

MENEDZSMENTDÖNTÉSEK TÁMOGATÁSA MUNKÁSOK KÜLÖNBÖZŐ KÉPZETTSÉGI SZINTJÉT FIGYELEMBE VEVŐ GYÁRTÓSOR-KIEGYENLÍTÉSI MODELLEKKEL

TATAY Viola¹, PhD-hallgató

Budapest Műszaki és Gazdaságtudományi Egyetem, Menedzsment és Vállalatgazdaságtan Tanszék

Dr. KOLTAI Tamás, Egyetemi tanár

Budapest Műszaki és Gazdaságtudományi Egyetem, Menedzsment és Vállalatgazdaságtan Tanszék

Absztrakt

A gyártósor-kiegyenlítés egymás után többször végrehajtandó, azonos tevékenységek szervezésével foglalkozik. A különböző műveletek végrehajtásának eredménye lehet egy termék (például járművek, elektronikai berendezések) vagy egy szolgáltatás (például egészségügyi szolgáltatás, ügyviteli feladatok végrehajtása). A gyártósor-kiegyenlítés célja a különböző tevékenységek munkahelyhez, munkáshoz rendelése úgy, hogy az bizonyos menedzsment céloknak a lehető legjobban megfeleljen. Ezt a hozzárendelést számos tényező befolyásolja, mint például a tevékenységek közötti logikai kapcsolatok, a tevékenységek végrehajtásának ideje, technológiai és minőségügyi előírások.

A tevékenységek munkahelyhez rendelésekor cél lehet a szükséges munkahelyszám minimalizálása, a ciklusidő minimalizálása vagy a hatékonyság maximalizálása. A különböző célú, optimális hozzárendelések matematikai programozási modellekkel egyszerűen meghatározhatóak. E tanulmány a gyártósor-kiegyenlítés modelljeit tekinti át és bemutatja a modellek menedzsment alkalmazási lehetőségeit. Kiemelten foglalkozik a dolgozók eltérő képzettségei miatt keletkező korlátoknak az optimális megoldásra kifejtett hatásával.

Abstract

Assembly line balancing (ALB) deals with the organization of identical tasks need to be performed several times one after the other. The result of the performance of the different activities can be a product (e.g. vehicles, electronic devices) or a service (e.g. health service, administrative processes). The objective of ALB is to assign these tasks to workstations and to workers considering several management objectives. This assignment can be influenced by several factors such as logical conditions between the activities, task time, technological and quality specifications.

The objective of the assignment of tasks to stations can be the minimization of the required number of workstations, the minimization of cycle time or the maximization of efficiency. Optimal assignments with different objectives can be determined with the help of mathematical programming models. This paper provides an overview of ALB models and shows, how the general conditions can be completed with workforce skill constraints.

¹ tatay@mvt.bme.hu

Bevezetés

A gyártósor-kiegyenlítés tevékenységek hatékony szervezésével foglalkozik. Ez a klasszikus termelési probléma mindenhol felmerül, ahol egymás után többször kell elvégezni ugyanazokat a feladatokat. Példaként említhető a különböző járművek (motorkerékpár, autó), elektronikai berendezések (hűtőszekrény, mobiltelefon) összeszerelése. Megjegyezzük, hogy gyakran szolgáltatások is kezelhetők gyártósorszerűen. Ilyen lehet például bizonyos egészségügyi szolgáltatások vagy ügyviteli feladatok szervezése. A gyártósor-kiegyenlítés célja a termék vagy szolgáltatás előállításához szükséges tevékenységek hatékony munkahelyhez rendelése. A cél mindig valamilyen gazdasági eredményhez kötött. Például minimalizálható a szükséges munkahelyszám, vagy a gyártás ciklusideje (Waters, 1996).

A feladatok munkahelyhez rendelését számos tényező befolyásolhatja. A ciklusidő, a tevékenységek közötti logikai kapcsolat, a gyártás technológiai és minőségi előírásai mind, mind hatással lehetnek a tevékenységek szervezésére. A különböző feltételeknek számos hozzárendelés megfelelhet. Az ilyen lehetséges megoldások közül matematikai programozási modellek segítségével kiválasztható egy valamilyen szempont szerinti legjobb megoldás. A gyártósor-kiegyenlítési probléma tipikusan bináris matematikai programozási modellekkel írható le. Ezek a modellek NP hard feladatok, így korábban fontos célkitűzés volt a modellek egyszerűsítése. A mai számítástechnikai és informatikai feltételek mellett viszonylag nagyméretű feladatok is könnyen megoldhatóak, így a kutatások fókuszába a modellek gyakorlati alkalmazása kerülhet.

A tanulmány célja a gyártósor-kiegyenlítés áttekintése és gyakorlati alkalmazásának elősegítése. Az egyszerű gyártósor-kiegyenlítési modellek menedzsment alkalmazási lehetőségeinek tárgyalása után részletesen bemutatjuk, hogy a munkások képzettségi szintje hogyan vehető figyelembe az optimális hozzárendelés meghatározásakor. Mindezeket egy egyszerű mintapélda segítségével szemléltetjük.

Gyártósor-kiegyenlítési modellek

A gyártósor-kiegyenlítési modellek két nagy csoportja különíthető el: az egyszerű és az általános gyártósor-kiegyenlítési modellek. A gyártósor-kiegyenlítéssel kapcsolatos kutatások jelentős része az egyszerű gyártósor-kiegyenlítési problémákkal (Simple Assembly Line Balancing Problem, továbbiakban rövidítve: SALBP) foglalkozik. Az SALB modellek alkalmazásakor a következő egyszerűsítő feltételezésekkel élünk (Becker and Scholl, 2006, Scholl and Becker, 2006):

- egy homogén termék tömegtermelését optimalizáljuk,
- a gyártási folyamat előre ismert, rögzített,
- a logikai kapcsolatokon kívül nincs más előírás a műveletek végrehajtására,
- a tevékenységidők determinisztikusak,
- minden tevékenység minden munkaállomáson elvégezhető,
- a cél a gyártósor hatékonyságának maximalizálása.

Amennyiben a probléma a fenti feltételezések közül valamelyiknek nem tesz eleget, akkor általános gyártósor-kiegyenlítési problémáról beszélünk (General Assembly Line Balancing Model, GALBM). Az általános gyártósor-kiegyenlítési probléma több ponton is eltérhet az egyszerűtől. Például a tevékenységidők lehetnek sztochasztikusak, egy tevékenységet nem lehet minden munkaállomáson elvégezni (mert például egy speciális eszköz, berendezés szükséges a művelet végrehajtásához), a munkahelyek követhetnek speciális (például U alakú) elrendezést. Az általánosított gyártósor-kiegyenlítési problémák nagymértékben különbözhetnek egymástól, attól függően, hogy az SALBP kikötések közül melyek sérülnek. A GALB problémák csoportosítását számos cikk tárgyalja (lásd például Becker-Scholl, 2006,

Boysen et al, 2006). GALB modellekkel a gyakorlati problémák jól leírhatóak. Ugyanakkor a GALBP modellek megoldása többnyire SALB modellekre vezethető vissza. Ebből fakadóan az egyszerű gyártósor-kiegyenlítési problémákkal kapcsolatos kutatások ma is fontos szerepet töltenek be. Mi a továbbiakban SALB modellekkel foglalkozunk.

