

## KITERJESZTETT TERMELÉSPROGRAMOZÁSI MODELL ERŐFORRÁS-KORLÁTOS ÜTEMEZÉSI FELADATOK MEGOLDÁSÁRA

**Kulcsár Gyula**

Egyetemi docens, Miskolci Egyetem, Informatikai Intézet, Alkalmazott Informatikai Tanszék  
3515 Miskolc, Miskolc-Egyetemváros, e-mail: [itkgy@uni-miskolc.hu](mailto:itkgy@uni-miskolc.hu)

**Kulcsárné Forrai Mónika**

Egyetemi tanársegéd, Miskolci Egyetem, Informatikai Intézet,  
Alkalmazott Informatikai Tanszék  
3515 Miskolc, Miskolc-Egyetemváros, e-mail: [aitkfm@uni-miskolc.hu](mailto:aitkfm@uni-miskolc.hu)

### **Összefoglalás**

A cikk a diszkrét gyártási folyamatok egy új ütemezési modelljét és annak gyakorlat-orientált megoldását foglalja össze. A korszerű gyártó-szerelő rendszerekre jellemző alternatív technológiai útvonalak, korlátozottan rendelkezésre álló gépek és gyártóeszközök, eltérő műveletvégzési és átállítási idők valamint határidős munkák együttes figyelembevételével gyakorlati igényekhez alkalmazkodó ütemezési koncepció kerül bemutatásra. Az erőforrások allokálásának, a belső rendelések, munkák és feladatok időbeli végrehajtásának részletes ütemezésére egy kiterjesztett rugalmas job shop modellt definiáltunk. A cikk ismerteti a vizsgált feladattípus legfontosabb jellemzőit és a kifejlesztett heurisztikus megoldási módszer koncepcióját, amely gyors szimulációs kiértékelésre alapozott lokális keresési technikát és többcélú optimalizálási modellt kombinál.

**Kulcsszavak:** termelés, ütemezés, szimuláció, kereső algoritmus, többcélú optimalizálás.

### **Abstract**

The paper summarizes a new model and a practise-oriented approach for solving scheduling problems of discrete manufacturing processes. An advanced scheduling approach is presented which is able to adapt to the requirements of real-life situations by taking into consideration the specific characteristics of modern manufacturing and assembly systems. These detailed constraints and capabilities of the actual resource environment include the alternative technological routes, the limited available machines, the unrelated processing time, the sequence dependent setup time and the jobs with due dates. An extended flexible job shop scheduling model is defined to solve the resource allocation problems and to create the fine schedule of the execution of jobs, tasks and operations. The paper describes the most important characteristics of the analysed problem class and shows the main approach of the developed heuristic solving method. Our approach combines a special searching technique based on fast execution simulation with a multi-objective optimization model.

**Keywords:** production, scheduling, simulation, searching algorithm, multi-objective optimization.

## 1. Bevezetés

Globalizált piaci környezetben a termelő vállalatoknak és termelési hálózatoknak egyre fokozódó kihívásokkal kell szembenézniük. Napjainkban a versenyképesség növelése érdekében rendkívül fontos, hogy a termelő vállalatok minél jobban alkalmazkodjanak a piaci körülmények gyors változásaihoz. Ennek érdekében a gyártási hatékonyságot és a szállító-készséget kell javítani a készletek alacsony szinten tartása mellett. A piaci igények elemzésének eredményére és az elfogadott külső (független vásárlói) rendelésekre alapozott termelés-tervezési döntések következtében meghatározott belső (függő gyártási) rendelések teljesítésének hatékonyságát és átláthatóságát a rövid távú részletes ütemezési feladatok átfogó megoldásával (finomprogramozásával) nagymértékben lehet javítani.

A mindenkor aktuális időintervallumra vonatkozó részletes termelési finomprogram kidolgozása során a gyártási erőforrás-környezetre, a munkák végrehajtási jellemzőire vonatkozó korlátozások figyelembevételével, a belső rendelések teljesítéséhez szükséges műveletek elvégzéséhez erőforrások hozzárendelését és a műveletek indítási időpontjának sorozatát kell megtervezni úgy, hogy az aktuális korlátfeltételek mellett a vállalat kitűzött céljai megvalósuljanak [7].

A diszkrét gyártási folyamatok sajátos jellemzője, hogy a végrehajtás alapelemét jelentő műveletek meghatározott gépeken, berendezéseken, munkahelyeken pontosan előírt sorrendben, különböző szerszámok, készülékek használatával, előzetesen megtervezett technológiai tervek alapján végezhetők el. A műveletek egymástól térben és időben rendszerint jól elkülöníthetők és közöttük rendszerint bonyolult kapcsolatrendszer áll fenn (pl. megelőzési relációk, stb.). A műveletek végrehajtását munkahelyek (pl. kézi munkahelyek, automatizált technológiai berendezések, stb.) valósítják meg. A munkahelyek rendszerint nagyobb irányítási egységekbe (gyártócsoportokba, gyártócellákba, gyártórendszerekbe, gyártóműhelyekbe) szervezve végzik el a hozzájuk rendelt feladatokat. Az ilyen komplex gyártórendszerek irányítási feladatait szoros integrációban célszerű megoldani. A korszerű gyártásirányítás legfontosabb feladataihoz tartoznak a következők:

- A termelési tervek rövidtávú feladatokra bontása, a feladatok végrehajtásának ütemezése (finomprogramozás).
- A feladatok végrehajtásához szükséges anyagi, személyi és információs feltételek biztosítása.
- A feladatok eszközökhöz rendelése és elindítása.
- A folyamatok valós idejű felügyelete, megfigyelése és irányítása.
- A végrehajtás minőségének biztosítása.
- Teljesítménymutatók számítása és az eredmények értékelése.
- Bizonytalanságok és váratlan események kezelése.

