

**Pécsi Tudományegyetem  
Közgazdaságtudományi Kar  
Gazdálkodástani Doktori Iskola**

**Egy új harmóniakereső metaheurisztika erőforrás-korlátos  
több-megvalósítási módú projektek ütemezésére**

**Doktori értekezés tézisei**

**Szendrői Etelka**

Témavezetők: Dr. Csébfalvi Anikó CSc, Phd  
Dr. Csébfalvi György CSc

**Pécs, 2011**

## Bevezetés

A disszertációban egy új harmóniakereső metaheurisztikát mutatunk be, amely a több-megvalósítási módú erőforrás-korlátos ütemezések halmazán a projekt időtartamát minimalizálja.

A mai projekt szemléletű világban a projektütemezési problémák egyre inkább előtérbe kerülnek, egyre fontosabbá válnak. Ezen belül is a több-megvalósítási módú projektek felé egyre nagyobb figyelem irányul a kutatók részéről, hiszen a tevékenységek többféle módon történő végrehajtása igen közel áll a mindennapi gyakorlathoz. A projektütemezés során különböző célokat valósítunk meg, a tevékenységek között fennálló elsőbbségi feltételek és az esetlegesen fellépő erőforráskorlátok figyelembevételével. Leggyakoribb cél a projekt hosszának minimalizálása, de emellett lehet cél az erőforrás felhasználás kiegyenlítése, vagy egyéb más kritérium.

A gazdasági és technikai fejlődés következtében a feladatok mérete és bonyolultsága folyamatosan nő, az erőforrások pedig korlátozott mennyiségben állnak rendelkezésre, ezért egyre nagyobb az igény arra, hogy a kutatók gyors, hatékony egzakt és heurisztikus megoldásokat dolgozzanak ki.

A gyakorlatban előforduló projektek nagyméretűek, sok és sokféle erőforrást igényelnek. Ugyanakkor számos esetben lehetőség van arra, hogy a projekt egyes tevékenységeit más-más módon hajtsák végre, s ezzel erőforrást, pénzt vagy időt takarítsanak meg. Fontos szempont a menedzserek számára, hogy az erőforrásokat megszakítások nélkül használják fel a projekt végrehajtása során, ne legyenek tétlen, várakozó, majd újrainduló erőforrások. A több-megvalósítási módú projektek esetében a cél úgy megválasztani a tevékenységek megvalósítási módjait, hogy az erőforráskorlátok betartása mellett a projekt időtartama legyen minimális. Másodlagos cél lehet az erőforrások megszakításmentes folyamatos felhasználása, azaz az erőforrásprofil kisimítása.

## A dolgozat felépítése

A bevezetést követően, a 2. fejezetben a projektütemezési modellek komponenseit, csoportosítási szempontjait, az egyes csoportok jellemző tulajdonságait részletezzük:

- Tevékenységek
- Erőforrások
- A projektek ábrázolása
- Ütemezés, kritikus út
- A projektütemezés céljai
- A megoldó algoritmusok összehasonlíthatósága – a PSPLIB tesztkönyvtár

A 3. fejezetben a szakirodalom áttekintésére kerül sor. Elsősorban a több-megvalósítási módú erőforrás-korlátos projektütemezési problémák megoldására készült tanulmányokat, eljárásokat vesszük sorra.

Az egzakt eljárások kisméretű feladatok esetében adnak optimális megoldást. Mivel a problémánk NP-nehéz természetű, elsősorban heurisztikus algoritmusok alkalmazása biztosítja, hogy elfogadható időn belül jó minőségű megoldást kapjunk. Az egzakt megoldó eljárások áttekintése azért fontos, mert gyakran alkalmazzák elemeiket a heurisztikus eljárásokban.

Dolgozatunk a metaheurisztikus megoldásokra helyezi a hangsúlyt, a megoldandó probléma NP-nehéz természete miatt. A szakirodalom legjelentősebb metaheurisztikáinak rövid ismertetése mellett, részletesen bemutatjuk az egyik leghatékonyabbnak bizonyuló metaheurisztikát, a harmóniakereső algoritmust.

A 4. fejezetben részletesen ismertetejük az új harmóniakereső hibrid algoritmusunkat, amely a több-megvalósítási módú erőforrás-korlátos projektek lehetséges ütemezéseinek halmazán a projekt teljes időtartamát minimalizálja.

A bemutatásra kerülő új metaheurisztika a Csébfalvi által kifejlesztett Sounds of Silence egy megvalósítási módú erőforrás-korlátos projektek minimális időtartamú ütemezésének meghatározására kifejlesztett harmóniakereső metaheurisztika továbbfejlesztése.

A továbbfejlesztett algoritmus lényege, hogy sikeresen bővítettük az algoritmust a több-megvalósítási módú projektek kezelésére. Lényeges újítása a modellnek az előoptimalizáló eljárás, amely mind az algoritmus gyorsaságában mind a megoldás minőségében jelentős javulást eredményezett. További minőségjavító komponensekkel is bővítettük a modellt és az algoritmust, melyek részletesen ismertetésre kerülnek a fejezetben.

