JÁRMŰIPARI FELSŐOKTATÁSI
ÉS KUTATÁSI EGYÜTTMŰKÖDÉS
TÁMOP-4.1.1.C-12/1/KONV-2012-0002

MAGYARORSZÁG KORMÁNYA
Európai Unió
Európai Szociális Alap
BEFEKTETÉS A JÖVŐBE
SZÉCHENYI 2020

Folyamatmodellezés és szimuláció

Készítette: Dr. Bohács Gábor

ISBN 978-615-5391-74-3

Folyamatmodellezés és szimuláció

1. modul: A folyamatmodellezés és szimuláció logisztikai alkalmazása

Ez a modul a folyamatmodellezés és szimuláció alapjait, módszereit vizsgálja logisztikai aspektusokon keresztül. A logisztikai rendszerek nagyfokú komplexitással rendelkeznek, így a folyamatok fejleszthetősége érdekében szükség van valamilyen rendszermodell előállítására.

Rendszermodelleket többféle módszerrel is előállíthatunk. Az összes gyakran alkalmazott modell mindegyikének bemutatása messze meghaladná a tananyag terjedelmét. Itt csak néhány jellemző modellt mutatunk be, és főleg az alkalmazás szempontjából.

A tananyagban kiemelten foglalkozunk a szimulációs modellekkel is. Ez az a módszer ugyanis, melyet jelenleg kiemelten használnak a logisztikai tervezés gyakorlatában.

1. lecke: Sorbanállási modell

Cél:

A tananyag célja, hogy a hallgató megismerje a sorbanállási modellek alkalmazásának fontosságát a logisztikában. Legyen tisztában a sorbanállás modellalkotás alapjaival

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- Bemutatni a sorbanállási modellek működését és használatát.
- Felsorolni és ábrával bemutatni a különböző sorbanállási modelleket.

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- Kiszolgáló központok,
- Beérezési ráta,
- Kiszolgálási ráta,
- Forgalmi intenzitás,
- Többcsatornás várakozó sorok,
- Korlátozott várakozó sorok.

1. Bevezetés a sorbanállási modellezésbe

Tevékenység:

A sorbanállási modell a hétköznapi életben is előforduló egyik kellemetlen jelenség a várakozás vizsgálatával foglalkozik. Definíció szerint a sorbanállás-elmélet különböző folyamatok eseményeivel kapcsolatos várakozási sorokat, sorbanállási időket a kiszolgálásra, és ezek összefüggéseit tárgyalja az alkalmazott matematikai eszközeivel. A műszaki tudományokban jelentősek a sorbanállási jelenségek. Az ilyen sorbanállás általános vonásait a következőkben lehet összefoglalni. A hálózatokban áramló egységek (objektumok) szabályos vagy szabálytalan időközönként áramlanak úgynevezett kiszolgáló központokba, ami műszaki vonatkozásban lehet raktárba érkezés, vagy megmunkáló géphez érkezés. A beérkezés után a kiszolgáló központ egy vagy több rendelkezésre álló kiszolgáló csatornáján (pl. párhuzamos műveletek, párhuzamos raktári fogadóhelyek) az egység továbbáramlik. A beérkező egységnek arra kell várni, hogy valamelyik kiszolgáló csatorna szabad legyen.

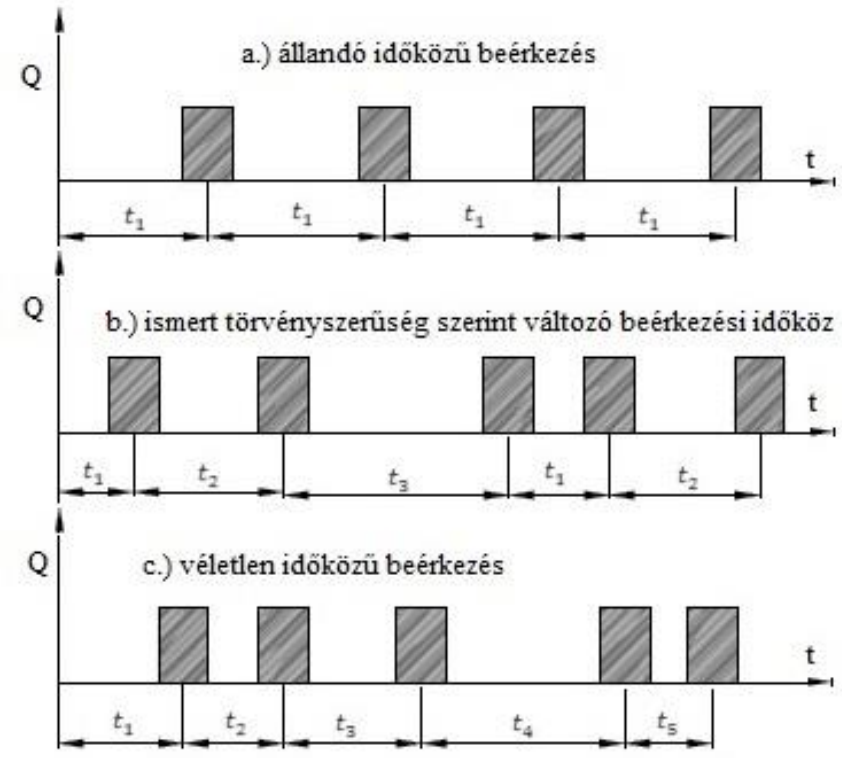
Kiszolgáló állomás vizsgálata

A kiszolgáló központba történő beérkezések közti eltelt idő lehet:

- egyenlő
- egyenlőtlen, de meghatározott törvényszerűség szerint változó
- egyenlőtlen, de csak az időközök valószínűségét ismerjük. Tehát a beérkezési időköz valószínűségi változó.

A kiszolgálást a kiszolgálási időtartammal tudjuk jellemezni, ami lehet:

- állandó
- változó, de meghatározott törvényszerűség szerint
- valószínűségi változó



1. ábra Beérkezési időközök [1]

A modell elemei általában két kimenetű elemek ezért állapotfüggvénnyel is jellemezhetőek.



2.a ábra Az igény felmerülése állapotfüggvénnyel ábrázolva



2.b ábra Várakozó sor



2.c ábra Kiszolgáló központ

Sorbanállási modellek vizsgálata

A folyamat során várakozó sor keletkezhet, ha a kiszolgáló egységekbe történő áramlás időköze, és a kiszolgálás időtartama szabálytalan. Sorbanállás keletkezhet akkor is a beáramlás időköze kisebb, mint a kiszolgálás időtartama, ekkor a tároló térben a beáramló várakozó anyagmennyiség, azaz várakozó sor folyamatosan növekszik.

Várakozó sor létrejöttének szempontjából először vizsgáljuk meg az állandó időközű beérkezés, és kiszolgálás esetét. Itt két esetet különböztetünk meg. Az egyik esetben a beérkezések időköze nagyobb, mint a kiszolgálásé, tehát a rendszerben a kiszolgáló oldalon szabad kapacitás keletkezik, ami alatt a kiszolgáló egység áll, vagy másik feladat elvégzésével tölti ki az álló időt. A másik esetben a beérkezés időköze kisebb, mint a kiszolgálásé, azaz a várakozó sor növekszik, ezt a kiszolgálási intenzitás (teljesítmény) növelésével tudjuk csökkenteni, vagy új párhuzamos kiszolgáló állomás üzembe helyezésével.

A gyakorlati életben a beérkezés időköze, és a kiszolgálás ideje nem meghatározott, hanem valószínűségi változó. Ekkor sztochasztikus folyamatról beszélünk, amelynek megoldása bizonyos feltételek mellett analitikusan végrehajtható. A várakozó sorba időegység alatt beérkező egységek száma, mint valószínűségi változó leggyakrabban Poisson féle eloszlást követ.

$$p_k = \frac{(\lambda t)^k}{k!}\exp(-\lambda t)$$

Vagy időegységben vizsgálva az alábbi sűrűségfüggvénnyel írható le.

$p_k = \frac{\lambda^k}{k!}\exp(-\lambda)$,ahol

$\lambda = \frac{1}{\overline{t_a}}$ a beérkezési ráta, és $\overline{t_a}$ a beérkezési időközök várható értéke.

A kiszolgálás eloszlása pedig leggyakrabban exponenciális eloszlást követ, amely az alábbi eloszlásfüggvénnyel jellemezhető,

$$P(\theta < t_b) = F(t_b) = 1 - \exp-(\mu t_b)$$

ahol $\mu = \frac{1}{\overline{t_a}}$ az egy időegységre jutó kiszolgálások száma, a kiszolgálási ráta ,és ahol $\overline{t_b}$ a kiszolgálási idő várható értéke. A várakozó sor vizsgálatának szempontjából a sorban várakozó egységek számának p_n valószínűsége a számottevő. Ha a sorbanállási jelenséget permanensnek tételezzük fel, akkor a sorban várakozó egységek számának valószínűsége

$$p_n = \left(\frac{\lambda}{\mu}\right)^n \left(1 - \frac{\lambda}{\mu}\right) = \psi^n(1-\psi)$$

Ahol, a sorbanállási rendszerre, és a kiszolgálásra jellemző tényezőt $\psi = \frac{\lambda}{\mu}$ forgalmi intenzitásnak nevezzük.

Továbbiakban a sorbanállási modell jelölését az eloszlási függvény tulajdonsága alapján csoportosítjuk, ha az eloszlási függvénnyel a sorbanállási modell analitikusan nem megoldható, akkor G jelzésű, ha megoldható, akkor M jelzésű (exponenciális eloszlás).

2. Sorbanállási modellek típusai

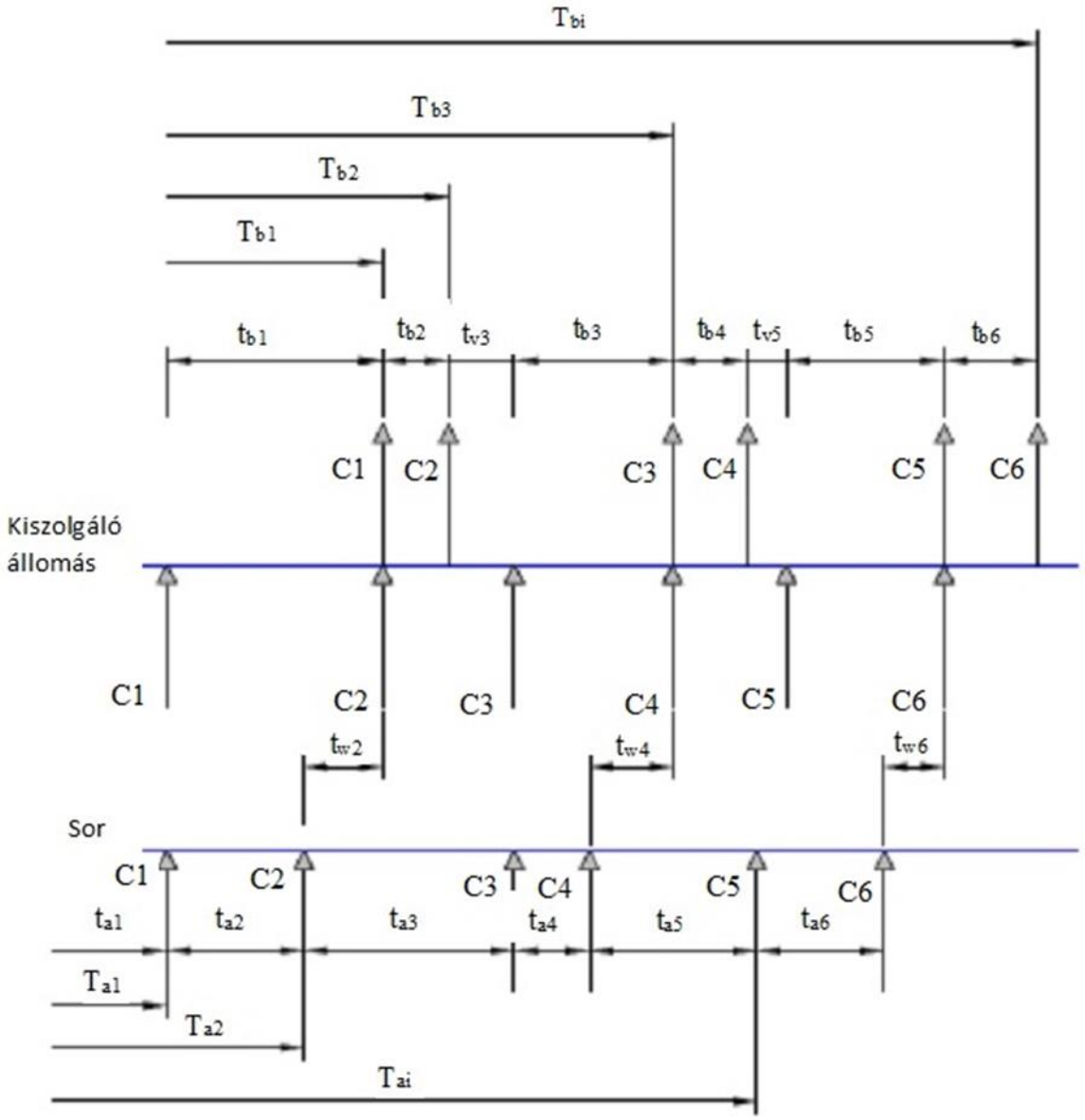
Egycsatornás várakozó sorok G/G/1 (M/M/1)

A várakozó sorok megnevezésében az első betű a beérkezési időköz, a második pedig a kiszolgálási időköz általános eloszlásfüggvényére utal. Az egyes szám pedig a kiszolgáló központok számát jelöli, egycsatornás kiszolgáló esetében tehát 1.

A kiszolgáló rendszerek időbeli dinamikájának vizsgálatára a 3. ábrán látható idődiagram használható. Az alsó tengelyen a kiszolgáló állomásra beérkező anyag igények, míg a felső tengelyen a kiszolgáló állomás működése van ábrázolva.
Az igények t_{a_i} időközönként jelennek meg, amik a kiszolgáló foglaltságának függvényében feldolgozásra továbbhaladnak, vagy várakoznak. A 3. ábrán jól látható, hogy C2 igény megjelenésekkor a kiszolgáló még C1 igények kiszolgálása folyik tb1 kiszolgálási ideig. A sor ezért tw2 ideig várakozni kénytelen. A kiszolgáló szabaddá válása után C2 igény kerül feldolgozásra, aminek ideje a várakozással együtt kisebb, mint C3 igény felmerülésének ideje tw2+tb2<ta3.

Hasonló elvek alapján, a soron C4, C6 igények felmerülésekkor várakozás alakul ki (tw4, tw6), míg C3, C5 igények beérkezése előtt a kiszolgáló állomás üresen várakozik (tv3, tv5).

A gyakorlatban egy sorbanállási rendszer üzemeltetésénél az a cél, hogy minimális veszteséggel üzemeljen, ehhez meg kell határozni az optimális forgalmi intenzitás, azaz a beérkezési, és kiszolgálási ráta hányadosát.



3. ábra Egycsatornás várakozó sor idődiagramja

Először határozzuk meg a sorban várakozó egységek várható számát $\bar{v}$ a sorban állási valószínűség P_n függvényeként, ha a rendszerben lévő egységek száma n>0

$$\bar{v} = \sum_{n=1}^{\infty}(n-1)P_n,$$

Forgalmi intenzitással kifejezve

$$\bar{v} = \frac{\psi^2}{1-\psi},$$

Innen a várható sorbanállási idő

$$t_f = \frac{\bar{v}}{\lambda},$$

A kiszolgálóhely szünetelésének várható értéke pedig

$$\bar{\rho} = \sum_{n=0}^{1}(1-n)P_n = P_0 = 1-\psi,$$

Az optimális forgalmi intenzitás meghatározásához a sor várakozási idejéhez k_1, és kiszolgáló állomás szüneteléséhez költségtényezőt k_2 rendelünk, és az így kapott költségfüggvény T időszakra

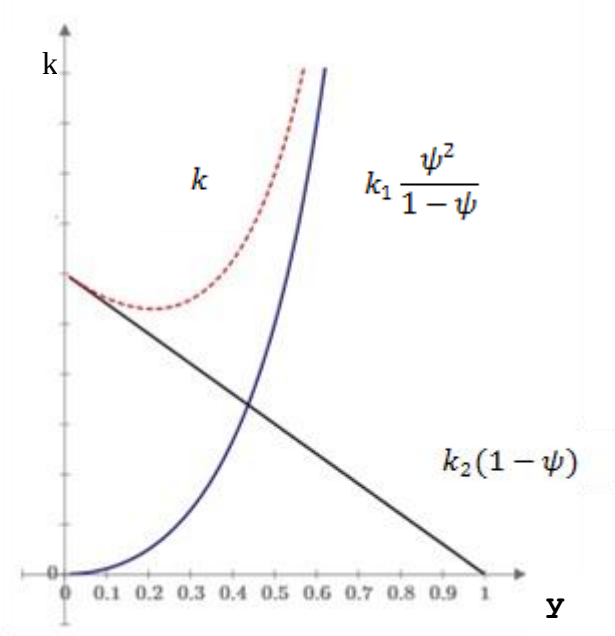
$$K = T(k_1\bar{v} + k_2\bar{\rho})$$

Behelyettesítve a forgalmi intenzitást, és időegységre rendezve az egyenletet

$$k = \frac{K}{T} = \left(k_1\frac{\psi^2}{1-\psi} + k_2(1-\psi)\right),$$

Költségfüggvény szélsőértékéből az optimális forgalmi intenzitás már meghatározható

$$\psi_{opt} = 1 - \sqrt{\frac{k_1}{k_1-k_2}}.$$

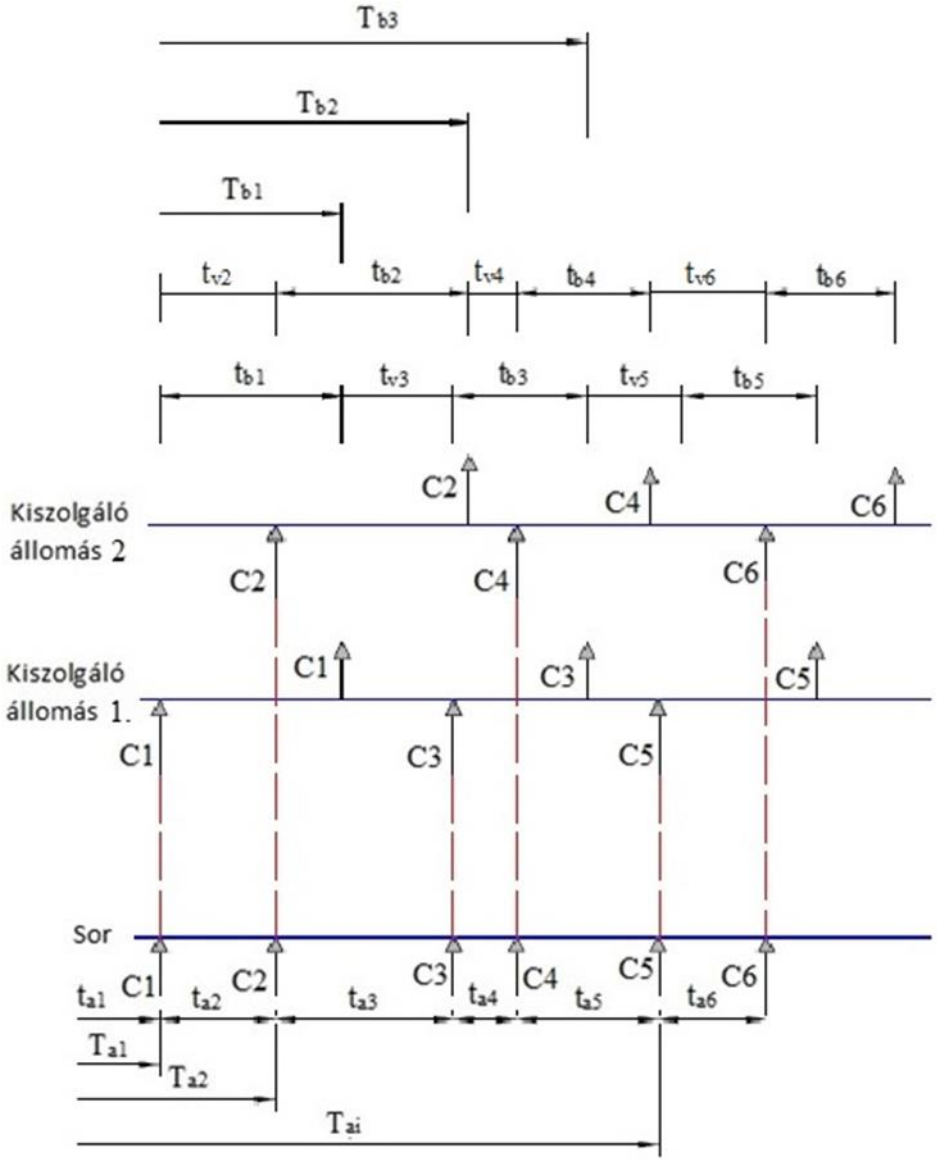


4. ábra Időveszteség szerinti költségfüggvények, és optimális forgalmi ráta

Többcsatornás várakozó sorok G/G/S (M/M/S)

A logisztikai folyamatok vizsgálata során gyakran előfordul, hogy nem tudjuk egy kiszolgálóhellyel megoldani a feladatot. Ilyenkor több kiszolgálóhelyet alkalmazunk. A kérdés itt, hogy hány állomást használjuk az optimális működéshez. Az 5. ábrán két csatornás kiszolgáló állomás látható, ahol a kiszolgálás előtti várakozás a nagyobb kiszolgálási kapacitás bevezetésével szüntettük meg.

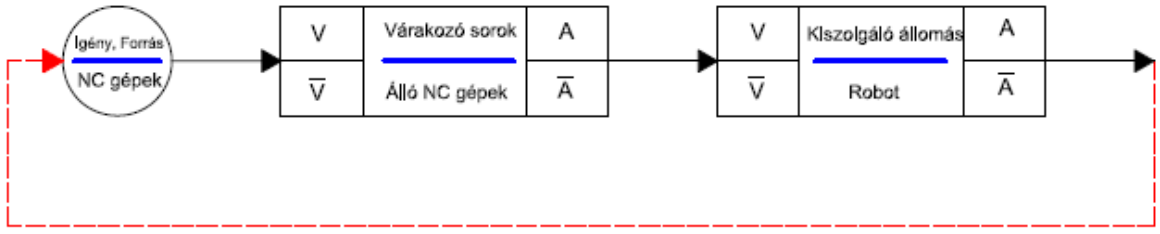
A nagyobb kiszolgálási teljesítménnyel viszont a kiszolgáló állomáson jelentkeznek termelésből kieső várakozó (üres) idők tv2, tv3, tv4, tv5, tv6. A kiszolgálók optimális számának meghatározása szintén a várakozó sor, és a kiszolgálók veszteségidejéhez rendelt költségfüggvények minimum értékének meghatározásával történik.



