



**XXXII. Országos
Tudományos Diákköri
Konferencia
2014**



**Termelő rendszerek teljes eszközhatékonyságának
modell alapú fejlesztését támogató eszközrendszer kidolgozása**

Készítette:

Ruppert Tamás

Gépészmérnöki BSc. 4. évfolyam

Témavezető:

Dr. Abonyi János

Egyetemi tanár

Folyamatmérnöki Intézeti Tanszék

Mérnöki Kar

Pannon Egyetem

Veszprém

Tartalom

Tartalmi összefoglaló.....	1
1. Bevezetés.....	2
2. A folyamatfejlesztés javasolt módszertana.....	8
2.1. A Folyamatmodell felépítése	10
2.2. Célfüggvény definiálása	11
3. A fejlesztett keretrendszer	12
4. Esettanulmány	16
4.1. A vizsgált folyamat	16
4.2. A folyamat paraméterei	18
4.2. Célfüggvény és az optimalizációs feladat	21
4.3 Analitikai vizsgálatok.....	22
4.3. A vizsgálat eredményei	23
5. Összegzés	26
Köszönetnyilvánítás.....	27
Irodalomjegyzék	28
Mellékletek.....	29
M1. Matlab forráskódok.....	29
M2. Visual Basic Kód.....	30

Tartalmi összefoglaló

Kutatásom célja a termelői és logisztikai rendszerek hatékonyságának, teljesítményének, megbízhatóságának és minőségének, azaz teljes eszközhatékonyságának fejlesztésére alkalmas eszközrendszer kidolgozása. A fejlesztett eszköztár a vizsgált folyamatok szimulációs modelljeinek optimalizálásával tesz javaslatokat azok fejlesztésére.

A dolgozat azokra a fejlesztésekre fókuszál, melyeket annak érdekében végeztünk el, hogy lehetőségünk legyen a folyamatok során előforduló hibák gyakoriságának, és ezek észlelési valószínűségének, valamint a minőségköltségek vizsgálatára. A kidolgozott rendszerrel a minőségköltségekre ható paraméterek hangolásával elemezni tudjuk a technológia fejlesztésének várható hatásait, a hibák feltárásával járó előnyöket és ezek költség és erőforrás igényeit. Ezen változók optimumának megtalálása nem triviális feladat. A fejlesztés elsődleges célja, hogy a teljes eszközhatékonysághoz kapcsolódó mutatószámok és költségelemek optimalizálásával javaslatot tudjunk tenni a fejlesztés ideális mértékére, fontosabb tényezőire. Ahhoz, hogy ezek a vizsgálatok illeszkedjenek a valós folyamatokhoz, figyelembe kell vennünk azok sztochasztikus jellegét is, amit az optimalizáló eljárás Monte-Carlo szimulációba integrálásával érhetünk el. A fejlesztett keretrendszer SIMUL8, Matlab és MS Excel programok együtteséből épül fel. A keretrendszer széles körű alkalmazhatóságát biztosítandó olyan Excel interfészt fejlesztettem, melynek segítségével a folyamatok teljes körűen paraméterezhetők, és a mutatószámok áttekinthetők. A kidolgozott módszertan alkalmazhatóságát egy általános technológiai folyamat elemzése kapcsán illusztrálom.

1. Bevezetés

Dolgozatom célja egy olyan keretrendszer és módszertan kidolgozása és fejlesztése, mellyel a termelői és logisztikai folyamatok teljes körű vizsgálatán és elemzésén túl összetett optimalizáló eljárást is végezhetünk. A kidolgozott keretrendszernek elég összetettnek és kellően átfogónak kell lennie ahhoz, hogy a módszertannal bármely termelői feladatot vizsgálni tudjunk a keretrendszer nagyobb mértékű változtatása nélkül.

Eddigi kutatásaim során folyamat-szimulációs technológiákat fejlesztettem és alkalmaztam, mely eszközök segítségével termelői és logisztikai folyamatokon végezhettem széleskörű vizsgálatokat. Munkám során felismertem, hogy a piacon megtalálható szoftverrendszerek vagy nem teszik lehetővé a teljes körű elemzést, vagy nem végezhető a szoftverrel paraméteroptimalás. Mivel az általunk eddig fejlesztett szimulációs keretrendszer sem teszi lehetővé az optimalást, így a jelenlegi dolgozat legfőbb célkitűzése a keretrendszer kibővítése volt annak érdekében, hogy a rendszer képes legyen paraméteroptimalásra. A továbbfejlesztett keretrendszer legfontosabb eredménye, hogy a szimulációs keretrendszert integráltuk a Matlab szoftvercsomaggal.

A termelői folyamatok fejlesztésének célja, hogy – a versenyképesség növelése érdekében – a fogyasztók által igényelt termékek és szolgáltatások előállításán és disztribúcióján javítsunk. A folyamatokban az anyagáramlás, a személyáramlás, az energiaáramlás és az információáramlás irányítása és felügyelete igen fontos és összetett feladat. A termelői szolgáltatások terén általában két versenyprioritás rendelkezik alapvető fontossággal, a költséghatékonyság és a megrendelések teljesítésének pontossága és átfutási ideje. A termékeket alapvetően nem raktárra kívánjuk termelni, de fontos szempont, hogy ki tudjuk elégíteni a vásárlói kívánságokat különösebb beavatkozások és várakozások nélkül. Ez a termelési rendszerektől rugalmasságot és nagy termelékenységet követel meg. A rövid átfutási idők elérése, a nagyfokú szállítási készség biztosítása, az alacsony készletszint fenntartása, az üzemi eszközök lehető legjobb kihasználása és egyenletes terhelése mind-mind fontos tényezők [1].

Minden folyamat esetén vizsgálnunk kell a hibák kialakulásának és azok észlelésének valószínűségét, tehát a folyamatunk minőségi mutatóit. Ezek a mutatók rendkívül összetettek és bonyolultak lehetnek. Nagyon fontos szempont, hogy a hibát észleljük még a folyamaton belül és ott is a lehető legkorábban, hiszen a hibaészlelésének minél későbbi bekövetkezése a hiba javítási költségének a 10 hatványával való növekedését vonja maga után [2].

Természetesen az optimális megoldás a hibamentes gyártás lenne, de ez a való életben lehetetlen feladat, így ezt az értéket csak közelíteni tudjuk. Azt, hogy a hibamentes működés és a hiba észlelésének mértékét milyen határokig tudjuk javítani, minden esetben a gyártási, logisztikai folyamat határozza meg. Ezen mutatók javítása komoly költségelem, hiszen minél tökéletesebb működést akarunk elérni, annál fejlettebb technológiát kell használnunk. Egy bizonyos határon túl ezeknek a mutatóknak a javítása a vállalat számára már nem gazdaságos, és az ezzel járó költség ellehetetlenítené a működésüket. Ennek tekintetében kijelenthető, hogy a minőségi paraméterek optimális értékének hangolása összetett feladat, a kidolgozott keretrendszer és azon belül is az optimálást is lehetővé tévő algoritmusok többek között erre a kérdésre is választ adhatnak.

Természetesen e módszertannak kapcsolódnia kell az eddig kidolgozott és egyre szélesebb körben alkalmazott folyamatmodell és mutatószámrendszer alapú fejlesztési technikákhoz. Ahhoz, hogy egy vállalat megőrizze versenyelőnyét folyamatosan fejlesztenie kell folyamatait, fel kell tárnia az esetleges változtatások lehetőségeit, a fejlesztések hatását, a kapcsolatot a folyamatokat jellemző mutatószámok között és nagy hangsúlyt kell fektetnie a minőségmenedzsmentjére is.

Dellana és Hauser [3] kiemeli, hogy a minőségmenedzsment stratégiai megközelítésében nincsen egyetértés. Emiatt a szerzők a TQM (Total Quality Management = teljes körű minőségirányítás) elveknek megfelelő, az Amerikai Egyesült Államok kormánya által alapított Malcolm Baldrige Nemzeti Minőség Díj szempontrendszerét használják. A szempontrendszer hét területet ölel fel: vezetés, információ és elemzés, stratégiai tervezés, emberierőforrás-fejlesztés és menedzsment, folyamatmenedzsment, üzleti eredmények, vevőorientáció és elégedettség.

Baird és társai [4] szerint a „TQM integráló, szervezetet átívelő filozófia, amely a folyamatok fejlesztését célozza meg, ami által megfelel vagy meghaladja a vevői elvárásokat”. A TQM-et Kaynak modelljének központi elemei alapján határozták meg: minőséggel kapcsolatos adatok és jelentések, a beszállítói minőségmenedzsment, szolgáltatás tervezése és folyamatmenedzsment.

Bates és társai [5] a termelési stratégia alatt az alábbiértéket: „a termelési funkció számára kialakított terv vagy vázlat, amely a távolabbi jövőig hatóan keretet adnak a termelési képességek akvizíciójának, a fejlesztésének vagy megszüntetésének”. A kutatásban a termelési stratégiára folyamatként tekintenek, pl. formális tervezés, kommunikáció, kapcsolat az üzleti stratégiával.

A termelési és logisztikai folyamatok esetén az egyik legjellemzőbb és összetettebb mutató a Teljes eszközhatékonyság (Overall Equipment Effectiveness - *OEE*). Az *OEE* egyetlen százalékos érték, mely azonban magában foglalja a rendszer számos elemét [6].