Az új, kifejlesztett algoritmus hatékonyságának bizonyítására részletes számítási eredményeket közlünk a PSPLIB tesztkönyvtár projektjein végrehajtott futtatások alapján.

Végül az 5. fejezetben az algoritmus további lényeges fejlesztési eredményéről számolunk be. A továbbfejlesztéssel felkészítettük az algoritmusunkat másodlagos kritérium szerinti optimalizálásra is. A másodlagos optimalizálási szempont rugalmasan változhat, azonban mi az erőforrás felhasználási hisztogram kisimítását fogalmazzuk meg másodlagos optimalizálási kritériumként. Ennek a továbbfejlesztésnek a futási eredményeit is ismertetjük az 5. fejezetben.

## **Kutatási célkitűzések, eredmények**

Dolgozatunkban a több-megvalósítási módú tevékenységek erőforrás-korlátos ütemezésének halmazán a projekt időtartamát minimalizáló, valamint az erőforrásprofil kisimító hatékony ütemező eljárást keresünk.

A kutatás során az alábbi feltételezések mellett a következő célokat tűztük ki:

- 1. Az adott problémára hatékony algoritmust kell létrehozni. A megoldást a heurisztikák között kell keresnünk.*

Az irodalmat tanulmányozva a több-megvalósítási módú projektek esetében a kutatók egybehangzó állítása az volt, hogy egzakt megoldások ezen a területen csak kisméretű problémák megoldása esetén működnek. A közölt eredmények beszámolnak sikeres egzakt algoritmusokról, de azt is megállapítják, hogy még a legsikeresebb egzakt algoritmusok is csődöt mondanak, nem találnak megoldást akkor, ha a tevékenységek száma meghaladja a 20 tevékenységet. Ennek tükrében az *egyetlen lehetséges alternatíva* csakis a *heurisztikus megoldás* lehet.

2. *A heurisztikák közül a metaheurisztikus algoritmusokat érdemes vizsgálni, a legmegfelelőbb és leghatékonyabb eljárás megtalálásához.*

A terület szakirodalmának áttekintése alapján megállapítható, hogy a kutatók szerint különböző típusú feladatok, célfüggvények esetén más-más, a területhez a legjobban illeszkedő heurisztikát célszerű alkalmazni. A metaheurisztikus eljárások egy keretet biztosítanak a feladatok megoldására, amely keretrendszerben, az egyes fázisokban bizonyos részfeladatok megoldását már korábban bevált, vagy új hatékony eljárásokkal, akár egzakt algoritmusokkal, lehet kombinálni.

A metaheurisztikák hatékonyságát számos kutató bizonyítja a szakirodalomban közölt futási eredményekkel. Ezeket a számítási eredményeket a jól ismert PSPLIB tesztalacson végezték el. A szakirodalom áttanulmányozása megerősítette azt a meggyőződésünket, hogy a több-megvalósítási módú erőforrás-korlátos projektütemezési probléma megoldására metaheurisztikát érdemes alkalmaznunk, leginkább a metaheurisztikus algoritmusok nyújtják azt a rugalmasságot és hatékonyságot, amelyet a feladat nehézségi foka megkövetel.

3. *A probléma jellegéhez, tulajdonságához legjobban illeszkedő metaheurisztikát kell találni. A metaheurisztika alkalmazásával cél egy rugalmas és lehetőleg a leghatékonyabb algoritmus megtalálása.*

A témában közzé tett metaheurisztikákat áttekintve úgy találtuk, hogy különböző területeken különböző típusú metaheurisztikák alkalmazása ad hatékonyabb eredményt. Az irodalom tanulmányozása során rátaláltunk, egy az utóbbi években Lee és Geem által kifejlesztett, igen hatékonynak bizonyuló metaheurisztikára, a harmóniakereső (Harmony Search) metaheurisztikára.

A Harmony Search algoritmus egy zenei inspirációjú metaheurisztikus optimalizáló eljárás, melynek célja a tökéletes harmóniára való törekvés zenei eljárásának felhasználásával az optimum, vagyis a tökéletes harmónia megtalálása. A zenei harmónia megfelel az optimalizálás megoldásvektorának, a zenei improvizációk pedig az optimalizáló technikák keresési sémáival analógok. A zenei improvizáció során minden zenész öltözködészerűen játszik valamilyen dallamot a saját hangszerén, és az eredmény, az együttes hangzás, a „harmónia-vektor” jó esetben valamilyen szép, a fülnek is tetsző, harmonikus dallam lesz.