Egy SALB modellel leírt gyártósor hatékonyságának növelése többféle matematikai programozási feladat segítségével megfogalmazható. Az SALBP modellek csoportosítását az 1. táblázat mutatja.

1. táblázat: Az egyszerű gyártósor-kiegyenlítési problémák osztályozása
Table 1. The classification of simple assembly line balancing models

		Ciklusidő (T_c)	
		paraméter	változó
Munkahelyek száma (m)	paraméter	SALBP-F	SALBP-2
	változó	SALBP-1	SALBP-E

Forrás: Scholl-Becker, 2006.

A klasszikus gyártósor-kiegyenlítési probléma SALBP-1 modellként került megfogalmazásra. Az SALBP-1 modellekben a ciklusidő (T_c) előre meghatározott, rögzített érték. Ekkor a cél a feladat végrehajtásához szükséges munkahelyek számának minimalizálása. A 2-es típusú gyártósor-kiegyenlítési problémában (SALBP-2) a munkahelyek száma előre ismert, rögzített. Itt a cél a ciklusidő minimalizálása, ami megegyezik a termelési ráta maximalizálásával (Baybars, 1986). A gyártósor-kiegyenlítési probléma eredetileg az 1-es modell szerint került megfogalmazásra (Bowman, 1960). A gyártósor-kiegyenlítés irodalmának bővülése, módszereinek fejlődése, eszköztárának kiszélesedése eredményeként jelent meg a 2-es típusú megközelítés.

Az SALBP-F modellek esetén a munkahelyek száma és a ciklusidő is előre rögzítve van. Ezzel a modellel arra a kérdésre kereshetjük a választ, hogy az adott idő alatt a rendelkezésre álló munkahelyekkel a feladat elvégezhető-e. Tehát az SALBP-F modelleknél csupán megengedett megoldást keresünk. SALBP-F modellek alkalmazása akkor fordul elő, amikor egy már kialakított rendszerben egy új feladat megvalósítását kell megvizsgálni. Az egyszerű gyártósor-kiegyenlítési feladat legáltalánosabb megfogalmazását az SALBP-E adja. Ez a munkahelyek számát és a ciklusidőt egyaránt minimalizáló feladat megoldását jelenti. Egy SALBP-E modell felírásakor előzetes ismeretek szükségesek a munkahelyek számára és a ciklusidőre vonatkozóan. Ebből adódóan egy SALBP-E feladat megoldását meg kell előznie egy SALBP-1 és egy SALBP-2 feladat megoldásának. Jelen dolgozat az SALBP-1 és SALBP-2 modellekre fókuszál.

A gyártósor-kiegyenlítés első analitikai felírása Bryton nevéhez fűződik. Salveson 1955-ben lineáris programozási feladatok megoldásával közelített a problémához. (Baybars, 1986). Azonban lineáris programozással a feladatok oszthatatlanságára vonatkozó feltétel nem kezelhető megfelelően. Az operációkutatás fejlődése lehetővé tette a gyártósor-kiegyenlítési feladatok megoldására alkalmasabb bináris modellek megoldását.

Az első bináris programozási modellt Bowman írta fel (Bowman, 1960). Bowman két egészértékű modellt is javasol. Az első modell változói egy adott állomáson egy adott művelet végrehajtásához szükséges időt jelölik. A célfüggvényben minden munkaállomáshoz exponenciálisan növekvő költség tartozik az elméletileg szükséges munkahelyszám fölött. Itt a munkahelyek számának minimalizálása a költségek minimalizálásával valósul meg. A második modell változói a tevékenységek végrehajtásának kezdési időpontját jelölik. A célfüggvény az utolsó tevékenység(ek) befejezési idejét minimalizálja, ami végső soron a feladat elvégzéséhez szükséges munkahelyszámot minimalizálja. White (1961) módosította

Bowman első modelljét és bevezette az x_{ij} bináris döntési változót, amelyet azóta is a legtöbb gyártósor-kiegyenlítési feladat tartalmaz (Patterson and Albracht, 1975).

Ezután a gyártósor-kiegyenlítéssel kapcsolatos kutatások a modellek, algoritmusok finomítására, tökéletesítésére fókuszáltak. Az egyszerű és a különböző általános modellek felírásakor elsősorban a megoldási algoritmus egyszerűsítésével foglalkoztak (lásd például Thangavelu és Shetty, 1971, Baybars, 1986). Ma már olyan matematikai programozási szoftverek állnak rendelkezésre, amelyek valós méretű gyártósor-kiegyenlítési modelleket is elhanyagolható időn belül képesek megoldani. Így ma a gyártósor-kiegyenlítési modellek gyakorlati alkalmazási lehetőségeivel kell foglalkoznunk.

A gyártósor-kiegyenlítési modellek gyakorlati alkalmazásának egy fontos kérdése lehet, hogy a modell képes-e figyelembe venni a gyártósoron dolgozók különböző képzettségét. A szakirodalomban kevés munka foglalkozik ezzel a problémával. Johnson (1983) – egy még a megoldási algoritmusokra fókuszáló cikkében – a feladatok különböző munkahelyen való végrehajtását vizsgálta. Egy bináris mátrixban összefoglalta, hogy mely tevékenységet mely munkahelyhez lehet hozzárendelni. Bár Johnson ezáltal két csoportra bontotta a feladatokat, a munkások képzettségbeli különbözőségét a modell nem vette figyelembe. Corominas és Plans (2008) egy motorkerékpár összeszerelő üzemben egy konkrét esetben foglalkozik a különböző munkásokkal. A cikkben kétféle munkást alkalmaznak: a régóta ott dolgozó, tapasztalt munkásokat és az átmeneti kapacitáshiányt orvosló, ideiglenesen ott dolgozókat. A modellben a kevésbé tapasztalt munkások miatt kétféle korlátozó feltételcsoport jelent meg. Egyrészt az új munkásoknak tovább tart a tevékenység elvégzése, ezért azon a munkahelyen, ahol ilyen munkás dolgozik nagyobb tevékenységidőt kell figyelembe venni. Másrészt az üzemben kikötötték, hogy egy új munkásnak egy régi mellett kell dolgoznia – a munka során felmerülő problémák mihamarabbi megoldása céljából. Ez a modell a munkások különböző képzettségét (vagy inkább jártasságát) ugyan figyelembe veszi, de a modell csak a konkrét esetre alkalmazható, általánosan nem fogalmaz meg képzettségi feltételeket. Nincs a szakirodalomban olyan munka, amely a gyártósor-kiegyenlítés matematikai modellezése során a munkások különböző képzettségének általános figyelembe vételével foglalkozik. Tanulmányunkban egy olyan módszert mutatunk be, amellyel bármely gyakorlati problémába beépíthető a munkások eltérő képzettségi szintje.