5. ábra Kétcsatornás várakozó sor beérkezési és kiszolgálási idődiagramja [1]

Egycsatornás várakozó sorok korlátozott számú kiszolgálóval G/G/1/m (M/M/1/m)

Olyan kiszolgáló rendszer ahol a várakozó sor korlátozott befogadó képességű, tehát csak meghatározott számú várakozó egység lehet a rendszerben. Egy alkalmazási példát véve egy m számú egymástól függetlenül működő NC szerszámgépet el kell látni munkadarabbal. A szerszámgépek munkadarab anyagellátását robot végzi. Tételezzük fel, hogy az NC gépek kiszolgálási igényei sztochasztikusan jelentkeznek és Poisson eloszlást követnek, míg a robot kiszolgálási ideje exponenciálisan jelentkezik. A várakozó sor, mivel korlátozott befogadó képességű, ábrázolása eltér az egycsatornás várakozó sor 2. állapotábrájától, és a 6. ábra szerint történik.



6. ábra egycsatornás várakozó sor korlátozott számú várakozóval G/G/1/m (M/M/1/m)

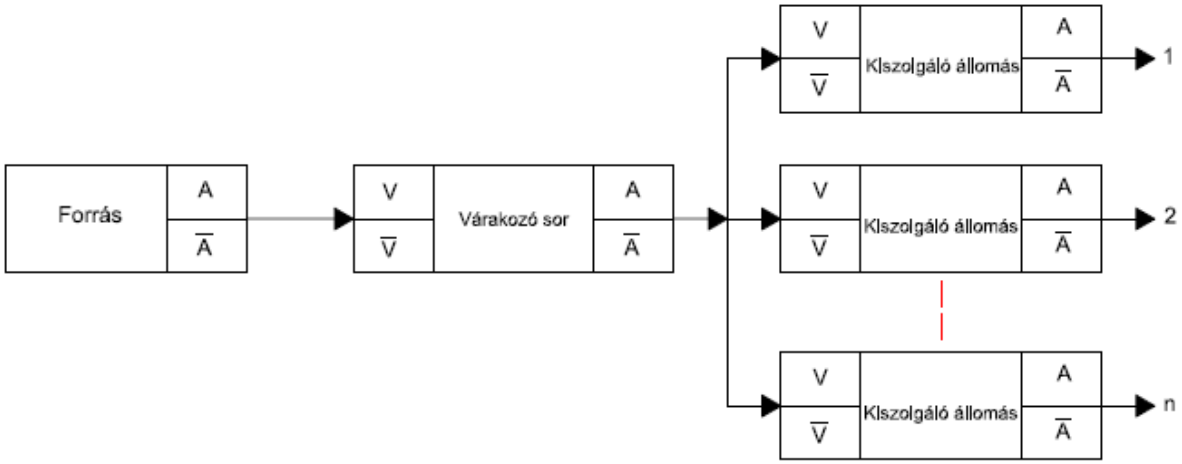
A vázolt sorbanállási rendszer működése a következő:

- Ha $n = 0$, a kiszolgáló állomás üzemen kívül van, hiszen ekkor minden NC gép dolgozik, a kiszolgálandó igények a forrásban vannak.
- Ha $1 \leq n \leq m$, a kiszolgáló állomás üzemel, esetleg várakozó sor is keletkezik.

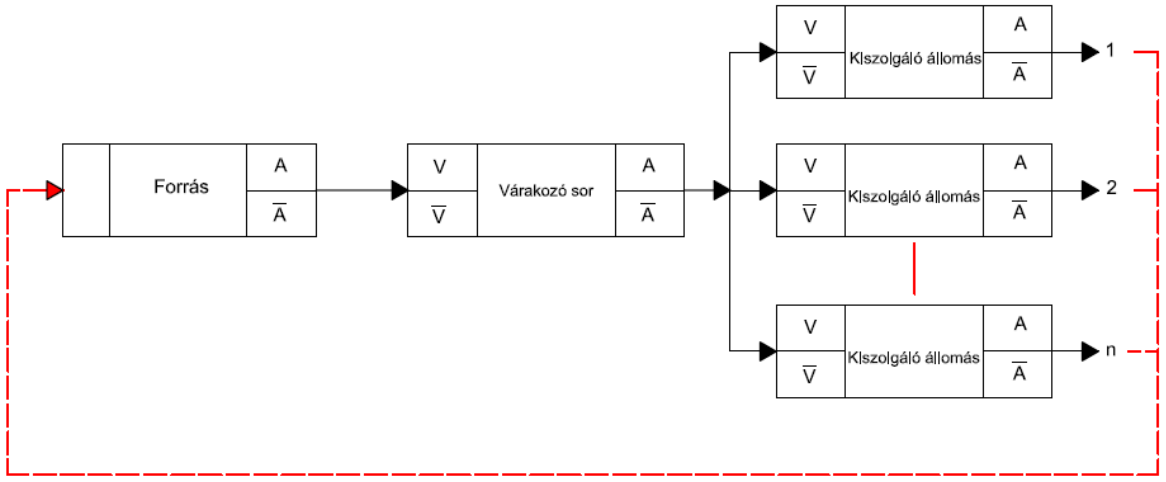
Többcsatornás várakozó sorok korlátozott számú kiszolgálóval G/G/S/m (M/M/S/m)

Az egycsatornás korlátozott várakozó sornak létezik többcsatornás kiszolgálása is, aminek állapotábráját a 7. ábra mutat. Az állapotábra megegyezik a többcsatornás korlátlan várakozó soréval, azzal a kivétellel, hogy a várakozó helyek korlátosak, amik jelölve is vannak.

A modell felhasználható az előző korlátlan többcsatornás fejezetpontban vizsgált feladathoz hasonlóan olyan problémák megoldásához is, amikor a kiszolgált igények visszatérnek a forrásba, tehát a sor korlátozását a forrásban lévő igények száma adja.



7. ábra Többcsatornás várakozó sor állapotábrája korlátozott kiszolgálóval



8. Többcsatornás várakozó sor korlátozott kiszolgálóval forrásban lévő elemek korlátosságával

Alkalmazás a logisztikában

A termelővállatok versenyképességük javítására egyre nagyobb hangsúlyt fektetnek a termelési költségek csökkentésére, a minőség növelésére, és flexibilis, gyorsan változtatható termelőrendszer kialakítására. Egy termék előállításnak költsége a gyártásszervezés, és a termelő eszközök számának optimális megválasztásával, és a termelési ráta növelésével jelentősen csökkenthető. A termelékenység mérőszáma a termelési ráta nagyban függ a termék átfutási idejétől. Az átfutási idő meghatározásánál viszont nem kerülhetjük meg a várakozási jelenséget.

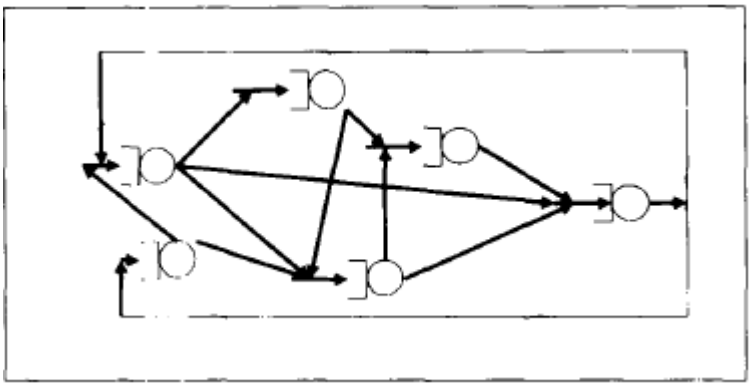
Termelő vállalatok fejlett termelőrendszerei, Flexible Manufacturing Systems (FMS), JIT, Flow lines vagy Computer Integrated Manufacturing (CIM) segítségével nagyban csökken a termelékenység növeléséhez szükséges átfutási idő. Ezek bevezetéséhez, elemzéséhez nyújt nagy segítséget a várakozó sor modell. Egy általános termelőrendszerben meghatározott

számú erőforrásokat találunk, amik folyamatokba rendezve különböző terméket állítanak elő. Ha a félkész termék a termelő folyamatban egy foglalt erőforráshoz érkezik, akkor várakoznia kell, amíg az erőforrás szabad nem lesz. Egy termelőrendszerben több várakozó sor alakulhat ki egymásba kapcsolódva, amik dinamikus rendszert alkotnak a sztochasztikus anyagbeérkezést figyelembe véve. Ezt a várakozó sorok alkotta hálózatot kívánjuk modellezni.

A modell működési szempontból kiszolgálókból, vásárlókból, hálózati csomópontokból és pufferekből áll. A gyártógépek kiszolgálóként, míg az alkatrészek vásárlóként vannak jelen. A gyártásközi pufferek lesznek a várakozó sorok, ahol az alkatrészek a szabad kiszolgáló egységre várakoznak. A munkaállomások a pufferekkel együtt csomópontokként működnek, amik az anyagáramlás előre definiált sorrendje szerint alkotnak kapcsolatot a többi munkaállomással. Egy munkaállomáson belül több kiszolgáló is működhet, de ha a beérkezett anyag feldolgozására nincs szabad kapacitás, akkor várakozó sor alakul ki az állomás pufferében. Az általános várakozó hálózat felépítése után nézzük meg, termelőrendszereken belüli alkalmazhatóságukat, és ezek fontosságát.

FMS

Az FMS, amely számítógép által vezérelt részben független munkaállomásokat jelenti, az anyagmozgató rendszerrel való kapcsolatán keresztül, alacsony raktárkészletet tesz lehetővé, aminek teljesítmény mérésére szolgálhat a sorbanállási modell. Ezenfelül a sorbanállási modell lehetővé teszi a termelő vállalatok számára, hogy a kiegyensúlyozott, stabil termeléshez meghatározzák a szükséges Kanban számát. A hozzákapcsolt anyagmozgató rendszer általában palettás egységekben működik, amin a megmunkálási feladatot elvégzik. Mivel egy ilyen rendszerben egyszerre jelenlévő paletták száma meghatározott, ezért a legkisebb átfutási idő eléréshez a rendszerből kiáramló paletták helyére egyből új palettának kell érkeznie. Ezért ezt a rendszert zárt várakozó sorként lehet modellezni, amihez matematikai algoritmusokat használó számítógépes elemző szoftverek léteznek CAN-Q (Computer Analysis of Networks of Queues).



9. Ábra Zárt várakozó sor hálózat FMS-ben

Tevékenység:

- Készítsen összefoglaló ábrát!
- Gyűjtse össze és rendszerezze a sorbanállási modellek típusait!

Önellenőrző kérdések

1. Mi a sorbanállási modellek tipikus felépítése?
2. Milyen költségeket értelmezünk a sorbanállási rendszerekben?
3. Mutasson be példát egy és többcsatornás várakozó sorra!

2. lecke: Fuzzy modellezés

Cél:

A tananyag célja, hogy a hallgató megismerje valószínűségi modell bizonytalanságát feloldó fuzzy logikai modellt. Megismerje a hagyományos Boole, és fuzzy logikai műveletek közti különbséget. Részletes rálátást kapjon a fuzzy logika működéséről, valamint az erre alapuló szimulációs technika alkalmazásáról.

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- A fuzzy halmazelmélet sajátosságait felsorolni,
- A fuzzy modell létrehozásának lépéseit megfogalmazni, és részletezni azokat,

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- Crisp halmaz (klasszikus alaphalmaz),
- Tagsági függvény,
- Metszetek,
- Fuzzy halmaz tartója, magja, magassága, normálisa/abnormálisa
- Fuzzifikálás/ defuzzifikálás.

1. Bevezetés a fuzzy logikába

Tevékenység: Korábbi matematikai tanulmányaira támaszkodva ismételje át a halmazműveletek tulajdonságait. Kommutativitás, asszociativitás, disztributivitás, idempotencia, elnyelési törvények, identitás, ellentmondás törvénye, a kizárt harmadik törvénye, De Morgan azonosságok.

Valós folyamatok, és problémák modellezésekkor akaratlanul beleütközünk a valószínűségi modellünk bizonytalansági korlátaiba. Erre a bizonytalansági tényezőre megoldás a Fuzzy logika alkalmazása. Ebben a leckében bemutatjuk a fuzzy halmazelmélet alapjait, valamint egy egyszerű hétköznapi modellen keresztül a fuzzy modellezés lépéseit.

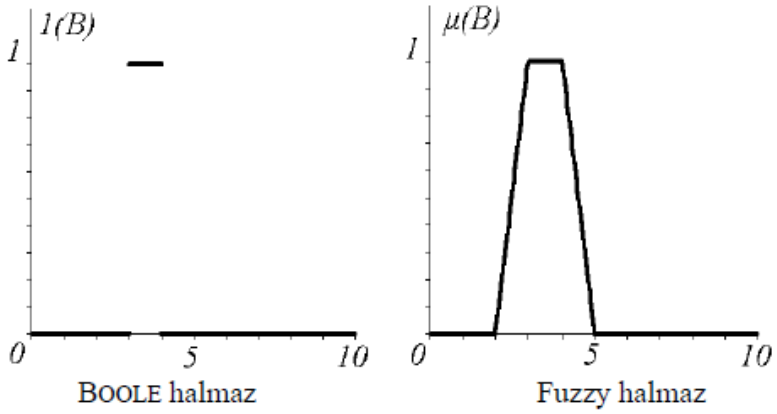
Háromértékű logika

Az egyszerű folyamatok leírására használt kétértékű logikát (0/1; igaz/hamis kimenet) már Arisztotelész is használta, de GEORGE BOOLE angol matematikus foglalta rendszerbe. Ezen kétértékű logikán alapuló matematikai eszközrendszert BOOLE – algebraként ismerjük. A mesterséges intelligencia, és bonyolultabb műszaki folyamatok leírására, viszont ez a logika a diszkrét kimeneti értékeinek korlátossága miatt nem alkalmas. A fuzzy logika megengedi a klasszikus logikában felvehető igaz (1), és hamis (0) értékek közti –

azaz [0,1] zárt intervallumban definiált – bármely valós értéket. A fuzzy logikával megoldhatóvá válik bonyolultabb műszaki feladatok megoldása.

Fuzzy halmazelmélet

A nem fuzzy halmazokra elterjedt a crisp halmaz megnevezés. A crisp halmazok karakterisztikus függvénye minden alaphalmazbeli elemhez 0-t vagy 1-et rendel hozzá. A karakterisztikus függvény fogalmát úgy általánosíthatjuk, hogy az alaphalmaz minden eleméhez valamely rögzített tartományból – ez általában az intervallum – rendelhető érték. Ezen érték nagysága a halmazbeli tagság mértékével arányos, azaz minél kisebb (nagyobb) mértékben tagja a halmaznak valamely elem, annál kisebb (nagyobb) az elemre vonatkozó függvényérték. Ezt a függvényt tagsági függvénynek, azt általa definiált halmazt pedig fuzzy halmaznak nevezzük.



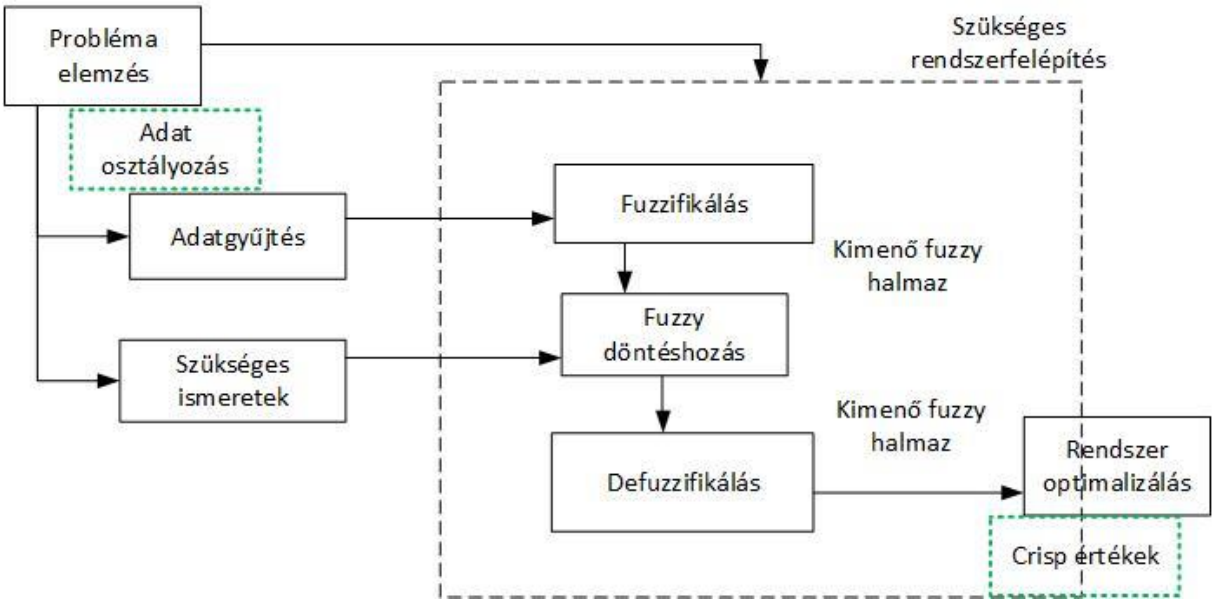
10. ábra Klasszikus BOOLE algebra, és a fuzzy halmaz adott [0,10] intervallumon eltérő értelmezése.

2. Fuzzy modellezés

Fuzzy modell létrehozása

Fuzzy alapú döntéstámogató rendszer létrehozásának lépései a 2. ábrán láthatóak. A lépések sorrendben

1. Probléma elemzés, adatgyűjtés
2. Be és kimeneti változók, és ezek tagsági függvényeinek meghatározása
3. Logikai kapcsolatok meghatározása be és kimenetek között
4. Fuzzifikálás
5. Hipotézisek kiértékelése
6. Aggregáció
7. Defuzzifikálás



11. ábra FIF modell felépítésnek sematikus rajza

Be és kimeneti értékek meghatározása

A modell létrehozásához meg kell határoznunk a be, és kimeneti változókat és ezek tagsági függvényeit. Ha a fuzzy változó z, és a bemeneti változók rendre x, y akkor a meghatározandó függvény $z = f(x, y)$.

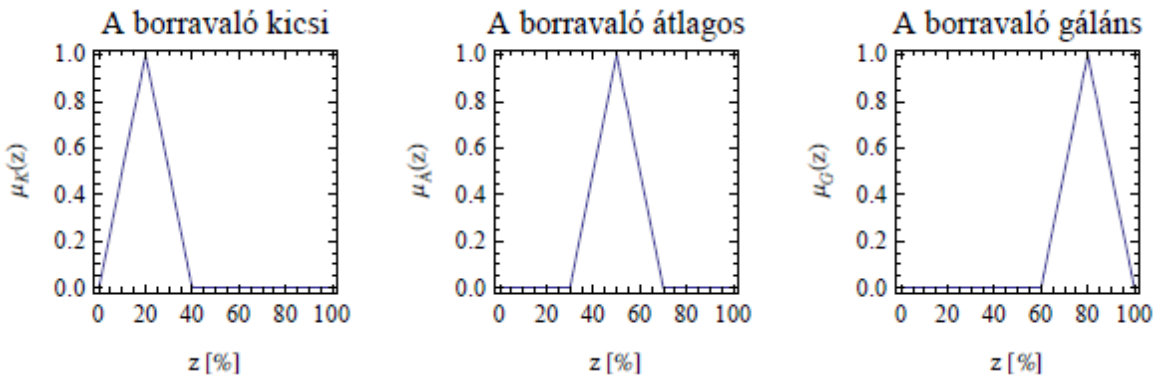
Egy egyszerű példát véve, megszeretnék határozni a borravaló mértékét egy vacsora után [2]. A fuzzy kimeneti változó ebben az esetben a borravaló mértéke, amit 3 halmazba rendezünk, ezek rendre

- A borravaló kevés (K)
- A borravaló átlagos (Á)
- A borravaló gáláns (G)

Miután z fuzzy változó meg kell adni a hozzá tartozó fuzzy halmazokat és az ezekre vonatkozó tagsági függvényeit. (z), mint fuzzy változó 3 fuzzy halmazhoz tartozhat, ezek tagsági függvényei,

$$\mu_K(z); \mu_{\acute{A}}(z); \mu_G(z)$$

A $\mu_K(z)$ tagsági függvény a z jellemző adott K (kicsi borravaló) halmazhoz való tartozásának mértékét adja meg. A skála meghatározásnál a könnyebb összehasonlítás miatt érdemes 0-10 esetleg 0-100 értékig terjedő skálát felvenni. Természetesen ettől eltérő skála felvétele is lehetséges. A tagsági függvények sarok pontjainak meghatározása szakértői vélemények alapján történik. A tagsági függvények értékeit a szakértői válaszok átlaga, vagy súlyozott átlaga adja a szakértői kompetenciák függvényében.



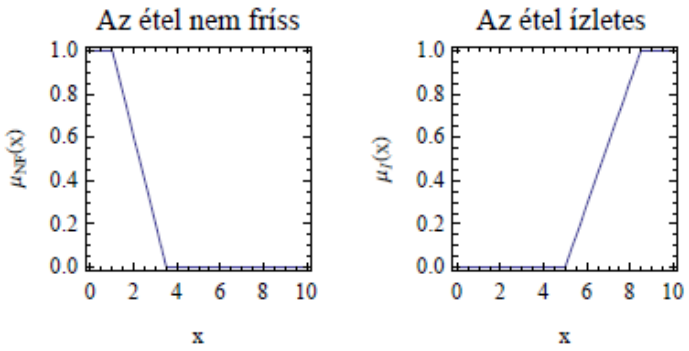
12. ábra Borravaló mértékéhez, mint fuzzy változóhoz tartozó tagsági függvények [2]

A bemeneti x, y értékekre vonatkozóan is meg kell adni a halmazokat, és a hozzájuk tartozó tagsági függvényeket. Az első bemeneti változó x, legyen az étel minősége, ami lehet

- Ízletes (I)
- Nem friss (NF)

A kiszolgálást egy 0-10 terjedő skálán mérjük. Az első bemeneti érték tagsági függvényei

$\mu_I(x); \mu_{NF}(x)$

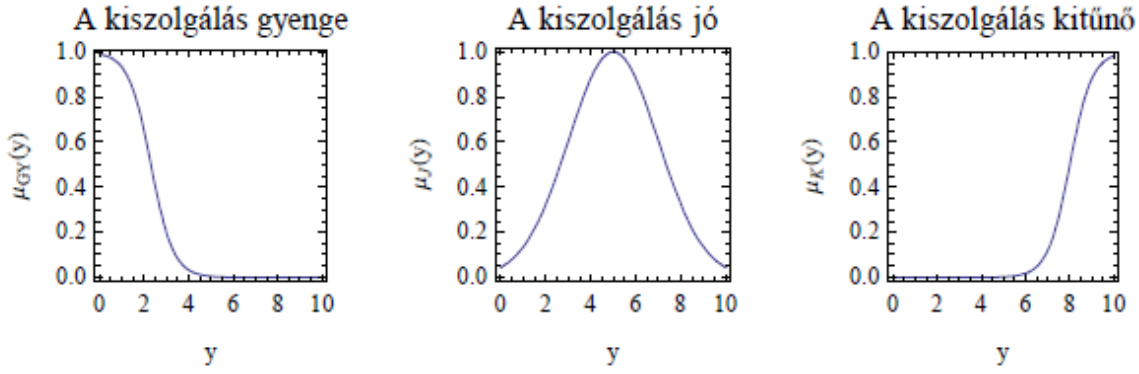


13. ábra Étel minősége, és tagsági függvénye, mint bemeneti érték

A második bemeneti változó legyen a kiszolgálás minősége, aminek halmazai

- A kiszolgálás gyenge (GY)
- A kiszolgálás jó (J)
- A kiszolgálás kitűnő (K)

Ezek tagsági függvényei $\mu_{GY}(y); \mu_J(y); \mu_K(y)$



14. ábra Kiszolgálás minősége, mint második bementi érték, és tagsági függvénye

Ezután második lépésben létre kell hoznunk azokat a logikai szabályokat, amik a fuzzy halmazok közti kapcsolatot határozzák meg.

