási időket. A munka aktuális besorolása (korlátozások: időablak, sorrend, erőforrás, útvonal) alapján ismertté válik az egyes munkák gépenkénti indítási időpontja és a befejezési időpontja.

Az időadatok felhasználásával előre definiált célfüggvények kiértékelésére és teljesítmény mutatók számítására kerül sor. Ez alapján történik meg az ütemterv minősítése.

A változó célokhoz való rugalmas alkalmazkodás úgy valósul meg, hogy előre definiálunk célfüggvényeket és ezek közül a felhasználó az ütemezési folyamat során kijelölheti az aktuális célfüggvényt vagy célfüggvényeket. Az ütemező az ütemterv minősítése alapján módosítja az ütemtervet. A folyamat ismétlésével javítható az ütemterv.

4.2 Adatmodell

A modellben indexszelt tömböket használunk az adatelérés gyorsítása érdekében. Ezekben a tömbökben a gyártási környezetre jellemző entitások (pl.: megrendelések, munkák, gépek stb.) indexekkel vannak helyettesítve. Ezek nem negatív egészszámok. Egy ilyen index megadja egy adott objektum adott tömbben elfoglalt pozícióját. Ezáltal lehetővé válik az, hogy egy adott tömb adott elemére történő hivatkozásban a tömb indexe ugyanannak vagy egy másik tömbnek az egyik értéke legyen. Az összetartozó értékeket a bázisvektorokban tárolt alapadatokra történő hivatkozások rendszere tartja össze. Ennek előnye főként az, hogy

egy tetszőleges elemből kiindulva – az indexelési szabályok betartásával – minden egyes vele kapcsolatban álló elem közvetlenül, keresés nélkül elérhető.

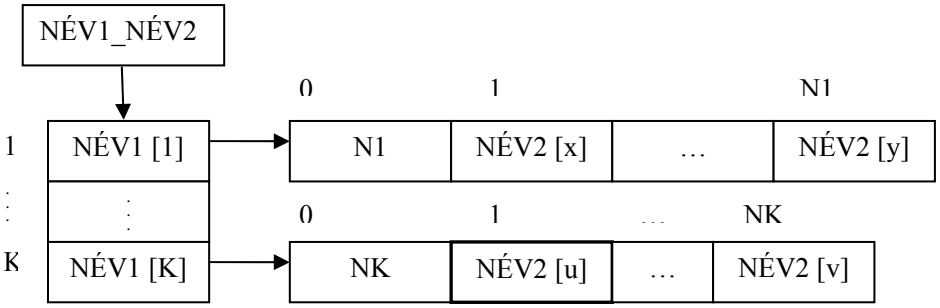
Az adatmodell legfontosabb entitásai a következők:

- A gyártható terméktípusok *p* $(p = 1,\ldots, N_P)$.
- A rendelés állomány belső rendelései *o* $(o = 1,\ldots, N_O)$.
- A rendelések felbontásából származó *i* $(i = 1,\ldots, N_J)$ munkák.
- A műhelyben lévő gépek (gépsorok) *m* $(m=1,\ldots, N_M)$.
- Gépcsoportok *mg* $(mg = 1,\ldots, 10)$.
- Technológiai lépések $(ts = 1,\ldots, 4)$.
- Végrehajtási lépések $(es = 1,\ldots, 10)$.
- Végrehajtási útvonalak $(r = 1,\ldots,26)$.

Egy soronként eltérő elemszámú kétdimenziós tömb általános szerkezetét a 2. ábra szemlélteti. Ilyen szerkezeteket használunk két különböző (*NÉV1 és NÉV2*) tömb elemeinek összerendelésére, ahol a *NÉV1* az alap tömböt, *NÉV2* pedig a hozzákapcsolt tömböt jelenti.

Az adatmodell entitásainak jellemző relációi a következők:

- Adott gépcsoportba tartozó gépek a műveletvégző képességek alapján *MG_M*.
- Adott termék szempontjából adott gépcsoportba tartozó gépek *P_MG_M*.
- Adott végrehajtási útvonalhoz tartozó gépcsoportok *R_MG*.
- Adott termék szempontjából használható útvonalak *P_R*.



