

## INTEGRÁLT TERMELÉSÜTEMEZÉSI ÉS MŰSZAKBEOSZTÁSI FELADATOK MEGOLDÁSA

**Kulcsár Gyula**

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

**Kulcsárné Forrai Mónika**

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

### **Absztrakt**

A cikk bemutatja egy integrált termelésütemezési és műszakbeosztási feladat megoldását. A feladat jellegzetessége, hogy összeszerelő gépsorok és alkatrészgyártó gépek működését kell összehangolni megfelelő ütemtervek készítésével. Ezen túlmenően az ütemezés és a műszakbeosztás egy integrált feladatát kell megoldani. A cikk összefoglalja a vizsgált feladat legfontosabb jellemzőit és a kifejlesztett megoldási módszer koncepcióját.

**Kulcsszavak:** gyártásirányítás, ütemezés, műszakbeosztás, optimalizálás, kereső algoritmus

### **Abstract**

The paper describes the solution of an integrated production scheduling and shift-assignment problem. The main characteristic of the problem is that the operation of assembly lines and component-part manufacturing machines must be coordinated by generating appropriate schedules. In addition, it is needed to solve an integrated task of scheduling and shift-assignment. The paper presents the main features of the problem and the approach of the developed solving method.

**Keywords:** manufacturing control, scheduling, shift-assignment, optimization, search algorithm,

### **1. Bevezetés**

A diszkrét termelési folyamatok ütemezési feladatainak megoldásához szükség van az adott termelési folyamat megfelelő modellezésére. A napi gyakorlat szempontjából rendkívül fontos, hogy olyan ütemező szoftverek legyenek bevethetők, amelyek megvalósítható és kellően hatékony (optimum közeli) ütemterveket készítenek. Az ütemező szoftverek eredményessége nagymértékben attól függ, hogy a termelés változó feltételeihez és igényeihez mennyire tudnak alkalmazkodni.

A termelő vállalatok szeretnék a versenyképességüket folyamatosan javítani. Ennek érdekében arra törekednek, hogy a termelési rendszerek és hálózatok minél jobban alkalmazkodjanak a piaci körülmények gyors változásaihoz. A vevők igényeinek kielégítése érdekében a gyártási hatékonyságot és a szállítókésztséget kell javítani. A gazdasági és finanszírozási szempontok a készletek alacsony szinten tartását, az erőforrások magas fokú kihasználtságát és a járulékos költségek csökkentését fogalmazzák meg.

A termelési főterv készítésével kapcsolatos döntések a piaci igények elemzésének eredményére és a valós vevői rendelésekre alapozzák a szükségletek kielégítését megvalósító belső (függő gyártási és beszerzési) rendelések kibocsátását. A helyben gyártandó munkadarabok és részegységek, valamint a

szerelt fődarabok és végtermékek elkészítésének hatékonyságát a célirányosan fejlesztett ütemező szoftverek nagymértékben fokozhatják.

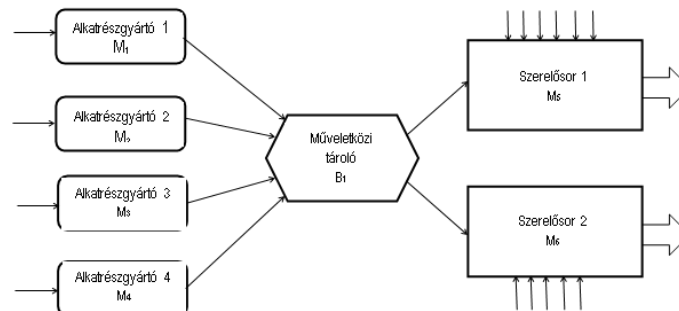
Az aktuális tervezési időhorizontra vonatkozó termelési ütemtervek kidolgozása több részfeladat együttes megoldását követeli meg. A belső rendelések teljesítéséhez szükséges műveletek elvégzéséhez erőforrások hozzárendelését és a műveletek indítási időpontját kell megtervezni olyan módon, hogy az aktuális korlátfeltételek mellett a kitűzött célok megvalósuljanak [5]. A korlátfeltételeket alapvetően a gyártási és a logisztikai erőforrások működésére, valamint a munkák és a műveletek végrehajtására vonatkozó szigorú előírások határozzák meg.

Cikkünkben bemutatunk egy integrált termelésütemezési és műszakbeosztási modellt, amelyet háztartási nagygépek gyártási és szerelési folyamatainak összehangolására fejlesztettünk ki.

## 2. A vizsgált termelési rendszer legfontosabb jellemzői

A vizsgált diszkrét termelési rendszerben párhuzamosan működő szerelősorokon folyik a végtermékek előállítás. A késztermékeket többféle komponensből (részegységekből és alkatrészekből) szerelik össze. A végtermék egyik alapvető alkatrésze helyben készül, míg a többit külső partnertől kerül beszerzésre.