- a) IF a kiszolgálás gyenge OR az étel nem friss THEN a borravaló kicsi
- b) IF a kiszolgálás jó THEN a borravaló átlagos
- c) IF a kiszolgálás kitűnő OR az étel ízletes THEN a borravaló gáláns

3. Fuzzy modell kiértékelése

Fuzzifikálás

Ha a bemeneti változókhoz értéket rendelünk, példánkban ezek legyenek most diszkrét (crisp) értékek $x = 7; y = 4$. Mivel az x és y fuzzy változók ezért a fuzzy modellben az őket reprezentáló értékeket a megfelelő tagsági függvények értékei szolgáltatják, azaz esetünkben például az első szabály a.) hipotézis alapján,

A kiszolgálás gyenge

$$\mu_{GY}(y) = \mu_{GY}(4) = 0{,}05$$

Az étel nem friss

$$\mu_{NF}(x) = \mu_{NF}(7) = 0$$

Ezt nevezzük a bemeneti adatok fuzzifikálásának Tehát a modell kiértékelés első lépése a bemeneti értékek fuzzifikálása a szabályok hipotézis oldalán szereplő fuzzy halmazok tagsági függvényei alapján.

Továbbiakban állítsuk elő a többi hipotézis szerinti tagsági függvények értékeit. Második szabályt tekintve a kiszolgálás jó,

$$\mu_J(y) = \mu_J(4) = 0{,}5$$

A harmadik szabály szerint, a kiszolgálás kitűnő,

$$\mu_K(y) = \mu_K(4) = 0$$

Az étel ízletes

$$\mu_I(x) = \mu_I(7) = 0{,}2$$

Hipotézisek kiértékelése

Következő lépésben a hipotézis szabályok alapján elvégezzük a fuzzy logikai műveleteket. Első OR szabály esetén maximum képzéssel

$$max\left(\mu_{GY}(y), \cup\mu_{NF}(x)\right) = max\left(\mu_{GY}(4), \cup\mu_{NF}(7)\right) = \max(0{,}05, 0) = 0{,}05$$

Második szabálynál, ha a kiszolgálás jó

$$\mu_J(y) = \mu_J(4) = 0{,}5$$

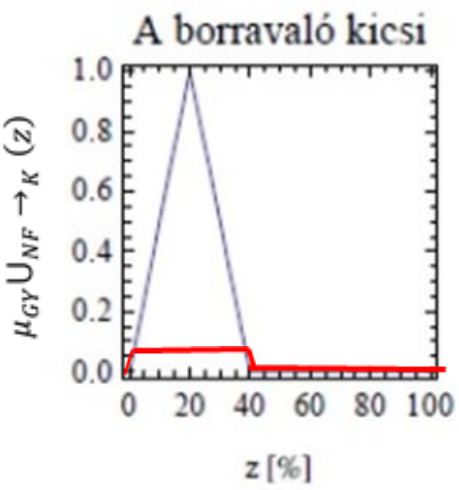
Harmadik szabálynál OR logikai műveletet esetében

$$max\left(\mu_K(y), \cup\mu_I(x)\right) = max\left(\mu_K(4), \cup\mu_I(7)\right) = \max(0{,}0{,}2) = 0{,}2$$

Fuzzy következtetés kiértékelése

A kiértékelés során, a hipotézis oldalon szereplő fuzzy halmaz tagsági függvényén hajtunk végre módosítást. A módosítást minimum képzéssel oldjuk meg (Mamdani-féle inferencia). Az első szabály esetében a kiszolgálás gyenge (GY) OR étel nem friss (NF) → borravaló kevés (K)

$$\mu_{GY} \cup_{NF} \rightarrow_K (z) = min\left(\mu_{GY} \cup_{NF} (y, x), \mu_K(z)\right) = min(0{,}05, \mu_K(z))$$

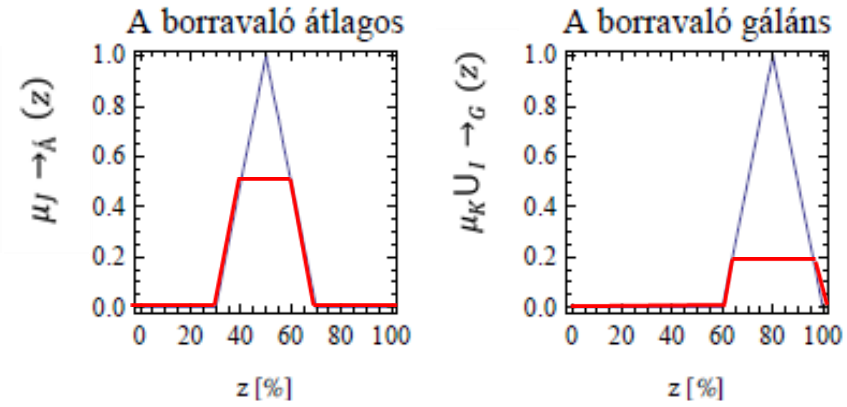


15. ábra Implikáció kiértékelése kicsi borravaló esetére

A többi szabályra is hasonlóan eljárva

$$\mu_J \rightarrow_{\acute{A}} (z) = min\left(\mu_J(y), \mu_K(z)\right) = min\left(0{,}5, \mu_K(z)\right)$$

$$\mu_K \cup_I \rightarrow_G (z) = min\left(\mu_K \cup_I (y,x), \mu_K(z)\right) = \min\left(0, \mu_K(z)\right)$$



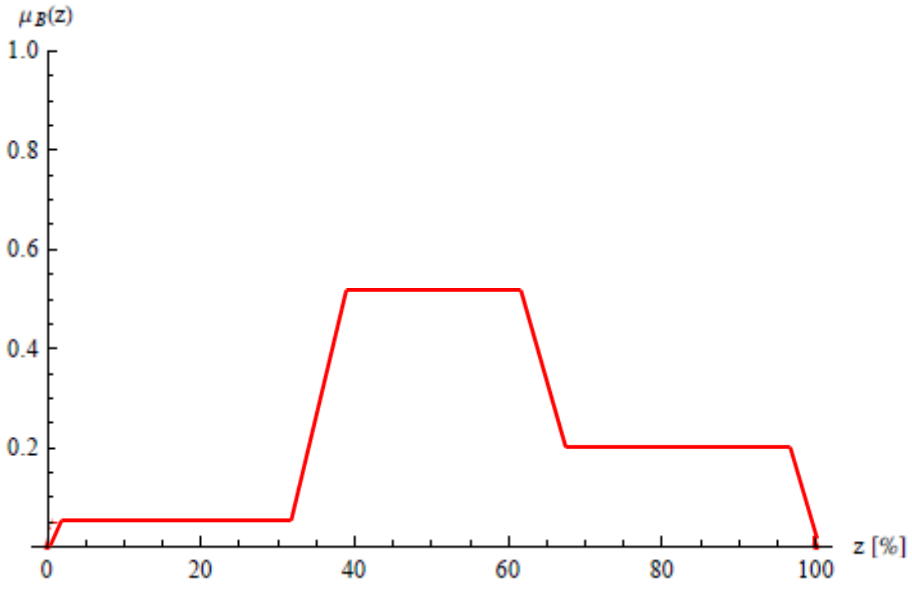
16. ábra Átlagos, és Gáláns borravaló módosított tagsági függvénye

Aggregáció

Fuzzy halmazokon értelmezett aggregációs operátorok több fuzzy halmaz megfelelő módon történő egyesítése által egyetlen fuzzy halmazt állítanak elő. A mi esetünkben az egyes borravaló halmazok OR kapcsolatát kell előállítani,

$$\mu_B(z) = \mu_{K\acute{U}G}(z) = max\left(\mu_{GY} \cup_{NF} \rightarrow_K (z), \mu_J \rightarrow_{\acute{A}} (z), \mu_K \cup_I \rightarrow_G (z)\right)$$

Példánkban a fuzzy változóra vonatkozó tagsági függvények aggregációja a következő,



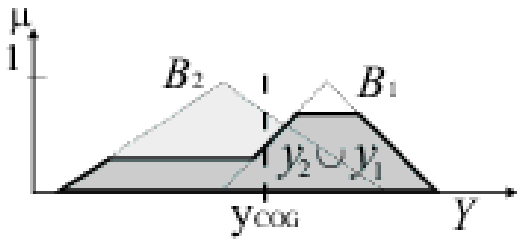
17. ábra Aggregált módosított fuzzy tagsági függvény

Defuzzifikáció

Defuzzifikációval az aggregált tagsági függvény pontjai közül kiválasztjuk azt az értéket, ami fuzzy halmazt az alkalmazástól függően a legjobban jellemzi. Az érték kiválasztására több matematikai módszer is ismert.

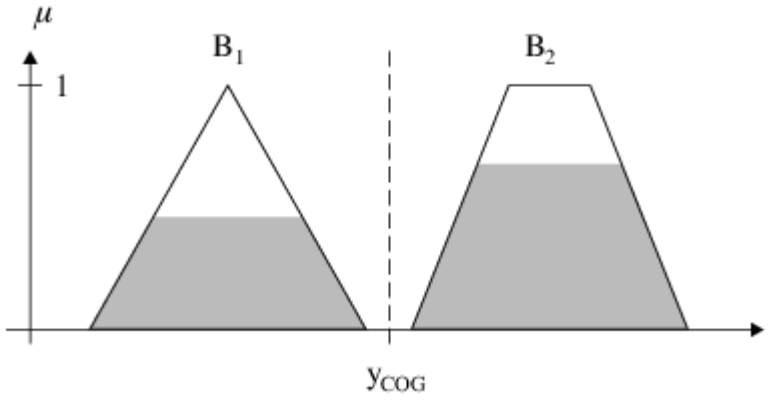
1. Súlyponti módszer (COG) Center of Gravity

Ez az egyik leggyakrabban használt defuzzifikációs módszer. Előnyei közé tartozik, hogy háromszög és trapéz alakú szabályoknál viszonylag egyszerűen számolható, valamint hogy közvetlen irányítás esetén majdnem mindig folytonos viselkedést eredményez: ha a megfigyelés s ezzel együtt a szabályok alkalmazhatóságának mértéke kis mértékben változik, az nem okoz nagy eltérést az úgynevezett crisp (éles, már „nem fuzzy") következmény értékében sem.



18. ábra Defuzzifikálás súlyponti módszerrel

Hátránya között említhető, hogy a módszer olyan értéket határoz meg, amelyre a konklúzió tagsági értéke nulla. Erre példa a XY ábrán látható. Egy irányított automatikus járműhaladás példáját tekintve az akadály jobbra vagy balra elkerülésénél a fuzzy következtetésnek a "kerüld el az akadályt jobbra vagy balra" utasítás lehet a nyelvi interpretálása. A súlyponti módszert alkalmazva viszont a jobbra és balra kerülés súlyponti átlaga pont az akadálynak ütközést eredményezi.



19. ábra Egymást nem metsző tagsági függvény konklúzió értéke nulla

2. Geometriai középponti módszer (COA)

Hasonló a súlyponti módszerhez, azzal a kivétellel, hogy míg a súlyponti módszer a több részkonklúzió által fedett területeket többszörösen számolja, addig COA (Center of Area) csak a következtetés alakját veszi figyelembe. Így természetesen az átlapolt felületek csak egyszeres súllyal esnek számításba. Hátránya, a súlyponti módszerhez képest, hogy bonyolult alakú részkonklúziók esetén igen nehezen számolható. A defuzzifikált érték COA eljárással az alábbi kifejezéssel számolható.

$$y_{COA}=\frac{\int_{y\in B*} B^*(y)ydy}{\int_{y\in B*} B^*(y)dy}$$

Diszkrét értékek esetén a függvény elemek véges összegeként is meghatározható.

$$y_{COA}=\frac{\sum_{i=1}^{m} B^*(y_i)y}{\sum_{i=1}^{m} B^*(y_i)}$$

3. Maximumok középértéke módszer (MOM)

A módszert leginkább véges elemszámú univerzum esetén alkalmazzák. Előnye, hogy egyszerűen számolható. Hátrányai közül a legjelentősebb, hogy nemfolytonos irányítási függvényt eredményez.

Ezenkívül ismert még a középső maximum módszer is, mint defuzzifikálás módszere. Az eljárás a következtetés legnagyobb tagságifüggvény-értékű elemeiből választja ki a középsőt (Center Of Maxima).

Tevékenység:

- Jegyzetfüzetébe gyűjtse ki a fuzzy modellezés legfontosabb fogalmait és az ezeket szemléltető ábráit.
- Gondoljon ki egy fuzzy következtetéssel megoldható hétköznapi példát.

Önellenőrző kérdések

1. Mit jelent a fuzzy halmaz?

2. Mutasson be egy kiválasztott fuzzy következtetést!

3. lecke: Folyamatleíró nyelvek

Cél:

A lecke célja, hogy megismertesse azokat a főbb folyamatleíró nyelveket, melyeket logisztikai modellezésnél is alkalmaznak.

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- Folyamatleíró nyelvek főbb típusait ismertetni
- A főbb leíró nyelvek előnyeinek és hátrányainak figyelembevételével alkalmazásukra javaslatot tenni

Időszükséglet:

A tananyag elsajátításához körülbelül 180 percre lesz szüksége.

Kulcsfogalmak

- IDEF diagramok
- UML
- EPC
- YAWL
- BPMN
- BPD
- BPEL

1. Folyamatleíró nyelvek

A folyamat, algoritmusok leírását a mérnöki/műszaki tudományok területén kezdték formalizálni az 1930-as években. A formalizálás lehetővé tette a folyamatok strukturált grafikus leírását, ami a szöveges leírás mellé nagyban elősegítette későbbi, harmadik (külső) fél általi megértését. A grafikus ábrázolás amellett, hogy könnyebben és gyorsabban értelmezhető, feldolgozhatóbb, mint egy többoldalas szöveg, információt sűrít, és jól alkalmazható az elágazások, ciklusok, döntési pontok egyértelmű reprezentálására, javítja az információ átláthatóságát.

Később a grafikus ábrázolást kiegészítő szöveges leírásokat formális szabályoknak eleget tevő leíró nyelvek váltották fel, amelyek már nemcsak leírásra, hanem modellezésre, ellenőrzésre is lehetőséget biztosítanak.

A műszaki területen legelterjedtebb folyamatleíró nyelvek a következők:

- IDEF diagramok
- UML
- EPC

- YAWL
- BPMN
- BPD
- BPEL

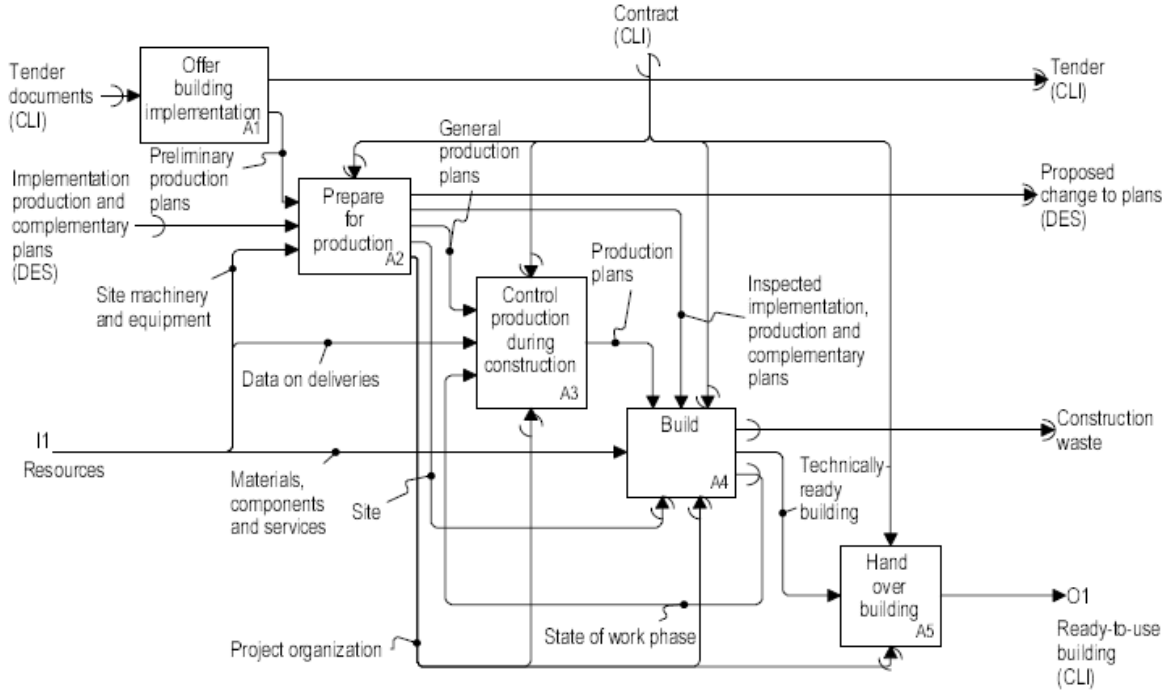
2. IDEF diagramok

Az Integrated Definition (IDEF) folyamatleíró nyelvet elsősorban folyamatok fejlesztéséhez, integrációjához, tervezéséhez és rendszerelemzéshez kapcsolódó tevékenységek leírására használják. Az IDEF leíró nyelvnek több változata is létezik, amiket számok jelölnek.

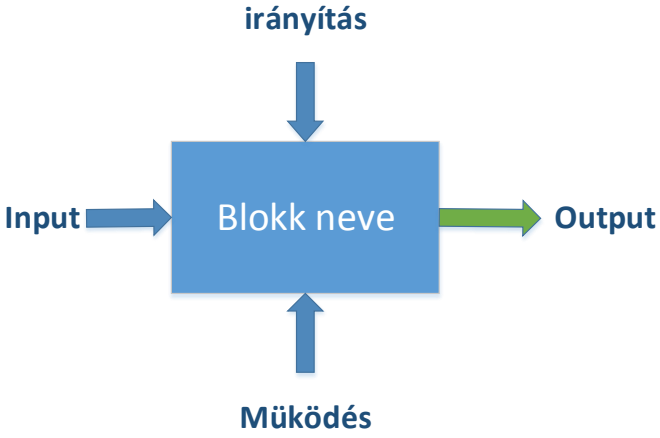
- IDEF0: Működés, funkció leírás,
- IDEF1: Információ áramlási folyamatok leírása,
- IDEF1x: Adat tervezés,
- IDEF3: Folyamatok leírása,
- IDEF4: tervezés.

Az IDEF0 leíró nyelv gyakran használt leíró nyelv, a modellkészítés első lépése, mivel a rendszer alapvető funkciói, tevékenységei közötti kapcsolati feltételeket állapítja meg. Ezenfelül részletes leírást ad a rendszer folyamatairól, tevékenységeiről. Az IDEF0 diagramokat a folyamatok tudásmenedzsmentjének kezelésére használják döntéstámogató céllal.

A 20 ábrán egy építkezés felső szintjének IDEF0 diagramja látható. A diagram egymással feltételi kapcsolatban lévő funkciókból áll. Minden funkciónak van ki- és bemeneti értéke, működési mechanizmusa és irányítása a 21 ábrán látható.



20. ábra építkezés folyamatának végrehajtása IDEF0 diagramban ábrázolva



21. ábra IDEF0 funkcióblokk felépítése

- Input: Azok az elemek, amik a részfolyamatot elindítják, triggerek,
- Irányítás (Controls): Befolyásolja, irányítja a tevékenységet,
- Működés (Mechanism): A tevékenység elvégzéséhez szükséges rendszer, ember, gépi feltétel,
- Output: A tevékenység, részfolyamat elvégzésének eredménye.

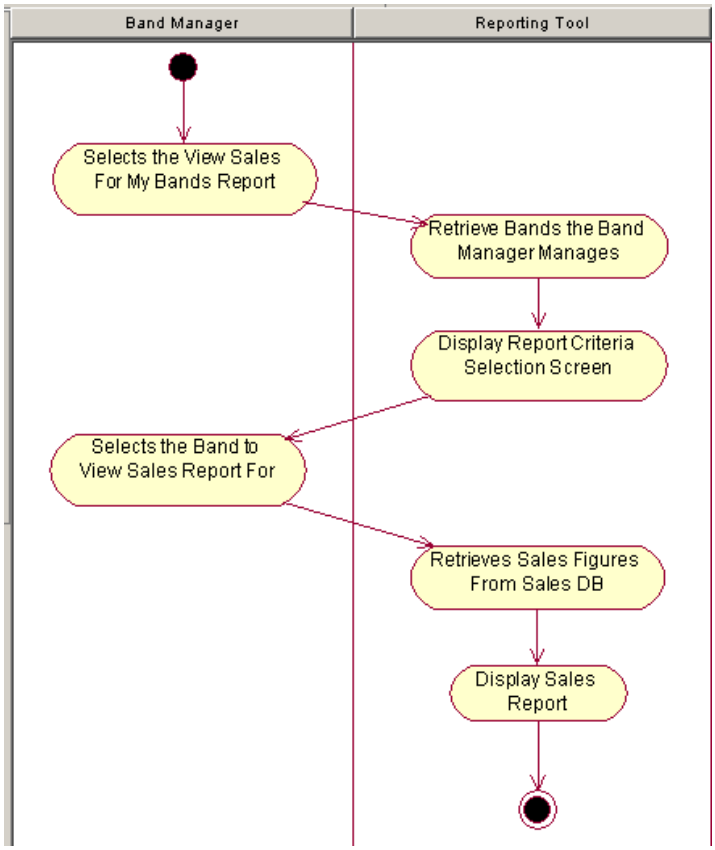
Az IDEF0 diagramok nem tartalmazzák a folyamatok időszükségletét, döntési logikáját, de ennek ellenére ez a módszer széles körben elterjedt mivel egyszerűen, és tömören képes modellezni a folyamathoz tartozó tevékenységek sorrendjét. További előnye, hogy több szinten képes leírni a folyamathoz tartozó tevékenységet, az aprólékos részletektől egészen a döntéshozáshoz szükséges teljes folyamatot tartalmazó szintig. Az IDEF0 diagram önmagában történő alkalmazásán felül további módszerekkel kombinálva is használható, ezzel az egyes módszerek specifikus előnyei jobban kihasználhatóak, hátrányaik kiküszöbölhetőek.

Hátránya, hogy túlzott tömörség esetén, harmadik fél számára kevésbé érthető. Valamint a tevékenységek időszükségletének és döntési logika hiányában nem alkalmas a folyamat szűk keresztmetszetének feltárására.