Hivatkozás egy adott elem értékére:
NÉV1_NÉV2[SOR_INDEX][OSZLOP_INDEX]
SOR_INDEX = (1,…, K)
OSZLOP_INDEX = (0,…, NÉV1_NÉV2[SOR_INDEX][0])

2. ábra: Kapcsolótömb általános szerkezete

Egy ütemterve – annak érdekében, hogy minden részlete indexelhető legyen – három részből épül fel. Ezek a következők:

1. Egy *J_A* kapcsolótömb mutatja a munkákhoz hozzárendelt útvonalakat az érintett konkrét gépekkel együtt a következő módon: *J_A[i][am]*. Ahol:

   i jelenti a munkát $(i = 1,\ldots, N_J)$,

   J_A[i][0] jelenti az *i* munkához hozzárendelt útvonalat,

   R_MG[J_A[i][0]][0] jelenti az útvonal bejárása során érintett gépek számát,

   J_A[i][am] $(am=1,\ldots, R_MG[J_A[i][0]][0])$ jelenti az *i* munka *am*-edik végrehajtási lépéséhez rendelt gépet.

2. Egy *M_WLOAD* kapcsolótömb mutatja a munkák sorrendjét az egyes gépeken a következő módon: *M_WLOAD[m][ai]*. Ahol:

   m $(m= 1,\ldots,N_M)$ jelenti a gépet,

   M_WLOAD[m][0] jelenti a munkák számát az *m* gépen,

   M_WLOAD[m][ai] $(ai = 1,\ldots, M_WLOAD[m][0])$ jelenti az *m* gép munkasorozatának *ai*-edik munkáját (2. ábra: *NÉV1 = M, NÉV2 = J*).

3. Egy *M_STET* tömb tárolja az *M_WLOAD* tömb munkáinak időadatait a következő módon:
M_STET[m][ai].ST jelenti a kezdési időpontot,
M_STET[m][ai].SetT jelenti az átállás időtartamát,
M_STET[m][ai].ProcT jelenti a műveletvégzés időtartamát,
M_STET[m][ai].ET jelenti a befejezési időpontot.

A vázolt adatmodell fontos tulajdonsága, hogy input adatai feltölthetők a termelésinformatikai rendszerekben jellemzően tárolt alapadatokból.

4.3 Szimulációs modell

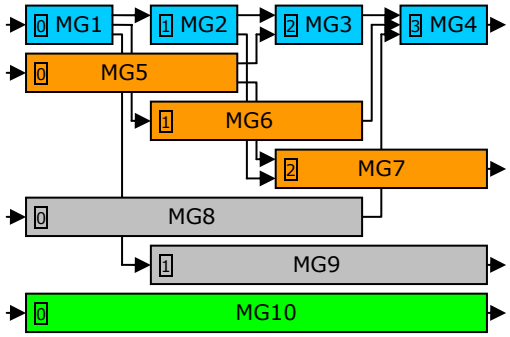
A szimuláció egy adott ütemterv munka-gép összerendeléseiből (*J_A)* és a munkák gépenkénti sorrendjéből (*M_WLOAD)* indul ki.

A gyártási folyamatot egy szabályalapú számítási eljárással szimuláljuk, amelynek során az ütemterv időadatait értékeljük ki (*M_STET*). A szimuláció során feltételezzük, hogy a gépek között korlátlan méretű átmeneti tárolók állnak rendelkezésre.

A számítási folyamat végrehajtásához definiáljuk a számítási lépések olyan sorrendjét, amely figyelembe veszi a következőket:

- Egy adott *i* munkának egy adott *m* közbenső géphez tartozó időadatainak számításához szükséges – többek között – az *i* munka előző géphez tartozó befejezési időpontjának ismerete és az *m* gép előző munkájának befejezési időpontja is.
- A gyártási környezet tartalmaz útvonal csatlakozásokat és szétválásokat (csomópontok), ezért egy adott *m* gépre több azonos típusú és különböző típusú gépről is érkezhetnek munkák.

Ezeknek megfelelően a gépcsoportok között sorrendi korlátozásokat fogalmazunk meg. A korlátozások leírhatók egy egyszerű (körútmentes) irányított gráffal, amelyben a csúcspontok jelentik a gépcsoportokat és az irányított élek mutatják a kötelező sorrendet.