A gyártásirányítási feladatok támogatására funkcionális komponensekből álló MES (Manufacturing Execution System) rendszerek alakultak ki. A MES célja a gyártási folyamatok végrehajtásának optimális irányítása, az aktuális üzleti és műszaki célok lehető legjobb megvalósítása. A MES hatóköre alapvetően a belső rendelések kibocsátásától a késztermék előállításáig terjed [2]. A MES alapvetően kettős szerepet játszik. Egyrészt aktuális adatok gyűjtésével, ellenőrzésével, elemzésével és felhasználásával gondoskodik a tervezett tevékenységek időbeli ütemezéséről, elindításáról, irányításáról, a végrehajtás felügyeletéről, másrészt a termelési aktivitásokról feladat-specifikus információkat szolgáltat a vállalat döntési folyamatainak támogatásához.

A MES által támogatott funkciók együttesen egy bonyolult, többszörösen visszacsatolt szabályozási rendszert alkotnak. A kereskedelmi MES rendszerek hatékony működéséhez a különböző modulok illesztése, konfigurálása és finomhangolása elengedhetetlen. Mivel a diszkrét gyártási folyamatok nagyon különbözőek, ezért hatékony irányításuk is különböző modelleket igényel. Ennek következtében a MES rendszerek funkcióinak és szolgáltatásainak továbbfejlesztése széleskörű elmélet- és gyakorlatorientált kutatás tárgyát képezi.

## **2. Termelésütemezési modellek**

A diszkrét termelési folyamatok műhelyszintű előidejű (prediktív) és valós idejű (reaktív) ütemezése mind elméleti, mind gyakorlati szempontból az erősen modellfüggő és komplex kombinatorikus optimalizálási feladatok közé tartozik. A szakirodalomban számos cikk foglalkozik a diszkrét gyártási folyamatok esetében használható ütemezési modellekkel és módszerekkel [1]. A témakörrel foglalkozó kutatók többsége főként operációkutatási szempontok szerint vizsgálja az ütemezési feladatokat [4], [11], [12].

A legegyszerűbb az egygépes ütemezési modelltípus, amelyben minden munka esetében egyetlen operációt kell végrehajtani, és csupán egy gép terhelhető. A párhuzamos gépes modellek esetén egyetlen operációt kell végrehajtani, de több gép dolgozhat párhuzamosan különböző munkákon.

A műhelymodellek esetében több munka, több operáció és több gép tartozik a rendszerbe. Minden munka több operációból állhat. Minden egyes operációt egy adott gép végezhet el (General Shop). A munkák operációinak száma és végrehajtási sorrendje lehet azonos (Flow Shop), munkánként eltérő (Job Shop), és nem korlátozott (Open Shop). A felsorolt alaptípusok kombináltan egyszerre is előfordulhatnak (Mixed Shop).

A rugalmas műhelymodellekben a rugalmas (flexible) jelző az ütemezési modellekhez kapcsolva arra utal, hogy míg az alap műhelymodellekben egy adott operáció végrehajtására egyetlen gép alkalmas csupán, addig a rugalmas modellekben az adott operációt egy adott gépcsoport (halmaz) bármelyik gépe elvégezheti. Ezáltal az alap műhelymodell kibővül operációhoz kapcsolódó gépválasztási feladattal, ugyanakkor továbbra is alapvető szerepet játszik a munkák gépenkénti sorrendjének és indítási időpontjának meghatározása. A gépcsoport gépei egymással párhuzamosan működő egyenértékű, részben azonos vagy különböző intenzitás- és más paraméterértékekkel rendelkező gépek lehetnek. Jellegzetes példái a rugalmas műhelymodelleknek a rugalmas egyutas (Flexible Flow Shop, FFS) és a rugalmas többutas (Flexible Job Shop, FJS) modelltípusok [5], [13].

A termelési folyamatok teljesítménye szempontjából alapvető fontosságúak az erőforrások allokálását és a munkák végrehajtási sorrendjének megválasztását definiáló döntéshozatali stratégiák. Az ütemezési modellek többsége csupán egyetlen elsődleges teljesítménymutató optimalizálására koncentrál. A speciális feladatváltozatokra kifejlesztett módszerek más célfüggvényekre erősen korlátozott mértékben használhatók, részletesebb modellekre pedig még komoly módosításokkal sem igazán alkalmazhatók. A többcélű optimalizálási szemlélet – amely az igény szerinti rugalmas gyártás szempontjából kiemelkedő fontosságú – sokkal kevesebb ütemezési modellben jelenik meg. Ilyen irányú kutatási eredményekről olvashatunk például a [10] és a [14] dolgozatokban.