A szerelősorok és az alkatrészgyártó gépek között egy közös műveletközi tároló van kialakítva. A technológiai fázisok végrehajtási sorrendje kötött: első fázis az alkatrészgyártás a második fázis a szerelés. Ez a két technológiai fázis adja a gépek csoportosításának rendező elvét. A végtermék gyártási folyamatának elvi vázlata az 1. ábrán látható.



1. ábra. A vizsgált termelési rendszer elvi vázlata.

A termelési rendszerben  $P_p$  ( $p=1, 2, \dots, N_p$ ) végtermék gyártható. Minden egyes  $P_p$  termékbe pontosan egy darab helyben gyártott alkatrész épül be. Az alkatrészek  $E_e$  ( $e=1, 2, \dots, N_E$ ) halmaza ismert.

Az alkatrészek és a termékek technológiai specifikációi egyértelműen meghatározzák az adott műveletek végrehajtására használható technológiai berendezések halmazait és ezek engedélyezett kombinációjaként adódó technológiai útvonalakat. Az ütemezési feladatban ezeket az előírásokat szigorúan figyelembe kell venni. Ismertnek tételezhetjük fel a rendszer anyagáramlási gráfját. Nem minden alkatrész gyártható minden gépen, és nem minden termék szerelhető minden szerelősoron. Az alkatrészgyártó gépekről egy adott munkadarab-sorozat már elkészült elemei kisebb egységekben továbbíthatók. Ezáltal ugyanannak a sorozatnak különböző munkadarabjai különböző helyeken (gépeken vagy tárolóban) lehetnek. Az átfutási idők pontosabb számítása érdekében az anyagmozgatási időket is célszerű bevonni a számításba.

A különböző gépek a különböző alkatrészekben és termékeken a műveletvégzést eltérő sebességgel valósíthatják meg. Bizonyos alkatrésztípusok több gépen történő egyidejű gyártására is lehetőség van

abban az esetben, ha különböző rendelések ugyanarra a terméktípusra vonatkoznak. Vannak olyan típusok is, amelyeknél ez nem valósítható meg.

Az alkatrészgyártó gépeket különböző alkatrészek gyártása között át kell állítani. Ennek bizonyos időigénye van. Az átállítás közben a gép nem végezhet műveletet. Az átállítási idők nagysága függ a konkrét géptől és a terméktípusok sorrendjétől. Általános esetben az átállítási idők nem szimmetrikusak, vagyis két termék gyártása közötti átállítás időtartama ellentétes sorrendet feltételezve nem feltétlenül azonos. Léteznek olyan terméktípusok is, melyek között nem szükséges átállítani a gépeket. A terméktípusok tehát egyértelműen besorolhatók diszjunkt halmazokba és a halmazok közötti átállítási idők definiálhatók, míg a halmazokon belüli átállítási idők rendre nullának tekinthetők.

Az aktuális ütemezési időhorizontra vonatkozóan a szerelősorok tervezett műszakjai egyedi bontásban adóttak. Ezek az adatok az ERP rendszer által generált szerelősori termelési terv részét képezik. Ezek mellett a termelési terv tartalmazza az aktuális belső rendelések részletes adatait is: terméktípus, darabszám, szerelősor-azonosító, szerelési sorozat indítási időpontja.

Az alkatrészgyártó gépek működése szintén műszakbeosztáshoz kötött, de itt az aktív műszakok kijelölése már az ütemező döntési hatáskörébe tartozik. Ezek alapján az ütemezés szempontjából a műszakbeosztás egyrészt korlátozási feltétel a szerelősorok esetében, másrészt döntési változó az alkatrészgyártó gépek esetében.

Az ütemezési feladatban fontos tényező a műveletközi tároló. Az ütemezési modell kidolgozásakor figyelembe kell venni a műveletközi tároló korlátozó hatásait. Ide tartozik például a műveletközi tároló kapacitása, amely terméktípusonként eltérő, mert a típusoknak eltérő befoglaló méreteik vannak. Továbbá a tárolót több gép egyszerre tölti és üríti különböző intenzitással és különböző típusú alkatrészekkel. Ha a tároló megtelik, akkor blokkolja az előtte lévő alkatrészgyártó gépeket, ezért azokat le kell állítani, majd később újra kell azokat indítani. Ezt el kell kerülni, mert az újraindításnak jelentős időszükséglete van.

Az ütemezési időhorizont kezdeti szakaszának tervezésekor azt is figyelembe kell venni, hogy a gépek a még be nem fejezett korábbi feladatokkal terheltek lehetnek.

### 3. Követelmények és kritériumok

A kiszállítási tervek alapján a végszerelés termelési terve adottnak tekinthető. Ezeket a vállalatirányítási rendszer (ERP) generálja. A megoldandó feladat az, hogy a szerelési fázist megelőző alkatrészgyártási fázis folyamatait megfelelően kell ütemezni.

