

## RUGALMAS TÖBBUTAS GYÁRTÓRENDSZER ÜTEMEZÉSI FELADATAINAK MEGOLDÁSA TÖBBCÉLÚ REAKTÍV MÓDSZERREL

**Halász András Péter**

BSc hallgató, Miskolci Egyetem, Informatikai Intézet, Alkalmazott Informatikai Intézeti Tanszék  
3515 Miskolc, Miskolc-Egyetemváros, e-mail: [halasz13@iit.uni-miskolc.hu](mailto:halasz13@iit.uni-miskolc.hu)

**Kulcsár Gyula**

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

**Kulcsárné Forrai Mónika**

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

### **Absztrakt**

A cikk bemutatja a többutas diszkrét gyártórendszerek egy továbbfejlesztett ütemezési modelljét és annak többcélú reaktív megoldási módszerét. A rugalmas gyártórendszerekre jellemző alternatív technológiai útvonalak, korlátozottan rendelkezésre álló gépek, eltérő műveletvégzési és átállítási idők, valamint határidős és eltérő prioritású munkák kezelésére alkalmas ütemezési koncepció kerül bemutatásra. Az erőforrások allokálásának, az operációk időbeli végrehajtásának részletes ütemezésére egy kiterjesztett rugalmas job shop modellt definiáltunk. A cikk ismerteti a vizsgált feladattípus legfontosabb jellemzőit és a kifejlesztett többcélú reaktív megoldási módszer koncepcióját.

**Kulcsszavak:** ütemezés, rugalmas gyártórendszer, optimalizálás, reaktív megoldás

### **Abstract**

The paper presents an advanced scheduling model and a multi-objective reactive solving method of flexible job shop manufacturing system. A scheduling approach is presented that can be able to manage alternative technological routes, limited available machines, unrelated processing times, sequence dependent setup times, in addition jobs with different due dates and priorities. 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 operations. The paper describes the most important characteristics of the analysed problem and shows the main approach of the developed multi-objective reactive solving method.

**Keywords:** scheduling, flexible manufacturing system, optimization, reactive solution

### **1. Bevezetés**

Napjainkban a piaci környezet gyors változásai a termelő vállalatok és termelési hálózatok számára egyre fokozódó kihívásokat generálnak. A piaci pozíció megtartása és növelése érdekében fontossá vált a termelő rendszerek és folyamatok hatékonyabb és rugalmasabb irányítása.

Ennek egyik eszköze a rövid távú, részletes ütemezési feladatok szoftveres megoldása, amely nagymértékben hozzájárulhat a termelési folyamatok hatékonyságának, rugalmasságának és átláthatóságának javításához.

Adott termelési rendszer esetében időről-időre visszatérő megoldandó feladat a mindenkori aktuális időhorizontra vonatkozó precíz ütemterv készítése. Ennek érdekében a gyártási erőforrásokat, a szükséges munkák végrehajtási jellemzőit és korlátozásait, valamint az aktuális célokat kell figyelembe venni. Ezek alkotják az ütemezési feladat modelljét, melynek összetevői időről-időre változhatnak. A modell alapján kialakuló gyártási műveletekhez erőforrásokat kell hozzárendelni, valamint a műveletek végrehajtási sorrendjét és indítási időpontjait kell megtervezni oly módon, hogy az aktuális korlátozások betartása mellett a megfogalmazott aktuális célok megvalósuljanak.

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 végezhetőek el. A műveletek egymástól térben és időben jól elhatároltak. A műveletek végrehajtását munkahelyek valósítják meg, melyek általában nagyobb irányítási egységekbe szervezve működnek.

Az ütemezési feladatok nagyon sokfélék lehetnek, mert erőteljesen követik a vizsgált konkrét gyártási rendszer jellemzőit és sajátosságait. A korszerű gyártásirányítás olyan funkciókat foglal magába, amelyek többszörösen visszacsatolt szabályozási rendszert alkotnak. Mivel a diszkrét gyártási folyamatok nagyon különbözőek, ezért hatékony irányításuk is különböző modelleket igényel. Ennek következtében a gyártásirányítás alapját képező ütemtervek készítésének modelljei és módszerei aktív kutatások és továbbfejlesztések tárgyát képezik.