Az egyszerű gyártósor-kiegyenlítési modellek és az eredmények alapján hozható menedzsmentdöntések

A tanulmányban alkalmazott jelöléseket a 2. táblázat tartalmazza. A feladatokat sorszámokkal látjuk el, így minden feladat egy i indexet kap. Összesen I feladatot kell munkaállomáshoz rendelni. A munkaállomásokat j indexszel látjuk el. A felírt bináris programozási modellben J állomást veszünk figyelembe a hozzárendeléskor. A gyártósoron az aktuális munkahelyszámot N -nel jelöljük.

A tanulmányban a következő bináris matematikai programozási modellt alkalmazzuk a SALBP-1 megoldásához:

$$\text{Min}(N) \tag{1}$$

$$\sum_{i=1}^I t_i x_{ij} \leq T_c \quad j = 1, \dots, J \tag{2}$$

$$\sum_{j=1}^J x_{ij} = 1 \quad i = 1, \dots, I \tag{3}$$

$$\sum_{j=1}^J j \cdot (x_{qj} - x_{pj}) \geq 0, \quad (p, q) \in R \quad (4)$$

$$N \geq \sum_{j=1}^J (j \cdot x_{ij}) \quad i \in L \quad (5)$$

$$x_{ij} = 0 \quad j < LJ_i \text{ and } j > UJ_i \quad i = 1, \dots, I. \quad (6)$$

A modell előírja, hogy a menedzsment által meghatározott ciklusidőt egyik állomás sem lépheti túl (2), minden feladatot végre kell hajtani (3) és a feladatok hozzárendelésekor a precedencia gráf logikai kapcsolatait teljesíteni kell (4). Az (5) feltétel és az (1) célfüggvény együtt előírják a munkahelyszám minimalizálását.

1. táblázat: A jelölések összefoglalása
Table 2. Summary of notation

Indexek:	
i	= feladatok indexe ($i=1, \dots, I$),
k	= feladatok részhalmazának indexe,
j	= munkaállomások indexe ($j=1, \dots, J$).
Paraméterek:	
I	= feladatok száma,
J	= munkahelyek száma a matematikai modellben,
N	= munkahelyek tényleges száma,
R	= egymással precedencia kapcsolatban álló feladatok indexpárjainak halmaza, vagyis $(p; q) \in R$, ha p feladat közvetlenül megelőzi a q feladatot,
t_i	= i feladat végrehajtásához szükséges idő (tevékenységidő),
s_j	= j állomáshoz rendelt feladatok végrehajtásához szükséges idő (állomásidő),
T_c	= gyártósor ciklusideje,
T	= gyártásra rendelkezésre álló idő,
LJ_i	= a legkorábbi olyan munkahely, amelyhez i tevékenység a megelőző tevékenységek miatt hozzárendelhető,
UJ_i	= a legkésőbbi munkahely, amelyhez i tevékenység a követő műveletek miatt hozzárendelhető,
Q	= gyártási mennyiség,
c_j	= j állomás kapacitás-kihasználtsága,
W	= speciális munkások száma,
z	= kellően nagy szám.
Halmazok:	
O	= az összes feladat halmaza,
L	= utolsó feladatok halmaza, tehát $i \in L$ ha i feladat nem előz meg egyetlen más feladatot sem,
P_i	= azon feladatok halmaza, amelyeket be kell fejezni mielőtt az i tevékenységet elkezdjük,
S_i	= azon feladatok halmaza, amelyek nem kezdhetőek addig el, amíg az i tevékenység nincs befejezve,
S	= speciális feladatok halmaza,
\bar{S}	= nem speciális feladatok halmaza (S halmaz komplementere).
Döntési változók:	
x_{ij}	= 0-1 döntési változó; $x_{ij}=1$, ha az i feladatot a j munkahelyhez rendeljük, $x_{ij}=0$ különben,
l_j	= 0-1 döntési változó; $l_j=1$, ha a j munkahelyhez alacsony képzettségű munkást rendelünk, $l_j=0$ különben,
h_j	= 0-1 döntési változó; $h_j=1$, ha a j munkahelyhez magasan képzett munkást rendelünk, $h_j=0$ különben,
e_j	= 0-1 döntési változó; $e_j=1$, ha a j , munkahelyen speciális munkást alkalmazunk, $e_j=0$ különben.

Megjegyezzük, hogy az (1)-(4) modell több utolsó tevékenység esetén is használható. Amennyiben csak egyetlen utolsó tevékenysége van a gyártósor-kiegyenlítési problémának, akkor elegendő az utolsó feladat munkahelyének sorszámát minimalizálni, tehát (1) és (5) a következő célfüggvénnyel helyettesíthető,

$$\text{Min} \left(\sum_{j=1}^J x_{c,j} \right), \quad (7)$$

ahol c az utolsó feladat sorszámát jelöli. A változók száma (6) segítségével csökkenthető, ahol

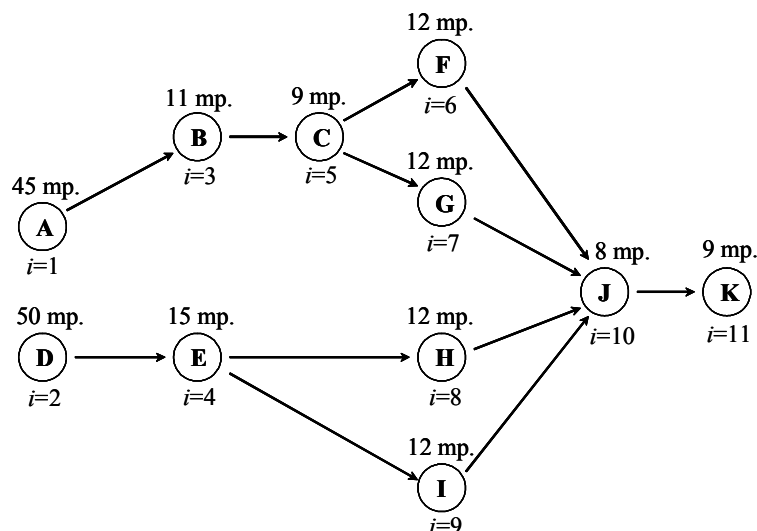
$$LJ_i = \left\lceil \frac{t_i + \sum_{k \in P_i} t_k}{T_c} \right\rceil, \quad (8)$$

$$UJ_i = J + 1 - \left\lfloor \frac{t_i + \sum_{k \in S_i} t_k}{T_c} \right\rfloor. \quad (9)$$