A bemutatott erőforrás-korlátos ütemezési feladat megoldása azt jelenti, hogy olyan ütemterv készítése a cél, amely a szigorú korlátozásokat nem sérti meg és a termelés-menedzsment által támasztott kritériumok tekintetében a lehető legjobb. A kritériumokat definiáló teljesítmény-mutatók nagyon sokfélék lehetnek. Több szempont együttes figyelembevétele nagyon nehéz feladat.

A szigorú technológiai, anyagmozgatási, tárolási és rendelkezésre állási korlátozások nagy számára és erősségére való tekintettel a túlterhelt időszakokban megengedett a határidők túllépése, így az ütemezési modellben a zéró csúszásokra vonatkozó korlátozások helyett minimalizálandó büntető-függvények jelennek meg a célfüggvények között.

A bemutatott ütemezési feladat megoldásával kapcsolatban a legfontosabb elvárások a következők:

- A szerelősorokra előírt termelési tervek megvalósíthatók legyenek csúszások nélkül, ennek érdekében lehetőleg minden szükséges munkadarab érkezzen meg időben a szerelősorokra.
- A gépek átállításának jelentős idő és költség vonzata van, ezért az alkatrészgyártás a lehető legkevesebb átállással valósuljon meg.

- A gépek leállítása és újraindítása időigényes és körülményes, ezért törekedni kell a műszakon belüli folyamatos munkavégzésre. A műveletközi tároló telítődése következtében kialakuló nemkívánatos várakozásokat a lehető legkisebb értéken kell tartani, ezért fontos a gyártás stabilitásának megőrzése minél alacsonyabb készlet szint fenntartása mellett.
- Az alkatrészgyártásban közreműködő dolgozók műszakbeosztását a gépek terhelésével összehangolva úgy kell kialakítani, hogy a beosztott dolgozók folyamatosan tudjanak dolgozni.

A bemutatott feladat jelentős mértékben eltér a hagyományos ütemezési feladatoktól. A szakirodalomban sok könyv és szakcikk foglalkozik az ütemezési modellekkel és módszerekkel. Ide tartoznak például az [1-5, 9-15] munkák. A bemutatott probléma sajátosságaihoz pontosan illeszkedő modellt nem találtunk. Megoldandó feladatunk a termelésütemezés és a műszakbeosztás egy integrált feladatként jelenik meg.

#### 4. A kidolgozott megoldási módszer

A korábban publikált saját modelljeink jellemzőit felhasználva [4], [6], [7], és kibővítve a vizsgált feladat sajátosságainak megfelelő elemekkel egy továbbfejlesztett feladattípust definiáltunk. A bemutatott ütemezési feladat a következő szimbolikus formában foglalható össze:

$$FF2, M_g, Q_{i,m}, Set_{px,py,m}, Cal_m, B_p | r_i^{O2}, d_i, A_{i,g}, Tr_{mx,my} | f_1, f_2, \dots, f_K \quad (1)$$

A szimbólumok jelentése a következő:

- $FF2$  – Az operációk (műveletek) száma kettő és sorrendjük kötött. A munkadarabok által felkeresett gépek sorozata (technológia útvonal) eltérő lehet (Flexible Flow Shop);
- $M_g$  – Funkciók szerint rendezett gépek halmazai (gépcsoportok), amelyekhez egy vagy több párhuzamosan működő gép vagy gépsor tartozik;
- $Q_{i,m}$  – A gépek/gépsorok munkáktól függő termelési sebességekkel működhetnek;
- $Set_{px,py,m}$  – Munkadarab-osztályok sorrendjétől és géptől függő átállítási időadatok;
- $Cal_m$  – A gépekhez tartozó műszakok (rendelkezésre állási időintervallumok);
- $B_p$  – Közös használatú, korlátozott méretű műveletközi alkatrésztároló, melynek tárolási kapacitása függ a tárolandó alkatrészek típusától;
- $r_i^{O2}$  – A munkák legkorábbi indítási időpontjai (indításra vonatkozó időbeli korlátozások). A szerelési folyamat esetében a termelési terv definiálja;
- $d_i$  – A munkák legkésőbbi befejezési időpontjai (teljesítési határidők);
- $A_{i,g}$  – Munkáknak definiálható a műveletvégzésre alkalmas gépek halmaza gépcsoportonkénti bontásban;
- $Tr_{mx,my}$  – Gépek közötti anyagmozgatási idők;
- $f_1, f_2, \dots, f_K$  – A kijelölt minimalizálandó célfüggvények listája (egyszerre több célfüggvény is előírható,  $K$  a célfüggvények aktuális száma).

A gyártásirányítás által támasztott igényeket az ütemezési modellben a következő célfüggvények együttes kompromisszumos optimalizálása fejezi ki:

- $f_1$ : minimális legyen a határidőt túllépő sorozatok száma,
- $f_2$ : minimális legyen a csúszások időösszege,
- $f_3$ : minimális legyen a legnagyobb határidő-túllépés (csúszás),
- $f_4$ : minimális legyen a gépátállítások száma,
- $f_5$ : minimális legyen a gépátállítások időösszege,

- $f_6$ : minimális legyen a gépek átlagos kihasználtsága: 100 – átlagos kihasználtság (maximális legyen a gépek átlagos kihasználtsága),
- $f_7$ : minimális legyen a sorozatok átlagos átfutási ideje,
- $f_8$ : minimális legyen a súlyozott műszakszám.