A jelenleg ismert többgépes és többoperációs műhelyszintű ütemezési modellek (pl. FFS, FJS) rendszerint nem támogatják egyidejűleg a rugalmasan átállítható gépeket, gépcsoportokat és ezek változó rendelkezésre állási időintervallumait, valamint a technológiai útvonal alternatívákat. A napjaink ipari gyakorlatában egyre fontosabbá váló

rugalmas és igény szerinti gyártási folyamatok irányítása szükségessé teszi az ismert ütemezési modellek további jelentős kiterjesztését annak érdekében, hogy olyan szoftvermodulokat tudjunk implementálni, amelyek a különböző technológiai alternatívákat és a korlátozott erőforrásokat egyidejűleg tartalmazó modellváltozatokat is képesek hatékonyan támogatni.

### 3. Kiterjesztett rugalmas Job Shop ütemezési modell

Az ütemezési feladatok formális leírásának eszközeként a szakirodalomban az  $\alpha|\beta|\gamma$  formalizmus [6] használata a legelterjedtebb, ahol  $\alpha$  az erőforrás-környezetet,  $\beta$  a korlátozásokat és végrehajtási jellemzőket, valamint  $\gamma$  az ütemezési célfüggvényeket definiáló szimbólumlista. A listákban szereplő paraméterekre számos javaslat van az irodalomban, és nagyon sokféle modell létezik [4].

Az általunk megfogalmazott és vizsgált probléma sajátosságai miatt nem sorolható be közvetlenül egyik ismert ütemezési feladatkategóriába sem, így egy új feladatosztályt definiáltunk az (1) formalizmussal, melyet kiterjesztett rugalmas job shop (Extended Flexible Job Shop, EFJS) ütemezési feladatnak neveztünk el.

$$FJ, M_g, Q_{i,m}, Set_{i,j,m}, Cal_m / R_i, D_i, Exe_i, A_{i,g} / f_1, f_2, \dots, f_K \quad (1)$$

Az EFJS feladat szimbólumainak jelentése a következő:

- $FJ$ : a műveletek (operációk) sorrendje kötött, de munkánként különbözhet.
- $M_g$ : a gyártórendszer egyetlen vagy akár több operáció együttes végrehajtására alkalmas gépcsoportokból tevődik össze, amelyekhez egy vagy több párhuzamos gép/gépsor tartozhat.
- $Q_{i,m}$ : a gépek/gépsorok munkáktól függő eltérő termelési sebességekkel működhetnek.
- $Set_{i,j,m}$ : munkák sorrendjétől és géptől függő gépátállítási idők.
- $Cal_m$ : gépekre előírt korlátozott (nem folyamatos) rendelkezésre állási időintervallumok. A gépek ezeken az időintervallumokon belül dolgozhatnak.
- $R_i$ : a munkák legkorábbi indítási időpontjai (indításra vonatkozó időbeli korlátozások).
- $D_i$ : a munkák legkésőbbi befejezési időpontjai (teljesítési határidők).
- $Exe_i$ : a munkákhoz tartozó gyártási folyamattípusok (a végrehajtási lépések típusainak és azok sorrendjének megengedett alternatívái).
- $A_{i,g}$ : a munkákhoz rendelhető gépek halmazai gépcsoportonkénti bontásban.
- $f_1, f_2, \dots, f_K$ : a kijelölt minimalizálandó célfüggvények listája (egyszerre több célfüggvény is előírható,  $K$  a célfüggvények aktuális száma).

Dolgozatunkban példaként megemlítünk néhány jellemző célfüggvényt, melyek a rugalmas és igény szerinti gyártásban előnyösen használhatók:

- Csúszó (határidőt túllépő) belső rendelések száma.
- Csúszó (határidőt túllépő) munkák száma.
- Legnagyobb csúszás [perc].
- Csúszások összege [perc].
- Átállások száma.
- Átállások idejének összege [perc].

- Átlagos gépkiszáratlanság (100 – átlagos gépkiszáratlanság) [%].
- Átlagos átfutási idő [perc], stb.

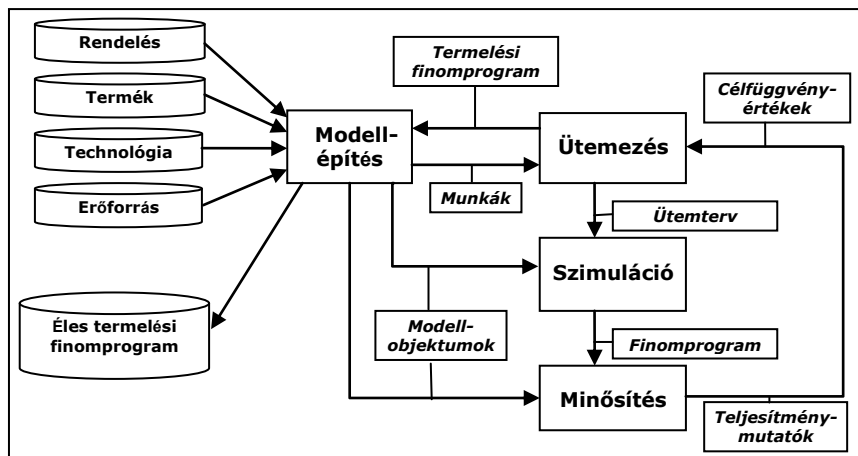
A feladatban kitűzött célfüggvények listája bővíthető, adott igényeknek megfelelően átalakítható. Az ütemezési célok egymással összefügghetnek, közöttük bonyolult, nehezen modellezhető kölcsönkapcsolat állhat fenn [3]. A megfogalmazott célok fontossága időben változhat, ezért a célfüggvények aktuális fontosságát prioritásértékek megadásával és finomhangolásával fejezhetjük ki az aktuális elvárásoknak megfelelően.