3. UML

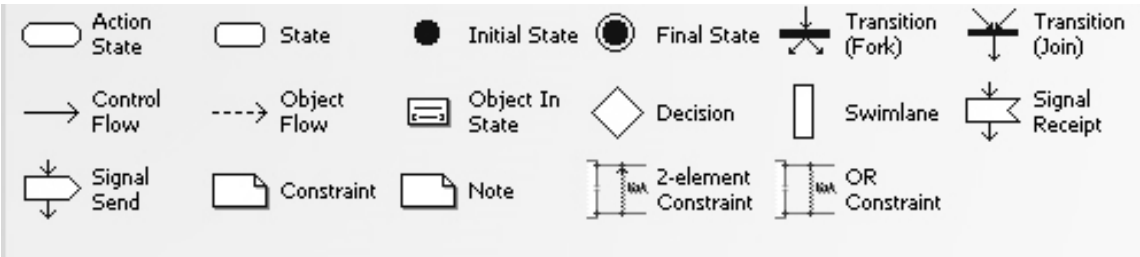
Az UML standardizált leíró nyelv (22. ábra), amellyel szöveges és grafikus formában adhatunk információt rendszerekről, szervezetekről (logikai összetevőkről, külső- és belső kölcsönhatásukról, stb.), üzleti folyamatokról, folyamatokon belüli vagy azok közötti kommunikációról, rendszerek viselkedéséről, feladatukról. 13 diagramtípus közül

választhatunk UML-ben, ezek három fő csoportba sorolhatók. Struktúradiagramokkal a modellezett rendszer elemeit, viselkedésdiagramokkal a rendszerben fellépő eseményeket, aktivitásdiagramokkal pedig magát a folyamatot lehet legjobban leírni.



22. ábra UML standardizált leíró nyelv - Activity diagram [3]

Az UML tevékenységdiagram elemeinek összefoglalóját és szimbólumait az alábbi ábra ismerteti (**Hiba! A hivatkozási forrás nem található.**).

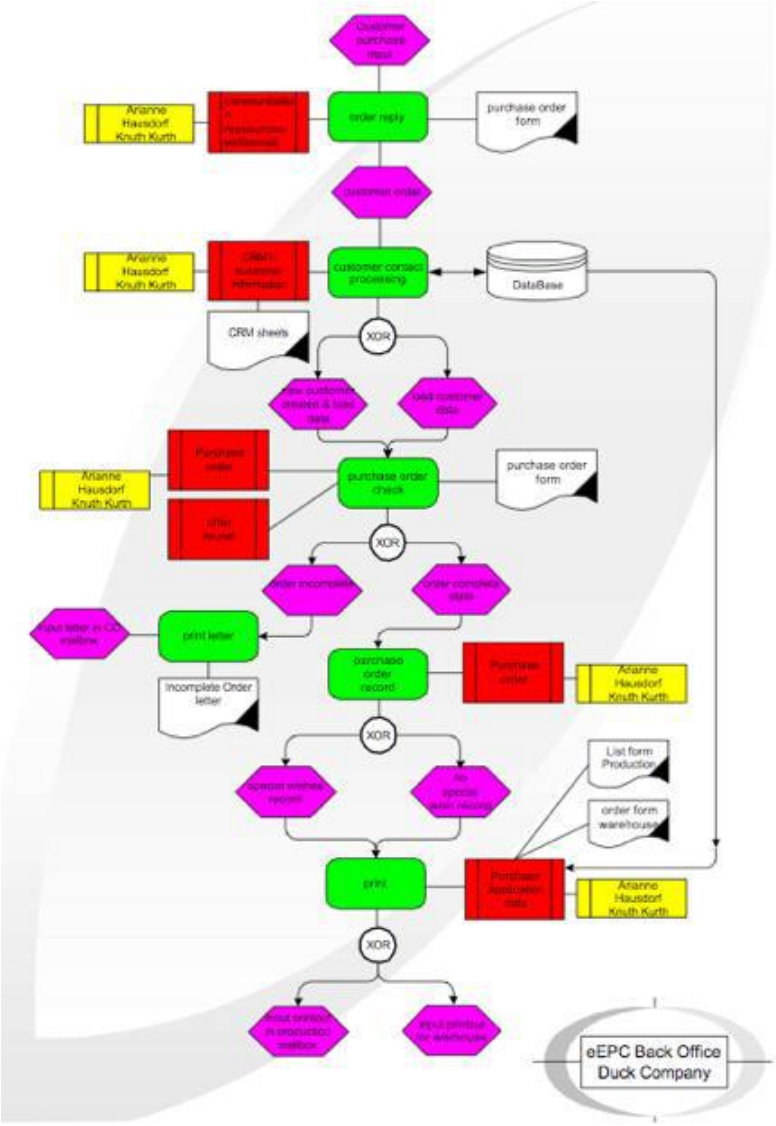


23. ábra UML tevékenységdiagram elemeinek listája

Az UML mint objektum orientált modellezési és szoftverfejlesztési technikák támogatására használt folyamatleíró nyelv széles körben elterjedt. Hátrányaként említhető viszont túlzott komplikáltsága, sokféle és gyakran redundáns diagramjai, és nem egyértelmű szemantikája.

4. EPC (Event Driven Process Chain)

Az EPC események és funkciók irányított gráfja (24. ábra). Különböző logikai kapcsolatok hozhatók létre benne, melynek révén alkalmas alternatív vagy párhuzamos lefutású folyamatok modellezésére. Ezekhez olyan logikai operátorokat használ, mint az OR, AND és a XOR. Egyik legnagyobb erőssége egyszerűsége és könnyen megérthető logikája. Az EPC felépítése és eszközkészlete alkalmassá teheti a nyelvet logisztikai folyamatok leírására is.

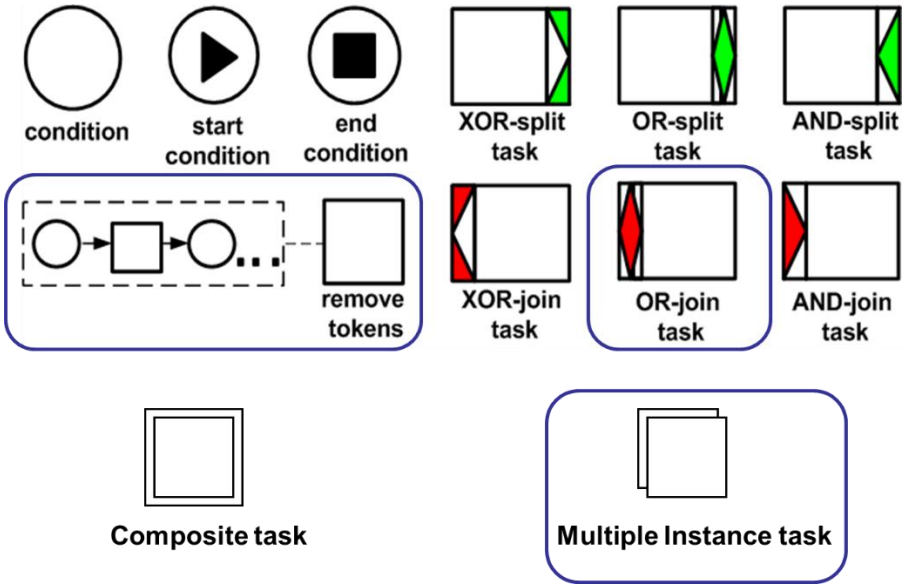


24. ábra EPC gráf

5. YAWL (Yet Another Workflow Language)

Kezdetben csupán a munkafolyamatokban előforduló tevékenységekre koncentrált, de mára már munkafolyamat leírásának teljes eszköztárát támogatja a folyamateditortól a folyamatmotoron át a modellező eszközig. Fejlesztésekkor különböző munkafolyamat mintákat vizsgáltak, és Petri-hálókat vették alapul, amelynek hiányosságait, mint folyamat megszakítás és a több esetű cselekvés lehetőségét kiküszöbölték. Ezt három fő építőelem-típus hozzáadásával érték el: VAGY kapcsolat, megszakítás és több esetű cselekvéssel. (25. ábra)

A YAWL alkalmas dinamikus modellezésre, azaz a bemenő adatok folyamat közbeni meghatározására.



25. ábra YAWL kapcsolati eszköztára megszakítás, több esetű cselekvés kiküszöbölésére

6. BPMN (Business Process Modeling Notation)

A BPMN szabványosított grafikus jelölési mód, mely lehetőséget ad az üzleti élet szereplőinek, hogy folyamataikat standardizált formában tehessék közzé. A grafikus jelölés segítségével kiküszöbölhetőek a résztvevők között felmerülő kommunikáció nehézségek.

A BPMN flowchart technikára épül és számos jelölésrendszert felhasználtak, mint EPC, UML, IDEF, BPSS, ebXML. A BPMN-ben folyamat objektumokat (Flow object), és összekapcsoló objektumokat (Connecting objects) különböztetünk meg. Az objektumoknak három alapeleme van,

- Event- esemény, körrel jelöljük, és ezzel jelezzük, hogy valami „történik” az üzleti folyamat során. Az eseményeknek oka, és hatása van, amivel a folyamatok működését befolyásolják.
- Háromféle eseménytípust különbözetünk meg, attól függően, hogy hogyan befolyásolja az üzleti folyamatot.

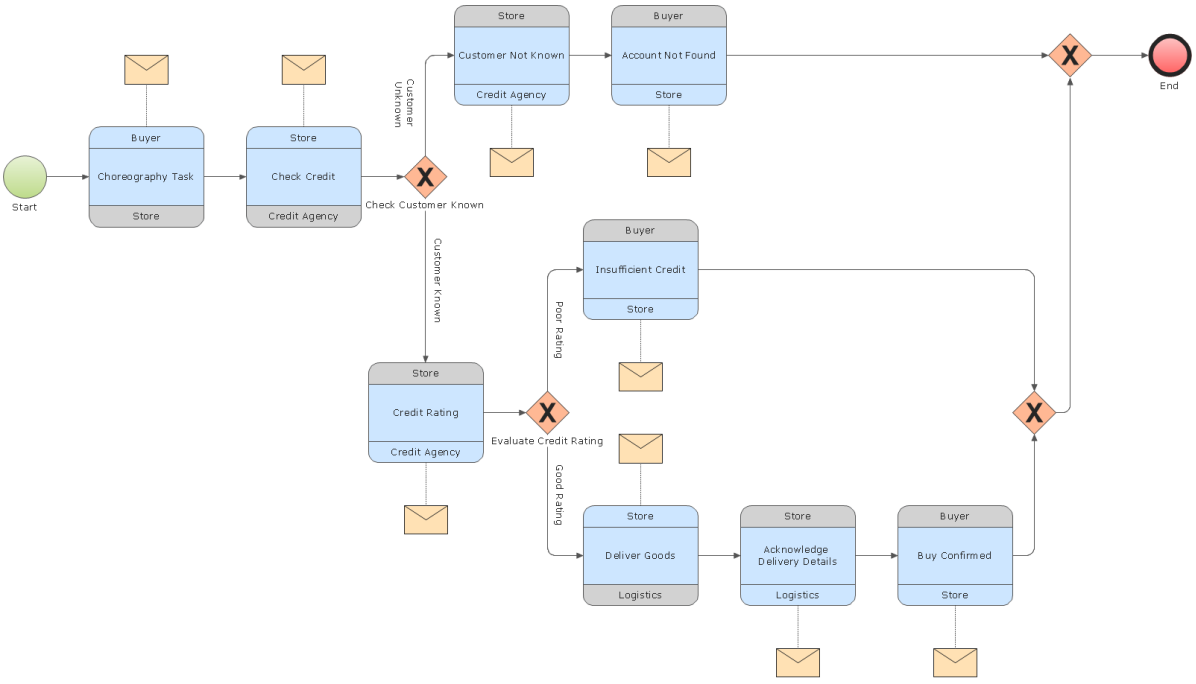


26. ábra Indító, köztes és záró esemény BPMN jelölése

- Activity – tevékenység, A tevékenységeket BPMN-ben kerekített sarkú téglalappal jelöljük, és a szervezet munkáját jelképezik.
- Gateway – átjáró, Rombusszal jelöljük, és döntési pontok, elágazási és összefogó pontokat ábrázolhatunk vele.

Az összekapcsoló objektumokkal pedig a folyamat objektumokat lehet összekötni a folyamat felépítése szerint. Az alapelemeken kívül precízebb modellek készítéséhez rendelkezésre áll a BPMN kibővített elemlistája, ami további altípusok bontva még konkrétabbá teheti az ábrát.

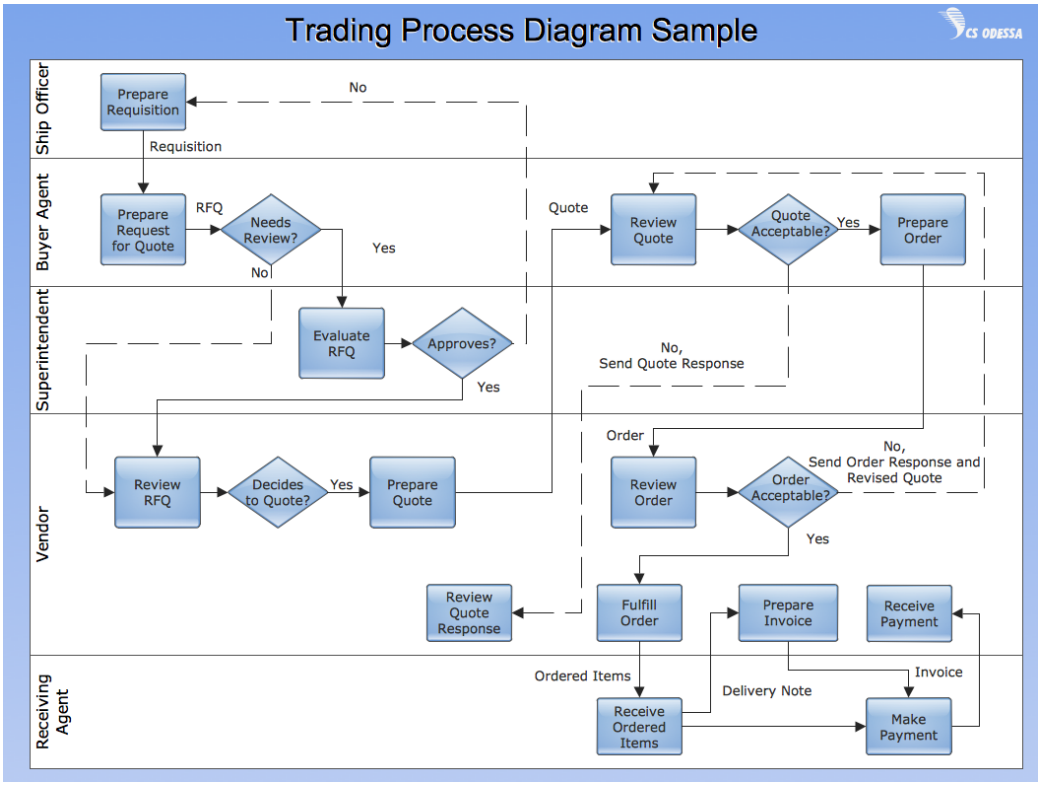
Hátránya, hogy nem tartalmaz eszközt szervezeti felépítés, erőforrások megjelenítésére, nem rendelkezik megszakításokkal, adat- és információmodellekkel, és adattároló elemekkel. Logisztikai alkalmazása erősen elterjedt, ahol komplex üzleti tevékenységeket kell formális keretek közt modellezni.



27. ábra BPMN diagram felépítése [4]

7. BPD (Business Process Diagram)

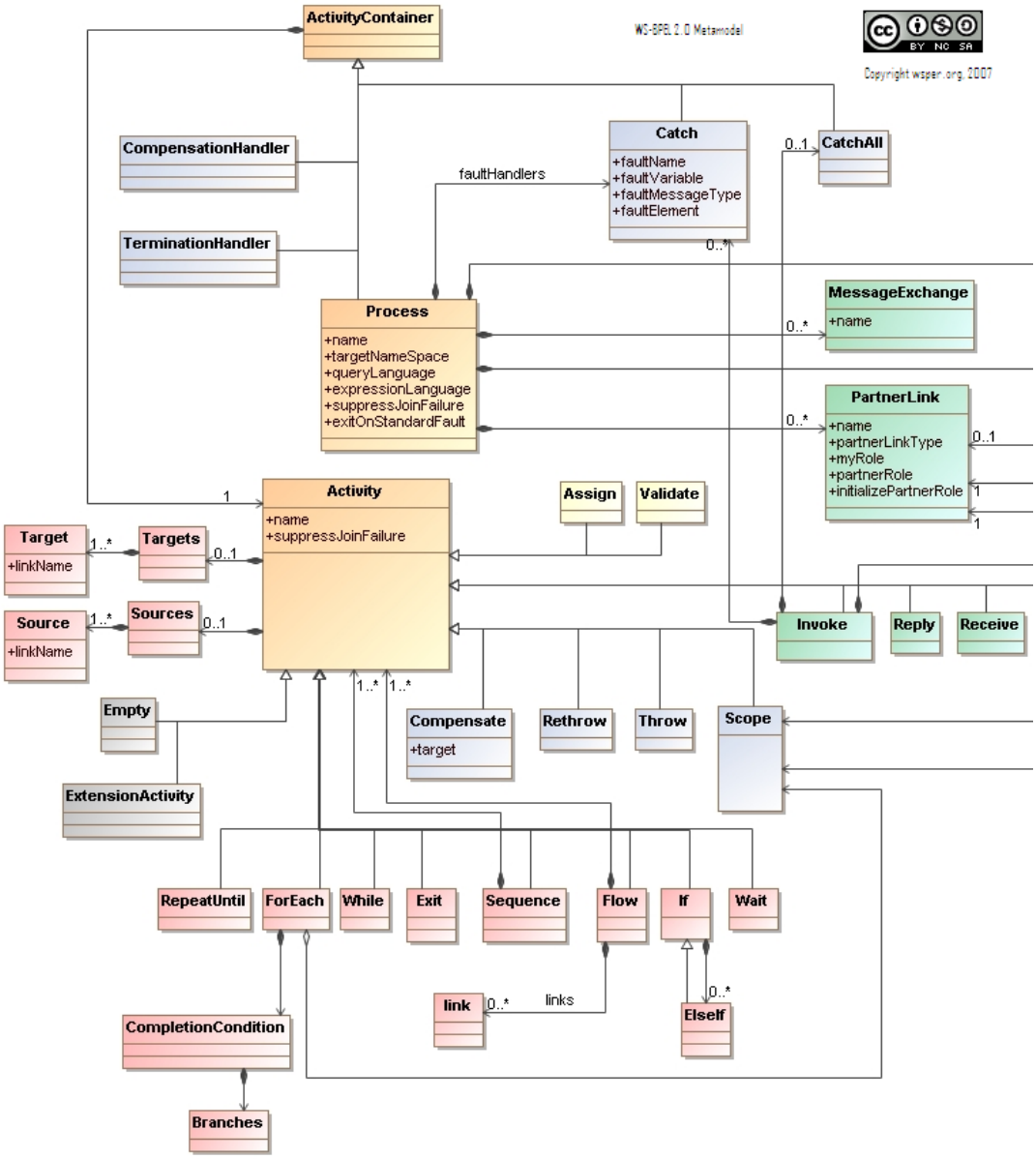
Irányított folyamatok ír le BPMN elemekkel. Alkalmas vállalati tevékenységek leírására. A folyamatok történhetnek szervezetek között (kollaboráció), szervezeti szinten, vagy kisebb egységen belül, ezeket munkafolyamatoknak is hívjuk. A különböző szintű BPD diagramokat érdemes külön ábrázolni, mert a diagram nehezen érhetővé válhat.



28. ábra. BPD diagram a folyamatban résztvevő elemek szintenkénti elkülönítésével [5]

8. BPEL (Business Execution Language)

A BPEL olyan szabványos leíró nyelv, amely üzleti partnerek folyamatainak leírására alkalmas, és webszolgátatáson keresztül segíti a folyamatok összehangolását, végrehajtását. A nyelve egy egyszerűbb programozási nyelvhez hasonló utasításkészlettel rendelkezik, amelyben támogatja a változók kezelését, adatműveleteket, külső folyamat és szolgáltatáshívásokat, de önmagában nem alkalmas semmilyen folyamat elvégzésére. Megjelenítési nyelve az XML, de grafikus ábrázolást nem tesz lehetővé és nem rendelkezik szabványosított folyamattervezési metodikával. Az egyes aktivitások végrehajtása egy külső, kiegészítő nyelven elkészített parancs meghívásával jár, ami leggyakrabban Java.



29. ábra BPEL diagram felépítése

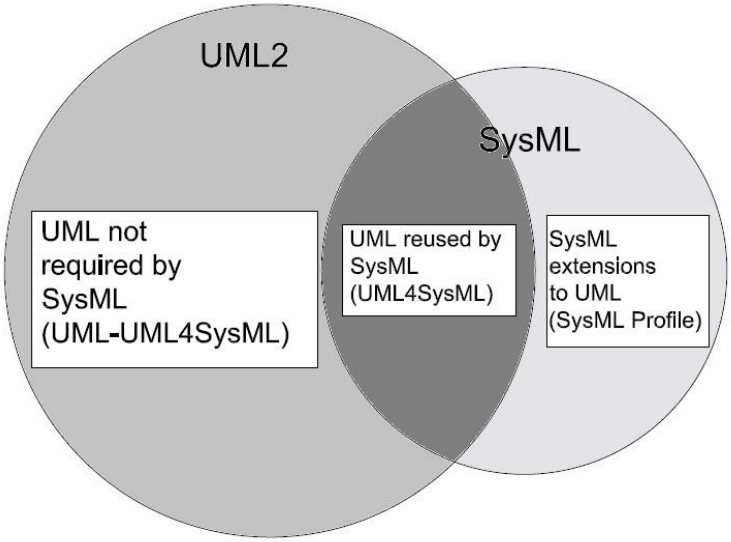
9. XPDL (XML Process Definition Language)

Az XPDL (XML Process Definition Language) a WfMC (Wokflow Management Coalition) által szabványosított leírónyelv. Az XPDL egy olyan XML sémát definiál, ami alkalmas a különböző leíró és modellezési eszközök közötti információcserére. Fontos kiemelni, hogy az XPDL a folyamat leírása mellett a folyamat végrehajtására is tartalmaz információkat. Ezzel szemben a BPEL csak a végrehajtásra koncentrál.

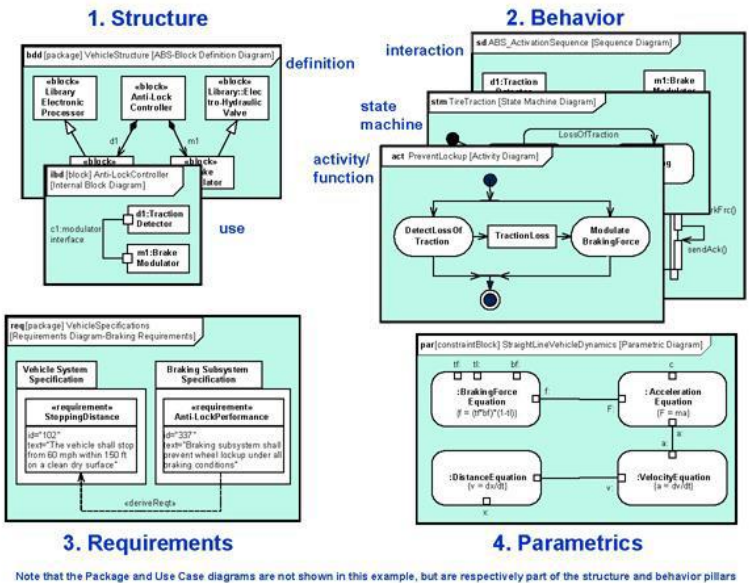
10. SysML (System Modeling Langauge)

A SysML egy általános célú leíró nyelv mérnöki alkalmazásokra kifejlesztve. Specifikációk, analízis, tervezés, felülvizsgálat és jóváhagyás területén nyújt támogatást. A SysML tulajdonképpen az UML kiterjesztett változata. Az UML 2 14 féle diagramtípusa közül hetet használ, ezen kívül további diagramtípusokat adtak hozzá a készletéhez (SysML extensions)

- Blokkok
- Értékek tulajdonsága
- Elrendezés
- Követelmények
- Parametrikus jellemzők
- Folyamatos áramok



30. ábra SysML és UML kapcsolata



31. ábra SysML diagram típusok

Tevékenység:

- Keressen példákat az Interneten az egyes modellek logisztikai alkalmazására!