Az ütemezési modellben a súlyozott műszakszámot használjuk célfüggvényként az aktív műszakok költségeinek összege helyett. A műszakokat kategóriákba soroljuk, és minden kategóriához hozzárendelünk egy relatív időegyenérték mutatót. A munkanap délelőtti értéke egy egész (1,00). Ez a referenciaérték, és ehhez viszonyítva adható meg a további kategóriák értéke. Az adott ütemtervre vonatkozó súlyozott műszakszámot az egyes betervezett műszakok kategória szerinti relatív időegyenértékének összegzésével állítjuk elő. A tényleges műszakköltség a súlyozott műszakszám és a műhely gazdasági modelljéből kiolvasható fajlagos költségmutató szorzataként adódik. Ez a megközelítés azért előnyös, mert függetleníti a megoldási módszert a konkrét gyártórendszer költségmodelljétől, mivel a műszakok relatív költségmutatóit a felhasználó állíthatja be.

A megfogalmazott célok fontossága időben változhat, ezért a célfüggvények aktuális fontosságát prioritásértékek megadásával fejezheti ki a felhasználó.

A bemutatott ütemezési feladattípus döntési változóinak kombinatorikus tulajdonságai miatt az NP-nehéz feladatosztályba tartozik. Az elméleti globális optimum keresése helyett, olyan megoldási módszereket fejlesztettünk ki, amelyek nagyméretű feladatok esetében is elfogadható időn belül több kritérium együttes figyelembevételével kompromisszumosan jó ütemtervet képesek előállítani.

Az alkatrészgyártás ütemezése során a szerelősorokra előírt termelési terv belső rendelkezéseinek teljesítését kell biztosítani. A szükséges munkák elvégzésére alkalmas gépeket kell kijelölni és az elvégzendő műveletek indítási időpontjait kell megtervezni.

A feladatban a bemenő adatok definiálják az aktuális gyártási erőforrás-környezetet, a belső rendelkezéseket, a korlátozásokat, valamint a célfüggvényeket és azok prioritásait. Ezekből kiindulva kell elkészíteni az alkatrészgyártás ütemtervét. Ez az ütemterv fogja előírni, hogy

- melyik gépen,
- mikor kell az átállítást/felkészítést elkezdni,
- mennyi idő alatt kell azt elvégezni.
- milyen alkatrészfajtaból,
- mikortól kezdve,
- mennyi idő alatt,
- mennyit kell gyártani.

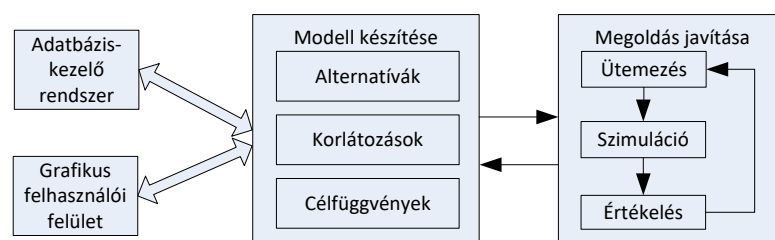
Az alkatrészgyártási igényeket a szerelősorok termelési programja adja meg. Ebből kinyerhetők az igényelt alkatrésztípusok, a szükséges darabszámok, az előírt rendeltetési helyek (szerelősorok) és a megkövetelt határidők. Egy alkatrésztípusra vonatkozó igény (munka - job) határidejét az érintett sorozat szerelősoron való indításának tervezett időpontja adja meg. Mivel a gépek szakaszosan dolgoznak, egy adott munka határideje szigorú értelemben a sorozat első munkadarabjára vonatkozik. A sorozat többi munkadarabjához saját belső határidőt rendelünk, melynek pontos értéke a szerelősor adott típusra vonatkozó ciklusidejéből és az aktuális műszakbeosztásából számítható ki.

Az igényelt alkatrészek pontos határidejének számítására egy olyan algoritmust készítettünk, amely a szerelősorok működését úgy szimulálja, mintha a szükséges összes alkatrész időben rendelkezésre állna. A koncepcióban az alkatrész helyett egy termelési vagy logisztikai egység is használható, amely az alkatrész egész számú többszörösét jelenti.