Nagyon fontos, hogy a feladat természete, az arra épülő matematikai modell és a megoldási módszer összhangban legyen. A HS algoritmus tulajdonságait megvizsgálva megállapítottuk, hogy a zenei analógiák a vizsgált problémákra is jól illeszthetők. A HS algoritmus nem gradiens alapú algoritmus, így kevesebb matematikai igénye van, hatékonyabb, mint a genetikai algoritmusok, mert nem használ kódolást és dekódolást, nem igényel kezdeti értékeket a döntési változókhoz, szimultán kezeli a vektorokat, tág keresési teret használ, globális optimumot ad. Az előbb felsorolt jellemzők alapján úgy véltük, hogy problémánk tulajdonságaihoz a leginkább illeszkedő metaheurisztika a harmóniakereső metaheurisztika.

A korábban említett tulajdonságok, tények után, eldöntöttük, hogy harmóniakereső megoldást alkalmazunk és megkíséreltünk hatékony és gyors algoritmust adni a problémánkra.

Az elvégzett munkánk legfontosabb eredményeit a következő tézisekben fogalmaztuk meg:

**1. Tézis A Sounds of Silence algoritmust továbbfejleszthető oly módon, hogy alkalmassá váljon a több-megvalósítási módú erőforrás-korlátos projektütemezési probléma megoldására**

A több-megvalósítási módú erőforrás-korlátos projektütemezési probléma megoldására heurisztikát kell alkalmazni. Az egzakt megoldások csak igen kisméretű feladatok esetében működnek. A gyakorlatban azonban nagyméretű feladatokat kell megoldani, ebben az esetben az egzakt megoldások már nem adnak megfelelő eredményt elfogadható időn belül. A problémát nehezíti az is, hogy a vizsgált modell megújuló és nem-megújuló erőforrásokat is tartalmaz.

A szakirodalom részletes tanulmányozása alapján arra a következtetésre jutottunk, hogy egy ismert harmóniakereső heurisztikát a Sounds of Silence algoritmust érdemes továbbfejleszteni. A Sounds of Silence algoritmus az erőforrás-korlátos projektek időtartamának minimalizálását oldotta meg. Az algoritmus működési elve lehetővé tette, hogy úgy bővítsük, hogy alkalmassá váljék a több-megvalósítási módú erőforrás-korlátos projektütemezési feladatok megoldására is.

A zene világában az erőforrásprofilok egy „többszólamú dallamot” alkotnak. Tegyük fel, hogy minden szólamban csak a „magas hangok” hallhatóak, így a projektütemezés nyelvén megfogalmazott probléma a következő lesz: Keressük a legrövidebb „Sounds of Silence (Csend hangjai)” melódiát az improvizáció során, vagyis a legrövidebb csendet! Természetesen a zenei analógiában szereplő „magas hang” a projektütemezésben az erőforráskorlát átlépését (túlmunka) jelenti. A zenei analógia nyelvén fogalmazva a több megvalósítási módú projektütemezési probléma a következőképpen írható le. Az MRCPS esetében minden  $i$  zenész (tevékenység),  $i \in \{1, 2, \dots, N\}$  jellemezhető egy diszjunktív többszólamú hanghalmazzal, és a nem-megújuló erőforrást egy adott energiatípusból származó, az előadáshoz szükséges „energiaként” interpretálva, a nem-megújuló erőforrást egy hang megszólaltatásához szükséges „fizikai energiának” tekinthetjük, vagy az előadás minőségének ellenőrzéséhez szükséges „spirituális” energiaként foghatjuk fel. Természetes feltételezés az is, hogy mindegyik energia fajtából a zenekar „teljes” energiája korlátozott és a teljes energiát az előadás során felhasználják.

A megoldandó matematikai modell a következő formulákkal írható le:

$$\min [X_{N+1}] = X_{N+1}^* \quad (1)$$

$$X_{N+1} \leq \bar{T} + 1 \quad (2)$$

$$\sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} X_{ims} = 1, \quad i = 1, 2, \dots, N \quad (3)$$

$$\sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} s * X_{ims} \leq X_{N+1}, \quad i \rightarrow N + 1 \in PS \quad (4)$$

$$\sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} (s + D_{im}) * X_{ims} \leq \sum_{m=1}^M \sum_{s=\underline{X}_{jm}}^{\bar{X}_{jm}} s * X_{jms}, i \rightarrow j \in PS \quad (5)$$

$$\sum_{i=1}^N \sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} W_{imst} * R_{imr} * X_{ims} \leq R_r, t = 1, 2, \dots, T, r = 1, 2, \dots, R \quad (6)$$

$$W_{imst} = \begin{cases} 1 & s \leq t \wedge t < s + D_{im} \\ ha & \\ 0 & t < s \vee t \geq s + D_{im} \end{cases} \quad (7)$$

$$\sum_{i=1}^N \sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} C_{imc} * X_{ims} \leq C_c, c = 1, 2, \dots, C \quad (8)$$

$$X_i = \sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} s * X_{ims}, i = 1, 2, \dots, N \quad (9)$$

$$M_i = \sum_{m=1}^M \sum_{s=\underline{X}_{im}}^{\bar{X}_{im}} m * X_{ims}, i = 1, 2, \dots, N \quad (10)$$

$$\mathbf{X} = \{X_1, X_2, \dots, X_N\} \quad (11)$$

$$\mathbf{M} = \{M_1, M_2, \dots, M_N\} \quad (12)$$

$$X_{ims} \in \{0, 1\}, i = 1, 2, \dots, N, m = 1, 2, \dots, M, s = \underline{X}_{im}, \dots, \bar{X}_{im} \quad (13)$$