Az *OEE* három egysége:

- **Teljesítmény:** Minden folyamatnak van egy úgynevezett szűk keresztmetszete, ami alatt azt a folyamatlépést értjük, amelynek a legnagyobb az átfutási ideje. A folyamaton feldolgozásra kerülő egységek elméleti maximális számát ezzel a folyamatlépési idővel határozzuk meg. Az *OEE* esetén a teljesítmény a ténylegesen feldolgozott egységek száma osztva az elméleti maximális értékkel. Ez az érték minden esetben egynél kisebb érték lesz.
- **Rendelkezésre állás:** A teljes működési idő az elméleti 100%-os kihasználtság, hiszen a legjobb (elméleti) esetben a leglesterheltebb folyamatlépés a munkaidő minden pillanatában dolgozik. Ahhoz, hogy megkapjuk a rendelkezésre állási mutatót, a vizsgált folyamatlépés tényleges kihasználtságát kell figyelembe venni a teljes vizsgálati időre nézve.
- **Minőség:** Az elméleti maximális kibocsátott érték megegyezne a bemenő egységek számával, azonban ahogy ezt már bemutattam, ilyen eset nem létezik és a hibákkal számolnunk kell. A minőségi mutató a ténylegesen is hibamentesen feldolgozott egységek száma osztva a bemeneti egységek számával.

A fenti három mutatóból számolható az *OEE* értéke:

$$OEE = \text{Teljesítmény} * \text{Rendelkezésre állás} * \text{Minőség} \quad (1)$$

Az *OEE* érték meghatározása részletesen:

$$OEE = \frac{P_{valódi}}{P_{elméleti}} \frac{T_i}{T_{teljes}} \frac{Q_{ki}}{Q_{be}} \quad (2)$$

Ahol,

- $P_{valódi}$ – ténylegesen feldolgozott egységek száma
- $P_{elméleti}$ – feldolgozható egységek elméleti maximális értéke
- T_i – adott folyamatlépés tényleges időbeli kihasználtsága
- T_{teljes} – teljes vizsgálat ideje
- Q_{ki} – az elkészült egységek száma
- Q_{be} – a bemeneti nyersanyagok száma

• Teljes termelési idő				
Rendelkezésre állás = B/A	A termelésre tervezett idő (A)			Nincs tervezett termelés
	Tényleges termelési idő (B)		Leállások	
Teljesítmény = D/C	Elméleti kibocsájtás a maximális sebesség mellett (C)		Veszteségek a sebességben	
	Tényleges kibocsájtás (D)			
Minőség = F/E	Valós kibocsájtás – összes termék (E)			
	Jó termék (F)	Minőségi veszteségek	Hatékonyságveszteségek	
OEE = Rendelkezése állás * Teljesítmény * Minőség				

1. ábra OEE bemutatása

Az ábra a teljes eszközhatékonyság összetételét ábrázolja, feltüntetve a veszteségeket és leállásokat.

Az OEE tehát megmutatja, hogy mennyire hatékonyan használjuk ki a folyamatunk termelő képességét az ideális helyzethez képest ($OEE = 100\%$). A gyakorlatban az OEE érték a vártnál sokkal alacsonyabb értéket mutat, hiszen mind a három érték egynél kisebb lesz. Az ideális és a valóságos teljesítmény közti különbség egyenlő a veszteségek összegével, és ezáltal közvetlenül megmutatja, hol van szükség a fejlesztésre. Az 1. ábra mutatja a teljes termelési időre vonatkoztatott leállásokat, kieséseket. Az ábra jól szemlélteti az OEE mind a három egységének veszteségét.

A fejlesztési ötletek bevezetése, kipróbálása a való életben költséges művelet. A „Mi lenne, ha...” kérdések megválaszolása nagy és összetett folyamatmodellek esetén költséges és kockázatos, ezért az igazán érett, korszerű, versenyképes szervezetek erre a célra folyamat-szimulációs eszközöket alkalmaznak. A folyamat-szimuláció lehetőséget teremt a tevékenység alapú költségelemzésre. E tekintetben Whicker és munkatársai [7], valamint Comeilla és társai [8] mutatnak be kutatási eredményeket.

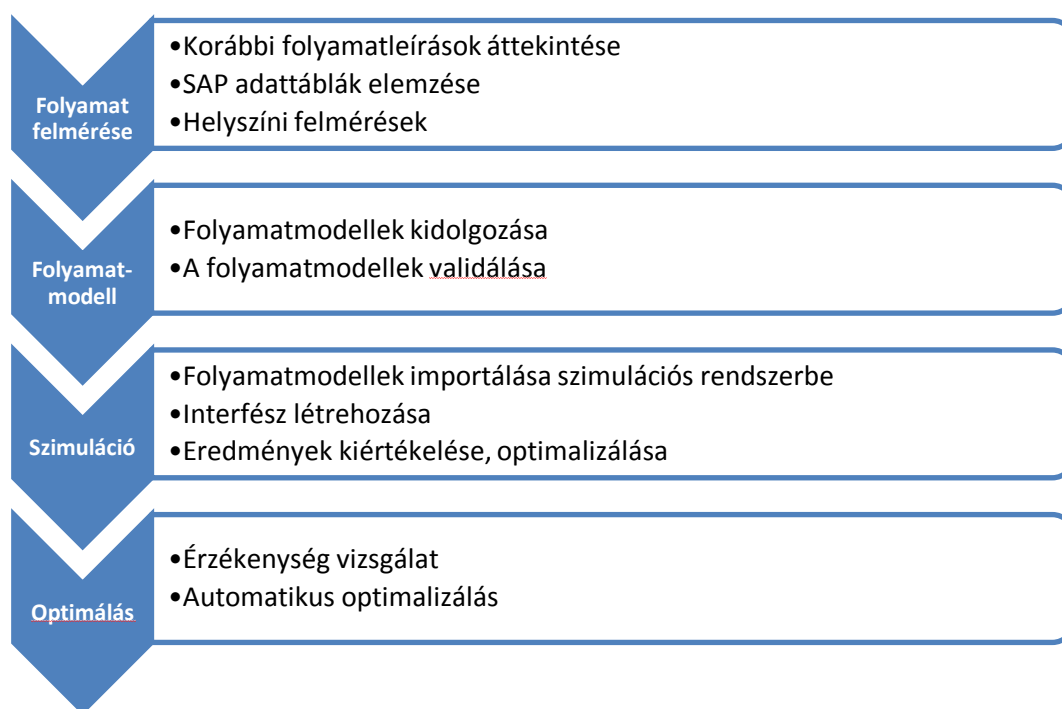
Az egyre erősödő versenykörnyezetben, sok bizonytalan vagy véletlenszerű elem fedezhető fel az ellátási láncban, mint vevői igények, átfutási idők, termelésingadozás. A sztochasztikus modellek ezeket a bizonytalan és véletlenszerű elemeket veszik figyelembe. Egyik meghatározó munka, amely az ellátási láncok sztochasztikus természetét veszi figyelembe Midler [9] nevéhez fűződik. Egy olyan dinamikus programozási modellt fejlesztett ki, ami az optimális irányítási technikák, szállítási módok, az árucikk áramok, vásárlók és beszállítók közötti utak optimális kombinációjának egy több periódusú tervezési horizonton történő meghatározására is alkalmas lehet.

A véletlenszerűség tehát fontos tényező a logisztikai folyamatok vizsgálatában, tehát a folyamat sztochasztikus jellegét figyelembe vevő elemzésre van szükség. Az elemzésre Monte-Carlo szimulációt javaslok.

A szimulációs vizsgálatok során generált mutatószámok halmazából értékes kimutatásokat tudunk felírni a vizsgált folyamatainkról. „Amit nem tudunk mérni, azt fejleszteni sem tudjuk.” Ezt a jelmondatot szem előtt tartva tekintjük át a már meglévő keretrendszert, amit a mostani kutatási fázisban fejlesztettem tovább.

A folyamatelemzési eljárás során 4 lépésre bontottam a fejlesztési lépéseket (2. ábra). A szimulációhoz interfészt fejlesztettem, melynek segítségével integrálhatóvá vált a vállalatirányítási rendszerrel, továbbá a paraméterezéshez és futtatáshoz a bonyolult szimulációs szoftver kezelői felülete helyett egy letisztultabb, felhasználóbarát felületet biztosít. Eddigi eredményeink több hazai, illetve nemzetközi konferencián is bemutatásra kerültek [10, 11, 12, 13, 14].

Jelen kutatási fázisban a logisztikai folyamatok mellett a termelő rendszerekre fókuszáltam, ahol a termékek átfutásának ideje mellett a minőségnek is kitüntetett szerepe van. Ebből kifolyólag alkalmazom az *OEE* – t, ahol a három egységből ez az egyik tényező. Annak érdekében, hogy az *OEE* minden tényezőjét vizsgálni tudjuk és ezen értékek javítására is javaslatot tudjunk tenni a termelésmenedzsment eszköztárát bővíteni kell a minőségköltségek kezelésével és az ezeket figyelembe vevő optimalizálással.



2. ábra. A módszertan lépései

A fejlesztést 4 lépésre bontottam, így felépítve a módszertant a kezdeti lépésektől a fejlesztési javaslatokat lehetővé tevő módszerig.