A munka fogalom (job) ebben a modellben adott számú egyforma munkadarabon előre meghatározott műveletek végrehajtását jelenti. A szerelésre vonatkozó belső rendelkezések nem bonthatók

meg, de az azonos típusra vonatkozó rendelések összevonhatók, így az ütemezés alapegysége célszerűen maga a belső rendelés. Minden egyes belső rendelésnek egy önálló munkát feleltetünk meg. A gyártórendszer és az anyagmozgató rendszer lehetővé teszi, hogy valamely gépen futó munkadarab-sorozat munkadarabjai egyesével vagy kisebb egységekben átszállításra kerüljenek a következő célállomásra. Ennek következtében adott munkának különböző munkadarabjain különböző műveletek is végrehajthatók egyidejűleg, ezzel jelentősen csökkenthető az átfutási idő. Adott munka (munkadarab-sorozat) adott gépen nem szakítható meg más munkával, de a műveletvégzés indokolt esetben szünetelhet azonos sorozat két munkadarabja között.

A megoldási módszer alapját egy iteratív keresési algoritmusba beágyazott szimulációs algoritmus adja (2. ábra).



2. ábra. A megoldási módszer elvi vázlatja.

Egy lehetséges megoldás reprezentációja alapvetően két fő döntési változócsoporthal adható meg:

1. Minden egyes munka első műveletének (az alkatrészgyártásnak) alkalmas géphez rendelése, valamint a munkák gépenkénti végrehajtási sorrendjének meghatározása.
2. Az alkatrészgyártó gépek műszakbeosztásának meghatározása a kialakuló terheléseknek megfelelően. Az engedélyezett műszakok (rögzített méretű időintervallumok) listájának adott géphez rendelése.

A felsorolt döntési változók az alkatrészgyártásra vonatkoznak, a szerelősorok termelési terve bemenő adat. A két fázis ütemtervét közös ütemtervbe kell beilleszteni mivel a termelési folyamatot együttesen határozzák meg. Ez tehát azt jelenti, hogy a teljes ütemterv egy része kötött. Az ütemezési folyamat eredményeképpen elkészül egy lehetséges termelési ütemterv.

A döntési változók értékének beállítása után lefut a teljes termelési folyamat szimulációja. Az ütemterv végrehajtását jellemző időpontok és időtartamok számítását egy szimulációs eljárás végzi el, mely figyelembe veszi a korábban ismertetett korlátozásokat és működési jellemzőket.

A szimuláció által számított időadatok és egyéb értékek felhasználásával egy értékelő algoritmus kiszámítja a vizsgált ütemtervre vonatkozó aktuális célfüggvény-értékeket.

A döntési változók értékeinek módosításától kezdve a szimuláció és a kiértékelés ismételt végrehajtásával hatékony keresési metaheurisztikára alapozva iteratív módon egyre jobb megoldás-változatok állíthatók elő.

Egy saját fejlesztésű többoperátoros és többcélú lokális kereső algoritmus (MOMOTS) foglalja keretebe a megoldási folyamatot. A MOMOTS algoritmus formális leírása a 3. ábrán látható.

A kezdeti  $s_0$  ütemterv a következő heurisztikus felépítő algoritmussal készül:

1. A munkákat sorba rendezi a belső határidők szerint nem csökkenő sorrendbe.
2. Ebben a sorrendben haladva, minden munkához az alkalmas gépek halmazából egyenletes valószínűséggel véletlenszerűen választ egy gépet.
3. Az aktuális munkát a kiválasztott gépen a már beütemezett munkák mögé helyezi el.
4. A gépek összes műszakját engedélyezi (bekapcsolja).

A keresési folyamat során az így felépített kezdeti  $s_0$  ütemtervből kiindulva megengedett módosítások ismételt végrehajtásával készül el a legjobbnak ítélt  $s^*$  ütemterv.

```

MOMOTS
{  $s_0 \leftarrow$  Kezdeti megoldás készítése;
   $s^* \leftarrow s_0$ ;
  Tabu_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 Tabu_List nem tartalmazza (  $s$  ) )
        { A Tabu_List bővítése új elemmel ( $s$ );
          if ( Tabu elemek száma > megengedett érték )
            A Tabu_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  $s^*$ ;
}

```

### 3. ábra. A MOMOTS algoritmus formális leírása.

Az iteratív javítás egy közbenső lépése során az  $s_0$  bázismegoldásából kiindulva egy bemenő paraméterben definiált számú szomszédos  $s$  ütemtervet készít az aktuálisan kiválasztott módosító  $N_c$  operátor alkalmazásával. A módosító (szomszédsági) operátorok a döntési változók értékeit módosítják. Ezek működési algoritmusai csak az alkatrészgyártó gépekre vonatkozó ütemtervrészeket változtatják meg, a szerelősorokét változatlanul hagyják.