LJ_i meghatározza azt a legkorábbi munkahelyet, amelyhez az i tevékenység hozzárendelhető. A precedencia kapcsolatok és a ciklusidő miatt ugyanis kizárható az i tevékenység nagyon korai munkahelyhez rendelése. Hasonlóan UJ_i meghatározza azt az utolsó munkahelyet, amelyhez i tevékenység még hozzárendelhető. Az így felírt modell összesen

$$\sum_{j=1}^J (UJ_j + 1 - LJ_j) \quad (10)$$

változót tartalmaz. Több utolsó feladatot tartalmazó gyártósor-kiegyenlítési feladat esetén általában egy minden utolsó feladatot követő látszattevékenységet vezetnek be (Talbot and Patterson, 1984). A cél ilyenkor a látszattevékenységet végrehajtó munkaállomás sorszámának minimalizálása. Az új tevékenység bevezetése miatt $I+1$ új bináris változó keletkezik. Az (1)-(6) modellben az utolsó tevékenységek sorszámát, mint súlyszámokat alkalmazzuk. Így esetünkben csak egyetlen új egészértékű változó (N) bevezetésére van szükség. Az SALB-1 modell megoldásának szemléltetésére vegyük az 1. ábrán látható mintapéldát (Koltai, 2003).



1. ábra: A mintapélda precedencia gráfja

Figure 1. Precedence graph of the sample problem

Az SALB-1 modellben – ismert ciklusidő mellett – cél a tevékenységeket oly módon rendelni munkahelyekhez, hogy a lehető legkevesebb munkahelyre legyen szükség.

Esetünkben egyetlen 7 órás műszak alatt 500 terméket kell előállítani, tehát a ciklusidő a következőképpen számítható,

$$T_c = \frac{T}{Q} = \frac{7 \cdot 60 \cdot 60}{500} = 50,4 \text{ mp/darab} . \quad (11)$$

Ahhoz tehát, hogy napi 500 darab termék készüljön el, egyenletes ütemben 50,4 másodpercenként egy termék összeszerelését be kell fejezni. A probléma optimális megoldását a 3. táblázat foglalja össze. Láthatjuk, hogy az optimum értéke 4, vagyis e paraméterek esetén legalább 4 munkahelyre van szükség.

3. táblázat: A mintapélda optimális megoldása az SALBP-1 modellel

Table 3. The optimal solution of the sample problem with SALBP-1

Modell	Opt.	Munkahelyek			
		1	2	3	4
SALBM-1	4	D	A	B,E,H,I	C,F,G,J,K

Az SALBP-1 megoldásával meghatározható, hogy egy termék összeszereléséhez minimálisan hány munkahelyre van szükség. A minimális erőforrás-szükséglet összefüggésben van a termék komplexitásával. A gyakorlatban különböző termékek gyártásának ütemezésekor a komplexitást gyakran figyelembe veszik és az azonos összetettségi csoportba sorolható termékek gyártását egymás után hajtják végre. Így az SALBP-1 megoldásával képet kaphat egy vállalat arról, hogy egy új termék mely már meglévő termékek gyártásához hasonlít leginkább.

A tanulmányban az SALBP-2 megoldásakor a következő bináris programozási feladatot alkalmazzuk:

$$\text{Min } T_c \quad (12)$$

$$\sum_{i=1}^I t_i x_{ij} \leq T_c \quad j = 1, \dots, J \quad (13)$$

$$\sum_{j=1}^J x_{ij} = 1 \quad i = 1, \dots, I \quad (14)$$

$$\sum_{j=1}^J j \cdot (x_{qj} - x_{pj}) \geq 0 \quad (p, q) \in R \quad (15)$$

$$x_{ij} = 0 \quad \begin{array}{l} j < LJ_i \\ j > UJ_i \\ i = 1, \dots, I \end{array} \quad (16)$$

A (12)-(16) modell megoldásakor a cél adott munkahelyszám mellett a ciklusidő minimalizálása (12), ami a termelékenység maximalizálását eredményezi. A (13), (14) és (15) feltételek az SALBP-1 modell (2), (3) és (4) feltételeivel megegyezően a ciklusidőre, a feladatok végrehajtására és a logikai kapcsolatokra vonatkozó korlátok. A változók számának csökkentését a (16) feltétel írja elő, ahol

$$LJ_i = \left\lceil \frac{t_i + \sum_{k \in P_i} t_k}{UB(T_c)} \right\rceil, \quad (17)$$

$$UJ_i = J + 1 - \left\lfloor \frac{t_i + \sum_{k \in S_i} t_k}{UB(T_c)} \right\rfloor. \quad (18)$$

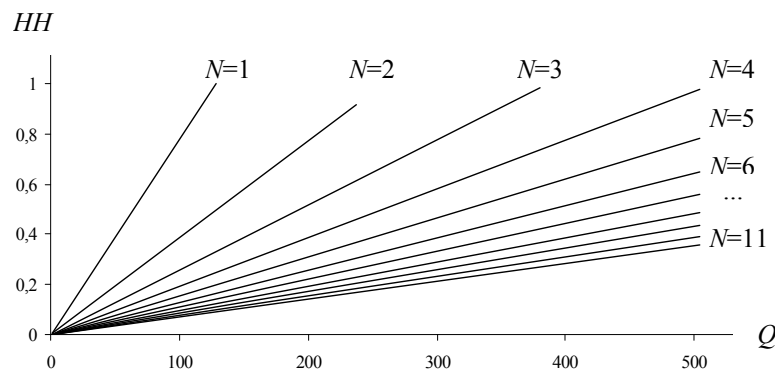
(17) és (18) – a (8)-hoz és (9)-hez hasonlóan – a ciklusidő egy felső becslésével ($UB(T_c)$) meghatározza azon munkahelyek sorszámát, amelyhez az i tevékenység legkorábban illetve legkésőbb rendelhető. Az SALBP-2 szemléltetésére tekintünk az 1. ábrán látható mintafeladatot optimális megoldását 6 munkahely esetén. A megoldást a 4. táblázat foglalja össze.