Kutatásom során fontos szempont volt, hogy összetett célfüggvényt tudjunk kezelni és soktényezős szimultán optimalizálást végezhessünk, mivel ilyen eszközt nem találtam, ebből következően a keretrendszert továbbfejlesztettem, hogy többváltozós optimalizálást is tudjunk vizsgálni. Új eredménynek számít a rendszer integrálása a Matlab szoftvercsomaggal, továbbá a bővített mutatószámok segítségével már teljes eszközhatékonysági mutatót is tudunk vizsgálni. Az így kialakított rendszer kellőképpen általános ahhoz, hogy sokoldalúan használni lehessen bármilyen folyamatmenedzsmenti probléma során anélkül, hogy a rendszeren komolyabb változásokat kellene alkalmaznunk, csupán a modellt kell a megfelelően felépítenünk.

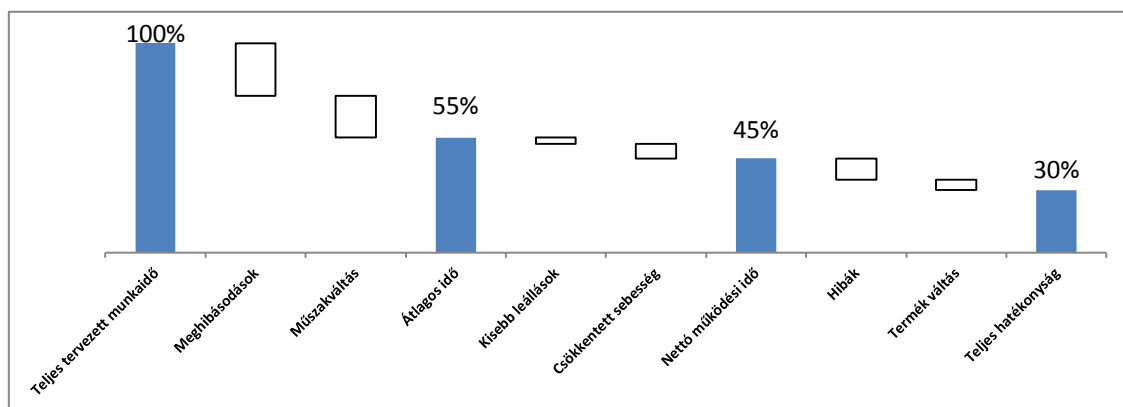
Dolgozatom második fejezetében részletesen bemutatom a folyamatoptimalizálásra kidolgozott módszertant, a harmadik fejezetben pedig a módszertan alkalmazására továbbfejlesztett keretrendszer felépítését. Végezetül pedig egy esettanulmányt ismertetek, amely egy összetett példán keresztül mutatja be a kidolgozott módszertan hatékonyságát.

2. A folyamatfejlesztés javasolt módszertana

Egy termelő szervezet kulcsmutatói optimumának megtalálása a keresleti, ellátási és belső termelési folyamatok paramétereinek összehangolását követeli meg a kereslet és kínálat, azaz a készletek minimalizálása és a hatékonyság növelése érdekében. Ezt szem előtt tartva hangolhatjuk a folyamatlépések paramétereit vagy az egységek beérkezésének gyakoriságát. Ezen paramétereket egy adott célfüggvény szerint kell optimalnunk, azonban a célfüggvény felírása nem triviális feladat, hiszen minden egyes folyamatlépés költségét és idejét definiálni kell és az optimalizáló paramétereknek alsó és felső korlátot is kell szabni.

A hatékony folyamatelemzés érdekében összetett módszertant dolgoztam ki. A cél az volt, hogy a módszertan a lehető legáltalánosabban alkalmazható legyen. A célfüggvény definiálásához meg kell állapítanunk az egyes folyamatlépések idejét, a beérkező anyagok gyakoriságának függvényében a várakozási időket, továbbá fel kell írunk a folyamat költségfüggvényét is. A folyamat egy éves átfutási idejének definiálásakor figyelembe vehetjük az emberi erőforrások esetén a szabadságok, a kényszerleállások és a nem produktív munkaidőket is. A harmadik ábrán látható, hogy egy folyamat esetén milyen veszteségekkel számolhatunk. Látható, hogy balról az első és az utolsó értékek közötti különbség a teljes veszteség és ezen értékek javítása a célunk.

Egy folyamatot sokféle mutatószámmal lehet jellemezni azonban ezek közül a legkifejezőbb mutató az úgynevezett teljes eszközhatékonysági mutató (*OEE*). Az *OEE* értéke egyetlen százalékos értékkel képes kifejezni a teljes folyamat teljesítőképességét, minőségét és kihasználtságát.



3. ábra. Egy rendszer teljes hatékonyság

Az ábráról leolvasható, hogy a folyamat teljes vesztesége 70%, ezen az értéken javíthatunk, amennyiben a különböző leállások és kiesések idejét csökkentjük.

A folyamatokat egy adott termelési kapacitásra tervezik meg. A gyakorlatban azonban a valós kibocsátás messze elmarad ezen elméleti kibocsátástól. Látszólag a végrehajtott fejlesztések sem hoznak hasznot: nagyobb termelési sebesség mellett több minőségi hiba fordul elő, és ha jobban ügyelnek a minőségre, akkor gyakoribb a leállás. Gyakran alkalmazzuk „a helyzet ilyen és már rég megtanultunk együtt élni vele” stratégiát [15]. Amennyiben a minőségileg jóváhagyott termék kibocsátása messze elmarad az installált kapacitástól, akkor rejtett termelési kapacitásról beszélünk. Ezen kapacitást ugyanis nem arra használják, hogy jó terméket állítsanak elő, hanem arra, hogy – akaratlanul vagy tudtukon kívül - különféle veszteséget termeljenek. A folyamatunk teljesítménye az ideális folyamattal összehasonlítva (mely mindig működik, maximális sebességgel és 100%-os minőséget termel) a rendelkezésre állásban, a teljesítményben és a minőségben termelődő veszteségek által határozható meg.

Az OEE három értéke a keretrendszer mutatóinak kombinációiból már vizsgálhatók, így végezhetünk modell alapú számításokat és vizsgálhatunk összetett „Mi lenne, ha...” típusú kérdéseket a teljes eszközhatékonyság javítása érdekében is.

A termelői rendszereknél jelenlevő bizonytalanságok, véletlenszerűségek miatt, hogy a valóságot minél jobban megközelíthessük a fenti vizsgálatokra Monte-Carlo szimulációs technikát javaslok, mely módszer segítségével a folyamatunk azon paramétereit is tudjuk vizsgálni, melyek bizonytalansági tényezőkkel rendelkeznek. Ezen paraméterekre normál eloszlást illeszthetünk, és a szimuláció többszöri kiértékelésével kaphatunk a valósághoz közeli értékeket.

A megfelelően felügyelt rendszer felveti az automatikus paraméteroptimalás lehetőségét, ennek tekintetében fejlesztettem tovább a keretrendszert. A fejlesztett interfész segítségével a logisztikai folyamat során szóba kerülő paraméterhalmazok mindegyikét a SIMUL8 programcsomagon kívülről hangolhatjuk, ebből következően a rendszer szabadságfoka igen magas. Az eljárást úgy alakítottam ki, hogy ne legyen rendszer specifikus, így a módszertan általánosan használható különböző feladatokra.

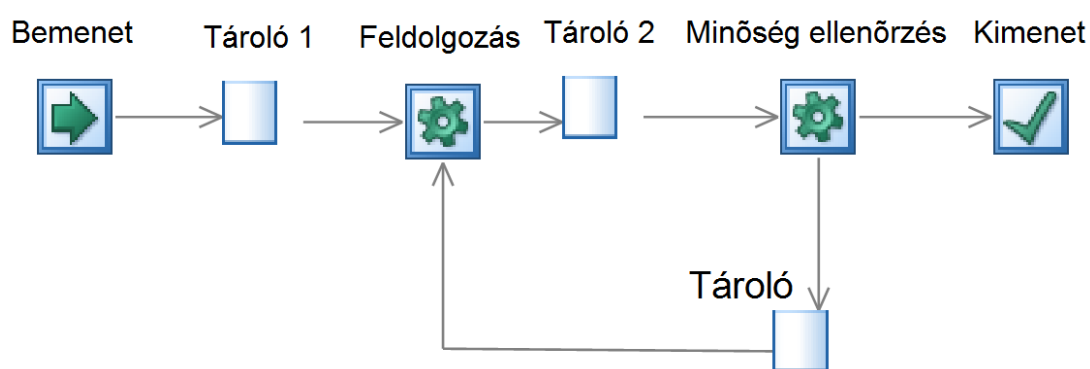
2.1. A Folyamatmodell felépítése

A folyamatmodellünket az első táblázat mintájára írhatjuk fel, ahol a négyzet szimbólum az egyes folyamatlépéseket, a háromszög szimbólumok pedig a tárolókat reprezentálják. A táblázatban rögzíthetjük az egyes objektumok legfontosabb jellemzőit.