A módosító operátorok a következők:

- $N_1$  egy véletlenszerűen kiválasztott munkát kiemel az ütemtervből és máshová illeszti be véletlenszerűen választott gép és pozíció szerint.
- $N_2$  a belső határidőt túllépő munkák közül véletlenszerűen kiemel egyet az ütemtervből és máshová illeszti be.
- $N_3$  egy véletlenszerűen választott gépen megváltoztatja a munkák végrehajtási sorrendjét, egy véletlen hosszúságú permutációciklust alkalmaz a sorrenden.
- $N_4$  egy véletlenszerűen választott gépen felcseréli két véletlenszerűen választott szomszédos munka sorrendjét.
- $N_5$  a belső határidőt túllépő munkák közül véletlenszerűen kiválaszt egyet és egyvel előre lépteti a végrehajtási sorrendben.
- $N_6$  egy véletlenszerűen választott gépen engedélyez egy véletlenszerűen választott letiltott műszakot.
- $N_7$  egy véletlenszerűen választott gépen letilt egy véletlenszerűen választott műszakot az engedélyezetttek közül.

- $N_8$  egy véletlenszerűen választott gépen engedélyez egy véletlenszerűen választott korábban letiltott műszakot és a rákövetkező korábban engedélyezett műszakot letiltja.

A módosító operátorok kiválasztását egy fontossági sorrendet kijelölő prioritáslista (*priority\_list*) és kvázi-véletlenszám generátor együttműködése határozza meg. A prioritáslista az operátorok kiválasztási valószínűségét írja le.

A lokális optimumból való kijutás elősegítése érdekében tabulistát (*Tabu\_List*) használunk. Ha a módosító operátorok által készített kiterjesztett  $s$  ütemterv szerepel a tabulistán, akkor az algoritmus azt nem értékeli ki, ellenkező esetben felkerül a tabulistára. Ha a megengedett tabuelemek száma elérte a maximális értéket, akkor a legkorábban felvett listaelem törlődik. A tabulista azt a célt szolgálja, hogy a keresési folyamat ne ragadjon bele egy lokális optimumba, hanem át tudjon rajta haladni és a közeljövőben ne térhessen vissza oda.

A szimulációt és a célfüggvények értékének kiszámítását követően, ha az aktuális ütemterv jobb, mint az adott kiterjesztés addigi legjobb ütemterve ( $s < s_k$ ), akkor megjegyzésre kerül ( $s_k \leftarrow s$ ). 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 ( $s^* \leftarrow s_k$ ).

A több összetevőből felépülő célfüggvény-rendszer kezelésére egy korábban kidolgozott matematikai modellt használtunk [6], [7], [8]. A módszer alapelve az, hogy két megoldás összehasonlításakor az egyik megoldásnak a másikkhoz viszonyított (relatív) jóságának számértéke alapján dönthető el, hogy melyik tekinthető jobb megoldásnak.

## 5. A MOMOTS algoritmus hatékonyságának vizsgálata

A bemutatott megoldási módszert alkalmaztuk jól ismert klasszikus ütemezési feladatok megoldására. A vizsgálat során a MOMOTS algoritmust a  $P|d_i|\sum T_i$  (Total Tardiness Problem on Identical Parallel Machines) típusú Shunji Tanaka által definiált benchmark feladat-instanciákon futtattuk. A feladatokat leíró bemeneti adatok és az optimális megoldást jellemző célfüggvény-értékek az alábbi weboldalon érhetők el:

<https://sites.google.com/site/shunjitanaka/pmtt>

A legnagyobb méretű Tanaka-benchmarkokon végzett futtatások eredményeit foglaljuk röviden össze. A vizsgált tesztfeladatokban 10 gép és 25 munka szerepelt. A vizsgált benchmark 125 feladat-instanciát tartalmazott ötös csoportokra osztva a műveleti idők és a határidők generálásának paramétereit alapján.

A vizsgált  $P|d_i|\sum T_i$  ütemezési feladattípus jellemzői a következők:

- Minden egyes munkához egyetlen művelet (operáció) tartozik.
- A gépek folyamatosan rendelkezésre állnak.
- Egy gép egyszerre csak egy munkán dolgozhat.
- Egy munkán egyszerre csak egy gép dolgozhat.
- A gépek párhuzamosan működhetnek és teljesen egyenértékűek.
- A munkák nem szakíthatók meg.
- A munkák egymástól függetlenek.
- A munkáknak nincs indítási időkorlátja (minden munka a 0 időponttól kezdve indítható).
- Nincs definiált átállítási idő a munkák között (a gépek átállítási ideje 0).
- Minden egyes munkának saját határideje és műveleti ideje van.
- Az ütemezés célja a határidő-túllépések (csúszások) összegének minimalizálása.



A MOMOTS algoritmus futtatásakor az alábbi keresési paramétereket használtuk:

- A legnagyobb megtehető lépésszám (leállási feltétel): 2000.
- Egy lépésben megvizsgálható szomszédos megoldások megengedett száma: 100.
- A tabulistán tárolt megoldások megengedett legnagyobb száma: 150.
- A szomszédsági operátorok prioritásai: az első öt ( $N_1-N_5$ ) operátor prioritása 1 és az utolsó három ( $N_6-N_8$ ) operátor prioritása 0. Itt most a műszakok beosztását módosító operátorok ki vannak kapcsolva, mert nincs rájuk szükség ennél a feladattípusnál.
- A célfüggvények prioritása: Ebben a speciális esetben a csúszások időösszegének ( $\sum T_i$ ) minimalizálása a cél, így az  $f_2$  célfüggvény prioritása 10 és a többi célfüggvény prioritása 0.
- Az iterációk (futtatások) száma: 10. Minden egyes feladat-instancián 10-szer futott le a MOMOTS algoritmus véletlenszerűen választott különböző kezdeti megoldásból kiindulva.