A bemutatott ütemezési feladattípus döntési változóinak kombinatorikus tulajdonságai miatt az NP-nehez feladatosztályba tartozik. Az elméleti globális optimum keresése helyett, olyan megoldási módszereket fejlesztettünk ki, amelyek nagyméretű feladatok (pl. 300 megrendelés, 3000 munka, 100 gép) esetében is elfogadható időn belül (legfeljebb 10 perc alatt) több kritérium együttes figyelembevételével kompromisszumosan jó megoldást képesek előállítani.

#### 4. Szimulációra és többcélú keresési módszerre alapozott megoldás

A termelési folyamatok finomprogramozásakor a termelő rendszer lehetőségeit, képességeit és korlátait figyelembe véve, a belső rendelések teljesítéséhez szükséges munkák időbeli végrehajtását kell megtervezni. Ennek során a szükséges erőforrások allokálását és a feladatok végrehajtásának indítási időpontját kell megtervezni úgy, hogy a vállalat magasabb szintjén megfogalmazott célok és a gyártásirányítás járulékos saját belső céljai egyaránt megvalósuljanak.

Az EFJS feladatosztályba besorolható feladatok megoldására egy integrált, többcélú, heurisztikus keresési technikára és szimulációra alapozott megoldási módszert fejlesztettünk ki. Az EFJS egyik speciális változatának, az ún. kiterjesztett rugalmas Flow Shop (Extended Flexible Flow Shop, EFFS) probléma modellezéséből és megoldásából indultunk ki [7], [8], [9]. A továbbfejlesztett megoldási koncepció elvi vázlatát az 1. ábrán látható.



1. ábra. Szimulációra alapozott termelésprogramozás.

A termelésinformatikai rendszer adatbázisában tárolt adatok és az interaktív felhasználói felületen beállított adatok felhasználásával egy modellépítő komponens definiálja a rendszerben lévő modell-objektumokat (entitásokat). Kiemelt fontosságú feladata a belső rendelkezésekhez kapcsolódó munkák és az érvényben lévő korlátozások definiálása. A korlátozások közül az előírt vagy számítható belső határidőket „puha” (adott esetben megsérthető) korlátozásoknak tekintve a csúszások és késések minimalizálása ütemezési kritériumként jelenik meg. A további (pl. technológiára, korlátozott erőforrásra, műveletvégrehajtásra vonatkozó stb.) korlátozások „kemény” (meg nem sérthető) előírásokká válnak. A szükséges rendelkezésre állási, alkalmazhatósági és megvalósíthatósági vizsgálatok elvégzését követően a modellépítő komponens felépíti a modellobjektumok között fennálló teljes kapcsolatrendszerrel, melyet indexelt többdimenziós adatmodellben tárol. Ennek elsődleges célja az, hogy a feladatmegoldás során minden egyes döntési helyzetben a választható alternatív lehetőségek (a döntési változók megengedett értékei) világosan és könnyen felismerhetők legyenek.

A belső rendelkezések ütemezési alapegységekre bontásával önálló munkák jönnek létre. Minden egyes munka (job) ütemezése önálló független döntési változók értékének megválasztásával valósul meg. Az ütemező modul minden egyes munkához hozzárendel egy megfelelő végrehajtási útvonalat, továbbá hozzárendel egy megfelelő gépet a kiválasztott útvonal minden egyes végrehajtási lépésének megfelelő gépcsoportból, és meghatározza minden érintett gépen a munkák végrehajtási sorban elfoglalt pozícióját. Ezáltal az ütemező algoritmus a munkákhoz hozzárendeli a konkrét feladatlistát (task list), és így a gépeken a gyártási sorozatnagyságok (batch size) és az azokat elválasztó átállítási műveletek (setup) a döntési változók függvényében ütemezés közben alakulnak ki. Nincs szükség előzetes sorozatnagyság tervezésre, ez a funkció az ütemező hatáskörébe tartozik. Az ütemezési (döntéshozatali) folyamat eredményeként elkészül egy termelési ütemterv.

Az ütemtervben szereplő feladatok végrehajtásához kapcsolódó időadatokat egy megfelelően gyors végrehajtás-vezérelt szimulációs algoritmus számítja ki. A szimuláció figyelembe veszi a gépek egyedileg dedikált rendelkezésre állási időintervallumait, a munkák sorrendje által meghatározott átállítási időket, a munka-gép párosítások alapján számítható megmunkálási időket és az egyéb kapcsolódó járulékos időket. A szimuláció közben alakul ki a feladatok gépenkénti tervezett – és következményként a munkák, valamint a megrendelések származtatott – indítási és befejezési időpontja.

A szimuláció által számított időadatokat és a végrehajtásra vonatkozó egyéb számértékek felhasználásával a termelési ütemterv kibővül és termelési finomprogrammá alakul át. Egy értékelő komponens kiszámítja a vizsgált termelési finomprogramra (mint megoldásra) vonatkozó aktuális célfüggvény-értékeket és teljesítménymutatókat.

Az ütemező modul iteratívan módosítja az aktuális ütemtervet, konzisztens változtatásokkal új megoldás-változatokat készít, majd szimulációt és kiértékelést követően a célfüggvény-értékektől és a leállási feltételtől függően tovább folytatódik a legjobb megoldás keresése.