1. táblázat. Folyamatmodell felírása

Folyamatlépés	Belépési pont	1. tároló	1. Folyamatlépés	2. tároló	2. Folyamatlépés	2. tároló	3. Folyamatlépés	Kilépési pont
								
Folyamatlépés átfutási ideje	$t_1/\text{egység}$	-	$t_2/\text{egység}$	-	$t_3/\text{egység}$	-	$t_4/\text{egység}$	-
Folyamatlépés egységenkénti költsége	$C_{IN}/\text{egység}$	$C_1/\text{egység/perc}$	$C_1/\text{egység}$	$C_2/\text{egység/perc}$	$C_2/\text{egység}$	$C_3/\text{egység/perc}$	$C_3/\text{egység}$	$C_{OUT}/\text{egység}$
Folyamatlépés költsége naponta	C_{FIX1}/nap	C_{FIX2}/nap	C_{FIX3}/nap	C_{FIX4}/nap	C_{FIX5}/nap	C_{FIX6}/nap	C_{FIX7}/nap	C_{FIX8}/nap

Amennyiben a fenti modellnél számításba szeretnénk venni azt is, hogy a valós folyamatok esetén előfordulnak hibák, abban az esetben a 4. ábra által mutatott folyamatmodellt írhatjuk fel. A példában a „Feldolgozás” folyamatlépés esetén a hibák $1 - \alpha$ valószínűséggel generálódnak, és annak érdekében, hogy ezek a hibás termékek ne jussanak ki a technológiai folyamatból a „Minőség ellenőrzés” lépésben vizsgáljuk meg a termékeket. Ebben a lépésben β valószínűséggel szűrjük ki a hibás termékeket és ezeket újra feldolgozzuk.



4. ábra. Folyamatmodell példa a minőség ellenőrzésre

A folyamat egy „Feldolgozás” és „Minőség ellenőrzés” lépésből áll, az első α valószínűséggel működik hibamentesen, az ellenőrzés pedig β valószínűséggel szűri ki a hibás egységeket

2.2. Célfüggvény definiálása

Az automatikus paraméteroptimalizálás során a legfontosabb feladat a megfelelő célfüggvény felírása. Az optimalni kívánt paramétereknek alsó-, és felső korlátot kell szabnunk, ezeket a korlátokat a célfüggvényben kell definiálnunk. Azt már megállapítottuk, hogy az OEE egy fontos jellemzője egy termelő rendszernek, így a felírt célfüggvényünknek követnie kell az OEE folyamatunkra jellemző mutatóit.

A 4. ábrán látható folyamat költségfüggvényét az alábbi módon határozhatjuk meg:

$$C_0 = C_{elad} Q_{ki} - \left[C_{vetel} Q_{be} + \int_0^t \sum_{i=1}^m (C_{tarol,i} Q_{tarol,i}) + \sum_{i=1}^n C_{proc,i} Q_{proc,i} + \sum_{i=1}^{n+m} C_{fix,i} + \frac{1}{1-\alpha} C_{alpha} + \frac{1}{1-\beta} C_{beta} \right] \quad (3)$$

α – Hibamentes működés valószínűsége [-]

β – Hibák észlelésének valószínűsége [-]

n - folyamatlépések száma (2 db)

m – tárolók (az egyes folyamatlépések közötti elemek) száma (3 db)

C – költségelemek $\left[\frac{\text{€}}{\text{db}} \right]$

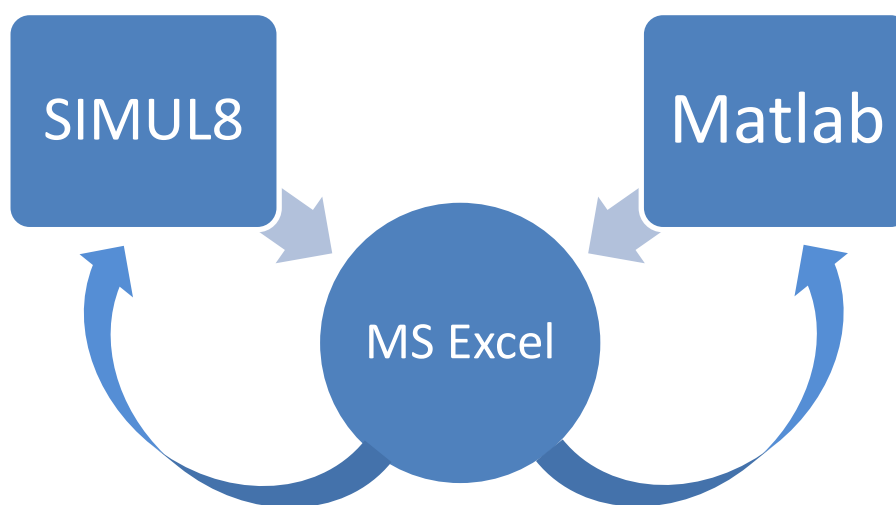
Q – feldolgozott egységek száma [db]

A minőségi jellemzők (α , β) hangolására a célfüggvényben az értékek felső határához az értékek reciprokjával szoroztam meg a hozzájuk tartozó költségelemeket. Számításba vettem mindegyik folyamatelem fix költségét, mely független az általa feldolgozott munkaegységek számától és a változó költségeket melyek már a feldolgozott, illetve a tárolt egységek számától is függ, továbbá természetesen a beszerzési költségeket. A profitot az eladási összeg és a költségek összegének különbségeként definiáltam.

3. A fejlesztett keretrendszer

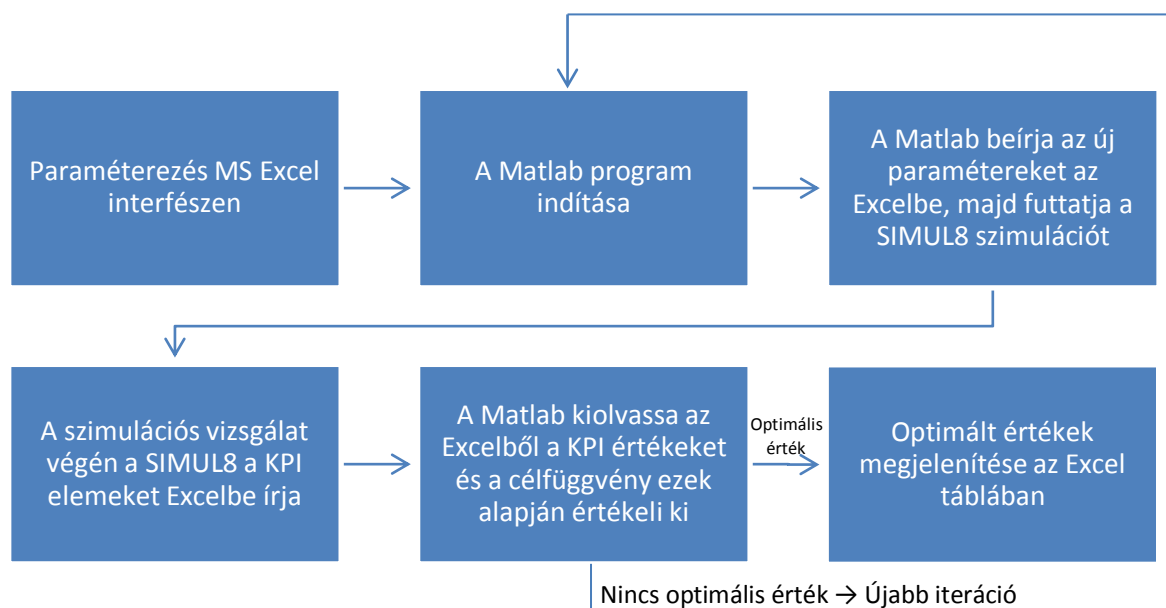
Kutatásom során fontos szempont volt, hogy összetett célfüggvényt tudjunk kezelni és soktényezős szimultán optimalizálást végezhesünk, ebből következően a keretrendszert továbbfejlesztettem, hogy többváltozós optimalizálást is tudjunk vizsgálni. A szimulációhoz interfészt fejlesztettem melynek segítségével integrálhatóvá vált a vállalatirányítási rendszerrel, továbbá a paraméterezés és futtatás a bonyolult szimulációs szoftver felülete helyett egy letisztultabb, felhasználóbarát platformot biztosít. Az interfész segítségével kibővíthető a szimulációs vizsgálatok figyelemmel kísérése is. A keretrendszer három fő szoftverből épül fel (5. ábra): Matlab, SIMUL8 és MS Excel.

A módszertanhoz továbbfejlesztett és kiegészített keretrendszer működésének algoritmusát a 6. ábra szemlélteti. A szimulációs vizsgálatokat a SIMUL8 programcsomag végzi és az Excel interfészen keresztül kommunikál a kezelővel. Ezen keresztül paraméterezzük a szimulációt és szintén az interfész segítségével kísérhetjük figyelemmel a vizsgált folyamatot. Ebből kiindulva döntöttünk amellett, hogy az optimalizálást az Excel interfészen keresztül valósítjuk meg, így az a teljes keretrendszer központi eleme lett.



5. ábra. A rendszer három fő szoftvereleme

A keretrendszer három fő programelemének a kapcsolata. A SIMUL8 és a Matlab az Excel interfészen keresztül kommunikál.



6. ábra. Optimalizációs iteráció

A rendszer folyamatábrája. A megfelelő paraméterezés után a Matlab programot elindítva lefut a szimulációs vizsgálat és a Matlab kiértékelése végén elindul ismét az iteráció az új paraméterekkel.