## 2. Tézis: Előoptimalizálási eljárás alkalmazásával egy optimális induló megoldást kapunk, amely lényegesen meggyorsítja, hatékonyabbá teszi az algoritmust.

Az eredeti Sounds of Silence algoritmusban az induló megoldás véletlenszerűen választott megvalósítási módokra alapozott ütemezésből indult ki. Az előoptimalizálással lényegesen csökkenthető a keresési tér, és általa az algoritmus sebessége nagymértékben megnő. Az előoptimalizálás során kizárjuk a keresési térből azokat a megvalósítási mód lehetőségeket, amelyek megsértik a nem-megújuló erőforráskorlátokat. Az előoptimalizálás során egy olyan optimális induló megoldást kapunk, ahol a megvalósítási módok közül azok lesznek hozzárendelve az egyes tevékenységekhez, amelyekre a nem-megújuló erőforráskorlát nem sérül.

Az eredeti változatban a repertoár feltöltési fázisban a tevékenységek kezdetének értékét normális eloszlásból generáljuk. A módosított változatban a relaxált megoldásból véletlenszerű perturbációval egy előoptimalizált repertoárt hozunk létre.

**3. Tézis: Kihhasználva a több-megvalósítási módból adódó lehetőséget, az optimalizálás során nyert ütemezést tovább javítjuk az ismert Forward-Backward Improvement eljárással, hogy tovább csökkentsük a projekt időtartamát.**

Az optimalizálás során kapott erőforrás-korlátos ütemezés, az egyes tevékenységekhez rendelt módok listájából és a tevékenységek kezdési időpontjainak listájából áll. A Forward-Backward Improvement eljárás alkalmazásával, lehetőség nyílik arra, hogy tovább csökkentsük a projekt időtartamát, anélkül, hogy megsértenénk az erőforráskorlátokat. Az eljárás alkalmazása során a korábbi lépésekben a tevékenységhez rendelt megvalósítási módok változhatnak.

**4. Tézis: A „head-tail” eljárás alkalmas arra, hogy tovább javítsuk a korábbi lépésben kapott ütemezést**

A „head-tail” eljárás alkalmazása során azt a tényt használjuk ki, hogy a nem-megújuló erőforrások esetében átstrukturálódhat a fix nem-megújuló erőforrás felhasználás, ha a projekt elején, illetve a végén lévő tevékenységeket mozgatjuk. A „head-tail” lokális kereső algoritmus csak egyetlen paraméterrel rendelkezik, a „head-tail” méretével. Ha ez a paraméter egyenlő eggyel, akkor csak a projekt kezdő és befejező elemeit használja fel az újraallokálási folyamatban. A „head-tail” algoritmus segítségével újraoszthatjuk a nem-megújuló erőforrásokat a projekt eleje és vége között, amennyiben az újraosztás eredményeképpen, a projekt időtartama rövidebb lesz. A „head-tail” algoritmus során egy kisméretű MILP problémát oldunk meg, amely a mai solver-ek segítségével rendkívül gyorsan eredményt ad. Minél nagyobb a „head-tail” mérete, annál nagyobb az esély arra, hogy a projekt időszükséglete csökkenthető, az erőforrások újraallokálásával. Ugyanakkor óvatosan kell bánni a „head-tail” méretének növelésével, mert a számítási költségek drámaian megnövekedhetnek a „head-tail” méretének függvényében. Így tehát egyensúlyt kell teremtenünk a költségek és a minőség között.

**Számítási eredmények**

Az algoritmus hatékonyságának és magas minőségének igazolására a közismert PSPLIB tesztalmaz elemein végeztünk futtatásokat. A J30MM tesztalmaz minden elemére elvégeztük a számításokat. Miért erre a tesztalmazra esett a választás? Az ok rendkívül egyszerű, mert jelenleg ez a legtöbb tevékenységet tartalmazó, nem-megújuló (elfogyó) és megújuló erőforrásokat is tartalmazó több-megvalósítási módú tevékenységeket magában foglaló projekthalmaz. Ez idő szerint a J30MM képezi a nehéz és nagy feladatokat. (Természetesen ezek a feladatok a valóságban létező nagy problémáktól messze vannak.)