Az ütemezési feladat döntési változóinak értékét egy többoperátoros és többcélú kereső algoritmus (*MOMOTS*) állítja be. Az általunk kifejlesztett algoritmus váza a 2. ábrán látható. A keresési folyamat során egy kezdeti ( $s_0$ ) ütemtervből kiindulva megengedett módosítások ismételt végrehajtásával alakul ki a végső ütemterv ( $s^*$ ). A kiindulási ütemterv döntési változói heurisztikus felépítő szabályok alkalmazásával a modellépítő algoritmus által megfelelően előkészített kapcsolótömbökből kiválasztott értékekkel inicializálódnak.

Az iteratív javítás egy közbenső lépése során, az aktuális kiterjesztés bázismegoldásából ( $s_0$ ) kiindulva az algoritmus paraméterben definiált számú kiterjesztett ( $s$ ) ütemtervet készít az aktuálisan kiválasztott módosító ( $N_c$ ) operátor alkalmazásával. A módosító operátorok alapvetően a munkákhoz rendelt útvonal típusra, konkrét gépekre és végrehajtási sorrendekre vonatkozó döntési változók értékeit módosítják különböző mértékben. A módosító operátorok kiválasztását futás közben dinamikusan változó prioritáslista (*priority\_list*) és kvázi-véletlenszám generátor együttműködése határozza meg. A prioritáslista az operátorok kiválasztási valószínűségét írja le a korábbi kiválasztások és az elért javító hatások függvényében. Ha egy kiterjesztett ( $s$ ) ütemterv szerepel a tabulistán (*Taboo\_List*), akkor az algoritmus azt nem értékeli ki, figyelmen kívül hagyja, ellenkező esetben felkerül a tabulistára, és ha a megengedett tabuelemek száma elérte a maximális értéket, akkor a legkorábban felvett listaelem törlődik. Szimulációs kiértékelést követően, ha a célfüggvények alapján a kiterjesztett ütemterv jobb, mint az adott kiterjesztés legjobb ütemterve ( $s < s_k$ ), akkor megjegyzésre kerül. A kiterjesztés legjobb ütemterve lesz a következő lépés kiterjesztésének a kiindulási bázisa ( $s_0 \leftarrow s_k$ ), és ha ez a megoldás jobb, mint a keresés során megtalált legjobb megoldás ( $s_k < s^*$ ), akkor ez kerül megjegyzésre ( $s^* \leftarrow s_k$ ).

```

MOMOTS
{
   $s_0 \leftarrow$  Kezdeti megoldás készítése;
   $s^* \leftarrow s_0$ ;
  Taboo_List  $\leftarrow$  NULL;
  while ( Leállási feltétel nem teljesül )
  {
    while ( Szomszédság kiterjesztésének feltétele teljesül )
    {
       $N_c \leftarrow$  Az aktuális szomszédsági operátor kiválasztása (priority_list);
       $s \leftarrow$  Szomszédos megoldás készítése (  $s_0, N_c$  );
      if ( A Taboo_List nem tartalmazza (  $s$  ) )
      {
        A Taboo_List bővítése új elemmel (  $s$  );
        if ( Taboo elemek száma > megengedett érték )
          A Taboo_List legkorábban felvett elemének törlése;
        if ( A szomszédság kiterjesztésének első eleme (  $s$  ) )  $s_k \leftarrow s$ ;
        else if (  $s < s_k$  )  $s_k \leftarrow s$ ;
      }
    }
     $s_0 \leftarrow s_k$ ;
    if (  $s_k < s^*$  )  $s^* \leftarrow s_k$ ;
  }
  return  $s^*$ ;
}

```

2. ábra. Többoperátoros, többcélú kereső algoritmus.

Az ütemezési feladatban szereplő különböző típusú, értékészletű és fontosságú összetevőkből felépülő változatos célfüggvény-rendszerek kezelésére egy új szemléletű matematikai modellt használunk. A módszer alapelve az, hogy két különböző megoldás összeha-

sonlításakor az egyik megoldásnak a másikhoz viszonyított (relatív) jóságának számértéke alapján döntjük el, hogy melyiket tekintjük jobb megoldásnak [7]. Az összehasonlító modell formális definícióját a (2), (3) és (4) képletek írják le.

$$f_k : S \rightarrow \mathfrak{R}^+ \cup \{0\}, \forall k \in \{1, 2, \dots, K\} \quad (2)$$

$$D : \mathfrak{R}^2 \rightarrow \mathfrak{R}, D(a, b) = \begin{cases} 0, & \text{ha } \max(a, b) = 0 \\ \frac{b - a}{\max(a, b)}, & \text{egyébként} \end{cases} \quad (3)$$

$$F : S^2 \rightarrow \mathfrak{R}, F(s_x, s_y) = \sum_{k=1}^K \left( w_k \cdot D(f_k(s_x), f_k(s_y)) \right) \quad (4)$$

A matematikai modellben alkalmazott szimbólumok jelentése a következő:

- $S$ : a megoldások halmaza.
- $f_k$ : a  $k$ . célfüggvény minimalizálandó alakban megadva.
- $K$ : a célfüggvények száma.
- $D(a, b)$ : a változás mértékét kifejező függvény, ahol  $a$  és  $b$  valós számok.
- $s_x, s_y$ : két különböző megoldás.
- $w_k$ : az  $f_k$  célfüggvény fontosságát kifejező nem negatív érték (prioritás).
- $F(s_x, s_y)$ : az  $s_y$  megoldás  $s_x$  megoldáshoz viszonyított relatív jósága.