4. táblázat: A mintapélda optimális megoldása az SALBP-2 modellel
Table 4. The optimal solution of the sample problem with SALBP-2

Modell	Opt.	Munkahelyek					
		1	2	3	4	5	6
SALBM-2	50	D	-	A	B,E	C,G,H	F,I,J,K

A 4. táblázatban látható, hogy az SALBP-2 modell optimális megoldása 50 másodperc. Ez a lehető legrövidebb ciklusidő, amellyel a termelési feladat 6 munkahely esetén elvégezhető. Azt is észrevehetjük, hogy valójában csak 5 munkahelyet használunk a 6-ból, hiszen a 2-es munkahelyhez nem rendelünk feladatot. Ez azt jelenti, hogy az optimum értéke nem javítható úgy, hogy egy olyan munkahelyről, ahol több tevékenység van egy vagy néhány tevékenységet átcsoportosítunk erre az üres munkahelyre. Ez azért van így, mert az optimális ciklusidő megegyezik a D feladata tevékenységidejével. Ez a tevékenység a rendszer szűk keresztmetszete. Hiába használnánk 5 vagy annál több munkahelyet, az optimum ezáltal nem javítható.

Az SALBP-2 eredményei segítségével különböző hatékonyság- és érzékenységvizsgálati számítások végezhetőek, amelyekkel fontos, a gyártósorokkal kapcsolatos termelésmenedzsment döntések támogathatóak. Az ilyen jellegű vizsgálatokhoz a 2. ábra grafikus segítséget nyújthat. Az ábra a hozzárendelés hatékonyságát (HH) mutatja a gyártási mennyiség (Q) és a munkahelyszám (N) függvényében. Különböző munkahelyszámok esetén megoldva az SALBP-2 modellt megkapható minden munkahely hatékonysági függvénye.



2. ábra: A hozzárendelés hatékonyságának vizsgálata SALBP-2 modellek eredményeivel
Figure 2. Analysis of the assignment efficiency with the help of SALBP-2 model results

Munkások különböző képzettségének figyelembe vétele

Gyakran előfordul, hogy különböző feladatok végrehajtása speciális képzettséget igényel és ilyen speciális képzettséggel nem rendelkezik minden munkás. A feladatok munkahelyhez és munkáshoz rendelésekor figyelembe kell vennünk, ha egy munkás valamely feladatot nem tud ellátni, vagy épp ellenkezőleg, csak ő képes ellátni. A munkások különböző képzettségi szintjét a következőkben három modell segítségével mutatjuk be. Mindhárom modellben definiálunk egy S speciális halmazt, amelybe olyan feladatok tartoznak, amelyek valamilyen speciális tulajdonsággal bírnak. Ez a speciális tulajdonság az egyes modellek esetében eltér egymástól. A speciális tulajdonsággal nem rendelkező feladatok az S halmaz komplementer halmazának (\bar{S}) elemei. Természetesen egy feladat egyszerre csak egy halmazhoz tartozhat, vagy S -hez, vagy \bar{S} -hez. A képzettségi szint optimális megoldásra gyakorolt hatását a következő három esetben vizsgáljuk:

- *Alacsony képzettségi szinttel rendelkező munkások alkalmazása a gyártósoron.* Ebben az esetben a munkások két típusát különböztetjük meg egymástól. A hagyományos képzettséggel bíró munkások bármilyen feladat elvégzésére képesek. Ezekhez a munkásokhoz bármilyen feladat hozzárendelhető, speciális és nem speciális feladat egyaránt. Az alacsony képzettségű munkások viszont csak az egyszerűbb, speciális feladatokat tudják ellátni. Tehát az S részhalmazba olyan egyszerű feladatok tartoznak, amelyek alacsonyabb képzettség birtokában is elvégezhetők. Ezt a szituációt az alacsony képzettségi szintet megfogalmazó korlátozó feltételek hozzáadásával írjuk le (angolul: Low Skill Constraint, továbbiakban rövidítve: LSC).
- *Magasan képzett munkások alkalmazása a gyártósoron.* Ebben az esetben a speciális, bonyolult feladatok ellátására csak a magasan képzett munkások képesek. A nem magas képzettségi szinttel rendelkező munkások csak a nem speciális feladatokat tudják elvégezni. A magasan képzett munkások a speciális (bonyolult) és nem speciális feladatokat is meg tudják csinálni. Tehát a feladatok S részhalmazába olyan tevékenységek tartoznak, amelyek bonyolultak, ellátásuk speciális szakértelmet, magasan képzett munkást igényel. Ezt a szituációt a magas képzettségi szintet megfogalmazó korlátozó feltételek hozzáadásával modellezzük (angolul: High Skill Constraint, továbbiakban rövidítve: HSC).
- *Speciális képzettségű munkások alkalmazása speciális feladatok ellátására.* Ilyenkor a feladatok és a munkások halmaza is két jól elkülöníthető csoportba oszthatók. Vannak speciális (S) és nem speciális feladatok (\bar{S}), speciális képzettséggel rendelkező és speciális képzettséggel nem rendelkező munkások. A speciális feladatot csak speciális munkás, a nem speciális feladatot csak nem speciális munkás végezheti el. Vagyis ebben az esetben a különböző feladattípusok nem keveredhetnek össze az egyes munkahelyeken és adott feladattípushoz kizárólag egyfajta munkás rendelhető. Ezt a szituációt a speciális képzettséget megfogalmazó korlátozó feltételek hozzáadásával írjuk le (angolul: Executive Skill Constraint, továbbiakban rövidítve: ESC).

Alacsony képzettségi szint figyelembe vétele (LSC)

Ebben az esetben a vállalat rendelkezik egy vagy néhány olyan alacsony képzettségű munkással, aki(k) csak egyszerűbb feladatok ellátására képes(ek). Az ilyen munkásoknak is munkát kell adni, ezért a feladatok munkahelyhez, munkáshoz rendelésekor ezt figyelembe kell venni. Az egyszerű feladatokat speciális feladatként definiáljuk, és az alacsony képzettségű munkásokhoz csak ilyen speciális feladatok rendelhetőek hozzá. A nem alacsony képzettségű munkás bármilyen feladatot meg tud csinálni. Az LSC feltételek leírásához bevezetjük az I_j bináris változót:

$$l_j = \begin{cases} 1 & \text{ha } j \text{ munkahelyhez alacsony képzettségű munkást rendelünk,} \\ 0 & \text{különben.} \end{cases}$$