Cikkünkben bemutatunk egy továbbfejlesztett rugalmas többutas (Flexible Job Shop) ütemezési modellt és egy diszkrét esemény-vezérelt szimulációs algoritmussal kombinált többcélú reaktív ütemezési módszert.

## 2. A témakörköz kapcsolódó ütemezési modellek rövid áttekintése

A diszkrét gyártási folyamatok előidejű és valós idejű ütemezése egyaránt az erősen modellfüggő és kombinatorikus optimalizálási feladatok közé tartozik. A szakirodalomban számos cikk foglalkozik a diszkrét gyártási folyamatok esetében használható ütemezési modellekkel és módszerekkel [1]. Számos cikkben főként operációkutatási szempontok szerint vizsgálják az ütemezési feladatokat [4], [9], [10].

Az általunk vizsgált ütemezési feladat az elvégzendő operációk száma alapján a műhelyszintű ütemezési modellek (Shop Scheduling Models) közé tartozik. Ezek közös jellemzője, hogy több munka, több operáció és több gép szerepel az ütemezési feladatban. Általános esetben minden munka egy vagy több operációból állhat és 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 vizsgált feladat a technológiai tervek alapján a rugalmas műhelymodellek kategóriájába sorolható be. 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 operáció végrehajtására több gép is alkalmas. Minden operáció esetében ismertnek tekinthető az alkalmas gépek halmaza, mely egy valódi részhalmaza lehet a rendszerben lévő gépek teljes halmazának. A rugalmas modellek esetében az alap műhelymodell

kibővül operációhoz kapcsolódó gépválasztási feladattal, ugyanakkor az alapmodellben is szereplő sorrendi kérdéseket is meg kell válaszolni.

A rugalmas modellekben a gépcsoport gépei egymással párhuzamosan működnek. Ezek lehetnek teljesen egyenértékűek, részben azonosak vagy teljesen különbözőek. A különbségek a műveletvégzési sebességben, az átállítási időkben és további egyéb paraméterértékekben jelennek meg. Az általunk vizsgált ütemezési probléma a rugalmas és többutas modell típusok (Flexible Job Shop, FJS) közé sorolható be a gépek képességei alapján. A rugalmas modellek jellegzetes változatairól jó áttekintést adnak a [3] és a [11] publikációk.

A vizsgált feladat az ütemezési célok szempontjából a többcélú modellek közé tartozik. Az ütemezési modellek többsége csupán egyetlen elsődleges teljesítménymutató optimalizálására koncentrál. A többcélú optimalizálást támogató modellek sokkal kevesebb publikációban jelennek meg. Több célfüggvény együttes figyelembevételét támogató modellekről ad áttekintést például a [8] és a [12] publikáció.

Kutató-fejlesztő munkánk során a rugalmas Job Shop ütemezési modellek (FJS) hatókörét abba az irányba bővítettük, hogy különböző tevékenységekre rugalmasan átállítható gépeket (Multi-Purpose Machines), egyedi rendelkezésre állási időintervallumokat, valamint munkánként alternatív technológiai terveket vettünk figyelembe a modell kidolgozásakor. Az ipari gyakorlatban felmerülő igények kielégítése érdekében szükséges az ismert ütemezési modellek ilyen irányú továbbfejlesztése.

### 3. A vizsgált ütemezési feladat formális leírása

Az ütemezési feladatok formális leírásának eszközeként a szakirodalomban az  $\alpha|\beta|\gamma$  formalizmus [4] 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 [2].

Az általunk megfogalmazott és vizsgált feladatosztály pontos meghatározásához az alábbi  $\alpha|\beta|\gamma$  formális leírást alkalmazzuk:

$$FJ, MPM, Cal_m \mid r_i, d_i, s_{i,j,m}, A_i, tech_i \mid T_{max}, \Sigma t_{setup} \quad (1)$$

A szimbólumok jelentése a következő:

<i>FJ</i>	Az operációk (műveletek) száma és sorrendje munkánként kötött. A munkadarabok által felkeresett gépek sorozata (technológiai terve) eltérő lehet (Flexible Job Shop);
<i>MPM</i>	Adott gép különböző operációk elvégzésére is alkalmas lehet (Multi-Purpose Machines). Egy gép egyidejűleg több különböző funkcionális gépcsoporthoz is tartozhat;
<i>Cal<sub>m</sub></i>	Adott géphez tartozó rendelkezésre állási időintervallumok sorozata egyedileg definiált. A gépek csak a megadott saját intervallumaikban dolgozhatnak;
<i>r<sub>i</sub></i>	A munkákra egyedileg előírt legkorábbi indítási időpont (indításra vonatkozó időbeli korlátozás) vonatkozik. A munka első operációjának indítása nem lehet korábbi a saját indítási korlátnál;
<i>d<sub>i</sub></i>	A munkákra egyedileg előírt legkésőbbi befejezési időpont (teljesítési határidő) vonatkozik;
<i>s<sub>i,j,m</sub></i>	Az operációk megkezdése előtt a gépeket megfelelően be kell állítani (fel kell készíteni). A beállításra vagy felkészítésre (átállításra) fordított időtartam nem hanyagolható el. Az átállítási idő függ az aktuális géptől, a legutóbbi munkától és az éppen következő

	munkától;
$A_i$	Munkaként egyedileg definiálható a műveletvégzésre alkalmas gépek halmaza. Ez a felbontás tovább finomítható operációk szintjéig;
$tech_i$	Adott munka operációinak végrehajtására egyedi előírás vonatkozik. Munkaként egyedileg definiált az alkalmazható technológiai tervek halmaza. Adott munkát többféle technológiai terv szerint is el lehet végezni. Adott technológiai terv az operációkat és azok sorrendjét definiálja;
$T_{max}$	Az ütemezés elsődleges célja a legnagyobb csúszás (tardiness) minimalizálása. Adott munka csúszása kifejezi a határidő után a befejezésig eltelt időt. Ha a munka elkészül határidőre vagy az előtt, akkor a csúszás értéke nulla;
$\Sigma t_{setup}$	Az ütemezés másodlagos célja az, hogy a gépek átállítási időtartamainak összege legyen minimális.

A bemutatott ütemezési feladattípus döntési változóinak kombinatorikus tulajdonságai miatt az NP-nehéz feladatosztályba tartozik. Ez viszonylag egyszerűen belátható, mivel ez a feladattípus magába foglal speciális esetként több olyan egyszerűbb feladattípust, amelyről ismert, hogy NP-nehéz, így az általánosabb probléma is NP-nehéz.

Ezt felismerve, nem törekedtünk a globális optimum elérésére, hanem olyan megoldási módszert fejlesztettünk ki, amely nagyméretű gyakorlati feladatok esetében is rövid idő alatt több kritérium együttes figyelembevételével kompromisszumosan jó megoldást képes előállítani.

#### 4. A kidolgozott megoldási módszer

A bemutatott ütemezési feladat megoldása három részből tevődik össze. Először minden egyes munkához hozzá kell rendelni egy alkalmas technológiai tervet az alternatív lehetőségek közül. Másodszer a munkák technológiai tervének végrehajtására alkalmas gépeket kell kijelölni és harmadszor a gépeken a műveletek végrehajtási sorrendjét és a hozzájuk tartozó konkrét kezdési időpontokat kell megtervezni. Ezekre a döntési és tervezési kérdésekre adott válaszokat az ütemterv tartalmazza.

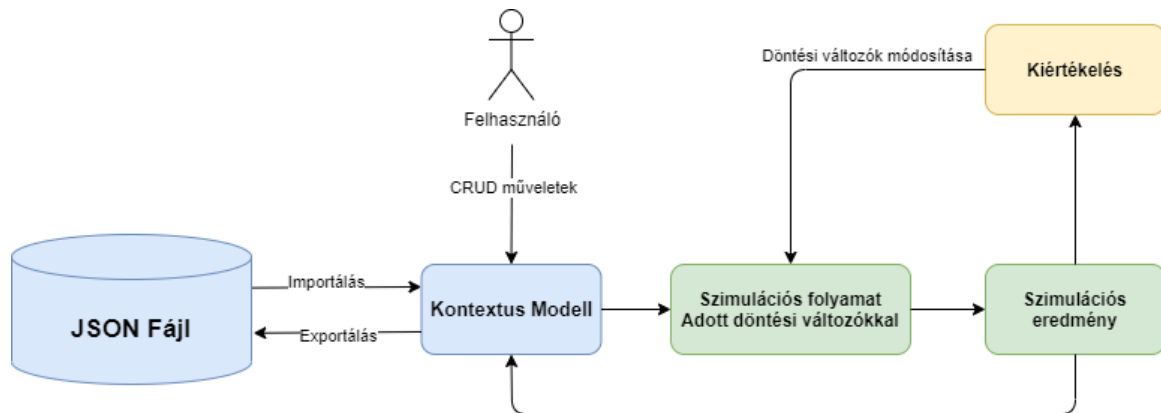
Az ütemterv fogja előírni, hogy a gyártási folyamatban