Ebben a halmazban a projektpéldányok 30 valós tevékenységet tartalmaznak, az egyes tevékenységek két megújuló és két nem-megújuló erőforrást igényelnek. Minden valós tevékenység három megvalósítási móddal rendelkezik. A J30 tesztalmaz 640 esetet tartalmaz, amely 64 eset 10 ismétlése. Néhány esetnek nincs lehetséges megoldása, ezeket kizártuk a vizsgálódásból. Ebben a halmazban nem ismert az összes optimális megoldás, így először ezeket az egzakt megoldásokat állítottuk elő egy korszerű MILP solverrel. (CPLEX 8.1). A megoldásokra szánt időt 900 sec időkorlátban állapítottuk meg. Az adott időkorlátban belül 382 esetben adott optimális megoldást a CPLEX, amely jól jelzi a több-megvalósítási módú projektütemezési feladatok nehézségét, NP-hard jellegét.

Az alábbi két táblázatban a J30MM halmaz futási eredményeit mutatjuk be. Az első táblázat a CPLEX által adott egzakt megoldások minőségi paramétereit mutatja, amelyben szembetűnő a megoldási idők magas szórása.

Megoldási idő CPLEX 8.1 (sec)				
Átlag	Szórás	Minimális idő	Maximális idő	Megoldott esetek
25.44	98.82	0.02	665.00	382

A második táblázat a hibrid algoritmusunk futási eredményeinek átlagát mutatja, mégpedig az algoritmus tudásának bővülésével kapcsolatosan. Standard repertoár alkalmazásával is a futási idők átlaga egy nagyságrenddel kisebb, mint az egzakt megoldások átlagos ideje. A megoldás minőségét az egzakt optimális megoldástól való eltérés %-ban adjuk meg. Jól látható, hogy az előoptimalizálás minőségileg és a megoldási idő szempontjából is lényegesen javított az algoritmus képességein. A további bővítése az algoritmusnak a head-tail eljárással, további javulást hozott a megoldás minőségében, bár egy keveset rontott a megoldási idő értékén. A táblázat utolsó sorából igazolódni látszik az az állításunk, hogy a head-tail méretével arányosan nő a számítás ideje is. A számok azt mutatják, hogy a head-tail méretét 2-re növelve a megoldás minősége nagyot javult, mindössze 0,5%-al tért el az optimális megoldás értékétől, azonban ez a minőségi javulás a számítási idő rovására történt.

„Soundss of Silence” eredmények				
Iterációk	Módszer	Megoldás minősége Eltérés (%)	Megoldási idő	
			$\mu$ (sec)	$\sigma$ (sec)
100	standard repertoár	8.14	3.581	7.781
100	elő-optimalizált repertoár	1.53	1.550	2.779
100	elő-optimalizált repertoár + head-tail -1	1.38	4.140	6.614
100	elő-optimalizált repertoár + head-tail -2	0.50	24.267	9.345

A következő táblázat annak a csoportnak a számítási eredményeit tartalmazza, amelyre a CPLEX az adott időkorláton belül nem tudott optimális eredményt adni, csak lehetséges megoldást. Ennek a csoportnak a futási eredményeit egy *Qualitymeasure* mutató bevezetésével mértük. Minden futtatást 30-szor végeztünk el, egymástól teljesen függetlenül. Ezzel statisztikailag szignifikáns eredményeket kaptunk, az eljárás robusztus jellegével kapcsolatosan. A hibrid algoritmusunk 8 esetben jobb eredményt szolgáltatott az egzakt megoldásnál, 7 esetben a szórás értéke 0.000 volt, ami azt jelenti, hogy ennél a hét esetnél a 30 független futás ugyanazt az eredményt adta. Ezek a tények mind a hibrid algoritmusunk robusztusságát és hatékonyságát mutatják.

Az alábbi táblázat mutatja, hogy ennek a csoportnak a megoldási idői is azt támasztják alá, hogy a hibrid algoritmusunk, gyors, hatékony, hiszen nagyságrenddel jobb eredményt szolgáltat a futási időket is tekintve, mint a CPLEX.



	Átlagos idő sec	Szórás	Min Idő sec	Max Idő sec
Átlagok:	4.9421	1.2922	0.1860	44.1380

**5. Tézis: Ha rögzítjük a tevékenységek megvalósítási módját, felvetődik az a lehetőség, hogy másodlagos optimalizálási kritériumokat fogalmazhatunk meg, a mozgatható tevékenységek halmazán.**

Ha rögzítjük a tevékenységek megvalósítási módjait, amelyeket az optimalizálás során kaptunk, valamint a projekt időtartamát, felmerül annak a lehetősége, hogy valamilyen másodlagos szempont alapján, további optimalizálási kritériumokat adhatunk meg. Ha rögzítjük a megvalósítási módokat és a projekt időtartamát, marad annyi szabadsági foka a tevékenységek mozgathatóságának, hogy másodlagos optimalizálási kritériumot kielégítő optimális megoldáshoz jussunk, természetesen az erőforráskorlátok megsértése nélkül. Olyan hozzárendelést kell definiálnunk, amiben nincs rejtett megszakítás, nincs rejtett konfliktus. A cél tehát egy konfliktusjavító modell alkalmazása. A dolgozatban szereplő konfliktusjavító modell hasonló, a Láng Blanka Phd dolgozatában (Pécs, 2010) szereplő konfliktus javító modellhez, hiszen a két kutatómunka közös töről fakad. Az idézett dolgozatban a szerző azonban csak egymódú ütemezésekre alkalmazza a megoldást, amit mi továbbfejlesztve, több-megvalósítási módú ütemezésekre alkalmazzuk.