Az  $F(s_x, s_y)$  kétváltozós függvény előjeles értékének felhasználásával az alapvető relációs operátorok értelmezése kiterjeszthető a többcélú keresési feladat  $S$ -beli  $s_x$  és  $s_y$  megoldásaira az (5) definíció szerint.

$$(s_y ? s_x) := (F(s_x, s_y) ? 0) \quad (5)$$

A formális leírásban a kérdőjel (?) tetszőleges relációs operátort ( $<, \leq, >, \geq, =, \neq$ ) helyettesíthet. A szimulációval előállított célfüggvény-értékekre alapozott, közvetlenül a lehetséges feladatmegoldások halmazán értelmezett relációs operátorok felhasználásával a különböző megoldások páronkénti összehasonlítása és értékelése egyszerűen megoldható. A módszer többféle feladat-megoldási stratégiában jól felhasználható az aktuális célokra legjobban megfelelő megoldás megtalálására.

## 5. Néhány jellemző futási eredmény

A kidolgozott ütemezési modellt és megoldó módszert C++ programozási nyelven implementáltuk. A modellünk sajátosságai miatt a szakirodalomban összehasonlításra alkalmas tesztadatokat nem találtunk, így saját fejlesztésű problémagenerátorral állítottunk elő tesztelésre alkalmas adathalmazokat.

A cikkben bemutatott heurisztikus megoldási módszer pontosságának vizsgálatához összehasonlítás céljából implementáltunk egy teljes leszámításra épülő egzakt ütemező algoritmust. A különböző összetételű és nehézségi fokú tesztfeladatok vizsgálata során azt tapasztaltuk, hogy a megvizsgált kisméretű feladatok megoldása során a többcélú heurisztikus kereső-algoritmus minden esetben optimális megoldást adott.

A cikkben vázolt megoldási módszerünk hatékonyságának vizsgálatához nagyméretű ütemezési feladatokat generáltunk többféle paraméterezéssel. Az ütemezési feladat néhány jellemző méretének a futási időre gyakorolt hatása magas nehézségi fokú feladatok eseté-

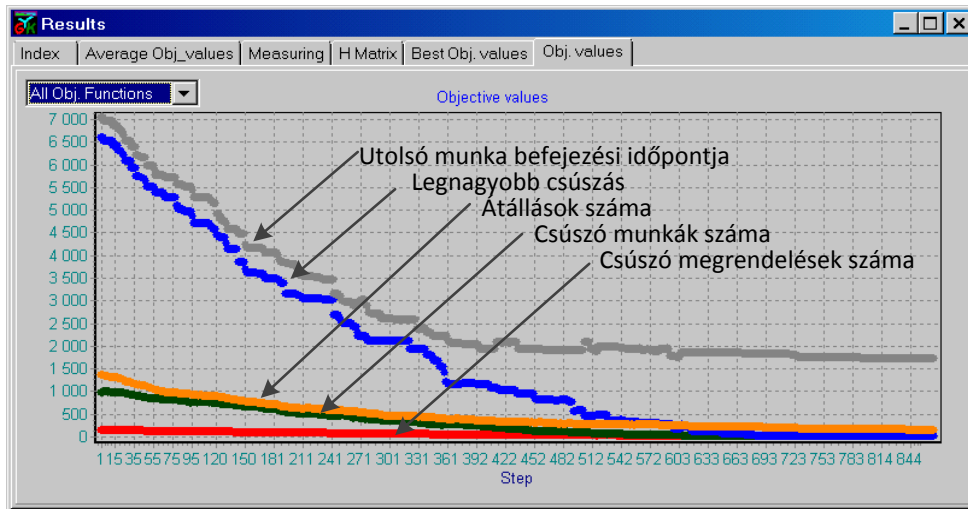


ben tájékoztató jelleggel az 1. táblázatban látható. (Tesztkörnyezet: Intel Core 2 DUO T9550 2,66 GHz CPU, 4GB RAM, Microsoft Windows Vista Business OS.).

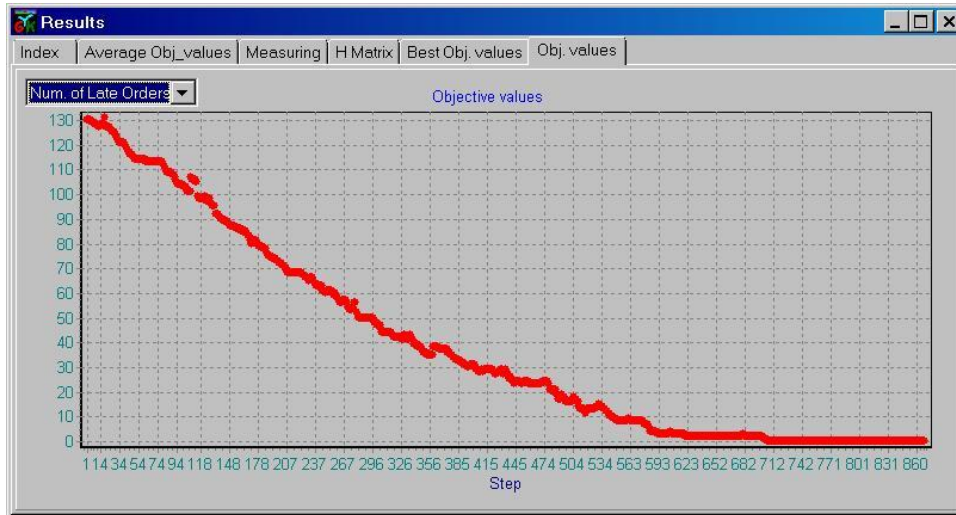
1. táblázat. Feladatméretek és futási idők.