3. ábra: Gépcsoportok kapcsolódásai

Minden egyes gépcsoporthoz hozzárendeljük a saját „befokát” (3. ábra). A befok az adott gépcsoportba érkező élek megengedett legnagyobb számát jelenti. A gépcsoportok keresett sorrendjét úgy kapjuk meg, hogy a befokok növekvő sorrendjének megfelelően rendezzük a gépcsoportokat. Az azonos befokú gépcsoportok között a további sorrendet a gépcsoportokra jellemző végrehajtási lépések határozzák meg úgy, hogy a több technológiai lépés megvalósítására képes gépcsoport lesz a magasabb prioritású:

Prioritás: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
Gépcsoport: {4, 3, 7, 2, 6, 9, 1, 5, 8, 10}

A számítási folyamatban a prioritások nem növekvő sorrendjében vesszük sorra a gépcsoportokat. Az adott gépcsoporthoz tartozó összes gépen (párhuzamos gépek) az *M_WLOAD* tömb által meghatározott sorrendben kiszámításra kerülnek a munkákhoz tartozó időadatok (műveletkezdési időpont, setup idő, műveleti idő, bejezési időpont). Az eredmények bekerülnek az *M_STET* tömbbe.

A számításban meghatározó szerepet kapnak a műveletkezdési időpontok. Az *i* munka kezdési időpontját a hozzárendelt *m* gépen a következő értékek együttesen határozzák meg:

- a legkorábbi kezdési időpontja az *i* munkának,
- a befejezési időpontja az *i* munkának az előző gépen,
- a befejezési időpontja az előző munkának az *m* gépen,
- az *i* munkához tartozó átállási idő az *m* gépen,
- az *m* gép felszabadulásának időpontja az előző ütemterv terhelése alól, és
- az *m* gépnek a rendelkezésre állási időintervallumai.

Definiáltunk egy olyan eljárást, amely elvégzi az *i* munka kezdési és befejezési időpontjainak az *m* gép rendelkezésre állási időintervallumaihoz való illesztését, figyelembe véve a gép utolsó terhelésének befejezési időpontját. Ennek során, az *i* munka által képviselt fix méretű terhelést (időablakot) az *m* gép első (vagyis legkorábban kezdődő) megfelelő méretű szabad időintervallumára balra illesztve helyezi el.

A szimuláció lefutásának eredményei az egyes munkák időadatai, amelyek az *M_STET* tömbben találhatók.

4.4 Ütemtervek értékelése

A kiszámított időadatok felhasználásával egy ütemterv jóságát, minőségét célfüggvények megfogalmazásával és azok kiértékelésével lehet számszerűsíteni.

Valós környezetben nagyon sokféle termelési cél megfogalmazható [6]. Ismeretes, hogy a termelési modellek széles körében az ismert termelési háromszög állapotváltozói: a lekötött készletszint, az erőforrás-kihasználás és a szállítókészség szükséges és elegendő mértékben meghatározza a termelési folyamat minősítését [14].

A modellünkben kiemelt szerepet játszanak a határidőhöz kapcsolódó célfüggvények, ugyanis a határidőket nem korlátozásként vesszük figyelembe, hanem célfüggvénybe építjük azokat.

Az *i* munka határidejét D_i -vel a tényleges befejezési időpontját C_i -vel jelölve definiálható a munka határidőtől való eltérése: $L_i = C_i - D_i$ (2) és késése: $T_i = \max(0, L_i)$ (3). Ezeket G_i -vel jelölve kapjuk a következő gyakran alkalmazott célfüggvényeket:

$$\gamma = \max_i(G_i) \ (4),\ \gamma = \sum_i G_i \ (5) \text{ és } \gamma = \frac{\sum_i G_i}{n} \ (6).$$

Továbbá, fontos célfüggvény még a késő munkák száma $\gamma = |\{i \mid T_i > 0\}|$ (7).

A munkákhoz hasonlóan a megrendelésekre vonatkoztatva is számítjuk a fenti célfüggvényeket.