A folyamat-szimulációk elkészítéséhez a SIMUL8 folyamatorientált szimulációs programcsomagot használtam, mely az egyik legelterjedtebb folyamat-szimulációs eszköz [16]. Megismerve a felhasználói igényeket egy olyan interfészt határoztam meg, amelynek segítségével a felhasználó nem közvetlenül a SIMUL8 programmal találkozik, hanem a széles körben alkalmazott Excel táblázattal. Ennek érdekében a szimulációs programcsomagot adaptálnom kellett az MS Excel programmal (7. ábra), melynek segítségével a paraméterezés már csak alap informatikai tudást igényel és az elhelyezett nyomógommbal egy kattintással elindítható a szimuláció. Az interfész segítségével kibővíthető a szimulációs vizsgálatok mutatóinak száma is. Az egyszeri vizsgálatokon túl Monte-Carlo szimulációt is vizsgálhatunk, melynek eredményei szintén a táblázatban kerülnek megjelenítésre.

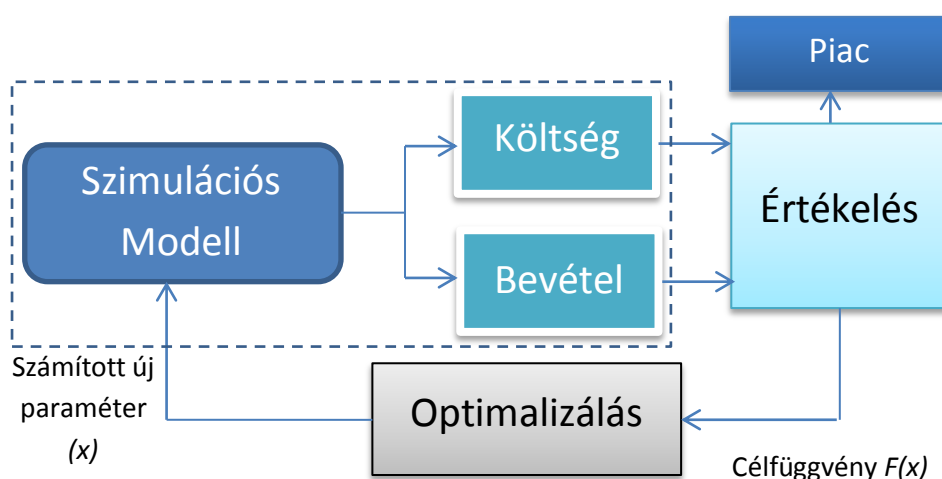
A szimulációs keretrendszernek képesnek kell lennie a Monte-Carlo szimulációra, így a rendszert úgy alakítottam át, hogy egyetlen bemenet helyett adott szórással a szimuláció akár több százszor is kiértékelje a modellt és az összes eredményt értelmezhető, áttekinthető formában megjelenítse.

	1	2	3	4
1	Indítás	Simul8 bezárása		
2				
3	Iterációk száma:	100		
4	Prioritás	min	max	
5	Process 1	40	40	
6	Process 2	80	80	
7	Process 3	100	100	
8	Process 4	100	100	
9	Process 5	40	40	
10	Párhuzamosság	min	max	
11	Process 1	3	3	
12	Process 2	3	3	
13	Process 3	2	2	
14	Process 4	1	1	
15	Process 5	1	1	
16	Folyamatidők, átlag	min	max	
17	1. process	0	0,05	
18	2. process	0	0,05	

7. ábra. MonteCarlogen munkalap

Az általam kidolgozott Excel munkalapon belül, meghatározhatjuk a szimuláció paramétereit, és az elhelyezett nyomógommbal elindíthatjuk a szimulációt.

A Matlab-ban írt eljárást futtatva az új paramétereket rögzíti az Excelben, majd a SIMUL8 folyamat-szimulációt már az új paraméterekkel meghívva végezzük el az új vizsgálatokat. Az így kapott eredmények szintén az Excel interfészen kerülnek megjelenítésre, majd a Matlab az újabb iteráció előtt a költség függvényt az interfésztől átvett költségelemekből számolja. Ezután az optimalizáló eljárás segítségével határozza meg az új paramétereket és végzi el az újabb vizsgálatot. A kialakított rendszer működését a 8. ábra szemlélteti.



8. ábra. Az optimalizálás sémája

A szimuláció által meghatározott költség és bevételi elemekből számolt célfüggvény alapján végezzük el az optimalizálást

A kapott keretrendszerrel már összetett vizsgálatokat végezhetünk és a többváltozós optimalizálás segítségével javaslatokat tehetünk bármilyen, általános típusú folyamat javítására, fejlesztésére. A megfelelő költségfüggvény definiálásával javaslatokat tehetünk, akár a teljes eszközhatékonyság (*OEE*) javítására is.

A negyedik fejezetben a bemutatott módszertant egy példán keresztül mutatom be, melynek segítségével rámutatok arra, hogy a rendszer segítségével összetett vizsgálatokat tudunk végezni és a megfelelő paraméterezéssel bármilyen típusú feladatot ki tudunk értékelni.

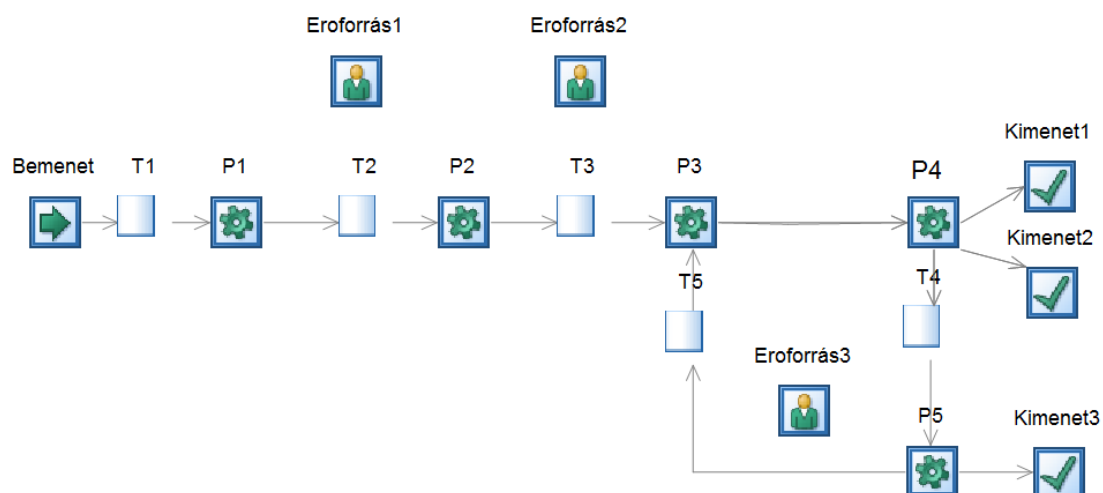
4. Esettanulmány

A kidolgozott módszertant, úgy alakítottam ki, hogy ne legyen problémaspecifikált és használható legyen későbbi fejlesztések, kutatások során. A dolgozatban egy általam kidolgozott példán keresztül mutatom be, hogy a rendszer széles körben használható és lefedi a teljes folyamatmenedzsmenti vizsgálatokat.

4.1. A vizsgált folyamat

A folyamat, amelyen vizsgálatokat végeztem (9. ábra) összesen 5 darab folyamatlépést és 5 darab tárolót tartalmaz, továbbá három kimenet is található:

- 1. kimeneten a kimenő megfelelő minőségű termékek jelennek meg
- 2. kimeneten a folyamatból kijutó hibák száma jelenik meg
- 3. kimeneten a selejtek száma, az újbóli feldolgozásra alkalmatlan termékek jelennek meg



9. ábra. Részegységek összeszerelésének folyamatmodellje

Az esettanulmányban vizsgált folyamatmodell felépítése. A folyamatban figyelembe vesszük és vizsgáljuk a felmerülő hibákat és azok észlelését is.

A modell jelölései:

- Bemenet – a modell belépési pontja, az alapanyag beszerzése
- T1, T2, T3, T4, T5 – az egyes folyamatlépésekhez tartozó tárolók jelölése
- P1 – Kitárolás folyamatlépése, alkatrész csomag előkészítése a további feldolgozásra
- P2 – Kicsomagolás folyamatlépése, alkatrészek és dokumentumok előkészítése
- P3 – Összeszerelés folyamatlépése, munkadarab és részegységek összeszerelése
- P4 – Minőség-ellenőrzés folyamatlépése, az összeszerelt egységek minőségének ellenőrzése
- P5 – Selejt kezelés folyamatlépése, ebben a lépésben ellenőrzik, hogy a hibás termékek még javíthatók-e
- Erőforrás1 – A P1 és P3 folyamatlépések erőforrása
- Erőforrás2 – A P2 és P4 folyamatlépések erőforrása
- Erőforrás3 – A P5 folyamatlépés erőforrása
- Kimenet1 – Ezen a kimeneten jelennek meg az elkészített egységek
- Kimenet2 – Ezen a kimeneten jelennek meg a folyamatból kijutó hibás egységek
- Kimenet3 – Ezen a kimeneten jelennek meg a selejtnak ítélt egységek

A folyamatmodell egy részegységeket összeszerelő folyamat modellje, amely összesen 5db folyamatlépést tartalmaz és a modellben megjelennek a hibás termékek is. A folyamatban a harmadik folyamatlépésnél jelennek meg a hibák $1-\alpha$ valószínűséggel, ha hibás termék keletkezik, akkor azt címkével látjuk el a SIMUL8 programon belül. A minőség-ellenőrzést a 4. folyamatlépés látja el. A minőség-ellenőrzés sem tudja 100%-osan kiszűrni az összes hibás terméket, így paraméterezhető egy β valószínűség, mely meghatározza, hogy milyen százalékban találjuk meg a hibás termékeket és $1-\beta$ valószínűséggel a hibás termék kikerül a folyamatból. Az észlelt hibás termékeket újbóli feldolgozásra küldjük, azonban lehetnek olyan rossz minőségű termékeink is, melyeket már nem tudunk újból feldolgozni. Ezt a valószínűséget szintén jellemezhetjük egy γ értékkel, melynek függvényében az 5. folyamatlépés dönt arról, hogy mely termékek mennek a selejt kimenetre és melyek újbóli feldolgozásra.