Önellenőrző kérdések

1. Mutassa be néhány szóban és ábrával az IDEF0 folyamatleíró nyelvet!
2. Mutassa be néhány szóban és ábrával az UML folyamatleíró nyelvet!
3. Mutassa be néhány szóban és ábrával a BPMN folyamatleíró nyelvet!
4. Mutassa be néhány szóban és ábrával az SysML folyamatleíró nyelvet!

4. lecke: Szimulációs módszerek

Cél:

A tananyag célja, hogy hallgató ismerje a szimulációs szoftverek alkalmazásának fontosságát a logisztikai folyamattervezésben, valamint tudja az egyes szimulációs módszerek elvi alapjait (DES, SD, ágens alapú). Legyen tisztában az egyes módszerek alkalmazásának előnyeivel, és hátrányaival.

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- Felsorolni a különböző szimulációs alapokat,
- Megadni a szimulációs alapok főbb jellemzőit,
- Az egyes alapok előnyeit és hátrányait figyelembe véve tudja milyen feladat esetében melyik alkalmazható előnyösebben.

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- Szimulációs platform,
- SD System Dynamics,
- DES, Discrete Event Simulation,
- DELS, Discrete Event Logistic System
- Rendszerváltozók
- ABS Agent Based Simulation,
- SCP Supply Chain Performance
- SCM Supply Chain Management
- LSCM Logistic Supply Chain Management

1. Szimuláció alapja

Definíció szerint a szimuláció a valóságban lezajló folyamatok modellezése, ahol a jellemzők modellezési pontossága a vizsgálatok céljainak megfelelő. A szimuláció céljai pedig a valóság másának megalkotása előre meghatározott szempontok szerint. Így a valós folyamat költséges létrehozása, vagy megváltoztatása nélkül képesek vagyunk következtetések levonni a tervezett állapotról. A költségkímélő tulajdonság, és a nem analitikus, valószínűségen alapuló szimulációt lehetővé tevő számítógépes háttér fejlődése miatt a folyamat szimuláció egyre jobban terjed.

A szimuláció alkalmazása elsősorban ott javasolt, ahol

- Analitikus módszerek nem állnak rendelkezésre, vagy használatuk túl körülményes a folyamat eredményének meghatározására,

- A valóságos rendszeren a kísérletezés veszélyes lenne,
- A valóságos rendszeren a kísérletezés hosszadalmas,
- A rendszer összetettsége miatt áttekinthetetlen,
- A rendszer kimenő eredménye nem észlelhető,
- A valóságos rendszeren kísérletezés, termeléskiesés miatt nem engedélyezett.

Szimulációt sokféle területen alkalmaznak, logisztikai folyamatok vizsgálatán kívül használják még pénzügyi, dinamikai, jármű modellek eredményeinek meghatározására. Egy adott rendszermodell működését három módszer határozhatja meg. Ezek sorrendben lehetnek rendszerdinamikai SD, esemény vezérelt DES, és ágens alapú modell.

2. SD System Dynamics

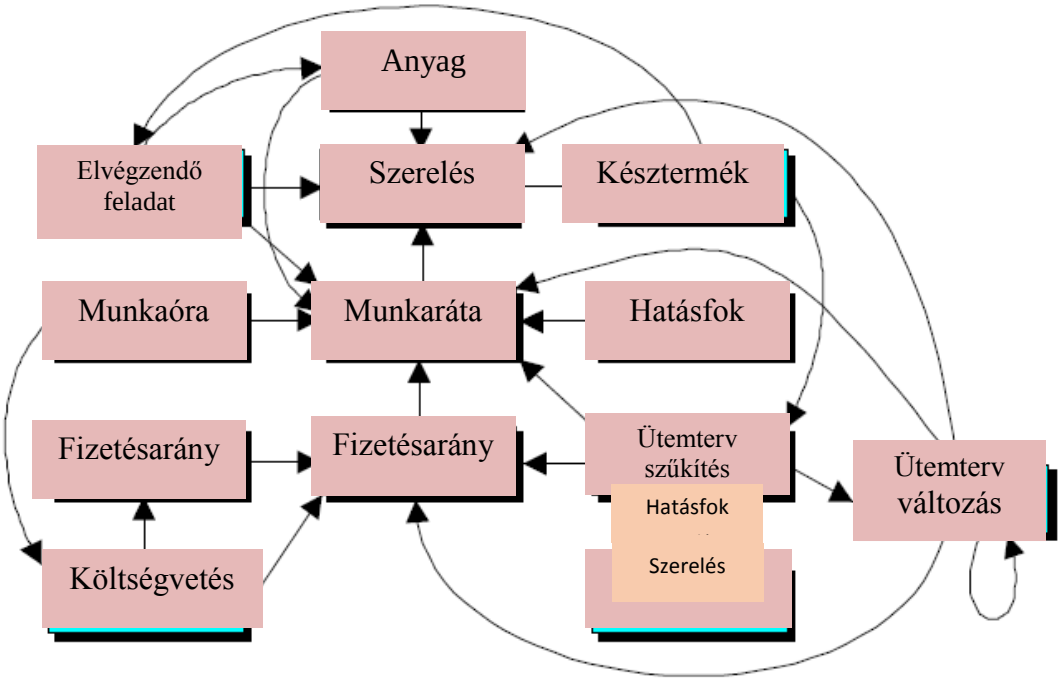
Az SD azaz rendszerdinamikai modellt a vizsgált rendszer összetevőinek változását nézi az idő függvényében. A rendszer, mint elemek halmaza egymással összefüggésben van, és időben folyamatosan változik. Ezért ezt a szimulációs technikát leginkább teljes logisztikai rendszerek időben hosszabb működésének modellezésére használják döntéshozási szinten.

Az SD modell jellemzője,

- A rendszer holisztikus megközelítése, az alrendszerekkel integrálva
- A visszaható hurkok lévén biztosítja a vállalati stratégia rendszerbe épülését
- Minimális adat szükséges a működéséhez
- A rendszer struktúrájára és stratégiára koncentrál

Rendszerdinamikai modell egy jellemző ábrázolási formája a hurok diagram CLD (Casual Loop Diagram) 1. ábra. A CLD a rendszerelemek egymásra gyakorolt hatását, és a változások okait is megmutatja. Az SD modell 3 fő elemből áll, készletekből, ezek áramlásából és késésekből. A 3 elem egymással kölcsönöcs kapcsolatban van, időben folyamatosan hatnak egymásra, változnak. A kimeneti értéket az elemek közti visszaható hurkok sűrűsége, és a hatás sebessége befolyásolja.

A diagramból könnyen, ránézésre kiolvasható a rendszer működése, és összefüggései, ezért gyakran alkalmazzák döntéshozáshoz stratégiai szinten.



32. ábra Hurok diagram (Casual Loop Diagram, CLD)

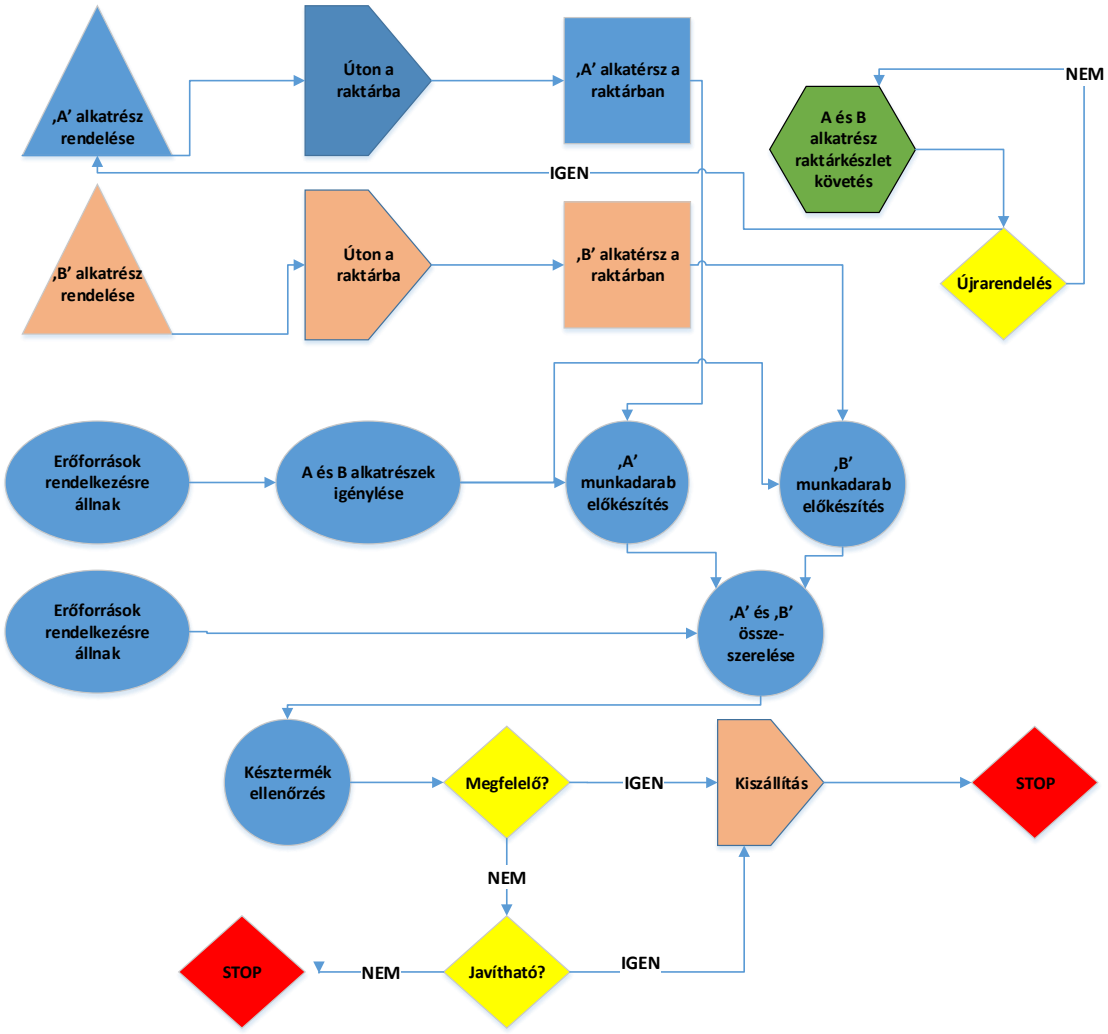
SD modell létrehozása

SD modell alkotásakor első lépésben az egyes elemek közt fennálló kapcsolatokat kell meghatározni, majd a kapcsolatokhoz függvényeket és numerikus értékeket rendelni. A rendszer kimeneti kezdeti értéke után a szimulációs futtatást pontosítani kell, amíg a kimeneti érték nem tükrözi elfogadhatóan a valóságot.

2. DES Discrete Event Simulation, Eseményvezérelt szimuláció

DES azaz eseményvezérelt rendszer modell állapotváltozók és események hálózata, ahol az állapotváltozók diszkrét időközönként változatják állapotukat. Ezek a diszkrét időközök az események bekövetkezésének időpontjai. Az események pedig olyan pillanatnyi történések, amik megváltoztathatják a rendszer állapotát. A rendszer állapotváltozása logisztikai folyamatot tekintve valamilyen változást idéz elő a rendszer erőforrásaiban, ami lehet a rendszerben lévő anyag mozgatása, megmunkálása, összeszerelése, WIP, Work-in-Process tárolása stb. A változás mértéke mindig egységekben mérhető, és a bekövetkezése pedig általában valamilyen tevékenység kezdeti, vagy végpontja.

Egy eseményvezérelt gyártási folyamat ábrázolása látható a 33. ábrán. A rendszer egyes állapotváltozóinak hatása akkor aktiválódik, ha a hozzá tartozó feltételrendszer teljesül. Például ebben az esetben "B" alkatrész akkor lesz a raktárban, ha szállítás feltétele teljesült, azaz "az alkatrész úton" állapotváltozó hatása befejeződött.



33. ábra DES diagram

4. SD és DES összehasonlítása ellátási lánc SC modellezésben

Az ellátási lánc teljesítményének (Supply Chain Perfomance) kutatása egyre több akadémikus, és vállalat érdeklődését keltette fel az utóbbi időben. Az ellátási lánc SC teljesítményét a SC struktúrája, és ennek menedzsmentje (Supply Chain Management) SCM határozza meg. Az SCM magában foglalja az anyag, és információ áramlásának biztosítását a rendszeren belül stratégiai, és funkcionális szinten, oly módon, hogy a költség, és szolgáltatás célok elérjék a korábban kitűzött értéket.

Az SC teljesítményének vizsgálatakor két szempontot mérünk. Az elégedettséget, hogy mennyire sikerül a vevő igényeket kiszolgálni, valamint a hatásfokot, hogy mennyire sikerül a belső erőforrásokat kihasználni. Az ellátási lánc modellezéshez előbb meghatározzuk az SC eleminek tulajdonságát, hogy stratégiai, vagy funkcionális szinten vannak-e jelen. 3. ábra.

A DES alkalmazása modellek objektum orientáltsága miatt akkor javasolt, ha a termelési rendszerhez kapcsolódó gyártási folyamatot kell modellezni, pl. gyártósor WIP meghatározása, vagy a szükséges megmunkáló erőforrás meghatározása.

A DES modellben a vállalati elvek, és visszaható mechanizmusok kevésbé érvényesülnek, ezért hosszú távú döntéstámogatásra inkább az SD modellt alkalmazzák, pl. beszállító kiválasztása.

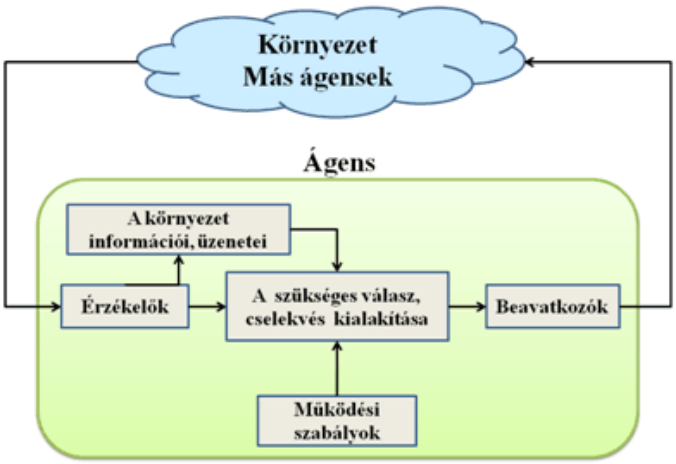
Alkalmazása akkor is javasolt, ha a döntéshozási fázisban a modell részletei nem fontosak, sokkal inkább az egyes folyamatok egymásra gyakorolt hatása.

Kutatási jelentésekből, amik különböző SC-t vizsgáltak kiderül, hogy Logisztikai ellátási lánc menedzsment (Logistic Supply Chain Management) LSCM modellezéséhez leginkább a DES az elterjedt. LSCM feladatok modellezésében, mint az ellátási lánc struktúra tervezés, integrálás, optimalizálás, árukészlet feltöltés, készlet meghatározás és menedzsment, költségcsökkentés, a DES az SD-hez viszonyítva sokkal inkább használt, kivéve a Bullwhip effektust BE ahol a holisztikus hosszabb távú vizsgálat a kedvezőbb.

5. ABS - ágens alapú szimulációk

Az ágens alapú szimulációk a szimulációs modellek legújabb csoportját képviselik. Ágensnek azt az aktív szoftver modult nevezzük, mely a környezetéből beérkező információ hatására, valamilyen beavatkozást végez. A következőkben a logisztikai alkalmazását tekintjük át.

A manapság egyre jobban elterjedő virtuális, és kiterjesztett vállalkozások számára az SCM egyik fő kérdése az SC strukturális változásaihoz való alkalmazkodás. Strukturális változást idézhet elő új termék bevezetése, új partner bevonása vagy az IT fejlődése. Strukturális változást idézhet még elő a kínálat-kereslet megváltozása is. Ezeknek a dinamikus változásoknak a modellezéséhez egy új platformot fejlesztettek ki. Az ellátási láncban lévő virtuális vállalatok hatását szimulálja az ágens-alapú modell. Az ágens alapú modell lényege, hogy a rendszerbe egymástól eltérő ágenseket (tagokat) helyezünk, amik az SC-ben lévő virtuális vállalatokat reprezentálják. Minden virtuális vállalat tevékenységét, azaz hatását az ellátási láncra egy ágens szimulálja. A rendszerben különböző tudású ágenssel modellezett virtuális vállalat különböző terméket állít elő. Minden ágens a saját termékének előállítására koncentrál, miközben egymásra hatást fejtenek ki. Fontos megjegyezni, hogy az ellátási láncon kívül a logisztikai folyamatok a valóságban is ágensekből állnak, targonca kezelő, csomagoló vagy a döntéshozó.



34. ábra Ágensek általános felépítése

Önellenőrző kérdések

1. Hasonlítsa össze a szimulációs modellek típusait!
2. Foglalja össze a SD modellek lényegét!
3. Foglalja össze a DES modellek lényegét!
4. Foglalja össze az ágens alapú modellek lényegét!

Tevékenység:

- Készítsen összefoglaló jegyzetet az egyes szimulációs modellek előnyeiről és hátrányairól.

5. lecke: Numerikus szimuláció a logisztikában

Cél:

A lecke célja, hogy bemutassa azokat a numerikus eljárásokat, melyekkel speciális szoftver igénybevétele nélkül is elvégezhető egy folyamat szimulációja.

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes

- A numerikus szimuláció lényegét visszaadni.
- A tanultakat példán keresztül alkalmazni.

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- Szimuláció
- Anyagáramlás
- Logisztikai rendszer
- Mezoszkopikus modell
- Többcsatornás szűrő

1. Bevezetés

Tevékenység:

A logisztika anyagok, információk, személyek, energia rendszereken belüli és közötti áramlásának tervezése, szervezése, irányítása, ellenőrzése. Az anyagáramlás négy csoportra bontható,

- Gyártás
- Raktározás
- Átalakítás
- Szállítás

Numerikus modellezés során konkrét számértékekre támaszkodunk modellünk létrehozásánál. Az anyagáramlási rendszerek modellezésére előző leckében már bemutattuk az eseményvezérelt DES, rendszerdinamika modellt (SD) valamint az ágens alapú szimulációt (ABS). A numerikus modelleket térbeli és időbeli tartomány szerint felbonthatjuk,

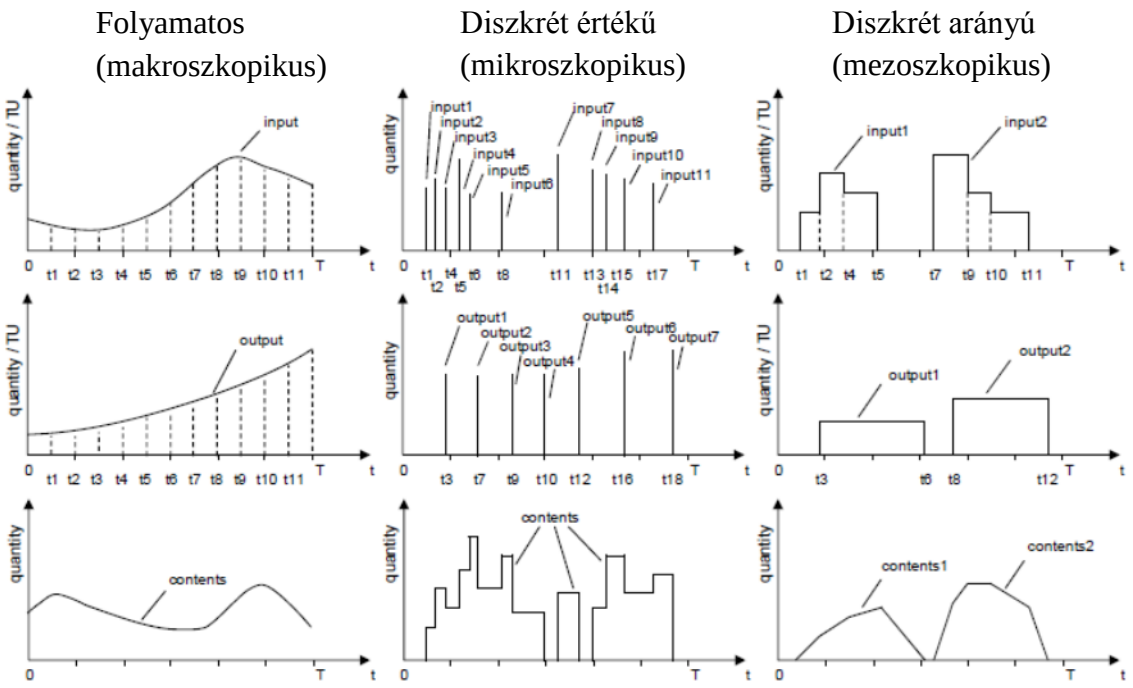
- Globális modellek (makro)
- Regionális modellek (mezo)
- Folytonos, vagy mikroléptékű modellek (mikro)

2. Anyagáramlási rendszerek modellezése mezoszkopikus megközelítéssel

Gyártás és logisztika rendszerek folyamatos rendszer dinamikai modelljeit makroszkopikusnak nevezhetjük, mivel ezek összevont általános modellek, amik nem támogatják a konkrét műveletszintű áramok elemzését. Viszont lehetőséget biztosítanak magasabb, döntéshozási szintű rendszervizsgálatra.

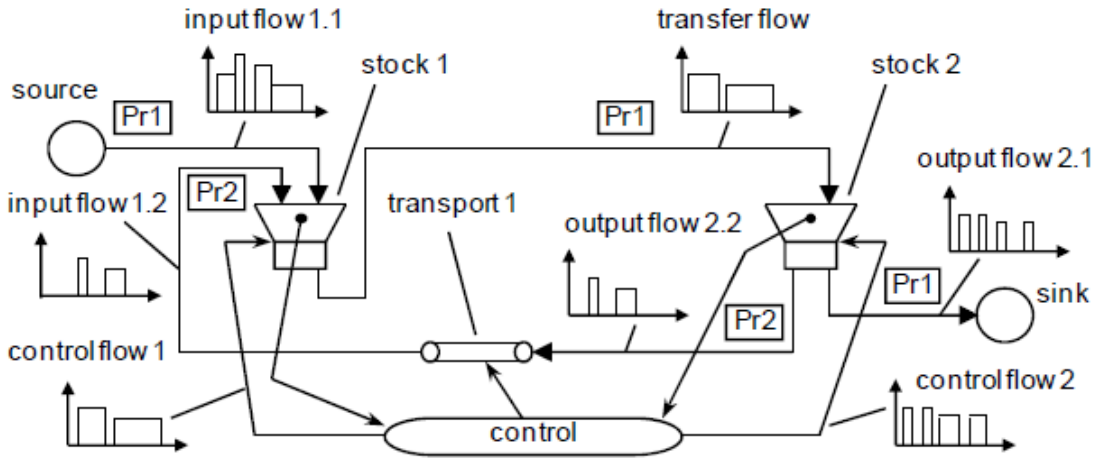
A DES modelleket pedig mikroszkopikusak tekinthetjük, mivel az elemek állapotváltozásait mutatják az idő függvényében diszkrét értékeket felvéve.