4.5 Heurisztikus ütemező algoritmus

Többféle heurisztikus megoldási módszert fejlesztettünk ki. A végrehajtási útvonal és gépek munkákhoz való rendelésének, valamint a munkák gépenkénti sorrendjének meghatározása alapján két csoportba sorolhatók az algoritmusok.

Az első csoportba tartoznak a felépítő jellegű heurisztikák. Ezek – a korábban ütemezett és nem módosítható hozzárendelésektől eltekintve – üres ütemtervből kiindulva, a kiválasztott soron következő ütemezendő munkához tartozó útvonalra, gépre és sorrendre vonatkozó döntéseket heurisztikus szabályok és szimulációs kiértékelés alapján hozzák meg.

A második csoportba tartoznak az iteratív javításon alapuló heurisztikák. Ezeknél a módszereknél egy kezdeti megvalósítható ütemtervből kiindulva megengedett módosítások és szimulációs kiértékelések iterálásával alakul ki a választott célfüggvénynek jobban megfelelő ütemterv.

A továbbiakban az ütemterv elkészítését – kombinált módszerrel - egy keresési feladatként fogalmazzuk meg. Majd egy módosított tabukeresési technikával oldjuk meg a feladatot. A megoldás során felhasználjuk a 4.2 - 4.4 fejezetben leírtakat.

Az ütemezés során a modellben szereplő korlátozások közül a munkákhoz rendelt befejezési határidőket puha (megsérthető) korlátozásnak tekintjük, a célfüggvényekben vesszük figyelembe azokat. Minden más korlátozást kemény (nem megsérthető korlátozásnak) tekintünk.

A keresési folyamat során egy kezdeti ütemtervből kiindulva megengedett módosítások ismételt végrehajtásával alakul ki a végső ütemterv. A kiindulási ütemterv elkészíthető felépítő jellegű heurisztikus szabály alkalmazásával, kézi ütemezéssel vagy a megfelelő tömbökből véletlenszerűen kiválasztott értékek felhasználásával.

A módosító operátor alapvetően kétféle változtatást idézhet elő az ütemtervben:

1. Egy kiválasztott munkát kiemel az ütemtervből, majd a hozzárendelt útvonalat és gépeket részben, vagy teljes egészében megváltoztatva újra beszúrja azt az ütemtervbe.
2. Egy kiválasztott gépen a munkák sorrendjét megváltoztatja.

Egy keresési folyamat egy közbenső lépése során az algoritmus bizonyos számú (K) kiterjesztett ütemtervet készít a módosító operátor alkalmazásával. Ha egy kiterjesztett ütemterv szerepel a tabulistán, akkor azt nem értékeli ki, figyelmen kívül hagyja. Ha egy kiterjesztett ütemterv nem szerepel a tabulistán, akkor felkerül a listára és a legkorábban felvett listaelem törlődik. A szimulációs kiértékelést követően, ha a célfüggvénynek a kiterjesztett ütemtervre vonatkozó értéke jobb, mint az adott kiterjesztés legjobb értéke, akkor megjegyzésre kerül. A kiterjesztés legjobb ütemterve lesz a következő lépés kiterjesztésének a kiindulási bázisa. Továbbá, ha ez az érték jobb, mint a keresés során megtalált legjobb érték akkor ez kerül megjegyzésre.

Az algoritmus paraméterei a következők:

- A tabulista maximális mérete.
- A lépések száma.
- Egy adott kiterjesztés során vizsgált szomszédos ütemtervek maximális száma.
- Egyidejűleg módosítható gépek maximális száma.
- A munkasorrendet megváltoztató permutáció ciklusának maximális hossza.

Az ismertetett alapalgoritmus további bővítő elemeket is tartalmaz. Ezek lehetővé teszik az algoritmus leállási feltételének összetettebb megfogalmazását (pl.: a futási idő elér egy megadott korlátot). Továbbá, lehetőséget biztosítanak a keresési paraméterek futás közben történő változtatására, ezáltal különböző vezérlési stratégiák alkalmazását teszik lehetővé.

6. FUTÁSI EREDMÉNYEK

A bemutatott adatmodellt és algoritmusokat Borland C++ Builder 5 fejlesztőrendszerben implementáltuk.