A folyamat minden lépésének folyamatidejét, a bemeneti intenzitást, az erőforrások számát és minden költségelemet, továbbá az ismertett minőségi elemeket (α , β , γ) is az Excel interfész segítségével tudjuk hangolni, ebből következőleg bármely paramétereken tudunk optimalást végezni. A példában az α , β és a bemeneti intenzitás értékeinek vizsgáltam az

optimumát a költségek függvényében. Ezeken az értékeken a valóéletben is tudunk javítani megfelelő fejlesztések segítségével.

4.2. A folyamat paramétere

Az esettanulmányban felírt példafolyamatot úgy paramétereztem, hogy az hasonlítson egy valós folyamathoz. A számítások során összesen egy hétnyi működést vizsgáltam. Az erőforrások számát az alábbi adatokkal paramétereztem:

- 1. típusú erőforrás: 3 fő
- 2. típusú erőforrás: 3 fő
- 3. típusú erőforrás (aki a selejteket újrafeldolgozásáról dönt): 1 fő

A folyamatlépések párhuzamossága alatt azt a számot értjük, amely megmutatja, hogy egyszerre mennyi munkadarab dolgozható fel az adott folyamatlépésben, az erőforrások függvényében. A folyamatlépések párhuzamossága:

- 1. folyamatlépés = 3
- 2. folyamatlépés = 3
- 3. folyamatlépés = 2
- 4. folyamatlépés = 1
- 5. folyamatlépés = 1

A bemeneti intenzitás határozza meg, hogy mennyi időegység alatt megy be a folyamatba egy munkadarab. A szimuláció során mindegyik paramétert percben határoztuk meg.

Folyamatlépések ideje

A folyamatlépések folyamatidejét külön-külön hangolhatjuk. Mindegyik értékét az interfészen keresztül paraméterezhetjük.

- 1. folyamatlépés = 1,20
- 2. folyamatlépés = 1,45
- 3. folyamatlépés = 2,40
- 4. folyamatlépés = 0,96
- 5. folyamatlépés = 2,00

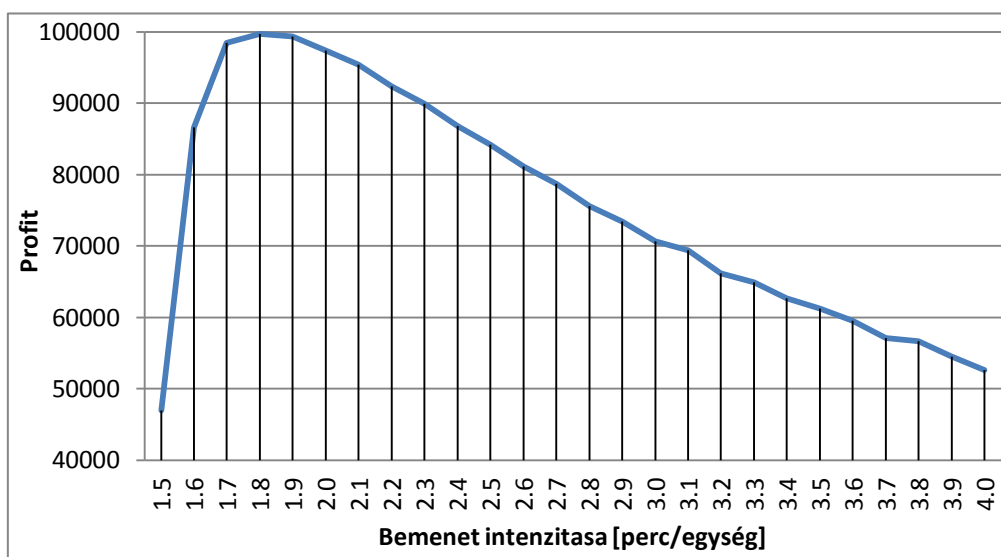
Amennyiben a sztochasztikus jelleget is figyelembe akarjuk venni megadható az egyes folyamatlépések idejének szórása is.

Költségek

A folyamat költségeinek jellemzői, hogy a beszerzési és eladási ár között 160%-os a haszonkulcs. A bemeneti intenzitás függvényében a folyamat által termelt profit az optimális intenzitástól eltérő értékek esetén (negatív és pozitív eltérés esetén is) eltér a maximális értéktől. Ezt a görbét a 10. ábra szemlélteti a 2. táblázatban szereplő adatok alapján.

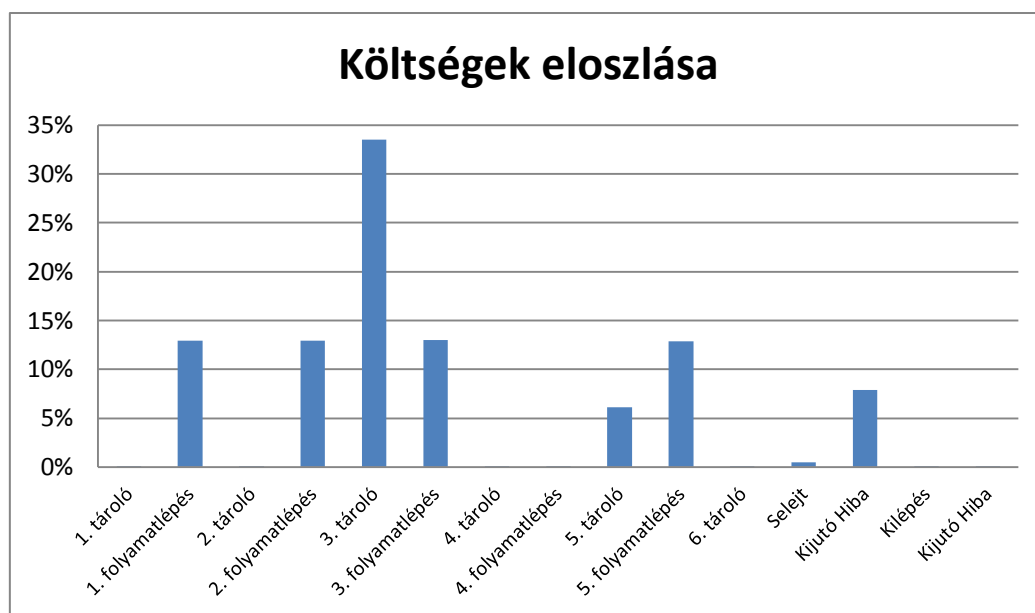
2. táblázat. A folyamat költségelemei

	Költségek	
	Fix költségek	Költség/egység
Bemenet	30	300
T1	30	10
P1	30	20
T2	30	10
P2	30	20
T3	30	10
P3	30	20
T4	30	10
P4	30	20
T5	30	50
P5	30	100
T6	30	50
Kimenet3	30	48
Kimenet2	30	1000
Kimenet1	30	480



10. ábra. A folyamat profitjának a bemeneti intenzitástól való függése

A görbe a profit alakulását mutatja a bemeneti intenzitás függvényében



11. ábra. Költségeloszlási ábra

Az ábrán látható, hogy a folyamat különböző elemeinek a költségei egy adott vizsgálat során milyen mértékű a teljes költségeket tekintve.

A költségek eloszlását a 11. ábra mutatja be. Az ábráról leolvasható, hogy az egyik legnagyobb költségelem a 3. tároló. Ez jelentheti azt is, hogy a bemeneti intenzitás túl nagy, vagy túl nagy százalékban keletkeznek hibás termékek, melyek a harmadik folyamatlépésben kerülnek újbóli feldolgozásra. Ez is magyarázhatja a harmadik tároló költségének emelkedését.

Hibavalószínűségek

A minőségi paramétereket az optimalizálandó paraméterek jelen feladatban. Ahogy azt már az előzőekben tisztáztuk az α értéke határozza meg a hibamentes munkadarabok százalékos mennyiségét a teljes mennyiséghez képest, β érték pedig a keletkezett hibák észlelésének valószínűsége.

A γ értéke meghatározza, hogy a hibás termékek milyen százalékban kerülnek újrafeldolgozásra, a maradék pedig selejtként kerül ki a folyamatból.