A két szimulációs alap között teremt átmenetet a mezoszkopikus modell. A mezoszkopikus modellben a diszkrét időbeli értékek helyett a vizsgált időintervallumban a modell az időintervallum kezdeti, és végértékének arányát veszi fel. A 35. ábrán egyfajta nyersanyagkészlet időbeli változása látható, ahol mindhárom alkalmazott modell fel van tüntetve. A rendszer be- és kimenő anyagáramát a rendszerben végbemenő dinamikai változások határozzák meg.



35. ábra Különböző modellek értékei anyagkészlet változás során [6]

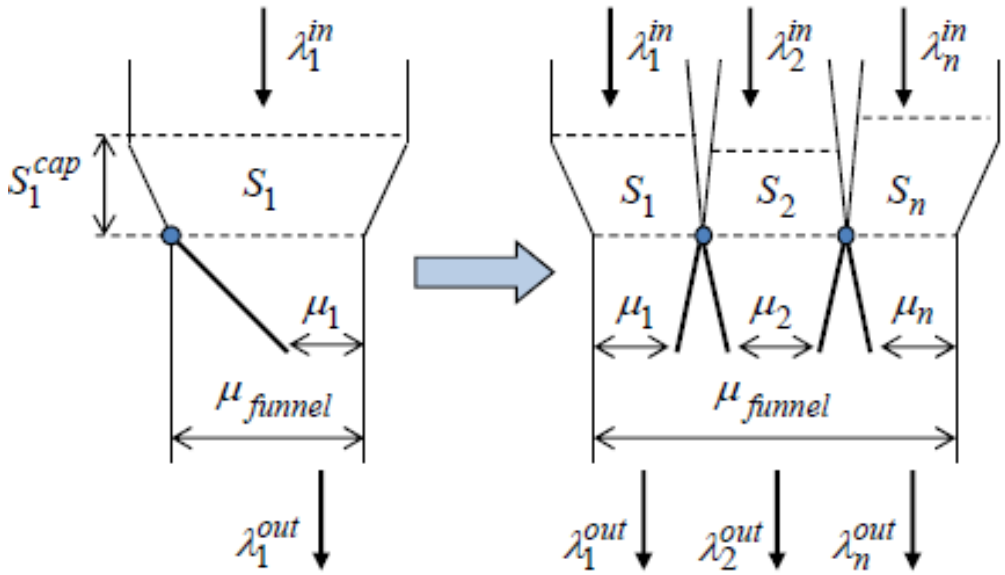
A mezoszkopikus modell lehetőséget ad (diszkrét modellhez hasonlóan) folytonos modell eseményeinek tervezésére, aminek jellemzői nem változnak ezen események között.
A 36. ábrán egy többcsatornás-szűrő mezoszkopikus modell látható. A párhuzamos csatornák használatával lehetőség nyílik több a szűrőben lévő feladat szimultán feldolgozására.



36. ábra Példa mezoszkopikus modell felépítésére [6]

A többcsatornás szűrő a mezoszkopikus modell fő elemeként tekintető. A teljesítménye egy ilyen többcsatornás szűrőnek az XY ábrán látható, ahol $\lambda_i^{in}(t)$ az i-edik csatornába beáramló anyag, $S_i(t)$ az adott időben a szűrőben lévő anyagmennyiség. $\mu_i(t)$ a maximális áteresztő képesség, és $\lambda_i^{out}(t)$ pedig a csatornából kiáramló anyag, ami többi tényezőtől függően három értéket vehet fel.

$$\lambda_i^{out} = \begin{Bmatrix} 0, ha\ \lambda_i^{in} = 0\ és\ S_i = 0 \\ \lambda_i^{in}, ha\ \lambda_i^{in} > 0\ és\ \lambda_i^{in} \leq \mu_i\ és\ S_i = 0 \\ \mu_i, ha\ S_i > 0 \end{Bmatrix}$$



37. ábra Egy és többcsatornás szűrő működése [6]

Egy többcsatornás szűrő a rendszerben jelenthet egy munkaállomást, gyártóterületet, raktárt vagy egy teljes gyártó vagy logisztikai vállalatot. Az egyes szűrők kapacitása közvetlenül függ az adagokban érkező anyagok feldolgozásához szükséges erőforrástól. A feldolgozáshoz kapcsolódó erőforrás lehet fogyasztási (áram, üzemanyag, víz, stb.) vagy újrahasználható (munkás, szerszám stb.). Ezeknek az erőforrásoknak az elosztása a mezoszkopikus modellben algoritmusokkal írható le.

3. Erőforrás elosztási módszerek [6]

Legyen egy kiszolgáló állomásnak legyen (egy többcsatornás szűrő a mezoszkopikus modellben) m számú erőforrás a birtokában, mindegyik egy lehetséges teljesítménnyel jellemezve $pl_k(k = 1, \dots m)$. Ahol a teljesítmény az időegység alatt felhasznált erőforrást jelöli. Így m számú erőforrás összteljesítménye

$$PL = \sum_{k=1}^{m} pl_k$$

A szükséges erőforrás az egyes feladatok feldolgozáshoz legyen $bd_i(i = 1, \dots n)$, ahol n a szűrőben megjelenő feladatok száma. A teljes erőforrás szükséglet tehát,

$$BD = \sum_{i=1}^{n} bd_i$$

A szükséges erőforrások bd_i változó értéke bármilyen elv alapján meghatározható, de leggyakrabban az állomás áteresztő képessége határozza meg $\mu_i(i = 1,..n)$.
Ha a kiszolgáló állomás mindegyik áramhoz az áteresztőképességet rendeli $\mu_i(i = 1,..n)$, akkor teljes erőforrás felhasználás

$$R = \sum_{i=1}^{m} \tau_i \mu_i$$

Itt egy matematikai probléma keretében keressük $\mu_i(i = 1,..n)$értékeit, hogy $R \leq PL$ feltétel teljesüljön.

$$\mu_i = bd_i \frac{PL}{\sum_{i=1}^{n}(vk_i bd_i)}$$

Ahol $vk_i(i = 1,..n)$ az egy elem feldolgozásához szükséges erőforrás átlagos mennyisége.

A módszer használata egy példán keresztül szemléltethető. Tegyük fel, hogy egy repülőtéren dinamikusan szeretnénk változtatni a nyitott check-in pontok számát. A nyitott deskek száma lesz tehát az erőforrás. Ennek nagyságát pedig minden pillanatban a hátralévő idő és a hátralévő utasok száma alapján határozzuk meg.

Önellenőrző kérdések

1. Hasonlítsa össze mikro- mezo- és makroszkopikus modelleket!
2. Foglalja össze az erőforrás elosztás lényegét!

Tevékenység:

- Állítson össze egy saját példát numerikus szimulációkra.

6. lecke: Szimulációs szoftverek típusai

Cél:

A tanagyag célja, hogy megismertesse a hallgatókat a leggyakrabban használatos szimulációs szoftverekkel.

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes néhány jellemző szimulációs szoftver funkcionalitását röviden összefoglalni.

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- Simul8,
- PlantSim,
- FlexSim,
- @RISK,
- Arena

1. Szimulációs szoftverek bevezetés

Tevékenység:

Mint az előző leckében megfogalmaztuk a szimuláció célja a valóság másának megalkotása előre meghatározott szempontok szerint. A szimulációs modell számítógépen futtatásának feltétele a szimulációs szoftver. A szimulációs szoftverek a modell előre meghatározott működését követik, a valóságot legjobban leképező kimeneti értéket generálva. Egy modell eredményét több módszer alapján kaphatjuk meg. Ilyen módszer az előző leckében bemutatott DES, SD, és ABS módszerek. Ezenkívül sztochasztikus folyamatok modellezésére használják még a Monte-Carlo módszert, valamint különféle valószínűség számítási módszereket. A szimulációs szoftverek által alkalmazott platformok összefoglalva,

- SD System Dynamics,
- DES, Discrete Event Simulation,
- Agent Based Simulation,
- Valószínűség számítás
- Monte-Carlo módszer

Ha determinisztikus folyamatról beszélünk, akkor a feladat analitikusan is megoldható megfelelő paraméter ismeretében.

A szoftver modell platformján kívül fontos jellemző a gyártó és anyagáramlási rendszer számítógépes szimulációs modelljének összetevői,

- Objektum,
- Tevékenység,
- Események,
- Sorok,
- Sajátosságok,
- Beállítások.

A leckében a logisztikai folyamat szimulációs programokon kívül más területen is elterjedt szimulációs szoftvereket is bemutatunk, bizonyítva a különböző folyamatok szimulációjának fontososságát. Jelen leckében az elsődlegesen logisztikai folyamatok szimulációjára kifejlesztett programok közül a Simul8, PlantSim és FlexSim szoftvereket ismertetjük részletesen.

Szimulációs szoftvereket a logisztikai minden területén alkalmaznak az optimális folyamatok eléréséhez. Ezen belül mégis fontos megemlíteni néhány főbb területet,

- Gyártás tervezés,
- Gyártás erőforrás allokáció,
- Raktározási folyamatok,
- Raktár tervezés,
- Raktárkészlet meghatározás,
- Ellátási lánc,
- Szállítás, anyagmozgatási folyamatok.

A következő fejezetekben a logisztikai alkalmazáson kívüli szimulációs szoftvereket mutatunk, be valamint három logisztikai folyamatok szimulálására optimalizált programot.

2. Szimulációs szoftverek csoportosítása

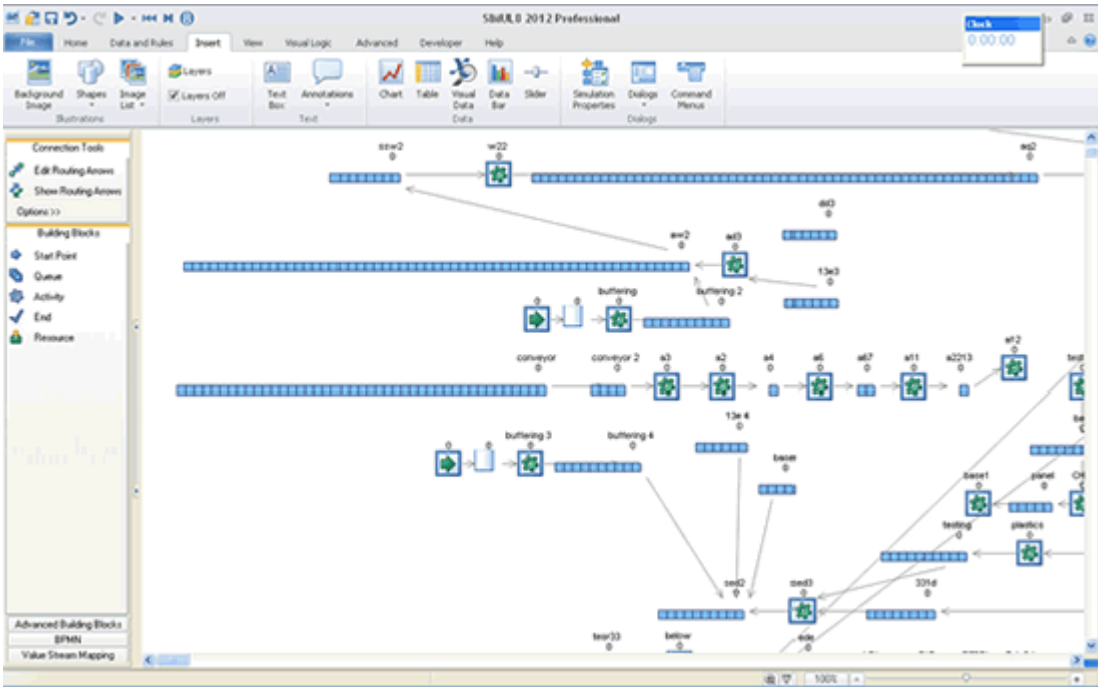
A szimulációs szoftvereket több tulajdonság alapján lehet csoportosítani. A szimulációs platform szerinti csoportosítás túlságosan nagy halmazokat enged a konkrét szeparációhoz, ezért jelen füzetbe a szoftvereket az alábbi főcsoportra bontjuk.

- Szoftver alkalmazása problémáknál
- Felhasználási terület
- Modell alkotás jellemzői

3. Simul8

A gyártás és anyagáramlási modellkészítés egyik ismert programja a Simul8. Alkalmazása gyártási területen többek között átfutási idő, erőforrás felhasználás optimalizálás, a gyártási folyamat szűk keresztmetszet, sorozat nagyság meghatározásakor előnyös. Könnyű folyamat elemző tulajdonsága miatt használják új termelési rendszerek bevezetésének vizsgálatára. Ezenfelül alkalmas gyártási költségek elemzésére is.

A Simul8 program kifejezetten a nagyvállalati igények számára lett kifejlesztve. Fejlesztésekkor fontos szempont volt a kommunikáció, és integráció vállalati információs rendszerekkel, CAD alapú műszaki tervező programokkal, technológiai adattárakkal, adatbázisokkal (SQL).



38. ábra Simul8 modell építő kezelő felülete.

Szoftver alkalmazási területei

A szoftver három fő alkalmazási területét vizsgáljuk meg sorrendben gyártás, szállítás-raktározás és a logisztikai alkalmazáson túlmutató üzleti folyamatok tervezését.

1. Gyártás

A gyártás területén a Simul8 szoftvert az alábbi problémák elemzésére használják

- Gyártósorok, és kapacitások valamint átfutási idők elemzésére,
- Új termelési rendszerek bevezetésének kérdésköreinek elemzésére,
- Új gyártósorok telepítése előtti vizsgálatára,
- Szűk keresztmetszetek vizsgálatára,
- Optimális gyártási sorrend, sorozatnagyságok meghatározására,
- Munkaerő-szükséglet elemzésére,
- Logisztikai folyamatok vizsgálatára,
- Gyártási költségek elemzésére.

2. Szállítás - Raktározás

- Optimális járműkihasználtság,
- Be- és kitárolási feladatok ütemezésére,
- Készlet, és raktárméret meghatározására,
- Anyagellátási folyamat tervezésére.

3. Üzleti folyamatok

- Ügyfélkiszolgálás elemzése,
- Alkalmazottak számának meghatározására,
- Munkafolyamatok újratervezése.

Főbb jellemzői

Mivel a Simul8 a viszonylag egyszerű kezelés miatt a felsőoktatásban igen elterjedt folyamat szimulációs szoftver. Több egyetem alkalmazza saját használatra oktatási céllal, illetve kutatási projektek megbízóinak folyamatainak modellezésére, optimalizálására. Az elterjedt alkalmazás miatt fontos részletesen kitérnünk a Simul8 szoftver főbb tulajdonságaira, és előnyeire.

1. "Mi lenne, ha..." szcenário készítése

Több megvalósítási ötlet eredményeinek, és hatásainak vizsgálatára nyílik lehetőség, anélkül, hogy időt és pénzt fektetnénk a tényleges fizikai megvalósításra

2. Problémafeltárás

A gyártósor, vagy logisztikai rendszer telepítése előtt már ki lehet szűrni az esetleges problémákat az áramlásban (szűk keresztmetszet, gyenge pont)

3. Vizuális modellezés és elemzés

Dinamikusan és vizuálisan mutatja a szimulációs eredményeket, a változások hatásait.

4. Költségelemzés

A gyártási folyamat és az abban történt változtatás hatásainak költség- és profit elemzése.

5. Kommunikáció a vállalati információs rendszerrel

Adatcsere az SQL alapú adatbázisokkal, Microsoft Excel és Visual Basic alkalmazásokhoz elérhető interface.

6. Interaktív Simul8 oktatás modell mintákkal

Előre definiált mintákkal segíti a gyors modell építést, és tanulást.

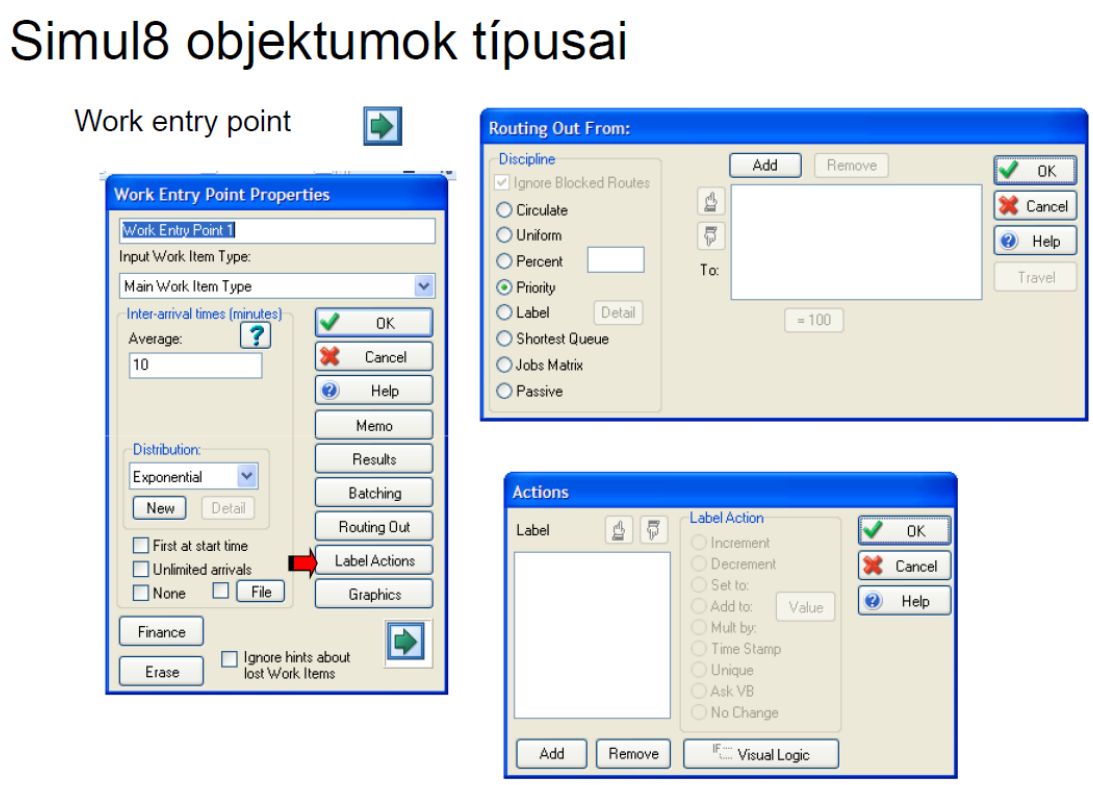
A Simul8 működése

A Simul8 alkalmazott objektumai a következőek

- Work entry point,
- Storage bin,
- Work center,
- Resource,
- Conveyor,
- Tanks and pipes.

Az objektumok működésük szerint külön konfigurálhatóak. Például a work entry point, mint anyag belépés objektumában a beérkezés átlagos időköze, eloszlása határozható meg. A

tulajdonság beállítás panelben van lehetőség az objektum kapcsolati rendszer logikájának meghatározására is.



39. ábra Simul8 objektum típusai [7]

4. @RISK

A @RISK szimulációs program Monte –Carlo szimulációs technikát alkalmazva egy táblázatba leírt modell lehetséges eredményeit, és azok előfordulási valószínűségét vizsgálja. Az egyes eredmények előfordulási valószínűségéből megállapítja a modell lehetséges forgatókönyveinek előfordulási kockázatát. Az egyes kockázati tényezők ismeretében, el döntjük dönteni melyik kockázatot vállaljuk, és melyiket kerüljük a siker érdekében.

Szoftver alkalmazási területei

A szoftver, mint folyamatok, és döntések eredményeinek sikeres kockázat csökkentő módszere, igen elterjedt. A @RISK kockázatelemző szoftvert minden területen alkalmazzák, ahol a kimeneti eredményt befolyásoló tényezők az időben változhatnak, azaz bizonytalanok. A bizonytalansági tényező a folyamatra vonatkozó döntéseket teszi kockázatossá. A kockázat, mint a folyamat különböző forgatókönyveinek előfordulási valószínűsége a jövőben a @RISK szoftverrel meghatározható. Elsődleges ipari felhasználási területei az alábbi táblázat tartalmazza.

felhasználási terület	Alkalmazási példa
Pénzügy	Kockázati érték meghatározás
	Nyugdíj tervezés
	Deviza értékelés
	Portfolió optimalizálás
Biztosítás	Kifizetetlen követelés felbecslés
Energia ipar	Olaj tartalék felbecslés
	Beárazás
Minőségbiztosítás, Six Sigma	DMAIC (Define, Measure, Analyze, Improve and Control) üzleti folyamat fejlesztés
	Vevői szolgáltatás fejlesztés
Gyártás	Gyár bezárás esettanulmány
	Új termék bevezetése
	Termék életciklus elemzés
Egészségipar	Megbetegedési becslés
	Új termék bevezetése
Kormány	Energiaforrás elrendezés
	Háborús esettanulmányok
Környezet	Veszélyeztetett fajok tanulmánya
	Környezetszennyezési becslések

1. táblázat @RISK felhasználási területei alkalmazási példákkal

Főbb jellemzői

A szoftver főbb jellemzői, előnyök

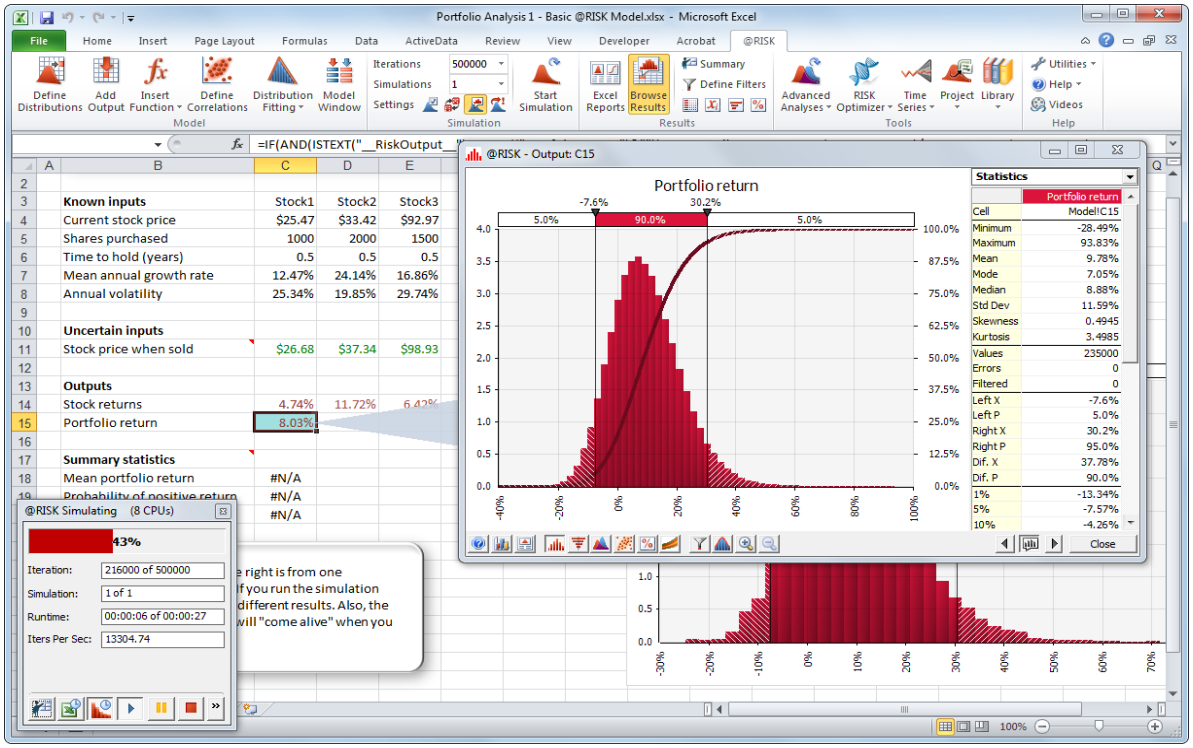
- Grafikus interfész
- Real-time modellezés
- Alkalmas Excel táblázatok kezelésére
- Excel VBA integráció

- 50 beépített valószínűségi függvény
- Genetikus algoritmuson alapuló optimalizáció

A szoftver előnyei között említhető a nagyfokú integráció Excel táblázatkezelővel, ami lehetővé teszi korábban létrehozott pénzügyi vagy egyéb táblázat modellek eredményeinek utólagos kockázatelemzését. A grafikus interfész és függvény megjelenítés segíti a folyamatot legjobban modellező eloszlási függvény kiválasztását. A program továbbá alkalmas a kimeneti eredmények előfordulási kockázatának ismeretében az optimális, azaz legkisebb kockázatú döntés meghatározására.