A tesztfeladatok vizsgálata során azt tapasztaltuk, hogy a MOMOTS algoritmus mind a 125 feladat-instancia esetében megtalálta az optimális megoldást. A 125 feladat-instanciából 112 példány esetében 10 futtatásból mind a 10 futtatás optimális megoldást talált (1. táblázat).

**1. táblázat.** A Tanaka-benchmark feladatok megoldásának eredményei.

<i>Az optimális megoldás megtalálásának gyakorisága 10 iterációban</i>		
<i>Az optimum elérésének száma</i>	<i>Instanciák száma</i>	<i>Instanciák aránya</i>
0	0	0
1	3	0,024
2	0	0
3	0	0
4	0	0
5	0	0
6	2	0,016
7	3	0,024
8	0	0
9	5	0,04
10	112	0,896

Az 1. táblázatból az is kiolvasható, hogy 5 instancia esetében 9-szer, 3 esetében 7-szer, 2 esetében 6-szor adódott optimális célfüggvény-érték. A 125 instanciából csupán 3 esetében fordult az elő, hogy 10 futtatásból egyszer találta meg az optimumot a kereső algoritmus.

A cikkben bemutatott kiterjesztett modell és a MOMOTS algoritmus – annak ellenére, hogy nem a tesztben szereplő  $P|d_i|\sum T_i$  feladattípusra koncentrálnak került kifejlesztésre – kiváló eredményeket ért el a Tanaka-benchmark feladatok megoldása során. Ezek az eredmények alátámasztják azt a következtetést, hogy a MOMOTS algoritmus a célul kitűzött nehezebb és összetettebb  $FF2, M_g, Q_{i,m}, Set_{px,py,m}, Cal_m, B_p|r_i^{O2}, d_i, A_{i,g}, Tr_{mx,my}|f_1, f_2, \dots, f_K$  feladatok esetében is feltételezhetően kiváló megoldásokat állít elő.

## 6. Összefoglalás és következtetések

A cikkben összefoglalt kutatómunkánk az igény szerinti rugalmas gyártás egy speciális termelésirányítási feladatának modellezésére és megoldására irányult. A probléma magába foglalja a munkák erőforrásokhoz rendelését, ütemezését és az erőforrások műszakbeosztásának meghatározását.

A modellezett erőforrás-korlátos gyártó-szerelő rendszer legfontosabb jellemzői közé tartoznak az alternatív technológiai útvonalak, a korlátozottan rendelkezésre álló gépek, a korlátozott méretű közös használatú műveletközi tároló, az eltérő műveletvégzési, átállítási és anyagmozgatási idők, valamint a szigorú belső határidős munkák. A megoldások előállítására egy továbbfejlesztett szimulációt magába foglaló többoperátoros és többcélú keresési algoritmust használtunk.

A döntéshozatali feladatot keresési feladatként fogalmaztuk meg, melyben definiáltuk a döntési változókat, az előírt korlátozásokat és a megfogalmazott célokat. Az alkalmazott szimulációs algoritmus magába zárja és elrejtja a probléma sajátos részleteit, ezáltal könnyebben kezelhetővé teszi a problémát.

A feladat bonyolultsága és a jó megoldást befolyásoló tényezők sokasága miatt a heurisztikus felépítő algoritmusok a különböző szélsőséges szituációk esetében nem alkalmazhatók. A többcélú keresési módszerek rugalmasságuknak köszönhetően képesek alkalmazkodni a változó feltételekhez és célfüggvény-rendszerekhez. Több célfüggvény egyidejű figyelembevétele nagyon fontos, mert lehetővé teszi a gyakorlati igények beépítését a modellbe. Példaként emeltük ki a súlyozott műszakszám alkalmazását a klasszikus célfüggvények mellett. A kiegészítő kritériumok a keresés kezdeti szakaszában nehezíthetik is a hatékony keresést kritikus feladatok esetében. A további kutatási terveink között szerepel annak vizsgálata is, hogy a kereső algoritmusok működése közben milyen lehetőségei vannak a célfüggvény-rendszerek szisztematikus és/vagy bizonyos határok közötti rugalmas változtatásának.

A bemutatott ütemezési modell fontos jellemzője, hogy képes korlátozott méretű műveletközi tároló figyelembevételére. Az elvégzett kísérletek azt mutatták, hogy különböző szituációkban a gépátállítások minimalizálásának kritériuma, a gépkihasználatok maximalizálásának kritériuma és a súlyozott műszakszámok minimalizálásának kritériuma ronthatja a határidők betartásának esélyét, míg más esetekben ugyanezek növelhetik is azt. Ez a hatás nagymértékben függ a rendelések tartalmától, a gépek képességeitől és a tárolóra vonatkozó előírásoktól is.