Feladat	Belső rendelések száma	Kapcsolódó munkák száma	Gépek száma	Futási idő
1.	190	1841	74	1 min 49 s
2.	231	2381	40	2 min 16 s
3.	305	3119	187	3 min 30 s
4.	306	3254	200	1 min 29 s
5.	388	3890	118	5 min 29 s
6.	393	2173	119	6 min 55 s

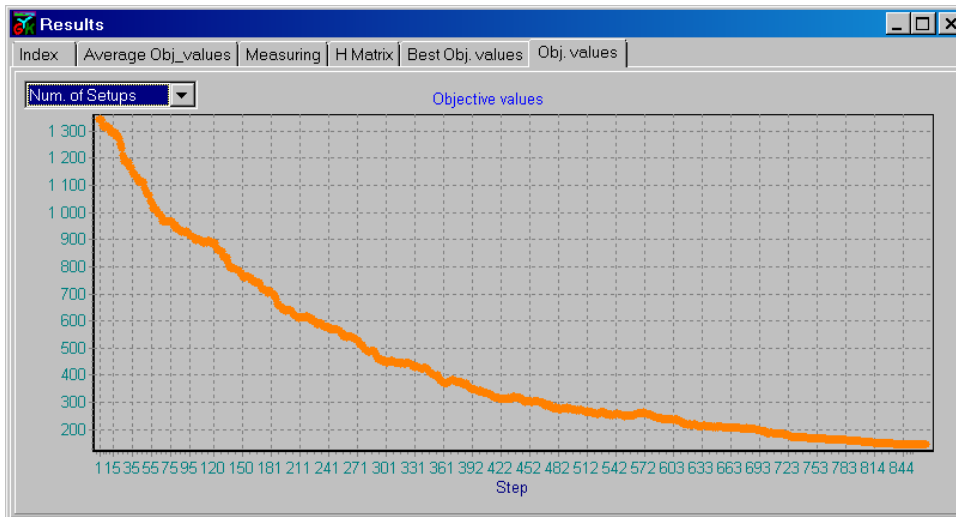
A tesztfeladatok megoldása során a következő minimalizálandó célfüggvényeket (és prioritásértékeket) használtuk: csúszó megrendelések száma (7), csúszó munkák száma (8), csúszasok összege (10), legnagyobb csúszás (10), átállások száma (7), utolsó munka befejezési időpontja (5). A célfüggvény-értékek együttes változását a keresés során megtett lépések számának függvényében – közös koordináta rendszerben megjelenítve – a 3. ábra mutatja. Külön megjelenítve a csúszó megrendelések számának és az átállások számának változása – szintén a keresés során megtett lépések számának függvényében – a 4. és az 5. ábrán látható.



3. ábra. A célfüggvény-értékek változása az 1. feladat megoldása során.



4. ábra. A csúszó megrendelések számának változása az 1. feladat megoldása során.



5. ábra. Az átállások számának változása az 1. feladat megoldása során.

Összehasonlítottuk az EFFE és az EFJS feladatok szimulációs algoritmusait kihasználva, hogy az EFFE feladatok az EFJS feladatok valódi részalmazát képezik. Korlátlan méretű műveletközi tárolókat feltételezve, az EFJS probléma szimulációjához szükséges algoritmus nagy feladat esetén akár 30-szor is lassabb lehet, mint az EFFE feladatra optimalizált szimulációs algoritmus. Ennek az elsődleges oka az, hogy míg az EFFE esetén elegendő a technológiai útvonalakból adódó logikai sorrendeket betartva egyetlen szekvenciában végigjárni az összes gépet az operációk időadatainak számításához, addig EFJS esetén kötelező az operációkhoz kapcsolódó eseményeket időrendben végigkövetni. Korlátozott méretű műveletközi tárolók és/vagy osztott erőforrások (pl. szerszámok, beállító szakmunkás stb.)

esetében a jelentős sebességkülönbség eltűnik. Valós ipari körülmények között sok esetben szükség van az EFJS modell által kínált kiterjesztések és bővítések bevetésére.

Az elvégzett vizsgálatok eredményeiből azt a következtetést fogalmaztuk meg, hogy a javasolt megoldási módszer nagyméretű feladatok esetében is hatékonyan alkalmazható, rugalmasan alkalmazkodik az aktuálisan előírt célfüggvény-rendszerhez és rövid időn belül szolgáltatja az eredményeket.

## **6. Összefoglalás**

A cikkben röviden vázoltuk a diszkrét termelési folyamatok alapvető jellemzőit és azok irányításával kapcsolatban felmerülő legfontosabb feladatokat. A gyártásirányításhoz kapcsolódó ütemezési feladatok nagyon sokfélék lehetnek, melyek összetett, nehezen megoldható többcélú optimalizálási problémákhoz vezetnek. Bemutattunk egy új ütemezési feladatosztályt (EFJS) és annak egy hatékony, többcélú keresési algoritmusra és szimulációra alapozott heurisztikus megoldási módszerét.