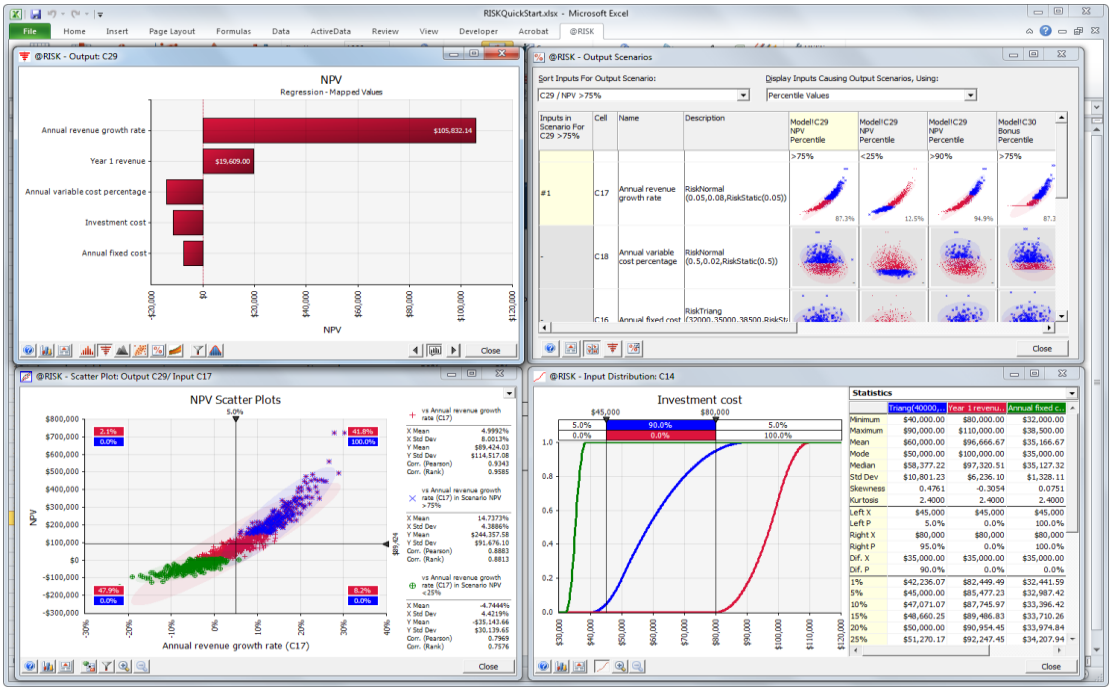
A @RISK működése

@RISK-el végzett elemzésnek három fő lépése van. Első lépésben a kész modellben használt bizonytalansági értékeket helyettesítjük a @RISK által felkínált több mint 50 lehetséges valószínűségi függvénnyel. Ezeknek az Excel alapú eloszlási függvénynek az alapján számítja ki a lehetséges kimeneti értékeket, egy előre definiált paraméter helyett.



40. ábra @RISK felhasználói felülete pénzügyi kockázat meghatározásra

A kimeneti cellánál beállítjuk az eloszlási függvényt, amit, a felhasználóbarát funkciót szem előtt tartva grafikus adatbázisból választhatunk ki. A szimuláció futtatása után következik a kockázat kielemzése. A program az egyes eredmények előfordulásának valószínűségét különféle diagramokban ábrázolja pl. hisztogram, halmozott görbék, szóródási, dobozdiagram.



41. ábra Kockázat kielemzése @RISK felületen

5. Arena szimulációs szoftver

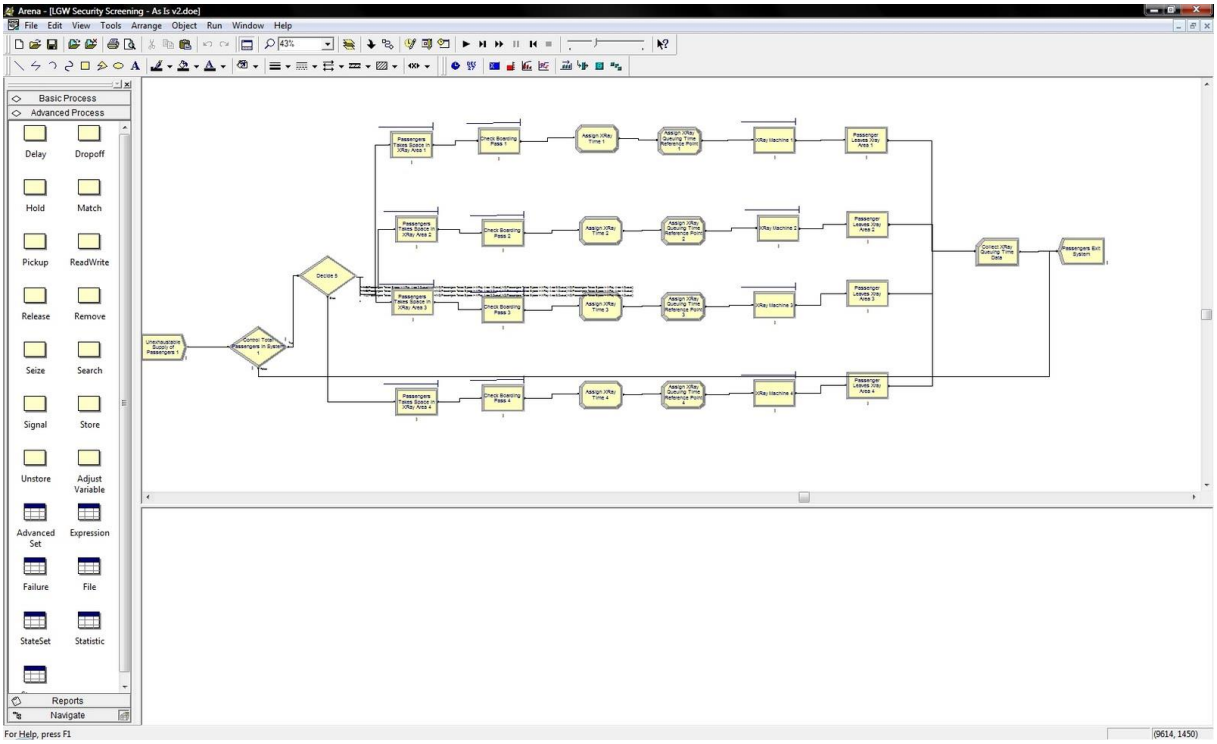
Az Arena a 2000-es években kifejlesztett DES, azaz esemény alapú szimulációs szoftver. A szoftverben a felhasználó objektumokat hozz létre, majd ezek összekapcsolásával modellezi a folyamatot, vagy a logikát. A DES technika ismertetésében (3. lecke) részletesen bemutattuk az egységek kapcsolatának jelentőségét. Az ezen alapuló Arena szoftverben is az egységek közötti definiált kapcsolat írja le a modellezni kívánt valós folyamat viselkedését.

Szoftver alkalmazási területei

Az Arena szoftver alkalmazásával olyan fontos kérdésekre kapunk választ folyamataink tervezésénél, mint

- Hogyan érhetünk el jobb teljesítmény célokat?
- Mikor szükséges csökkenteni, vagy növelni az erőforrásokat?
- Milyen hatásai lesznek egy döntésnek?
- Hogyan reagáljuk váratlan vészhelyzetekre, abnormális szituációra?

Ezek a kérdések az üzleti folyamatok tervezésénél elsődleges és gyakran felmerülő kérdések, ezért a szoftvert mind logisztikai, és üzleti folyamatok tervezése során használják. Emellett egészségügyi alkalmazása is fontos, mint az állami egészségügyi rendelkezésekre reagálás, vagy betegellátásban a sorban állási idő szimulálása. Az objektum tárban található elemekből könnyen drag-and-drop elven építhetjük fel a modellünket. Minden szimulációs modellhez 2D vagy 3D animáció is elérhető a jobb átláthatóságért.



42. ábra Arena szoftver eszköztára modell felülettel.

Fontos felhasználási területe még a programnak a kormányzati és katonai alkalmazás, mivel DES eseményvezérelt platformja lehetővé teszi a "mi történik, ha?" ("What-if?") kérdésre a válaszadást. Alkalmazási területei a kormányzati, és katonai területen belül,

- Vészhelyzeti tervezés,
- Katonai eszköz javítás, és karbantartás szimulációja,
- Törvények, és rendelkezések hatásainak szimulációja.

Főbb jellemzői

Folyamatábra modellezéshez a szoftver beépített előre meghatározott építő blokkokat tartalmaz, szükségtelenné téve a bonyolult egyedi programozást. A folyamat változékonyságát modellezni tudó széles statisztikai eloszlásfüggvények állnak rendelkezésre, hogy a lehető legpontosabb eredményt kapjuk. A program fontos jellemzője, hogy az eredmények 2D és 3D formában is megjeleníthetőek a könnyebb megértés érdekében. A szoftver gyártási folyamat optimalizációra is alkalmas, mint megfelelő számú gyártásközi pufferek meghatározása.

Alkalmazásának előnyei között említhető

- Probléma feltárás és megoldás,
- A szűk keresztmetszetek gyors meghatározása,
- Pénzügyi előrejelzések pontonosságának fejlesztése,
- Átfutási idő csökkentése ezáltal nagyobb termelékenység elérése,

- Jobb erőforrás kihasználás, valamint raktárkészlet optimális meghatározása, költségcsökkentés,
- Hibaráta negatív hatásainak kielemzése.



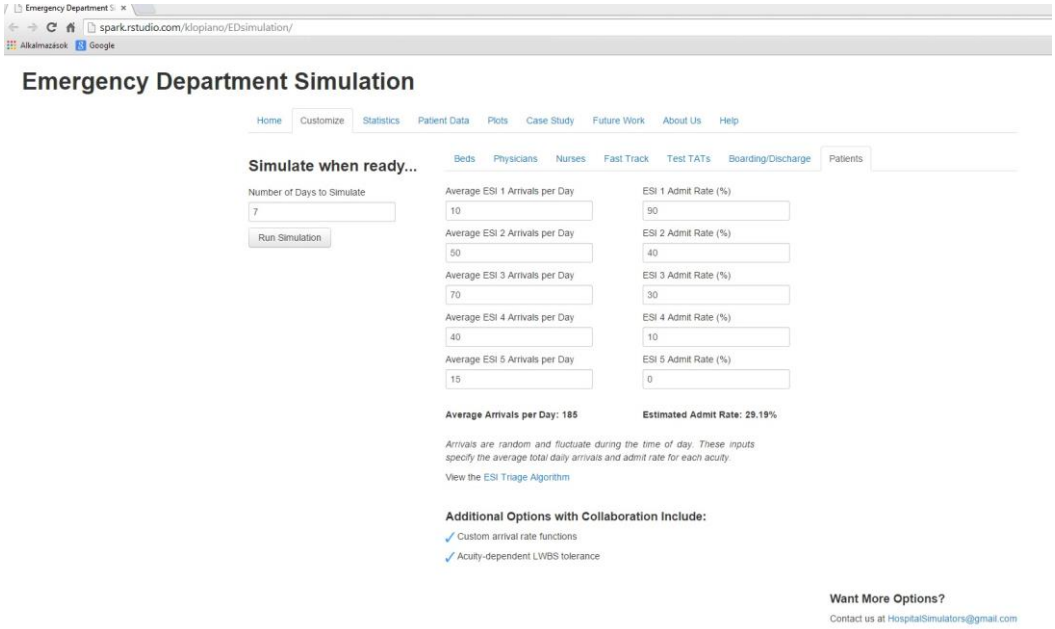
43. ábra Arena logisztikai alkalmazásban, fuvarok közlekedési ágazat bontásban szimulálva.

6. Emergency Department Simulator (Sürgősségi betegellátó szimulátor)

Az Emergeny Department Simulator (ED Simulator) egy olyan átfogó szimulációs eszköz, amellyel egy sürgősségi osztály dinamikus betegáramlását lehet modellezni.

A szimuláció online felületen keresztül érhető el, könnyű hozzáférést biztosítva a világ bármely pontjáról. A modellben a betegellátás minden fontos paramétere előre meghatározható, mint

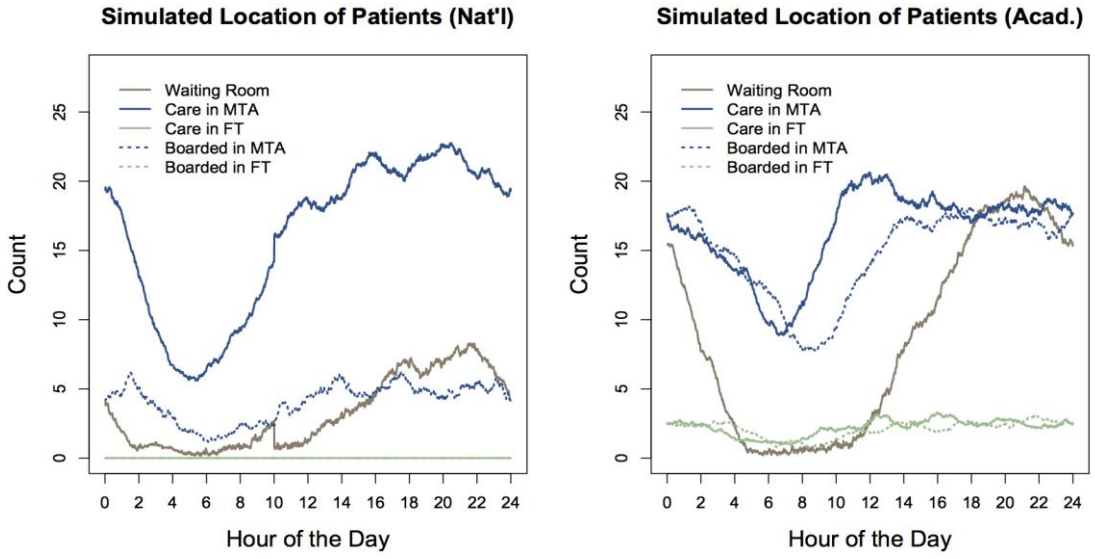
- Elérhető betegágyak száma,
- Orvosok, és ápolók munkaideje,
- Labor tesztek ideje lelet kiadáshoz,
- Beérkező betegek sérülés szerinti csoportosítása,
- Betegek érkezésének várható értéke.



44. ábra Szimuláció paraméter beállítás online felületen

A szimulációs végrehajtása után a folyamat kimeneti paraméteri között információt kapunk többek között a,

- Betegek helyzetéről idődiagramban ábrázolva,
- A betegellátó erőforrásainak kihasználásáról (orvosok, ápolók, betegágyak stb.),
- Várakozó idők alakulásáról,
- A betegellátáshoz köthető összes költség alakulásáról.



45. Ábra A betegek helyzetének idődiagramban való ábrázolása

5. PlantSim

A PlantSim anyagáramlási szimuláció végrehajtására alkalmas rendszer. A többi korábban ismertetett szoftverrel ellentétben a PlantSim kifejezetten a gyár, gyártósor és ehhez

kapcsolódó folyamat szimulációra és optimalizációra lett kifejlesztve. Bemutatása azért, is fontos, mert gyártástervezés terén komplex megoldást nyújt, és az egyik legelterjedtebb szoftver.

DES, azaz diszkrét, esemény-vezérelt szimulációs rendszerről beszélünk, ami a többi szoftverhez hasonlóan a folyamatosan előre haladó időben csak a lényeges események diszkrét pillanatait vizsgálja. Ilyen esemény lehet például, amikor a munkadarab megérkezik egy szállítószalag elemre, vagy elhagyja azt.

Szoftver alkalmazási területei

A gyártási folyamat Plant Simulation alapú szimulációjával kimutathatók az alacsony vagy túlzott raktárkészletek, az alul vagy túlterhelt szállítószalagok, és a rendszer szűk keresztmetszetei, amelyek a gyártás hatékonyságának visszaesését eredményezhetik. Használatával optimalizálható az anyagáramlás, az erőforrás felhasználás és a logisztika a gyár minden szintjén a teljes gyártóüzemtől a helyi gyártó sorokig.

Példaként egy esett tanulmányt mutatunk be, ahol a gyártási logisztika folyamatain hajtottak végre egy elemzést a kamionforgalom csökkentésére a gyár területén Plant Simulation alapon, aminek az eredményeként 10%-kal csökkent a kamionok száma a gyár területén, 12,7%-kal csökkent a kamionok által a gyár területén töltött idő, és jelentősen csökkentek a belső közlekedés várakozási idői.

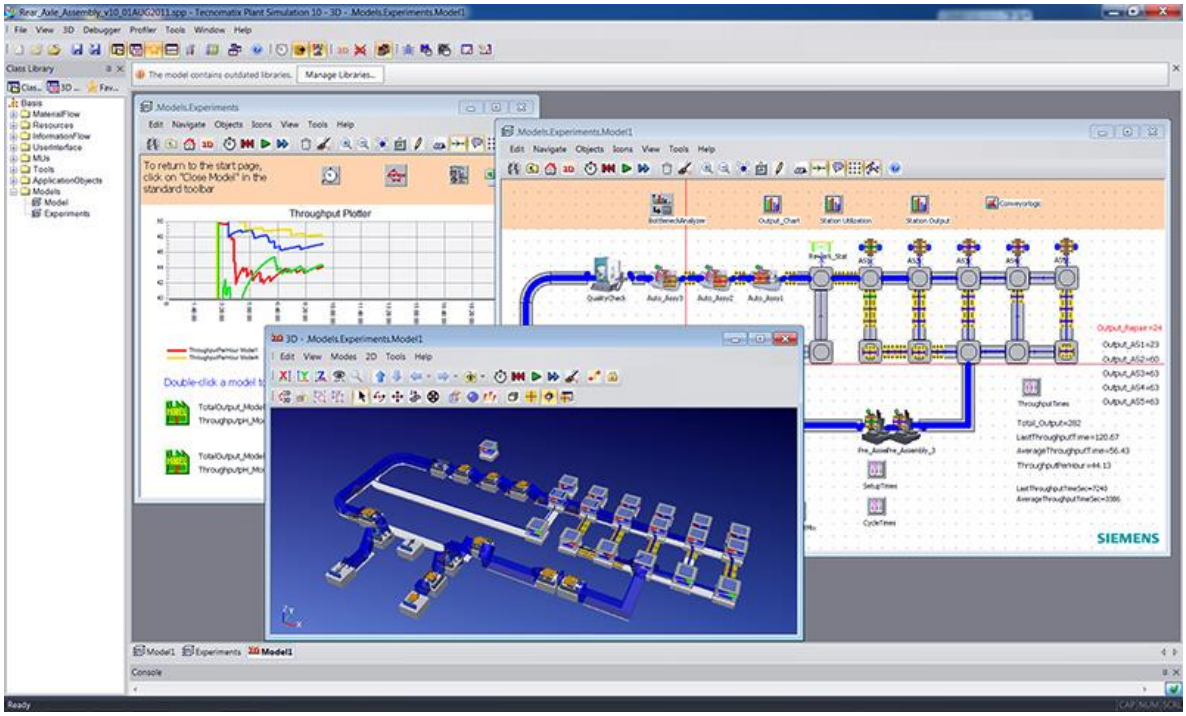
A fent említett előnyei miatt számos világméretű termelővállalat alkalmazza a szoftvert gyártási folyamatainak optimalizálására. A PlantSim több iparágra biztosít egyedi specifikációjú csomagokat, az adott területhez kapcsolódó objektum könyvtárral, mint

- Kikötői folyamatok modellezéshez,
- Raktár és logisztikai modellezéshez,
- Értékáram elemzéshez,
- Járműgyártás modellezéshez.

Főbb jellemzői

- Komplex gyártórendszerek és vezérlési rendszereik szimulációja
- Objektum-orientált, hierarchikus gyármodellek készítése, beleértve az üzleti logikát, a logisztikát és a gyártási folyamatokat
- Objektum könyvtárak a legjellemzőbb modellezési feladatok gyors, hatékony elvégzéséhez
- Grafikonok és diagramok a kimenet, az erőforrások és a szűk keresztmetszetek elemzéséhez
- Átfogó analíziseszközök, beleértve az automatikus szűk keresztmetszet elemzést, a Sankey-diagrammokat és a Gantt-grafikonokat
- 3D megjelenítés és animáció
- Az angol és a német mellett magyar nyelvű felhasználói felület is elérhető
- Genetikus algoritmus a rendszerparaméterek optimalizálásához

- Nyitott architektúra, integrációs lehetőségek (ActiveX, CAD, Oracle, SQL, ODBC, XML, stb.)



46. ábra PlantSim felhasználói felülete

5. FlexSim

A FlexSim szoftvercsalád DES alapú szimulációs és vizualizációs szoftvereket foglal magában, amiket az üzleti, és gyártási folyamattervezés különböző területein széles körben alkalmaznak. Menedzserek, döntéshozók munkáját könnyíti meg a vizsgált folyamat, rendszer ábrázolása dinamikus élethű 3D környezetben. A 3D vizuális megjelenítés lehetővé teszi, a modell egyszerű megértését, és könnyen azonosíthatóvá a szűk keresztmetszetek, leállási hibalehetőségek, és megszakításokat. A már bemutatott Arena szoftverhez hasonlóan itt is lehetőség van a "what-if?" szcenárió megválaszolására.