- melyik gépen,
- mikor kell az átállítást/felkészítést elkezdni,
- mennyi idő alatt kell azt elvégezni.
- melyik munka melyik operációját (műveletét),
- mikortól kezdve,
- mennyi idő alatt kell elvégezni.

Az ütemtervben megadott „menetrend” képezi a tényleges gyártási folyamatok irányításának alapjelét. A gyártórendszer tervezett működésének előfeltételeit az ütemterv alapján lehet biztosítani.

A munka fogalom (job) ebben a modellben adott munkadarabon (vagy több egyforma munkadarabon) előre meghatározott műveletek technológiai terv szerinti végrehajtását jelenti. Egy munka egyidejűleg csak egy gépen lehet, és egy gép egyidejűleg csak egy munkán dolgozhat.

Az ütemtervek automatikus elkészítésére létrehoztunk egy új ütemező szoftvert. A megoldási módszer alapját egy saját fejlesztésű diszkrét esemény-vezérelt szimulációs algoritmusba beágyazott reaktív-elven működő szabály-alapú többcélú ütemező algoritmus adja. Az aktuális bemeneti adatok felhasználásával a megoldást jelentő ütemterv automatikusan elkészül (1. ábra).



1. ábra. A megoldási módszer elvi vázlatja.

Az esemény-vezérelt szimuláció legfontosabb vezérlő paraméterei a következők:

- A szimuláció időhorizontjának kezdetét jelölő dátum és idő: Ez az érték adja a referencia-időpontot a szimuláción belüli események növekményes időpontjaihoz.
- A munkák várakozósorára vonatkozó döntési szempontok (változók) prioritásértéke: A prioritások megváltoztatásával szabályozhatja a felhasználó az esemény-vezérelt reaktív ütemezési logika működését. A szempontokhoz megadott prioritások figyelembevételével dönt a reaktív ütemező algoritmus arról, hogy adott szituációban a várakozó sorban lévő munkák közül melyikkel foglalkozzon az adott gép (erőforrás).
- A technológiai tervek választását befolyásoló szempontok (döntési változók) prioritása: Ezek a számértékek a technológiai tervek dinamikus megválasztását támogatják.

A szimuláció végrehajtásához a működést szabályozó felsorolt paramétereken túl szükség van az alapadatokra. Ide tartoznak a végrehajtásra váró munkák és az azok végrehajtásában szerepet játszó gépek (erőforrások). A kapcsolódó adatok ezeken keresztül elérhetők (terméktípusok, technológiai tervek, operációk stb.).

A szimulációba beépített reaktív szabály-alapú ütemező algoritmus működése az alábbiak szerint foglalható össze:

1. A szimulációs eljárás első lépése az adatok ellenőrzése és a teljes virtuális gyártórendszer inicializálása. Ezt a szimulációs mag úgy végzi, hogy meghívja az ezért felelős metódusokat:
  - A megoldást reprezentáló objektum adatszerkezetének kialakítása és feltöltése kezdeti értékekkel.
  - A munkák és az erőforrások dinamikus értékeinek beállítása (induló rendszerállapot beállítása).
2. Külső ciklus:
  - Létrehozza az esemény időhorizontot.
  - Kiválasztja a szabad erőforrásokat és egy listán tárolja. Szabad erőforrásnak minősül az az erőforrás, amelyik a vizsgált időpontban nem dolgozik egyik munkán sem. Minden egyes erőforrásnak van egy *InProcess* változója. Ha az erőforrás dolgozik, akkor ez a változó tárolja a végrehajtás alatt lévő munka azonosítószámát, egyébként nulla értéket tárol.
  - Eseménykezelő (belső) ciklus:
    - A legkorábbi időpontú még le nem kezelt esemény kiválasztása.