- $\gamma = 70\%$

4.2. Célfüggvény és az optimálási feladat

Az ismertett folyamatmodell esetén a minőségi paramétereknek kerestem az optimális megoldását. Egy bizonyos határ felett mind a hibamentes valószínűség, mind a hibák észlelésének valószínűsége elér egy bizonyos költség szintet, mely a folyamat szempontjából már gazdaságtalan. Ennek a szintnek az elhelyezkedését a célfüggvény határozza meg:

$$C_0 = C_{eladás} Q_{ki} - \left[C_{vételi} Q_{be} + \int_0^t \sum_{i=1}^m (C_{tároló,i} Q_{tároló,i}) + \sum_{i=1}^n C_{process,i} Q_{process,i} + \sum_{i=1}^{n+m} C_{fix,i} + C_{selejt} Q_{selejt} + \frac{1}{1-\alpha} C_{alpha} + \frac{1}{1-\beta} C_{beta} \right] \quad (4)$$

α – Hibamentes működés valószínűsége [-]

β – Hibák észlelésének valószínűsége [-]

n - folyamatlépések száma (2 db)

m - tárolók száma (3 db)

C – költségelemek $\left[\frac{Ft.}{db} \right]$

Q – feldolgozott egységek száma [db]

A függvényben a változó és fix költségek mellett figyelembe vettem a selejtek költségét is. A két érték (α , β) felső korlátját az: $\frac{1}{1-\alpha} C_{alpha} + \frac{1}{1-\beta} C_{beta}$ definiáltam. Az alsó határt az adja, hogy ha túl nagy számban jelenik meg hiba a folyamatban, akkor csökken az eladás és növekszik a selejtek száma, az újra feldolgozandó egységek száma, és ami a legnagyobb költségelem: a folyamatból kijutó selejtes termékek száma.

4.3 Analitikai vizsgálatok

A fejlesztett interfészen széleskörű analitikai vizsgálatokat is végezhetünk a paraméterezett folyamatunkon. A rendszert úgy alakítottam ki, hogy egy külön munkalapon egyetlen gombnyomással betölti és meghatározza az új paraméterekkel 3. táblázat és a 4. táblázat által leírt mutatókat. A táblázatokban szereplő mutatókat az alábbi összefüggésekkel definiáltam:

$$\text{Kapacitás} = \frac{1}{\text{Feldolgozási idő}} \quad (5)$$

$$\text{Holtidő} = \text{Ciklusidő} - \text{Feldolgozási idő} \quad (6)$$

$$\text{Kihasznátság} = \frac{\text{Térfogatáram}}{\text{Kapacitás}} \quad (7)$$

$$\text{Folyamat kapacitás} = \text{MIN}(\text{Kapacitás}) \quad (8)$$

$$\text{Térfogatáram} = \text{Folyamat kapacitás} \quad (9)$$

$$\text{Ciklusidő} = \frac{1}{\text{Térfogatáram}} \quad (10)$$

$$\text{Teljes holtidő} = \sum \text{Holtidő} \quad (11)$$

3. táblázat. Analitikai számítások eredményei

	Feldolgozási idő [perc]	Kapacitás [egység/perc]	Kapacitás óránként [egység/óra]	Holtidő	Kihasznátság	Kapacitás/munkahét [egység/hét]
1. folyamatlépés	1,20	0,83	50,0	1,20	0,5	2000
2. folyamatlépés	1,45	0,69	41,4	0,95	0,6	1655
3. folyamatlépés	2,40	0,42	25,0	0,00	1,0	1000
4. folyamatlépés	0,96	1,04	62,5	1,44	0,4	2500
5. folyamatlépés	2,00	0,50	30,0	0,40	0,8	1200

4. táblázat. Analitikai számítások eredményei

Folyamat kapacitás	Térfogatáram	Ciklusidő	Teljes holtidő
0,42	0,42	2,4	3,99

A munkahetet 5 napban és napi 8 órában határoztam meg.

4.3. A vizsgálat eredményei

Az előzőekben definiált feladat vizsgálatát és optimalizását végeztem el az általam fejlesztett módszertan és keretrendszer használatával. A vizsgálat során három paraméter optimumát kerestem:

- Bemeneti intenzitás
- Hibamentes működés valószínűsége
- Hibák észlelésének valószínűsége

Az optimalizálódó paramétereknek az alábbi kezdeti értékeket definiáltam:

- $Bemeneti\ intenzitás = 2,4 \left[\frac{egység}{perc} \right]$
- $\alpha = 99,95 \text{ [\%]}$
- $\beta = 99,99 \text{ [\%]}$

A keretrendszert a fenti értékekkel paraméterezve (12. ábra) kezdtem meg a vizsgálatot. A folyamatunk a 4.2. alfejezetben felírt költségfüggvény értéke a fenti kezdeti értékekkel:

- $y = 90.530 \left[\frac{€}{hét} \right]$ – Költség függvény, tehát ez az érték a rendszer által termelt profit egy hét alatt (5 munkanap, napi 8 óra termelés)

	1	2	3	4	5	6
1	Bemeneti intenzitás	2,4			Folyamatidő átlag	Folyamatidő szórás
2	Alpha	99,95		1. process	1,2	0,05
3	Beta	99,99		2. process	1,45	0,85
4	Gamma	30		3. process	2,4	3
5				4. process	0,96	0,56
6				5. process	2	1,54
7						
8	Resource 1	3				
9	Resource 2	3				
10	Resource 3	1				
11		Finance				
12		Fix	Cost-per			
13	Entry	30	300			
14	1. storage	30	10			
15	1. process	30	20			

12. ábra. Paraméterezés
Példa az Excel interfész paraméterezésére

Az optimalizálás előtti teljes eszközhatékonysági mutató számítása:

$$OEE = \frac{EF_1 + EF_2}{2} * \frac{\text{Tényleges teljesítmény}}{\text{Elméleti teljesítmény}} * 1 - \frac{\text{Hibák}}{\text{Kimenet}} \quad (12)$$

Ahol:

- EF_i – Erőforrások kihasználtsága
- *Elméleti teljesítmény* – A folyamat elméleti maximum feldolgozási teljesítménye:

$$\left(\frac{2400}{\text{Szűk keresztmetszet folyamat ideje}} \right)$$
- *Tényleges teljesítmény* – A folyamaton ténylegesen feldolgozott egységek száma
- *Hibák* – A vizsgálat során keletkezett összes hiba száma
- *Kimenet* – A vizsgálat során létrejövő összes kimenet és a selejtek száma

$$OEE = \frac{0,55 + 0,34}{2} * \frac{995}{\frac{2400}{1,3}} * \left(1 - \frac{1}{1000} \right)$$

$$OEE = 0,445 * 0,539 * 0,999 = 0,24$$

$$OEE = 24\%$$

Mivel a feladat összetett és a többváltozós optimálás miatt a vizsgálat teljes időtartama néhány óra. A vizsgálat eredménye:

A változók optimált értékei:

- *Bemeneti intenzitás* = $1,8223 \left[\frac{\text{egység}}{\text{perc}} \right]$
- $\alpha = 99,2460 \text{ [\%]}$
- $\beta = 99,9179 \text{ [\%]}$

A kapott profit érték:

- $y = 102.220 \left[\frac{\text{€}}{\text{hét}} \right]$

OEE számítása az optimált értékekkel:

$$OEE = \frac{0,72 + 0,45}{2} * \frac{1311}{\frac{2400}{1,3}} * \left(1 - \frac{11}{1000} \right)$$

$$OEE = 0,585 * 0,71 * 0,989 = 0,41$$

$$OEE = 41\%$$

Látható, hogy a paraméterek hangolásával javulást értünk el a költségfüggvény tekintetében, tehát a profitot növelni tudtuk a kezdeti feltételekhez képest. A növekedés számszerűsítve: 113%-os. Az *OEE* értékben is javulást értünk el (24% → 41%). Ahhoz, hogy ezt a javulást elérjük a két minőségi változó (α , β) értékén kis mértékben csökkenést kell alkalmaznunk, a bemeneti intenzitást pedig csökkentenünk kell. Ezek a változtatások eszközölhetők, hiszen új beszállítókkal a bemeneti intenzitást növelhetjük.

Az előző példával illusztráltam a bővített keretrendszer új funkcióit és lehetőségeit. Az eddigi és a mostani kutatási eredményekkel a keretrendszer a széleskörű folyamat-szimulációs elemzéseken túl, már automatikus paraméteroptimalásra is alkalmazhatóvá vált. A rendszer abban különbözik a piacon fellelhető szoftvercsomagok közül, hogy kihasználja a SIMUL8 nyújtotta széleskörű folyamat-elemzést és a Matlab program csomag által elérhetővé vált optimalási eljárásokat, továbbá a fejlesztéseknek köszönhetően mindezen funkciókra kiterjeszthető a Monte-Carlo szimuláció is. Az Excel interfész segítségével ezek a funkciók egy széles körben alkalmazott felületen keresztül paraméterezhető, futtatható és a rendszer fontos elemeként a teljes kimutatások, mutatók is az Excel táblázatban kerül megjelenítésre.

5. Összegzés

A kutatómunkámban termelői rendszerek kapacitásainak, teljesítményének, minőségének és költségének jellemzésére használt mutatószámok összefüggéseit és bizonytalanságát is kezelő logisztikai és termelői rendszer elméleti háttérének kidolgozásával kapcsolatos eredményeket foglaltam össze. A felvázolt keretrendszer kialakítása megköveteli a folyamatok részletes elemzését és figyelemmel kísérését. A modell alkalmazását egy termelői folyamat optimalizálási feladatának vizsgálata kapcsán mutattam be.