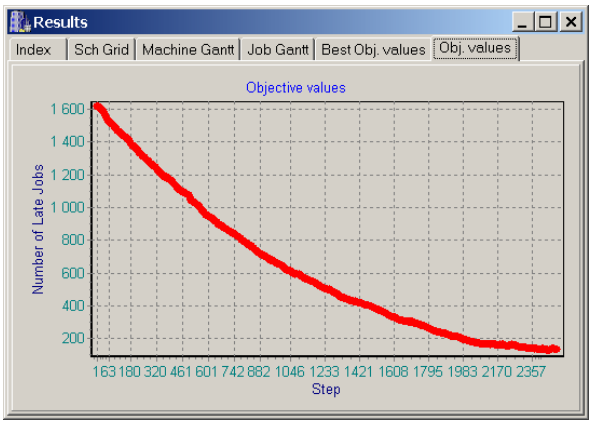
Az elvégzett tesztelések közben azt tapasztaltuk, hogy a futási idő alakulását alapvetően befolyásolja a feladat mérete, és nehézségi foka. Az ütemezési feladat jellemző méreteinek és az alkalmazott keresési paramétereknek a futási időre gyakorolt együttes hatása – magas nehézségi fokú feladatok esetén – az 1. táblázatban látható. (Tesztkörnyezet: Intel Pentium 4 2,8 GHz CPU, 512MB DDR333 RAM, Microsoft Windows XP.)

1. táblázat. **Ütemezés feladatok**

	NO	*NJ*	*NM*	*L*	*K*	**Futási idő**
1	22	78	11	100	100	**2 s**
2	300	1052	51	2000	25	**1 min 9 s**
3	700	2455	151	2500	10	**2 min 35 s**

Az 1. táblázatban az *NO* a megrendelések számát, *NJ* a munkák számát, *NM* a gépek számát, *L* a lépések számát, *K* a vizsgált szomszédos ütemtervek számát jelenti.

Ezekben a leegyszerűsített keresési feladatokban az *L* és *K* konstansok, a kiválasztott célfüggvény pedig a késő munkák száma. A célfüggvény változása a megtett lépések számának függvényében a 4. ábrán látható.



4. ábra: A késő munkák számának változása az 3. feladat megoldása közben

Az eddig elvégzett vizsgálatok igazolták, hogy a javasolt megoldási módszer nagyméretű feladatok esetében is hatékonyan alkalmazható. Rugalmasan alkalmazkodik a kiválasztott célfüggvényhez és kivárható időn belül szolgáltatja az eredményt. További hatékonysági vizsgálatokat jelenleg is végzünk a kidolgozott algoritmusok értékelése és továbbfejlesztése érdekében.

7. ÖSSZEFOGLALÁS

A cikkben az igény szerinti tömeggyártás műhelyszintű ütemezési feladataival foglalkoztunk. Az erőforrások allokálásának és a munkák ütemezésének modellezésére egy új kiterjesztett rugalmas Flow Shop ütemezési modellt vezettünk be. Az alternatív útvonalak, korlátozottan rendelkezésre álló párhuzamos gépek, sorrendfüggő átállási idők, terméktől függő termelési sebességek valamint határidős munkák együttes figyelembevételével valós gyártási környezethez alkalmazkodó ütemezési koncepciót mutattunk be. Ismertettünk egy heurisztikus megoldási módszert, amely lokális keresési technikát valamint indexelt adatmodellen alapuló gyors kiértékelést kombinál.

KÖSZÖNETNYILVÁNÍTÁS

A szerző ezúton mond köszönetet a kutatási és fejlesztési munka támogatásáért a Magyar Tudományos Akadémia Termelésinformatikai Kutatóhelyének (alapítva a Miskolci Egyetem Alkalmazott Informatikai Tanszékén, Grant No. MTA – TKI 06108). Az ismertetett eredmények a „VITAL” nevű projekthez (Nemzeti Kutatási és Technológiai Hivatal, Grant No.: 2/010/2004) kapcsolódó kutatási munkák során születtek.