- Ha több azonos időpontú lekezelésre váró esemény is van, akkor lekezelésre kerülnek azok az események, amelyek valamely erőforráson egy munka befejezésére vonatkoznak. Majd ezt követően az eseménykezelő ciklus újabb iterációval indul.
  - Ha az előző feltétel nem eredményezett esemény-lekezelést, akkor egy újabb kiválasztás (halmazszűkítés) következik, melynek célja, hogy megtalálja azokat a munkára készen álló szabad erőforrásokat, amelyek munkákat fogadó várakozósorában vannak várakozó munkák. Ezen erőforrásokat hozzáadja egy tárolóhoz minden iterációban, mivel a korábbi eseményekre reagáló kezdésre alkalmas erőforrások továbbra is elérhetőek maradnak. Erre a lépésre azért van szükség, mert ha a munka számára kiosztott legkorábbi megengedett kezdés idő követelménye nem teljesült, akkor az kikerül a kiválasztási versenyből. Viszont a következő iteráció egy újabb időpillanatot fog eredményezni, amikor újra ellenőrizni kell, hogy teljesült-e az indítási feltétel.
    - Az adott erőforrás vizsgálatára egy eljárás indul, amely megállapítja, hogy a várakozó sorában a legkorábbi indíthatósági időpont milyen értékkel bír. Ha az nagyobb a jelenlegi esemény időértékétől, akkor veszi a soron következő erőforrást. Ha pedig ez a feltétel nem teljesül, akkor a nyertes erőforrás tekintetében lekezeli a kezdési eseményt. Ezt követően kilép az iterációból.
  - Abban az esetben, ha továbbra sem került lekezelésre esemény, akkor megkeresi az összes erőforrás legutolsó befejezési időpontját. Megvizsgálja, hogy ez megegyezik-e a jelenlegi esemény időpontjával. Mivel az eseménykezelés nem valósult meg, ennek az egyetlen oka az lehet, hogy a várakozósorokban lévő munkák legkorábbi kezdési időpontja még nem múlt el és arra várakoznak.
    - Ha a feltétel teljesül, akkor megkeresi a legkorábban kezdhető munkát az erőforrások várakozósoraiból. Majd az első nyertes munka jelenlegi erőforrása tekintetében lekezeli a kezdési eseményt és kilép az iterációból.
  - Ha egyik elágazás sem vezetett eredményre, akkor a belső ciklusmag megismétlődik, és az előző minimum operáció-végzési értéket eldobva, a következő minimum adja majd meg a következő iteráció esemény időértékét. Ez a belső mag addig dolgozik, amíg nem kezel le minden végzési eseményt, vagy nem indít el egy kezdési eseményt.
3. A szimuláció külső ciklusa addig ismétlődik, amíg van éppen dolgozó erőforrás vagy van aktívan várakozó szabad erőforrás.
4. Az algoritmus az események lekezelésekor meghozott döntések naplózott eredményéből összeállítja a teljes ütemtervet (megoldást).

A megoldási módszer kiemelkedően fontos részei az eseményt kezelő metódusok. Az események típusa alapján megkülönböztethetők az operációk elkezdését és befejezését jelző események. A bemutatott esemény-kiválasztó koncepcióban a befejezés eseménye nagyobb precedenciával bír. Ennek akkor van jelentősége, ha azonos időpontban több eseményt kell lekezelni.

Egy adott operáció befejezési eseményét lekezelő metódus megkeresi az éppen végző munkapéldányt, annak az operációs indexét egyel inkrementálja, majd a munkát elhelyezi a következő operációjának alkalmas erőforrásainak várakozósoraiba. Végül az erőforrást felszabadítja. A vezérlő reaktív ütemezési stratégia lényege az, hogy egy adott munka virtuálisan minden olyan gép várakozási sorába bekerül, amelyik alkalmas az adott soron következő operációjának elvégzésére.