A Sounds of Silence algoritmus további bővítésként a rejtett erőforrás felhasználási konfliktusokat elsőbbségi kapcsolatok beépítésével oldja fel. A „forward-backward listáütemező eljárás” eredménye egy olyan ütemezés, amelyben az összes olyan tevékenység, amely mozgatható, mozgatható lesz az erőforrás korlátok megsértése nélkül. Az így kapott erőforráskorlátoknak megfelelő megoldáshalmazon elvégezhetjük a másodlagos kritérium szerinti optimalizálást.

**6. Tézis: Másodlagos optimalizálási kritériumként fogalmazható meg az erőforráskiegyenlítési-hozzárendelési probléma**

Az erőforrásprofil alakjának „szépsége” nem csak esztétikailag fontos, hanem a projektvezetés szempontjából is. A menedzsment érdeke az, hogy egy munkát lehetőség szerint ugyanaz a munkás végezzen el, menet közben ne változzék a tevékenységet végrehajtó erőforrás. Az is rendkívül fontos szempont, hogy az erőforrások lehetőség szerint egyenletesen legyenek felhasználva, ne legyenek tétlen várakozási idők, majd újbóli munkavégzés, hiszen ez a gyakorlatban általában plusz költségekkel jár. Modellünk szempontjából másodlagos optimalizálási kritériumként azt a célt tűztük ki, hogy a megvalósítási módokat, az ütemezés sorrendjét és a projekt időtartamát rögzítve, minimalizáljuk az erőforrás egységek indítási-újraindítási eseményeinek számát az előző tézisben megfogalmazott mozgatható tevékenységek halmazán, mely mozgatható tevékenységek nem sértik az erőforráskorlátokat. A másodlagos optimalizálási kritériumnak megfelelő erőforrás kiegyenlítési probléma a következő formulákkal írható le:

$$\min \left[ LM = \sum_{r=1}^R \sum_{i=1}^T CU_{rt}^+ \right] = LM^* \quad (15)$$

$$X_i + D_i \leq X_j, \quad i \rightarrow j \in PS^* \quad (16)$$

$$X_i = \sum_{t \in T_i} X_{it} * t, \quad T_i = \{\underline{X}_i, \underline{X}_i + 1, \dots, \bar{X}_i\}, \quad i \in \{1, 2, \dots, N\} \quad (17)$$

$$\sum_{t \in T_i} X_{it} = 1, \quad X_{it} \in \{0, 1\}, \quad i \in \{1, 2, \dots, N\} \quad (18)$$

$$A_t = \{i \mid X_i \leq t < X_i + D_i, i \in \{1, 2, \dots, N\}\}, \quad t \in \{1, 2, \dots, T\} \quad (19)$$

$$U_{tr} = \sum_{i \in A_t} R_{ir}, \quad t \in \{1, 2, \dots, T\}, \quad r \in \{1, 2, \dots, R\} \quad (20)$$

$$U_{tr} - CU_{tr}^+ + CU_{tr}^- = U_{t-1r}, \quad t \in \{2, 3, \dots, T\}, \quad r \in \{1, 2, \dots, R\} \quad (21)$$

$$U_{1r} - CU_{1r}^+ = 0, \quad r \in \{1, 2, \dots, R\} \quad (22)$$

$$X_{it} \in \{0, 1\}, \quad t \in T_i, \quad i \in \{1, 2, \dots, N\} \quad (23)$$

Az algoritmust bővítettük egy MILP formulán alapuló erőforrás kiegyenlítő-hozzárendelő eljárással. Hogy ne növekedjék az algoritmus számítási idő igénye túlzottan, a hibrid algoritmusunk „karmestere” csak akkor hívja meg ezt az eljárást, ha az optimalizálás során az éppen aktuális „legjobb” megoldást cseréljük le egy még jobbra. Ekkor a javított (5. tézis) erőforrás-korlátos megoldáshalmazon, - rögzítve a tevékenységek megvalósítási módját, az ütemezés sorrendjét és a projekt időtartamát az éppen aktuális optimális megoldásnak megfelelően - megoldjuk az erőforrás kiegyenlítési problémát. A tevékenységek megvalósítási módjának valamint a projekt időtartamának rögzítésével marad még valamennyi szabadságunk, hogy a tevékenységek mozgatóásával az erőforrásprofilot kisimítsuk.