A kidolgozott módszertan hatékonyan integrálja a különböző szimulációs és vizualizációs eszközöket, mint pl. a SIMUL8 programcsomag. A SIMUL8-en alapuló szimulációs vizsgálat azon képessége, hogy a döntési változók függvényében lehetővé válik a fejlesztési célokat definiáló célfüggvény kiértékelése, felvetette az automatizált paraméteroptimalizálás lehetőségét. Ezért a kialakított keretrendszert integráltam a Matlab programcsomaggal és segítségével paraméteroptimalásra alkalmas eszközt fejlesztettem ki.

A fejlesztett keretrendszer lehetővé tette a SIMUL8-Matlab integrációt és ez egy új fejezetet nyithat a folyamat-szimuláció alapú fejlesztésben. A Matlab integrációjával lehetőség adódik további toolbox-ok alkalmazására is.

Köszönetnyilvánítás

A publikáció a TÁMOP-4.2.2.C-11/1/KONV-2012-0004 Nemzeti kutatóközpont fejlett infokommunikációs technológiák kidolgozására és piaci bevezetésére projekt társfinanszírozásával készült.

Szeretném megköszönni Király Andrásnak a SIMUL8 és Matlab integrációjához való hozzájárulását, továbbá Kulcsár Tibornak az optimalizálási algoritmus terén nyújtott segítségéért.

Végezetül szeretném megköszönni Dr. Abonyi Jánosnak a fáradhatatlan és folyton ösztönző munkáját, továbbá hihetetlen problémamegoldó képességét!

Irodalomjegyzék

- [1] Dr. Kulcsár Béla, Ipari Logisztika, LSI Oktatóközpont, 1998.
- [2] Dr. Kövesi János, Kockázat és megbízhatóság, Oktatási segédanyag Budapesti Műszaki- és Gazdaságtudományi Egyetem Gazdasági- és Társadalomtudományi Kar Ipari Menedzsment és Vállalkozásgazdaságtan Tanszék, 2010
- [2] Dellana, S.A., Hauser, R.D. Corporate Culture's Impact on a Strategic Approach to Quality. Mid-American Journal of Business, 2000, p. 12.
- [3] Baird, K., Hu, K.J., Reeve, R. The relationship between organizational culture, total quality management practices and operational performance. International Journal of Operations & Production Management, 2011, 789-814
- [4] Bates, K.A., Amundson, S.D., Schroeder, R.G., Morris, W.T. The Crucial Interrelationship between Manufacturing Strategy and organizational Culture, Management Science, 1565-1580, 1995
- [5] A LeanDesign szakmai oldalak jegyzetei alapján (leandesign.hu/), 2013.10.25.
- [6] Whicker, L., Bernon, M., Templar, S., Mena C., Understanding the relationships between time and cost to improve supply chain performance, Int. J. Production Economics, 2009, 641–650.
- [7] Comellia, M., Feniesa, P., Tcherneva N., A combined financial and physical flows evaluation for logistic process and tactical production planning, Application in a company supply chain, Int. J. Production Economics, 2008, 77–95.
- [8] Midler, J. L. A stochastic multiperiod multimode transportation model. Transportation Science, 1969, 8-29.
- [9] T. Ruppert, G. Belvárdi, Z. Gyozsán, J. Abonyi, Zs. Ulbert, Logisztikai tevékenységet támogató folyamatmodellek fejlesztése és Monte-Carlo szimulációval történő kialakítása, Műszaki Kémiai Napok 2012, 2012, 47-53.
- [10] Zs. Ulbert, T. Ruppert, G. Belvárdi, Z. Gyozsán, J. Abonyi, Interactive Balanced Scorecard based on Simulation of Production and Services Systems, Factory Automation 2012, 2012, 88-91.
- [11] Műszaki Kémiai Napok 2012, Veszprém 2012. április 24-26.
- [12] XXXI. Országos Tudományos Diákköri Konferencia, Kecskemét 2013. április 24-26.
- [13] WASET 2012, Amsterdam nemzetközi konferencia, 2012. július 26-27.
- [14] A Blom Consultancy leírásai nyomán (blomconsultancy.nl) 2013.10.12.
- [15] A Simtech Logisztika Kft. adatai szerint (simtech.hu) 2012.03.24.

Mellékletek

M1. Matlab forráskódok

```
%ExcelIn - Excel cellatartomány, formai megadása: 'b13:c29' - 'b2:b3'

function Start_simulation_v4(ExcelIn, ExcelOut)

    global exeFile simulFile runningTime excelFile excelCell_out excelworksheet waitingTime
    CostCells Fix h excelFile2;

%Simul8
    exeFile = 'Simul8_start_v5.exe';
    simulFile = 'd:\Dropbox\Ruppert\ITDK_2013\Program\Tobbvaltozos\Simulation_model1_v3.s8';
    runningTime = '2400';

%Excel
    excelFile = 'Options.xls';
    excelFile2 = 'd:\Dropbox\Ruppert\ITDK_2013\Program\Tobbvaltozos\Options.xls';
    excelCell_in = ExcelIn;
    excelCell_out = ExcelOut;
    excelworksheet = 'Simulation';
    h = actxserver('Excel.Application');
    waitingTime = 45;

%Fix költség számolása
    Fix = 0;
    CostCells = xlsread(excelFile, excelworksheet, excelCell_in);
    x0 = xlsread(excelFile, excelworksheet, excelCell_out);
    SizeCost = size(CostCells);
    for i = 1:SizeCost(1)
        Fix = Fix + CostCells(i,1);
    end

%Optimalizálás indítása
    [x,fval] = fminsearch(@Opt_sim_v4, x0);
end

function y = Opt_sim_v4(x)

global exeFile simulFile runningTime excelFile excelCell_out waitingTime CostCells Fix Check
excelworksheet h excelFile2;
x
%Excel-be irás
    writeInExcel_v3(excelCell_out,x);
    if (Check == 1)

%Simul8 futtatása
        Check = 0;
        winopen(excelFile);
        system([exeFile,' ',simulFile,' ',runningTime,' ',excelFile2,'
',num2str(waitingTime),' &']);
        pause(15);
        h.Workbooks.Item(excelFile).Save;
        h.Workbooks.Item(excelFile).Close;
        pause(30)

%KPI kimentése
        KPI = xlsread(excelFile, 'MonteCarlo', 'b1:b17');
        pause(5)
        SizeKPI = size(KPI);

%Valtozo költség számolása
        Cost = 0;
        HibaCost = 0;
        for i = 1:(SizeKPI(1) - 1)
            Cost = Cost + (CostCells(i,2) * KPI(i));
        end

%Hiba költség számítása
        HibaCells = xlsread(excelFile, excelworksheet, 'b2:b3');
        SizeHiba = size(HibaCells);
        for i = 1:(SizeHiba - 1)
            HibaCost = HibaCost + (1/HibaCells(i) * 100)
        end

%Bevetel számítása
        Revenue = CostCells(SizeKPI(1),2) * KPI(SizeKPI(1));

%Profit számolása
        y = Revenue - Cost - Fix - HibaCost
        y = y * (-1);
    end
end
```

[illegible]

M2. Visual Basic Kód

```
Imports System.Timers

Imports Microsoft.Office.Interop.Excel
Imports Microsoft.Office.Interop

Module Module1

    Private WithEvents mySimul8 As SIMUL8.S8Simulation
    Private counter As Integer
    Private ExcelFileName As String
    Private inFileName As String
    Private params() As String
    Private args() As String

    Dim waitEvent As System.Threading.AutoResetEvent = New
System.Threading.AutoResetEvent(False)

    Sub Main()
        args = System.Environment.GetCommandLineArgs()

        mySimul8 = GetObject("", "SIMUL8.S8Simulation")
        mySimul8.Open(args(1))
        ExcelFileName = args(3)
        Dim paramNames() As String = System.IO.File.ReadAllLines(args(1))

        Console.WriteLine("Run simulation")
        mySimul8.ResetSim(0)
        mySimul8.RunSim(CInt(args(2)))
        waitEvent.WaitOne()
        mySimul8.Close()
        mySimul8.Quit()
    End Sub

    Private Sub mySimul8_S8SimulationEndRun() Handles mySimul8.S8SimulationEndRun
        Dim app As New Excel.Application
        Dim Sheets As Excel.Worksheet
        Dim Sheets2 As Excel.Worksheet

        app.Visible = False 'Visible is False by default, so this isn't necessary
        Dim xl As Excel.Application
        xl = CreateObject("Excel.Application")

        Dim w As Workbook
        w = xl.Workbooks.Open(ExcelFileName)
        xl.Visible = False
        Sheets = w.Worksheets("MonteCarlo")
        Sheets2 = w.Worksheets("Simulation")
        Sheets2.Cells(9, 5).Value = mySimul8.ResultsCount
        For n = 1 To mySimul8.ResultsCount
            Sheets.Cells(n, 1).Value = mySimul8.Results(n).Name
            Sheets.Cells(n, 2).Value = mySimul8.Results(n).Value
        Next n
        w.Close(SaveChanges:=True)
        xl = Nothing
        app.Quit()
        app = Nothing
        waitEvent.Set()
    End Sub
End Module
```