Egy adott gép (erőforrás) operációkezdési eseményét lekezelő metódus kiválasztja indításra a döntési szempontok (változók) prioritásai és a legkorábbi kezdési időpontok alapján a várakozó munkák közül a legkedvezőbbet. Ezt a kiválasztott munkát leveszi az összes többi alkalmas erőforrás virtuális várakozósoráról. A terhelendő erőforrást lefoglalja, majd kiszámolja a kezdéshez és az elvégzendő operációhoz tartozó időértékeket. Ezek számítása során az algoritmus figyelembe veszi az adott operáció megszakíthatóságára vonatkozó technológiai előírást, az erőforrás rendelkezésre állási időintervallumait, az utolsó gyártott típust, az átállási időt és a műveleti időt. A megkapott időértékeket naplózza. Végül a kiszámított befejezési időponttal felülírja az erőforrás felszabadulását jelző változót, valamint az erőforrás megjegyzi a kezelt terméktípust legutóbbi terméktípusként.

A bemutatott reaktív ütemezési stratégia kiemelkedően fontos összetevője az a kiválasztási logika, ami a felszabaduló erőforrás következőként indítandó munkájának kiválasztását végzi. A kidolgozott módszerünk abban különbözik a szakirodalomban publikált módszerektől, hogy nem egyetlen szabály, vagy számított érték alapján valósul meg a kiválasztás, hanem több különböző prioritású szempont együttes figyelembevételével oldjuk meg a kiválasztást. Ennek lényege az, hogy döntési szempontok (változók) alapján prioritásokat figyelembe véve a relatív összehasonlítás módszerét alkalmazva dönti el az algoritmus, hogy két jelölt közül melyik a kedvezőbb.

A relatív összehasonlító módszer alapelve az, hogy két különböző jelölt elem összehasonlításakor az egyik elemnek a másikhoz viszonyított (relatív) jóságának számértéke alapján döntjük el, hogy melyiket tekintjük kedvezőbbnek [5], [6], [7]. Az összehasonlító modell formális definícióját a (2), (3) és (4) képletek írják le.

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

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

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

A matematikai modellben alkalmazott szimbólumok jelentése a következő:

- $S$ : a jelöltek halmaza.
- $f_k$ : a  $k$ . szempontoz tartozó nemnegatív érték, ahol a kisebb érték a kedvezőbb.
- $K$ : a szempontok száma.
- $D(a, b)$ : a változás mértékét kifejező függvény, ahol  $a$  és  $b$  nemnegatív valós számok.
- $s_x, s_y$ : két különböző jelölt elem.
- $w_k$ : az  $f_k$  szempont fontosságát kifejező nem negatív érték (prioritás).
- $F(s_x, s_y)$ : az  $s_y$  jelölt elem  $s_x$  jelölt elemhez viszonyított relatív jósága.

Az  $F(s_x, s_y)$  kétváltozós függvény előjeles értékének felhasználásával az alapvető relációs operátorok értelmezése kiterjeszthető a többcélú kiválasztási feladat  $S$ -beli  $s_x$  és  $s_y$  jelölt elemekre 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. Az (5) definíció azt fejezi ki, hogy az  $s_y$  jelölt elem és az  $s_x$  jelölt elem közötti relációt úgy kell értelmezni, mint ahogy az  $F(s_x, s_y)$  érték viszonyul a nullához. Például a  $<$  relációjel azt az eldöntendő kérdést fejezi ki, hogy a baloldali operandus kedvezőbb elem-e, mint a jobb oldali operandus. Az  $s_y < s_x$  kifejezés logikai értéke akkor igaz, ha az  $F(s_x, s_y)$  érték kisebb mint nulla.

A kiválasztható elemek halmazán értelmezett relációs operátorok felhasználásával a különböző jelöltek páronkénti összehasonlítása és értékelése egyszerűen elvégezhető. Ez az összehasonlítási módszer többféle kiválasztási stratégiában jól felhasználható az aktuális szempontoknak legjobban megfelelő elem megtalálására.

## 5. Összefoglalás

Cikkünkben bemutattunk egy továbbfejlesztett reaktív ütemezési módszert, az igény szerinti rugalmas gyártás hatékonyságának növelésére. A vizsgált probléma magába foglalja a munkák technológiai tervének kiválasztását, a műveletek erőforrásokhoz rendelését és ütemezését. A bemutatott modell magába foglalja az erőforrás-korlátos gyártási környezet legfontosabb jellemzőit (pl. az időben és számban korlátozottan rendelkezésre álló erőforrásokat, a sorrendfüggő átállítási időket stb.), valamint a munkák végrehajtási jellemzőit (pl. alternatív technológiai terveket) és korlátozásait (pl. indítási időkorlátokat, határidőket stb.). Az ütemtervek előállítására egy továbbfejlesztett szimulációba beépített többcélú reaktív ütemezési logikát használtunk.