Az algoritmus, függetlenül az erőforrásprofil simításától, egy másik fontos gyakorlati problémát is megold, mégpedig azt a jogos követelményt, hogy egy tevékenység egységnyi erőforrásigényét pontosan egy erőforrás elégítse ki. Ezzel a követelménnyel a dedikált erőforrás hozzárendelési probléma megoldására is felkészítettük az algoritmust.

Az algoritmus kihasználja azt a tényt, hogy az erőforrás kiegyenlítés és a dedikált erőforrás hozzárendelési probléma egymástól teljesen függetlenül kezelhető, mivel az adott erőforrás kiegyenlítő mérték invariáns az erőforrás egységek permutációjára, a dedikált erőforrás hozzárendelés nem képes megváltoztatni az erőforrás kiegyenlítési mértéket. Az alkalmazott kisimító eljárás a folyamatos munkát részesíti előnyben és minimalizálja az induló-újrainduló dedikált erőforrás egységek számát.

### *A másodlagos kritériummal kibővített optimalizáló heurisztika számítási eredményei*

A bővített algoritmus képességének bizonyítására a már korábban bemutatott J30MM halmazon, annak is a J30MM-10-1 példányára vonatkozó eredményeket adtuk meg.

Az erőforrás kiegyenlítés hatását az alábbi táblázat adataival szemléltetjük:

*Az újraindítási események száma az erőforrás kiegyenlítés tükrében*

Erőforrás	Kiegyenlítés	
	előtt	után
1	55	24
2	57	27
	<b>112</b>	<b>51</b>

A táblázatból kitűnik, hogy a simító eljárás több mint 50%-os csökkenést eredményezett az induló-újrainduló erőforrás egységek számában.

Az eredmények alapján megállapítható, hogy a hibrid eljárásunk elfogadható időn belül képes a több-megvalósítási módú projektek esetében minimális időtartamú erőforrás-korlátos ütemezést adni, valamint képes arra, hogy egy tevékenység egységnyi erőforrásigényhez pontosan egy erőforrás egységet rendeljen, és az erőforrásprofil tekintetében úgy simítsa ki az erőforrás hisztogramot, hogy minél jobban megközelítse az ideálisnak tekintett téglalap alakot.

## **Összegzés**

Disszertációnkban egy új harmóniakereső metaheurisztikát ismertettünk, amely a több-megvalósítási módú projektütemezési problémák esetén képes optimális, minimális időtartamú erőforrás-korlátos ütemezést adni. Hibrid algoritmusunk az egy-megvalósítási módú erőforrás-korlátos projektek időtartamát minimalizáló Sounds of Silence algoritmus továbbfejlesztése, bővítése. Munkánkkal bebizonyítottuk, hogy a harmóniakereső heurisztika Sounds of Silence algoritmus megfelelő továbbfejlesztéssel, adaptálható a több-megvalósítási módú projektek esetére. Alkalmassá tettük az algoritmust, a forward-backward improvement, a head-tail eljárásokkal arra, hogy még pontosabb és gyorsabb eredményt adjon. Részletes, reprodukálható számítási eredményeket közöltünk a nemzetközileg, a kutatók körében ismert és elfogadott teszthalmazon. Ezek az eredmények mind azt mutatták, hogy algoritmusunk hatékony, robusztus algoritmus.

A konfliktusjavító modell adaptálásával alkalmassá tettük az algoritmust arra, hogy másodlagos kritériumokat is megfogalmazzunk célfüggvényként. Másodlagos célként az erőforrás kiegyenlítési és hozzárendelési probléma megoldásának képességével növeltük az algoritmus tudását. Ezzel a gyakorlathoz még közelebb álló probléma megoldására is alkalmassá vált hibrid algoritmusunk.

A közelmúltban a kutatókban kialakult az igény arra, hogy nagyobb méretű ismert paraméterekkel rendelkező teszthalmazok jöjjenek létre, amelyek standardként szolgálnak arra, hogy összehasonlíthatókká váljanak a közzétett eredmények. A Gentben folyó kutatások, Vanhoucke és Peteghem vezetésével, már létrehoztak egy 50 és 100 tevékenységből álló halmazt (MMLIB50, MMLIB100, és MMLIB+) de a dolgozat megírásakor még nem álltak rendelkezésre futási eredmények és még nem ismertek a teszthalmazok pontos paraméterei. A tesztprojektek tanulmányozása alapján állítható, hogy tartalmaz valódi kihívást jelentő feladatokat. Különösen igaz ez az MMLIB+ halmazra, amely 50 és 100 tevékenységből álló projekteket tartalmaz. Minden tevékenységhez 2 vagy 4 megújuló és nem-megújuló erőforrás rendelhető hozzá, és minden tevékenységnek 3, 6 vagy 9 megvalósítási módja van. A Gentben folyó kutatások új kihívásokat jelentenek, amelyek egyben további kutatási irányt is megszabnak számunkra. A valódi kihívást jelentő, 50 és 100 tevékenységet tartalmazó halmazok esetén is vizsgálhatjuk az algoritmusunk hatékonyságát, versenyképességét.