A kidolgozott elméleti modellek és módszerek alapján implementáltunk egy továbbfejlesztett termelésprogramozó szoftvert. A számítógépes alkalmazás a betöltött aktuális belső rendelések, gyártási erőforrás-környezet, korlátozások és végrehajtási jellemzők alapján a definiált célfüggvény-rendszernek megfelelően képes automatikusan generálni a feladat megoldását jelentő részletes termelési finomprogramot.

A vizsgált termelésütemezési feladat többcélú keresési feladattá alakításával és a problémátér szimulációra alapozott transzformációjával létrehozott modell és megoldási módszer széles feladatkörben felhasználható.

A végrehajtás-szimulációra alapozott problémátér-transzformáció légyege, hogy az ütemező modul csupán egy „egyszerűsített” ütemtervet készít, amely a munkák és gépek megengedett összerendeléseit és a munkák végrehajtásának gépenkénti sorrendjét tartalmazza. A gyártórendszer jellemzőit és korlátozásait figyelembe véve, a gyártási folyamatokat szimuláló algoritmus rendeli hozzá az ütemtervhez a részletes termelési finomprogramot azáltal, hogy kiszámítja a gyártási feladatok és munkák pontos időadatait. A redukált problémátérben a lehetséges megoldásokat reprezentáló döntési változók egyszerűbben kezelhetők annak köszönhetően, hogy a szimuláció magába foglalja a konkrét ütemezési probléma gyártórendszer-függő jellemzőit, így a megoldó szoftvermodul általánosabb érvénnyel fejleszthető.

A megoldások egymáshoz viszonyított (relatív) minőségének számszerűsítésére kifejlesztett matematikai modell a termelésütemezési problémáktól teljes mértékben független, ezáltal felhasználható ismert keresési algoritmusokkal és metaheurisztikákkal kombinálva tetszőleges kombinatorikus többcélú optimalizálási feladat megoldására.

## **7. Köszönetnyilvánítás**

A kutató munka a Miskolci Egyetem stratégiai kutatási területén működő Mechatronikai és Logisztikai Kiválósági Központ keretében valósult meg.

## 8. Irodalom

- [1] Allahverdi, A. Ng. C. T., Cheng, T. C. E., Kovalyov M. Y.: *A Survey of Scheduling Problems with Setup Times or Costs*, European Journal of Operational Research, 187, 2008, pp. 985-1032.
- [2] Barkmeyer, E., Denno, P., Feng, S., Jones, A., Wallace, E.: *NIST Response to MES Request for Information*, NISTIR 6397, National Institute of Standards and Technology, 1999, pp. 1-124.
- [3] Bikfalvi, P., Kulcsár, Gy., Erdélyi, F., Tóth, T.: *Performance Analysis of some Manufacturing Systems based on Multi-Objective Approach*, Proceedings of the 15th International Conference on Machine Design and Production (UMTIK 2012), Pamukkale, Denizli, Turkey, 19th–22th June 2012, CD, Paper No. 55.
- [4] Brucker P.: *Scheduling Algorithms*, 5th ed, Springer, 2007, p. 371.
- [5] Defersha, F. M., Chen, M.: *A Parallel Genetic Algorithm for a Flexible Job-Shop Scheduling Problem with Sequence Dependent Setups*, International Journal of Advanced Manufacturing Technology, 49 (1). (2012), pp. 263-279.
- [6] Graham, R. L., Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G.: *Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey*, Annals of Discrete Mathematics, 5, 1979, pp. 287–326.
- [7] Kulcsár, Gy.: *Ütemezési modell és heurisztikus módszerek az igény szerinti tömeggyártás finomprogramozásának támogatására*, Doktori (PhD) értekezés, Miskolci Egyetem, Miskolc-Egyetemváros, 2007, p. 146.
- [8] Kulcsár, Gy., Erdélyi, F.: *A New Approach to Solve Multi-Objective Scheduling and Rescheduling Tasks*, International Journal of Computational Intelligence Research, 3 (4), 2007, pp. 343-351.
- [9] Kulcsár, Gy., Kulcsárné, F. M.: *Detailed Production Scheduling Based on Multi-Objective Search and Simulation*, Production Systems and Information Engineering, A Publication of the University of Miskolc, Vol. 6, 2013, pp. 41-56.
- [10] Loukil, T., Teghem, J., Tuyttens, D.: *Solving Multi-Objective Production Scheduling Problems Using Metaheuristics*, European Journal of Operational Research, 161, 2005, pp. 42-61.
- [11] Pinedo, M. L.: *Scheduling: Theory, Algorithms, and Systems*, 4th ed, Springer, 2012, p. 696.
- [12] Pinedo, M. L.: *Planning and Scheduling in Manufacturing and Service*, 2th ed, Springer, 2009, p. 537.
- [13] Quadt, D., Kuhn, H.: *A Taxonomy of Flexible Flow Line Scheduling Procedures*, European Journal of Operational Research, 178, 2007, pp. 686-698.
- [14] Smith, K. I., Everson, R. M., Fieldsend, J. E.: *Dominance Measures for Multi-Objective Simulated Annealing*, Proceedings of Congress on Evolutionary Computation, Portland, Oregon, USA, 19-23 June 2004, pp. 23-30.