Ha alacsony képzettségű munkás dolgozik a j munkahelyen, akkor ehhez a munkahelyhez, csak a feladatok S részhalmazából rendelhető feladat. Ugyanakkor egy olyan munkahelyhez, amelyhez nem alacsony képzettségű munkást rendelünk bármilyen feladat hozzárendelhető. Ezt a szituációt a következő egyenlőtlenségek segítségével fogalmazhatjuk meg:

$$\sum_{i \in S} x_{ij} \leq z(1-l_j) \quad j=1, \dots, J, \quad (19)$$

$$\sum_{i=1}^n x_{ij} \geq l_j \quad j=1, \dots, J, \quad (20)$$

$$\sum_{j=1}^m l_j \geq W, \quad (21)$$

ahol z egy elegendően nagy szám, W pedig az alacsony képzettségű munkások minimális számát adja meg. Ha $l_j=1$, vagyis a j munkahelyhez alacsony képzettségű munkást rendelünk, akkor a (19) egyenlőtlenség jobboldala zéró. Ekkor a feltétel csak úgy teljesül, ha a (19) baloldala is zéró, vagyis nem rendelünk ehhez a munkahelyhez nem speciális feladatot. Ha $l_j=0$, vagyis a j munkahelyhez nem rendelünk alacsony képzettségű munkást, akkor (19) jobboldala pozitív. Ebben az esetben (19) baloldala lehet pozitív és zéró is. $l_j=0$ két esetben fordulhat elő: vagy rendelünk az adott munkahelyhez munkást és az ő képzettségi szintje nem alacsony, vagy ehhez a munkahelyhez nem is rendelünk munkást. Azt szeretnénk elérni, hogy ha egy munkahelyhez nem rendelünk feladatot, akkor oda munkást se rendeljünk. Ezt a (20) egyenlőtlenség írja elő. (20) baloldala zéró, ha a j munkahelyhez nem rendelünk feladatot. Az egyenlőtlenség csak úgy teljesülhet, ha $l_j=0$, vagyis egy olyan munkahelyhez ahova nem rendelünk feladatot, nem rendelhetünk speciális munkást sem. Azt, hogy a rendelkezésre álló összes alacsony képzettségű munkást alkalmazzuk (21) írja elő.

Tehát ha az SALBP-1 vagy SALBP-2 modelleket kiegészítjük a (19)-(21) feltételekkel, akkor az alacsony képzettségű munkások csak speciális – ebben az esetben egyszerű, alacsony képzettségi szintet igénylő – feladatot, míg a nem alacsony képzettségi szinttel rendelkező – általános – munkások bármilyen feladatot elvégeznek. Az így kapott modellek személtetésére oldjuk meg az 1. ábrán látható gyártósor-kiegyenlítési mintapéldát. Az eredményeket az 5. táblázat foglalja össze. A táblázat első oszlopa mutatja, hogy mely modellre vonatkoznak az adott sor eredményei. A második oszlopban a bináris programozási feladat optimumát tüntettük fel. Ez SALBP-1 esetében a minimális munkahelyszámot, SALBP-2 esetében a minimális ciklusidőt jelenti. A harmadik oszlopban a speciális, alacsony képzettségű munkások száma (sz) és a speciális tevékenység(ek) ($tev(k)$) kerültek feltüntetésre. A következő oszlopokban a feladatok munkahelyhez rendelése látható. A táblázatban a speciális munkahelyeknél a cellát árnyékkoltuk, a speciális tevékenységeket vastagon szedtük.

Mindkét alapproblémánál (SALBP-1 és SALBP-2) feltüntettük az alacsony képzettségi szint figyelembe vétele nélküli optimális megoldásokat is. A második SALBP-1 modellnél előírtuk, hogy egyetlen alacsony képzettségű munkást kell alkalmazni a feladatok elvégzésekor és a J tevékenységet speciális tevékenységként definiáltuk. Láthatjuk, hogy az egyetlen speciális feladatot az alacsony képzettségű munkához rendeltük hozzá. Láthatjuk továbbá, hogy a képzettségi szint figyelembe vétele 6-ra növelte a minimális munkahelyek számát. A következő modellben ugyanúgy egyetlen alacsony képzettségű munkást alkalmaztunk, de a J mellett a C tevékenységet is speciális tevékenységként adtunk meg.

Ekkor a C tevékenységet rendeltük az alacsony képzettségű munkáshoz, a J tevékenységet pedig egy nem alacsony képzettségű munkáshoz. A két feladatot a precedencia kapcsolatok miatt nem lehet egy munkahelyhez rendelni. Érdekes, hogy itt több a speciális tevékenységek száma, mint az előző esetben, de az optimum kedvezőbb értéket vesz fel (5). Az utolsó SALBP-1 modellben megtartottuk speciális tevékenységnek a C és J tevékenységet, de 2 alacsony képzettségű munkás alkalmazását írtuk elő. Látható, hogy a speciális tevékenységeket két különböző, alacsony képzettségű munkás hajtja végre, továbbá a minimális munkahelyek száma 7-re nőtt. Az SALBP-2 modelleknél hasonló eredményekre jutottunk, azzal a különbséggel, hogy az alacsony képzettségi szinttel kapcsolatos előírások nem változtatták meg az alapmodell 50 másodperces optimális ciklusidejét.

**5. táblázat: Az alacsony képzettségi szint figyelembevétele
SALBP-1 és SALBP-2 modellekben**

Table 5. Consideration of LSC in SALBP-1 and SALBP-2 models

Modell	Opt.	Spec.tev. (sz;tev(k))	Munkahelyek						
			1	2	3	4	5	6	7
SALBP-1	4	-	D	A	B,E,H,I	C,F,G,J,K	-	-	-
SALBP-1	6	1; J	D	A	B,C,E	F,G,H,I	J	K	-
SALBP-1	5	1; C,J	D	A	B,E,H,I	C	F,G,J,K	-	-
SALBP-1	7	2; C,J	D	A	B,E	C	F,G,H,I	J	K
SALBP-2	50	-	D	-	A	B,E	C,G,H	F,I,J,K	-
SALBP-2	50	1; F	A	D	B,E	C,H	F	G,I,J,K	-
SALBP-2	50	1; F,E	A	D	B	E	C,F,H,I	G,J,K	-
SALBP-2	50	2; F,E	A	D	B,C,G	E	F	H,I,J,K	-

Magas képzettségi szint figyelembe vétele (HSC)

Ebben az esetben a vállalat rendelkezik egy vagy néhány olyan magasan képzett munkással, akik az általános feladatokon túl a bonyolultabb, speciális feladatok ellátására is képesek. Feltételezzük, hogy az ilyen munkások száma maximálva van. Ilyenkor tehát a speciális feladatok S halmazába a magas képzettségi szintet igénylő feladatok kerülnek. A magas képzettségi szinttel rendelkező munkások minden feladatot meg tudnak csinálni: speciális és nem speciális feladatot egyaránt. A magas képzettségi szinttel nem rendelkezők csak nem speciális tevékenységet végezhetnek. Megjegyezzük, hogy a HSC modell az alacsony képzettségi szintű (LSC) eset duálisaként is felfogható. Az LSC modellben az alacsony képzettségűek csak a speciális feladatokat tudják ellátni – amit kevésbé komplikált, egyszerűbb feladatként definiáltunk –, a nem alacsony képzettségi szinttel rendelkezők bármilyen feladatot el tudnak végezni. A HSC modellben a magas képzettségi szinttel nem rendelkezők csak a nem speciális feladatokat tudják megcsinálni, akik magas képzettségi szinttel rendelkeznek, bármilyen feladatot meg tudnak oldani: speciális – ebben az esetben bonyolult – és nem speciális feladatot egyaránt. Az LSC modellben az alacsony képzettségű munkások számára alsó korlátot, a HSC modellben a magasan képzett munkások számára felső korlátot adunk meg.