A tesztprojektek letölthetők a <http://www.projectmanagement.ugent.be/mrcpsp.html> oldalról. Ezek az új teszthalmazok további, jövőbeli kutatások alapját képezhetik.

Érdemes megfontolni az algoritmus további bővítésének lehetőségét is. Egyik ilyen irány lehet a több-megvalósítási módú erőforrás-korlátos projektütemezés időtartamának minimalizálása, valamint az erőforrásprofil kisimítása mellett a nettó jelenérték maximalizálási kritérium beépítése is a modellbe. Az utóbbi 5-10 évben a szakirodalom egyre intenzívebben foglalkozik ezzel a területtel több-megvalósítási módú projektek esetében is.

További lehetséges fejlesztési irányok:

- Terveink között szerepel az egész J30MM halmazra adni futási eredményeket a kibővített heurisztika esetében is.
- Az alkalmazott erőforrás kiegyenlítési modell helyett más kiegyenlítési kritériumok alkalmazása. Számos ilyen modell ismert a szakirodalomban. Érdemesnek tartjuk összehasonlítani a különböző modellek alkalmazása esetében kapott futási eredményeket.

## Publikációk

### Referált publikációk:

**Etelka Szendrői:** *A robust hybrid method for the multimode resource-constrained projectscheduling problem* Pollack Periodica, Vol 5, No. 3, 2010, 175-184,  
DOI:10.1556/Pollack.5.2010.3.15

**Szendrői Etelka:** *Egy hibrid eljárás a több megvalósítási módú erőforrás-korlátos projektütemezési probléma megoldására,* SZIGMA, XLI., 3-4,2010, 177-194

### Referált konferencia kiadványok:

**György Csébfalvi, Anikó Csébfalvi, Etelka Szendrői:** *A harmony search metaheuristic for the resource-constrained project scheduling problem and its multi-mode version,* in Project Management and Scheduling 2008, F.S. Serifoglu, Ü. Bilge, (Editors), Istanbul, Turkey, 56-59, 2008

**Etelka Szendrői :** *A hybrid method for the multi-mode resource-constrained project scheduling problem,* Conference paper, CC 2009, The First International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering, B.H.V. Topping, Y. Tsompanakis, (Editors), Civil-Comp Press, Stirlingshire, United Kingdom, paper 3,2009, doi:10.4203/ccp.92.3

**Etelka Szendrői:** *A Hybrid Method for the Multi-Mode Resource-Constrained Project Scheduling Problem with Strip Packing like Resource Constraints,* In: B. H. V. Topping, J. M. Adam, F.J.Pallares, M.L. Romero (Editors) Proceedings of the Seventh International Conference on Engineering Computational Technology Valencia, Spain, 2010.09.14-2010.09.17., Civil-Comp Press, Stirlingshire, United Kingdom, paper 93,2010, doi:10.4203/ccp.94.93

**Szendrői Etelka :** *Több megvalósítási módú erőforrás-korlátos projektütemezési probléma megoldása harmóniakereső metaheurisztikával,* Informatika a Felsőoktatásban 2011, Debrecen, 2011.8.-24-2011.8.26, Paper 23, pp 229-238, ISBN: 978-963-473-461-1

### Konferencia előadások:

**Szendrői Etelka:** *Egy erőforrás kiegyenlítő eljárás több megvalósítási móddal jellemezhető tevékenységi hálókra,* XXVI. Operációkutatási Konferencia, Győr, 2004. március 26-28.

**Etelka Szendrői:** *A new resource leveling MILP model for multi-mode projects based on global measure,* Veszprém Optimization Conference: Advanced Algorithms (VOCAL 2004), December 13-15, 2004, Veszprém, [http://www.dcs.vein.hu/vocal\(2004\)](http://www.dcs.vein.hu/vocal(2004))

**Etelka Szendrői:** *Resource Balancing MILP model for multi-mode projects,* 17th EURO Mini Conference: Continuous Optimization Industry, Pécs, 2005, June 29-July 1, 2005 University of Pécs

**Anikó Csébfalvi, György Csébfalvi, Etelka Szendrői:** *A harmony search metaheuristic for the multi-mode resource-constrained project scheduling problem*, Veszprém Optimization Conference: Advanced Algorithms (VOCAL 2008)

**Etelka Szendrői:** *A hybrid algorithm for the multi-mode resource-constrained project scheduling problem*, Konferencia előadás, Fifth International Phd&DLA Symposium, Pécs, 2009

**Etelka Szendrői:** *A Hybrid Method for the Multi-mode Resource-Constrained Project Scheduling problem with resource-leveling*, Sixth International Phd&DLA Symposium, Pécs 2010