IRODALOM

[1] ACERO-DOMINGUEZ M. J., PATERNINA-ARBOLEDA C. D., 2004, *Scheduling Jobs on a K-Stage Flexible Flow Shop Using a TOC- (Bottleneck) Procedure*, Proceedings of the 2004 Systems and Information Engineering Design Symposium.

[2] BARKMEYER E., DENNO P., FENG S., JONES A., WALLACE E., 1999, *"NIST Response to MES Request for Information"*, NISTIR 6397, National Institute of Standards and Technology: pp. 1-124.

[3] BRUCKER P., 1998, *Scheduling Algorithms*, Springer-Verlag, Berlin.

[4] BAUER A., BOWDEN R., BROWNE J., DUGGAN J., LYONS G., 1991, *Shop Floor Control Systems – From design to implementation*, Chapman &.Hall, UK.

[5] YAMADA T., 2003, *Studies on Metaheuristics for Jobshop and Flowshop Scheduling Problems*, Ph.D Thesis, Kyoto University, Japan.

[6] KULCSÁR GY., 2006, *Az igényszerinti tömeggyártás termelésprogramozásának modellezése és heurisztikus megoldása*, GÉP, A Gépipari Tudományos Egyesület Műszaki Folyóirata, LVII. évfolyam 2006/10 szám, pp. 3-13.

[7] KULCSÁR GY., HORNYÁK O., ERDÉLYI F., 2005, *Shop Floor Decision Supporting and MES Functions in Customized Mass Production*, Conference on Manufacturing Systems Development - Industry Expectations, Wroclaw, Poland, pp. 138 – 152.

[8] KURNAZ A., COHN Y., KOREN Y., 2005, *A Framework for Evaluating Production Policies to Improve Customer Responsiveness*, CIRP Annals, Volume 54/1, pp. 401-406.

[9] LINN R., ZHANG W., 1999, *Hybrid Flow Shop Scheduling: A Survey*, Computers and Industrial Engineering, 37, pp. 57-51.

[10] MCKAY K. N., WIERS V. C. 1999, *Unifying the Theory and Practice of Production Scheduling*, Journal of Manufacturing Systems, Vol. 18, No. 4, pp. 241-248.

[11] MCCLELLAN M., 1997, *Applying Manufacturing Systems*, St. Lucie Press, Florida.

[12] TÓTH T., ERDÉLYI F., 2004, *Research and Development (R&D) Requirements for Up-to-date Production Planning & Scheduling (PPS) Systems*, The Eleventh International Conference on Machine Design and Production, pp. 13 – 15, Antalya, Turkey.

[13] TÓTH T., ERDÉLYI F., 1997, *The Role of Optimization and Robustness in Planning and Control of Discrete Manufacturing Processes*, Proceedings of the 2nd World Congress on Intelligent Manufacturing Processes & Systems, Budapest, Hungary, pp. 205-210.

[14] TÓTH T., 2004, *Termelési rendszerek és folyamatok*, Egyetemi tankönyv, Miskolci Egyetemi Kiadó, Miskolc.

[15] TÓTH T., 1998, *Tervezési elvek, modellek és módszerek a számítógéppel integrált gyártásban*, Egyetemi tankönyv Miskolci Egyetemi Kiadó, Miskolc.

[16] TÓTH T., 1994, *Heurisztikus módszerek termelés-programozási feladatok megoldására*, Oktatási segédlet, Miskolci Egyetem, Miskolc.

[17] TAILLARD E., 1990, *Some Efficient Heuristic Methods for the Flow Shop Sequencing Problem*, European Journal of Operational Research, 47, (1) pp. 65-74.

[18] ZWEBEN M., FOX M. S., 1994, *Intelligent Scheduling*, Morgan Kaufmann Publishing, San Francisco.

[19] SAUER J., 2006, *Modeling and solving multi-site scheduling problems*, Book section in: van WEZEL W., JORNA R., MEYSTEL A., *Planning in Intelligent Systems: Aspects, Motivations and Methods*, pp. 281 - 299, Wiley.

[20] VÍZVÁRI B., 2004, *Ütemezés elmélet*, 9. fejezet az IVÁNYI A. (szerk.), *Informatikai algoritmusok*, 364-415, ELTE Eötvös Kiadó.