A speciális feladatokat (S) csak a magasan képzett munkások végezhetik. Azok, akik nem rendelkeznek magas képzettségi szinttel, csak a nem speciális feladatokat (\bar{S}) tudják ellátni. A HSC feltételek leírásához bevezetjük a h_j bináris változót:

$$h_j = \begin{cases} 1 & \text{ha } j \text{ munkahelyhez magasan képzett munkást rendelünk,} \\ 0 & \text{különben.} \end{cases}$$

Azon munkahelyekhez, ahova magas képzettségi szinttel rendelkező munkást rendelünk, bármely feladat hozzárendelhető. Azokon a munkahelyeken, ahol nem magasan képzett munkások dolgoznak, csak nem speciális feladatok fordulhatnak elő. Ezt az esetet következő feltételekkel írhatjuk elő.

$$\sum_{i \in S} x_{ij} \leq zh_j \quad j = 1, \dots, J, \quad (22)$$

$$\sum_{i=1}^n x_{ij} \geq h_j \quad j = 1, \dots, J, \quad (23)$$

$$\sum_{j=1}^m h_j \leq W, \quad (24)$$

ahol z egy elegendően nagy szám, W pedig a magas képzettségi szinttel rendelkező munkások maximális száma. Ha j munkahelyhez rendelünk legalább egy speciális tevékenységet, akkor a (22) feltétel baloldala pozitív. Ekkor a (22) feltétel csak úgy teljesülhet, ha magas képzettségű munkást rendelünk a j munkahelyhez, vagyis $h_j=1$. Ha (22) baloldala zéró – vagyis az adott munkahelyhez nem rendelünk speciális feladatot –, akkor h_j értéke lehet 0 és 1 is. Amennyiben a j munkahelyhez nem rendelünk speciális feladatot, akkor vagy nem rendelünk semmilyen feladatot a j munkahelyhez, vagy csak nem speciális feladat van azon a munkahelyen. (23) segítségével előírható, hogy magas képzettségű munkást csak ahhoz a munkahelyhez rendeljünk, ahol van valamilyen feladat. A (24) maximálja a magas képzettségű munkások számát.

**6. táblázat: A magas képzettségi szint figyelembevétele
SALBP-1 és SALBP-2 modellekben**

Table 6. Consideration of HSC in SALBP-1 and SALBP-2 models

Modell	Opt.	Spec.tev. (sz;tev(k))	Munkahelyek					
			1	2	3	4	5	6
SALBP-1	4	-	D	A	B,E,H,I	C,F,G,J,K	-	-
SALBP-1	4	1; H	D	A	B,E,H,I	C,F,G,J,K	-	-
SALBP-1	5	1; H,K	A	D	B,C,E,G	F,I	H,J,K	
SALBP-1	4	2; H,K	D	A	B,E,H,I	C,F,G,J,K	-	-
SALBP-2	50	-	D	-	A	B,E	C,G,H	F,I,J,K
SALBP-2	50	1; I	D	A	B,C,F	E	I	G,H,J,K
SALBP-2	50	1, I,F	A	D	B	C,G	E,F,I	H,J,K
SALBP-2	89	1; I,F,A	D	E,H	-	-	A,B,C,F,I	G,J,K

A HSC modell szemléltetésére tekintjük a mintafeladat kiegészítését a (22)-(24) feltételekkel. Az eredményeket a 6. táblázatban foglaltuk össze. A táblázat az alkalmazott modellt, az optimum értékét, a magas képzettségi szinttel rendelkező munkások számát, a speciális feladatokat és az optimális hozzárendelést tartalmazza. A második SALBP-1 modellnél láthatjuk, hogy amennyiben egyetlen magas képzettségű munkásunk van és a H tevékenységet speciális tevékenységként definiáljuk, akkor a magas képzettségi szintet is figyelembe vevő modell megoldása nem változik az eredeti feladat megoldásához képest. Ez azért van így, mert a magas képzettségű munkás el tudja látni a H speciális feladatot is. Amennyiben a H tevékenység mellett a K tevékenység is speciális (harmadik SALBP-1 modell) – és továbbra is egyetlen magasan képzett munkásunk van –, akkor a minimális munkahelyszám 5-re nő. De ha 2 magas képzettségű munkást tudunk alkalmazni erre a két speciális feladatra (negyedik SALBP-1 modell), akkor a modell megoldása megegyezik az alap SALBP-1 modell megoldásával. Az SALBP-2 modellek esetében láthatjuk, hogy az első

három modellben az optimális ciklusidő ugyanaz. Hiába írjuk elő a speciális feladatokat és hogy van magas képzettségű munkásunk, az optimális ciklusidő nem csökken 50 másodperc alá (a szűk keresztmetszetet alkotó D tevékenység miatt). Az utolsó SALBP-2 modellben az egyetlen magasan képzett munkásnak kell ellátnia 3 speciális feladatot. Ez a precedencia kapcsolatok miatt azt eredményezi, hogy az optimális ciklusidő 89 másodpercre nő.

Speciális képzettség figyelembe vétele (ESC)

Előfordulhat, hogy vannak a vállalatnál olyan – speciális – munkások, akik csak a speciális feladatok ellátására képesek, a nem speciális feladatokat nem tudják ellátni. A nem speciális munkások pedig kizárólag a nem speciális tevékenységeket tudják elvégezni. Tehát ebben az esetben speciális feladatot csak speciális munkás, nem speciális feladatot csak nem speciális munkás végezhet. Az ESC feltételek leírásához bevezetjük az e_j bináris változót:

$$e_j = \begin{cases} 1 & \text{ha } j \text{ munkahelyhez speciális képzettségű munkást rendelünk,} \\ 0 & \text{különben.} \end{cases}$$

Azon a munkahelyen, ahol speciális tevékenység van, csak speciális munkás dolgozhat. Nem speciális feladatot csak nem speciális munkás végezhet. A feladatok és tevékenységek ily módon történő elkülönítését a következő feltételekkel érhetjük el:

$$\sum_{i \in S} x_{ij} \leq z e_j \quad j = 1, \dots, J, \quad (25)$$

$$\sum_{i \in S} x_{ij} \leq z(1 - e_j) \quad j = 1, \dots, J, \quad (26)$$

ahol z egy elegendően nagy szám. Ha speciális tevékenységet rendelünk a j munkahelyhez, akkor (25) baloldala nagyobb mint zéró. Ekkor (25) feltétel csak úgy teljesülhet, ha $e_j=1$, vagyis speciális munkást rendelünk a j munkahelyhez. Ha nem speciális feladatot rendelünk a j munkahelyhez, akkor (26) baloldala nagyobb mint zéró. Ahhoz, hogy (26) teljesüljön, a jobboldalnak is pozitívnak kell lennie. Ekkor $e_j=0$, vagyis nem rendelünk speciális munkást a j munkahelyhez.