Még tovább nehezíti az ütemezési feladatok megoldását, ha a klasszikus ütemezési feladatokra jellemző gépválasztási és sorrendi döntési változók halmazába bekerülnek a műszakbeosztások előírására vonatkozó döntési változók is. Ez a kiterjesztés a modell és a megoldási módszer szempontjából további kihívásokat támasztott. A modellben a döntési változókat leíró objektumok, a keresési algoritmusban pedig a módosító (szomszédsági) operátorok újraértelmezését és kibővítését kellett megoldani.

Az elvégzett vizsgálatok eredményei azt mutatták, hogy a bemutatott megoldási módszer ipari feladatok esetében is hatékonyan alkalmazható, rugalmasan alkalmazkodik az aktuálisan előírt célfüggvényekhez és rövid időn belül szolgáltatja az eredményeket.

## 7. Köszönetnyilvánítás

A cikkben ismertetett kutató munka az EFOP-3.6.1-16-00011 jelű „Fiatalodó és Megújuló Egyetem – Innovatív Tudásváros – a Miskolci Egyetem intelligens szakosodást szolgáló intézményi fejlesztése” projekt részeként – a Széchenyi 2020 keretében – az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

## Irodalom

- [1] Allaoui, H., Artiba, A.: *Scheduling two-stage hybrid flow shop with availability constraints*, Computers and Operations Research, Vol. 33, pp. 1399-1419, 2006.  
<https://doi.org/10.1016/j.cor.2004.09.034>

- [2] 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, pp. 985-1032, 2008. <https://doi.org/10.1016/j.ejor.2006.06.060>
- [3] Brucker, P.: *Scheduling Algorithms*, E5th ed, Springer-Verlag Berlin Heidelberg, 2007, ISBN 978-3-540-69515-8.
- [4] Gharbi, A., Haouari, M.: *Optimal parallel machines scheduling with availability constraints*, Discrete Applied Mathematics, Vol. 148, pp. 63-87, 2005. <https://doi.org/10.1016/j.dam.2004.12.003>
- [5] Kaabi, J., Harrath, Y.: *E A survey of parallel machine scheduling under availability constraints*, International Journal of Computer and Information Technology, 3 (2), pp. 238-245, 2014.
- [6] Kulcsár, G.: *Ütemezési modell és heurisztikus módszerek az igény szerinti tömeggyártás finomprogramozásának támogatására*, E PhD értekezés, Miskolci Egyetem, 2007.
- [7] Kulcsár G., Erdélyi F.: *A New Approach to Solve Multi-Objective Scheduling and Rescheduling Tasks*, International Journal of Computational Intelligence Research, 3 (4), pp. 343-351, 2007. <https://doi.org/10.5019/j.ijcir.2007.115>
- [8] Kulcsár G., Kulcsárné Forrai M.: *Detailed Production Scheduling Based on Multi-Objective Search and Simulation*, Production Systems and Information Engineering, 6, pp. 41-56, 2013.
- [9] Lei, D.: *Multi-objective production scheduling: a survey*, The International Journal of Advanced Manufacturing Technology, Vol. 43, Issue 9-10, pp. 926-938, 2009. <https://doi.org/10.1007/s00170-008-1770-4>
- [10] Linn, R., Zhang, W.: *Hybrid flow shop scheduling: a survey*, Computers and Industrial Engineering, Vol. 37, No. 1-2, pp. 57-61, 1999. [https://doi.org/10.1016/S0360-8352\(99\)00023-6](https://doi.org/10.1016/S0360-8352(99)00023-6)
- [11] Ma, Y., Chu, C. B., Zuo, C. R.: *A survey of scheduling with deterministic machine availability constraints*, Computers & Industrial Engineering, Vol. 58, pp. 199-211, 2010. <https://doi.org/10.1016/j.cie.2009.04.014>
- [12] Pinedo, M. L.: *Planning and Scheduling in Manufacturing and Service*, 2nd ed., Springer-Verlag New York, 2009. [https://doi.org/10.1007/978-1-4419-0910-7\\_3](https://doi.org/10.1007/978-1-4419-0910-7_3)
- [13] Pinedo, M. L.: *Scheduling Theory, Algorithms, and Systems*, 3rd ed., Springer-Verlag New York, 2008.
- [14] Quadt, D., Kuhn, H.: *A taxonomy of flexible flow line scheduling procedures*, European Journal of Operational Research, Vol. 178, pp. 686-698, 2007. <https://doi.org/10.1016/j.ejor.2006.01.042>
- [15] Wang, W.: *Flexible flow shop scheduling: optimum, heuristics, and artificial intelligence solutions*, Expert Systems, Vol. 22, No. 2, pp. 78-85, 2005. <https://doi.org/10.1111/j.1468-0394.2005.00297.x>