A bemutatott szimuláció alkalmazásával az ütemezési feladatot reaktív szabály-alapú kiválasztási feladatok kombinációjaként fogalmaztuk meg. A dinamikusan változó kiválasztási részfeladatok megoldására egy többszempontú relatív összehasonlításon alapuló matematikai modellt használtunk. Az esemény-vezérelt szimulációs algoritmus sokféle gyártórendszer esetében használható. A kiválasztási szempontok és azok prioritásának változtatásával könnyen adaptálható a módszer tetszőleges diszkrét gyártórendszer reaktív irányítására.

A reaktív ütemezési feladatok bonyolultsága miatt az egyszerű szabály-alapú algoritmusok nem képesek kellő rugalmassággal kezelni a kialakuló szituációkat.

A többcélú keresési módszerek rugalmasak ugyan, de a viszonylag nagyobb futási időigényük miatt nem alkalmasak a dinamikus rendszerek valós idejű on-line irányítására. Az általunk javasolt megközelítés egyesíti a többcélú keresés hatékonyságát és rugalmasságát, valamint a szabály-alapú ütemezés gyorsaságát. A módszerünkben alkalmazott kiválasztási szempontok és azok prioritása a felhasználó által beállítható, ennek köszönhetően a módszer a változó feltételekhez és célokhoz igazítható.

A bemutatott ütemezési módszer előnyei közé tartozik az, hogy automatikusan korrigálja a gyártórendszerben esetleg fellépő hibákat, kezeli a váratlan eseményeket és bizonytalanságokat, növeli a rendszer robusztusságát és segíti a digitális ikerpár koncepció bevezetését. A módszerünk használható off-line és on-line irányítási környezetben egyaránt. Off-line irányítási környezetben a reaktív módszerünk előre lefuttatható a szimuláció segítségével és az eredmény egy prediktív megoldásként kiadható végrehajtásra. On-line irányítási környezetben a döntési logika a valós idejű visszacsatolt események és állapotok alapján működhet.

## 6. 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] 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 2008, 187:985-1032. <https://doi.org/10.1016/j.ejor.2006.06.060>
- [2] Brucker, P.: *Scheduling algorithms*, 5th ed, Springer, 2007, p. 371.
- [3] 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 2012, 49(1):263-279. <https://doi.org/10.1007/s00170-009-2388-x>
- [4] 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 1979, 5:287–326. [https://doi.org/10.1016/S0167-5060\(08\)70356-X](https://doi.org/10.1016/S0167-5060(08)70356-X)
- [5] Kulcsár, Gy.: *Ütemezési modell és heurisztikus módszerek az igény szerinti tömeggyártás finomprogramozásának támogatására*, Doktori (PhD) értekezés, Miskolci Egyetem, Miskolc-Egyetemváros, 2007, p. 146.
- [6] Kulcsár, Gy., Erdélyi, F.: *A new approach to solve multi-objective scheduling and rescheduling tasks*, International Journal of Computational Intelligence Research 2007, 3(4):343-351. <https://doi.org/10.5019/j.ijcir.2007.115>
- [7] 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 2013, 6:41-56.
- [8] Loukil, T., Teghem, J., Tuytens, D.: *Solving multi-objective production scheduling problems using metaheuristics*, European Journal of Operational Research 2005, 161:42-61. <https://doi.org/10.1016/j.ejor.2003.08.029>
- [9] Pinedo, M. L.: *Scheduling: Theory, algorithms, and systems*, 4th ed, Springer, 2012, p. 696.
- [10] Pinedo, M. L.: *Planning and scheduling in manufacturing and service*, 2th ed, Springer, 2009, p. 537.
- [11] Quadt, D., Kuhn, H.: *A taxonomy of flexible flow line scheduling procedures*, European Journal of Operational Research 2007, 178:686-698. <https://doi.org/10.1016/j.ejor.2006.01.042>
- [12] 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.