7. táblázat: A speciális képzettségi szint figyelembevétele SALBP-1 és SALBP-2 modellekben

Table 5. Consideration of ESC in SALBP-1 and SALBP-2 models

Modell	Opt.	Spec.tev.	Munkahelyek					
			1	2	3	4	5	6
SALBP-1	4	-	D	A	B,E,H,I	C,F,G,J,K	-	-
SALBP-1	5	G	D	A	B,C,E,I	G	F,H,J,K	-
SALBP-1	5	G,D	A	D	B,C,E,F	G	H,I,J,K	-
SALBP-1	6	B,D,G,J,K	D	E	A	B	C,F,H,I	G,J,K
SALBP-2	50	-	D	-	A	B,E	C,G,H	F,I,J,K
SALBP-2	50	K	A	B,C	D	E,H	F,G,I,J	K
SALBP-2	50	K,B	D	A	B	C,E,H,I	F,G,J	K
SALBP-2	61	K,B,D,F,H	A	B,D	C,E,G,I	F,H	J	K

Az ESC modell szemléltetésére tekintünk a 7. táblázatban összefoglalt mintapéldák megoldásait. A táblázat az 5. és 6. táblázathoz hasonlóan foglalja össze az adatokat, azzal a különbséggel, hogy ebben a modellben nincs a munkások számára vonatkozó előírás. A

második SALBP-1 modellben G tevékenység – ami eredetileg több más tevékenységgel együtt került végrehajtásra – speciális. A táblázatból látható, hogy a minimális munkahelyszám 5-re nőtt, mert G tevékenységnek külön munkahelyet kell biztosítani. Ahogy növeljük a speciális feladatok számát, úgy nő a speciális munkások iránti igény is. Az utolsó SALBP-1 modellben az 5 speciális tevékenység miatt 6-ra nő a minimális munkahelyszám. Az SALBP-2 modellben az 50 másodperces ciklusidő csak több speciális tevékenység megadásánál romlik. A táblázat utolsó sorában feltüntetett esetben az 5 speciális feladat miatt 3 speciális munkahelyre van szükség, ami 61 másodpercre növeli az optimális ciklusidőt.

Összefoglalás

A tanulmányban ismertettük a gyártósor-kiegyenlítési modelleket. Részletesen tárgyaltuk az SALBP-1 és SALBP-2 modelleket és rávilágítottunk azok gyakorlati jelentőségére. Az SALBP-1 modell segítségével a gyártás erőforrásigényéről kaphatunk információt. Az SALBP-2 modell eredményei alapján különböző hatékonysági számítások végezhetőek.

A tanulmányban bemutattuk, hogy a munkások különböző képzettségi szintje hogyan vehető figyelembe a gyártósor-kiegyenlítési modellekben. Három különböző esetet ismertettünk részletesen. Az első esetben az alacsony képzettségi szinttel rendelkező munkásokhoz csak az egyszerűbb feladatok rendelhetőek. A második esetben a magas képzettségi szinttel rendelkező munkások tudtak csak minden tevékenységet végrehajtani: az egyszerűbb és a bonyolultabb feladatokat is. Az utolsó esetben szigorúan elkülönítettük a speciális feladatokat a nem speciálisaktól és a speciális munkásokat a nem speciálisaktól. Az SALB modellek gyakorlati jelentőségét és a munkások különböző képzettségi szintjeit figyelembe vevő modellek leírását egy egyszerű mintapéldával szemléltettük. A kis esettanulmány segítségével beláttuk, hogy a képzettségi szintet figyelembe vevő modellek megoldása az eredeti (SALBP-1 vagy SALBP-2) modell megoldásánál sosem lehet jobb.

Természetesen a bemutatott képzettségi szinteket leíró korlátozó feltételek tetszés szerint kombinálhatóak és építhetőek be bármilyen gyártósor-kiegyenlítési modellbe. A munkások eltérő képzettségi szintjének figyelembe vételével a gyakorlat igényeit jobban kielégítő modellek megfogalmazását teszi lehetővé.

A tanulmányban egyszerű képzettségi szinteket definiáltunk, speciális és nem speciális feladatokkal. A valóságban előfordulhat, hogy nem kettő, hanem több csoportba sorolhatóak a munkások (képzettség szerint) és a tevékenységek (specialitás szerint). A jövőbeli kutatásaink fókuszába a többszintű tevékenységek és képzettségi szintek vizsgálata és modellezhetősége áll.

Irodalomjegyzék:

1. Baybars, Í. (1986): A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem, *Management Science*, Vol. 32, pp. 909-932
2. Becker, C. and Scholl A. (2006): A Survey on Problems and Methods in Generalized Assembly Line Balancing, *European Journal of Operational Research*, Vol. 168, pp. 694-715
3. Boysen N., Flidner M. and Scholl A. (2008): Assembly Line Balancing: Which Model to Use When?, *International Journal of Production Economics*, Vol. 111, pp. 509-528
4. Bowman E. H. (1960): Assembly Line Balancing by Linear Programming, *Operations Research*, Vol. 8, pp. 385-389
5. Corominas A, Pastor F. and Plans, J. (2008): Balancing Assembly Line with Skilled and Unskilled Workers, *Omega*, Vol. 36, pp. 1126-1132

6. Johnson, R. V., (1983): A Branch and Bound Algorithm for Assembly Line Balancing Problems with Formulation Irregularities, *Management Science*, Vol. 29, pp. 1309-1324
7. Koltai T. (2003): A termelésmenedzsment alapjai II, *Műegyetemi Kiadó*
8. Patterson, J. H. and Albracht J. J. (1975): Assembly-Line Balancing: Zero-One Programming with Fibonacci Search, *Operations Research*, Vol. 23, pp. 166-174
9. Scholl, A. and Becker (2006): State-of-art Exact and Heuristic Soution Procedures for Simple Assembly Line Balancing, *European Journal of Operational Research*, Vol. 168, pp. 666-693
10. Talbot, F. B. and Patterson, J. H. (1984): An Integer Programming Algorithm with Network Cuts for Solving the Assembly Line Balancing Problem, *Management Science*, Vol. 30, pp. 85-99
11. Thangavelu, S. R. and Shetty. C. M. (1971): Assembly Line Balancing by Zero-One Integer Programming, *AIIE Transactions*, Vol. 3, pp. 61-68
12. Waters, D. (1996): Operations Management. Producing Goods and Services, *Addison-Wesley*
13. White W. W. (1961): Comments on a Paper by Bowman, *Operations Research*, Vol. 9, pp. 274-276