Szoftver alkalmazási területei

Folyamatok és rendszerek széles skálájának modellezésére alkalmas, a felhasználási területeket, és azokra alkalmazási példákat az alábbi XY. Táblázat foglalja össze

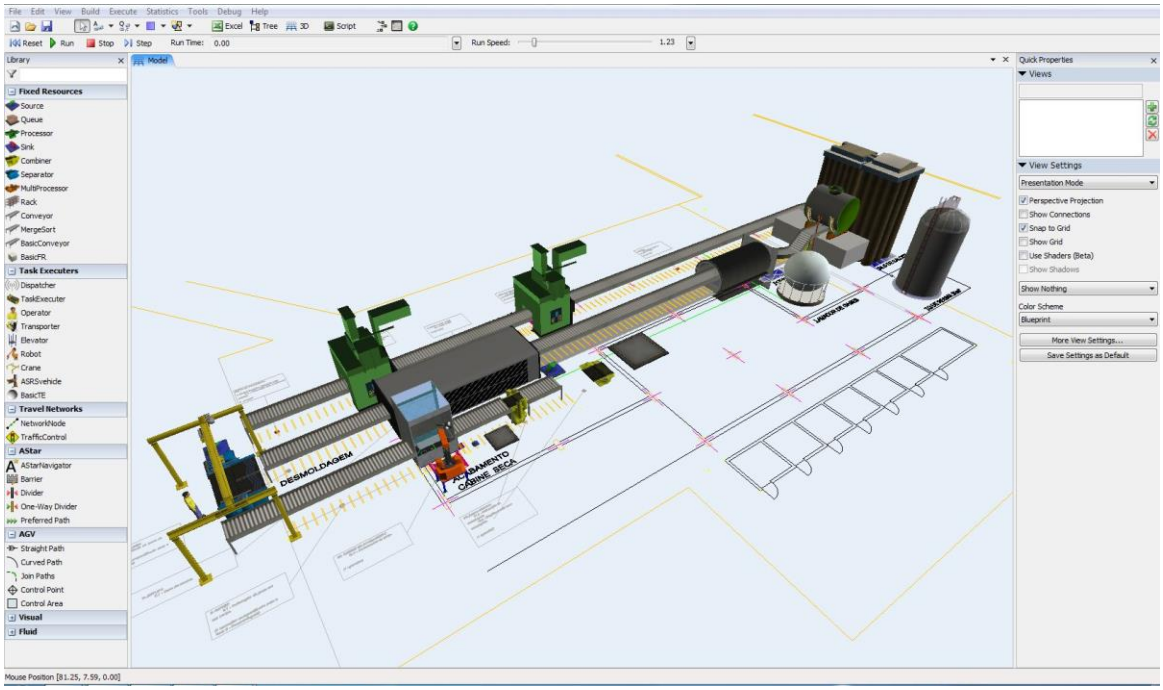
Gyártás	Acél gyártás, Gyártás műveletek, mint töltés, címkézés, csomagolás stb. szimulációja, Elektronikai hardver eszközök gyártása,
Raktározás és elosztás	Be és kirakodás konténerszállító hajókról kikötőben, Konvejor rendszerek elrendezése, Elosztó központok műveleteinek

	szimulációja, Order picking,
Szállítás	Autópálya csomópont forgalom áramlásának szimulációja, Forgalom torlódás határátkelőknél,
Egyéb	Bányászat és feldolgozás, Ételkészítés, és kiszolgálás gyors- étterem láncokban, Látogatók áramlása vidámparkban, Adatáramlás közös online tároló helyen (Shared-Acces Network Storeage SANS)

2. táblázat. A Flexsim szoftver felhasználási területei alkalmazási példákkal

Főbb jellemzői

A teljes körű 3 dimenziós szimuláción kívül fontos jellemzője a szoftvernek a könnyen átírható bemeneti változók, amik lehetővé teszik többféle forgatókönyv szimulálását rövid idő alatt.



47. ábra FlexSim kezelőfelülete automata anyagmozgató rendszer és munkaállomás kapcsolatának modellezésére.

7. lecke: Anyagáramlási csomópont terminál modellezése

Cél: Ez a tananyag terjedelmi okokból befejező fejezete. Célja, hogy egy átfogó példán bemutassa az előző leckékben tanultakat

Követelmények:

Ön akkor sajátította el megfelelően a tananyagot, ha képes a példát átlátva a megvalósítás célját és lépéseit összefoglalni.

Időszükséglet:

A tananyag elsajátításához körülbelül 120 percre lesz szüksége.

Kulcsfogalmak

- konténer terminál
- szimuláció
- UML folyamatleíró nyelv

1. Bevezetés az anyagáramlási terminálok kérdésköreibe

A példa egy releváns publikáció [8] átfogó eredményeit mutatja be. A mai felgyorsult fogyasztói társadalom igényeinek kielégítésére hatalmas mennyiségű árut kell a gyártóktól a vevőkig áramoltatni. A gyártás és felhasználás térbeli elválasztása nagy terhelést helyez a logisztikára. A nagy távolságú összetett áruszállítás láncok kiszolgálására roppant méretű intermodális terminálok jöttek létre. Összetett áruszállítási lánc a különböző közlekedési ágak áruszállításban való együttműködése révén valósul meg. Mivel az áruáramlás jelentős része konténer egységekben történik, ezért ezt az intermodális áruszállítási rendszert vizsgáljuk. A különböző csomópontok közül a legnagyobb volumenű, legdinamikusabb és összetettebb a kikötői konténer mozgatás szervezése. Ennek a bonyolult rendszernek a modellezésére mutatunk be egy esettanulmányt.

A példa egy kikötői konténer terminál menedzselésének hatékonyságára tesz javaslatot, szimulációs alapú döntéstámogató modellel. Először a terminálban lejátszódó folyamatokat vizsgáljuk meg, amik modellünk megértéséhez szolgálnak és a szükséges bemenő adatokat is ezután tudjuk megállapítani. Azután a folyamatleíró nyelvek című leckében bemutatott objektum orientált UML folyamatleíró nyelv segítségével ábrázoljuk a terminál folyamatait. Végül pedig magát a szimulációt és a konklúziókat mutatjuk be.

2. Kikötői konténer terminál folyamatai

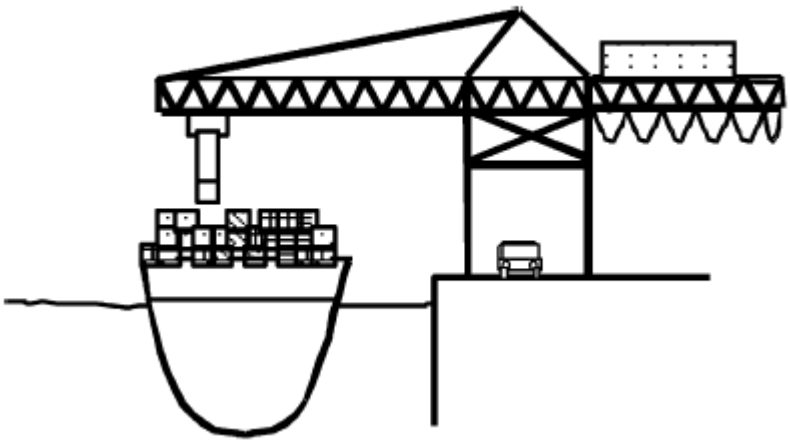
A konténer terminál egy olyan csomópont, ahol a konténerek be és kilépése a rendszerből különböző közlekedési áruszállítási ágazatokon történik, azaz interfészként funkcionál közúti, vasúti és vízi áruszállítás között, amivel fontos szerepet tölt be az intermodális áruszállítási láncban.

Átrakodás folyamatai

Az áru beérkezése valamely közlekedési eszközön, konténer egységekben történik. A vizsgált esetben, a terminálban hajóról be- és kirakodás történik vagy teherautóra, vagy vonatra. A felépíteni kívánt modellhez szükséges megvizsgálnunk a rakodáshoz kapcsolódó folyamatokat és azok időszükségletét.

A rakodás gyakran használt eszközei a kikötői bakdaruk. A kirakási folyamat az alábbi műveltekre bontható, amihez meghatározhatjuk a standard időket.

- Daru mozgás ~3 perc/konténer (macska, emelő)
- Daru javítás, kiesőidő
- Konténer ki-be emelés
- Konténer rögzítés szpréderhez. ~5 perc/konténer
- Konténer rendezés



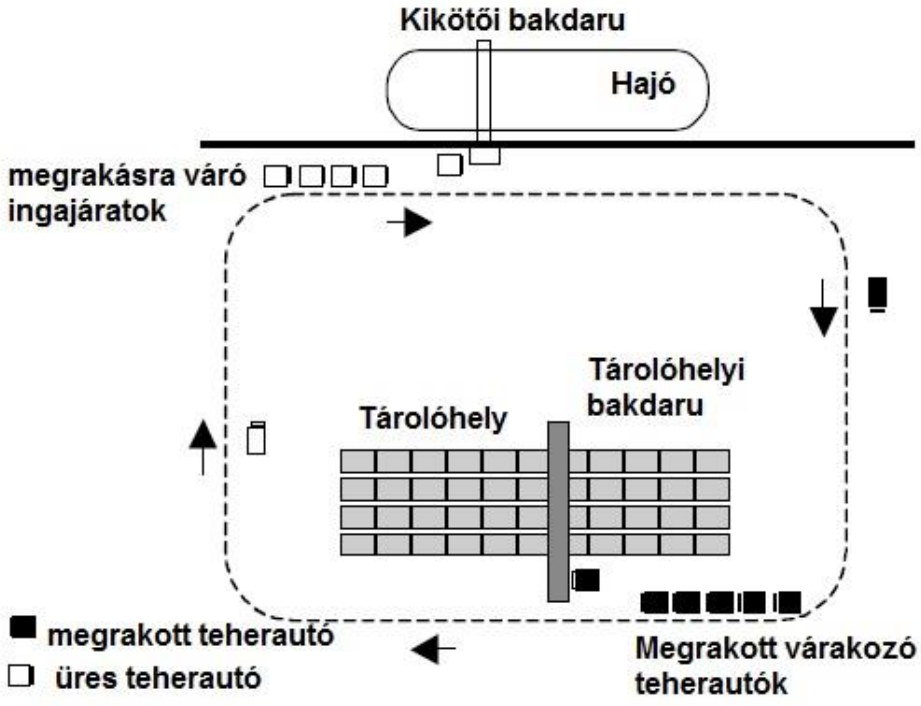
48. ábra konténer rakodás bakdaruval hajóról a daru alatt várakozó tehergépkocsira.

Tárolás, konténer rendezés folyamatai

A kikötői terminálban az anyag be- és kirakodását kívül az időbeli különbségek áthidalására tárolótér is kialakításra került, ahol a konténerek tárolják a terminálba történő be- és kiáramlás között. A konténereket három szintig halmazolják az erre kialakított eszközzel, ami lehet

- Bakdaru (gumikerekes, vagy sínen futó)
- Homlok, vagy teleszkópgémes targonca

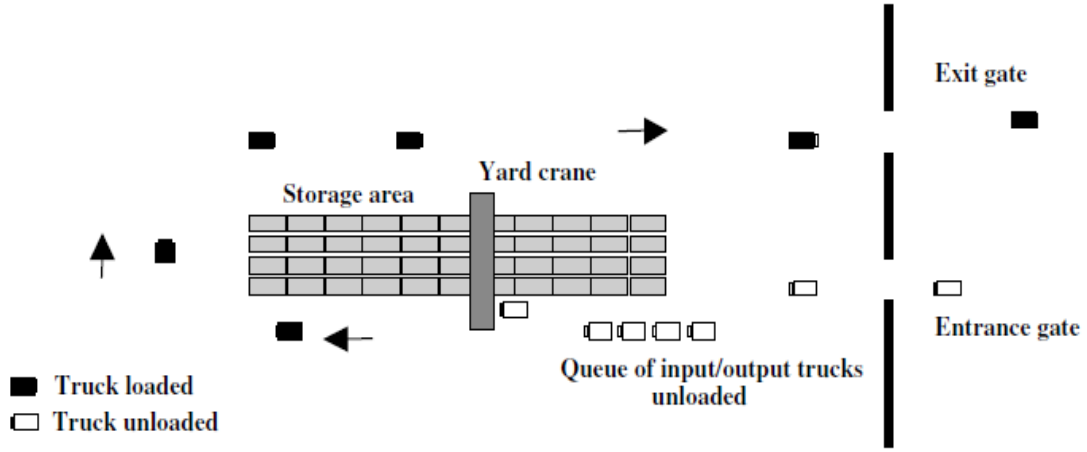
A tárolótér folyamatai a 49. ábrán láthatóak.



49. ábra A tárolótér folyamatainak ábrája a konténer mozgatást végző bakdaruval

Ingajáratok folyamatai

A kirakodott konténereket központi tároló területre viszik, a mozgatást ingajáratokkal oldják meg, lásd 49. ábra. A teherautók folyamatosan járnak a kikötői daru és a tárolótér között egy zárt hurkot alkotva. A rendszerben lévő járművek mennyiségének optimális meghatározása elengedhetetlen a terminál termelékenységének szempontjából. Ha túl sok ingajármű közlekedik a tárolótér és a hajórakodó bakdaru között, akkor a kikötői oldalon üres járművek gyűlnek fel várakozó sort alkotva. Ellenben ha túl kevés a rendszert kiszolgáló jármű, akkor a konténermozgató daruk oldalán jelentkezik szabadkapacitás, és a be- és kirakodás ütemterve csúszik, a hajónak a kikötőben kell várakoznia jelentős költséget eredményezve.



50. ábra A konténer mozgató daruk kiszolgálást végző ingajáratok működési ábrája

A be- és kirakodás és a tárolótér közötti anyagáramlási kapcsolatot már egy szállítójármű is meg tudná oldani, de a fent említett esetleges várakozás kialakulása miatt daru, és ingajárat oldalon is szükségessé teszi egy több elemből álló dinamikus rendszer kiépítését. A modell dinamikáját a rendszerben lévő változó mennyiségű teherautó adja. Egy kikötői daru, és egy tárolótéri daru között az anyagáramlási hurkot megvalósító teherautókat csoportba vesszük. A csoportban lévő teherautók számát úgy választjuk meg, és módosítjuk, hogy a két kiszolgáló állomás (daru) mindig e legyen látva feladattal, azaz ne keletkezzen szabad kapacitás daru oldalon.

Az ingajáratok folyamatában előforduló kiesőidők elhanyagolhatóak, mivel a legtöbb ilyen probléma a kiszolgáló daruknál fordul elő. Mégis fontos megemlíteni néhány tényezőt, mint a műszaki hibából, és a rossz helyre szállításból adódó kiesőidőket. Továbbá előfordulhat túlzott teher autóforgalom esetén torlódás a kikötő területén. Ez akkor fordulhat elő, ha több daru egyszerre szolgál ki egy hajót.

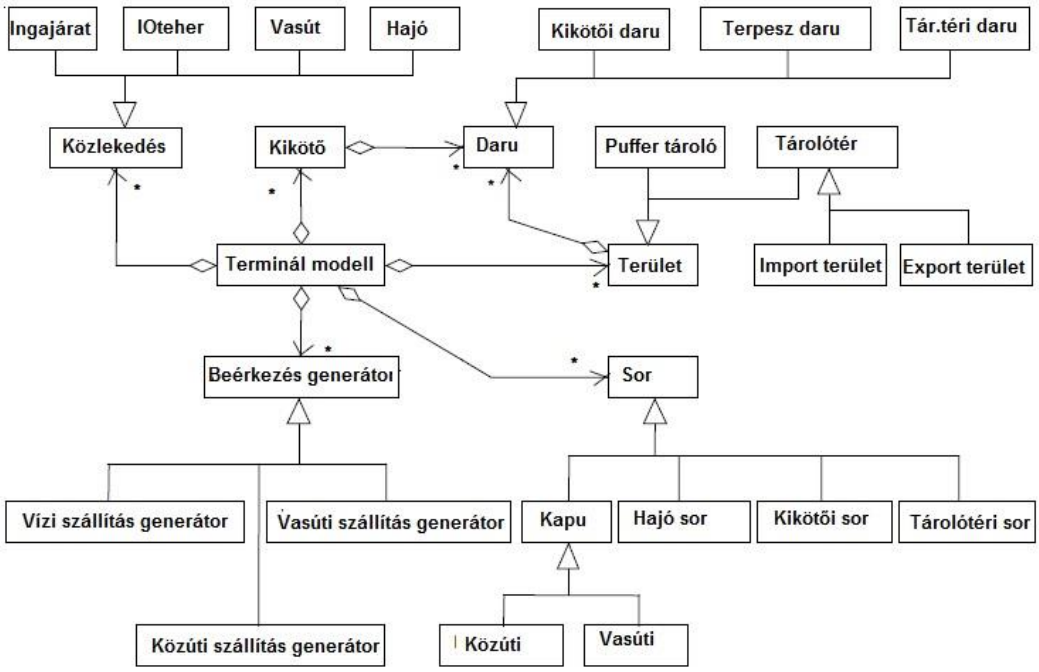
3. Osztott eseményvezérelt szimuláció - A konténer terminál logikai modellje

A modell készítésének első lépésében a folyamatokat csoportokba soroljuk. Az egyes csoportok közötti kölcsönhatások leírását az formális leíró nyelv szemlélteti (UML).

Terminál modell	A konténer terminál jelenti a rendszerünkben	
Közl.	Főcsoportja az összes szállítójárműnek	Főcsoport
Ingajárat	Területek Point-to-point kiszolgálása a Teherautókkal	
IOteher	Konténerek Input/output mozgatása a terminál területéről teherautókkal	
Vasút	Vasúti szállítás	
Hajó	Vizi szállítás	
Daru	Főcsoportja az összes anyagmozgató darunak	Főcsoport
Kikötői daru	Hajó rakodását végző daru	
Terpesz daru	Konténer szállítást is végző konténermozgató daru	
Tárolótéri daru	Tárolótéren konténermozgatását végző daru	
Terület	Főcsoportja az összes terminálban megtalálható területnek	Főcsoport
Tárolótér	Konténerek tárolására kijelölt hely (import, export)	
Puffer tároló	Átmeneti tároló konténereknek	
Sor	Főcsoportja az összes várakozó sornak a terminálban	Főcsoport
Kapu	Együttes pontok ahol közlekedési eszközök elhagyják a terminált	
Közúti kapu	Teherautók be- és kihajtási pontja	
Vasúti kapu	Vasúti szállítás be- és kihajtási pontja	
Kikötői sor	Kikötői rakodási során várakozó ingajáratok	
Tárolótéri sor	Tárolótéri anyagmozgatás során ingajáratokban jelentkező várakozó sor	
Hajó sor	Hajó kikötésére váró ingajáratok	
Érkezés generátor	Beérkezéseket generáló csoport (vízi, közút, vasút)	Főcsoport

3. táblázat A modellben szereplő csoport részletes leírása

Az 51. ábrán a modell felépítése látható. A hierarchikus modell tetején a terminál modell áll, ami az összes a terminálban végbemenő folyamatot tartalmazza. Alatta található a hozzákapcsolt hat főcsoport, ami azonos technológiai folyamatokat (csoportokat) foglalnak magukban (közlekedés, kikötő, daru, terület, sor, beérkezés generátor). Az egyes csoportokról részletes leírás található a 3. táblázatban.

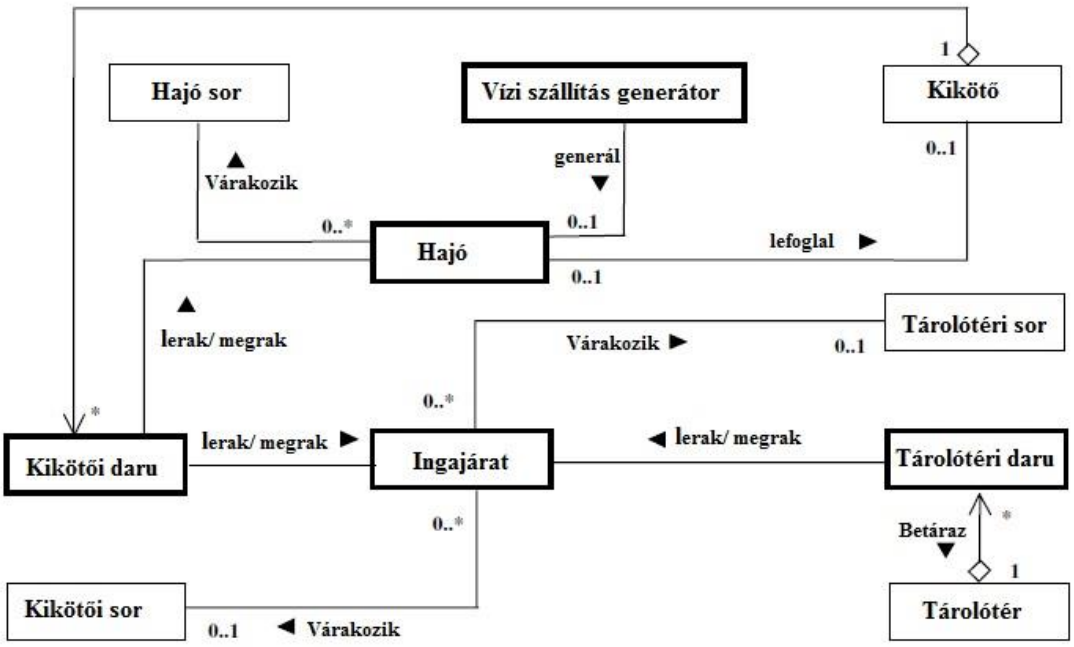


51. ábra A konténert terminál infrastruktúrájának kategória diagramja

Párhuzamos folyamatok

A valóságban az egyes elemek (daru, ingajáratok teherautói, stb.) függetlenül és párhuzamos végzik a feladatukat, ezért a szimulációnál szükséges az egymás mellett futó párhuzamos cselekvés, amit az UML leíró nyelv támogat.

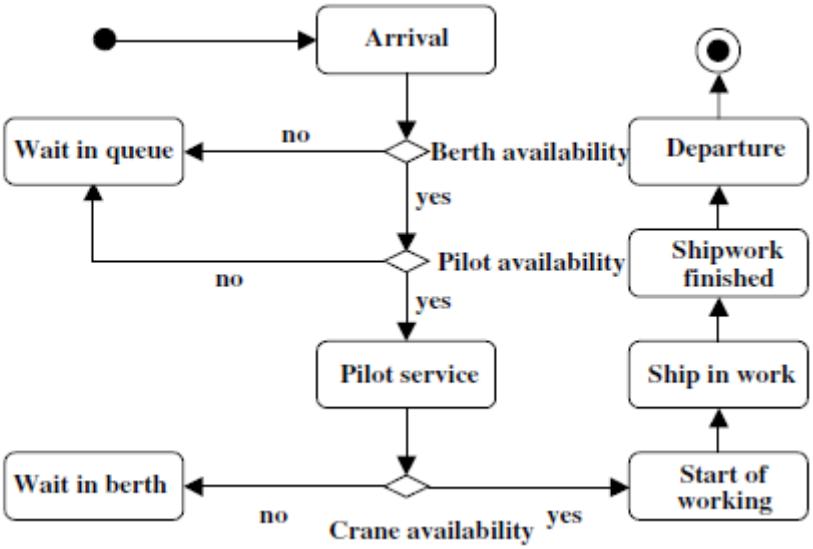
A terminálban zajló folyamatok közül a konténermozgatás a legjelentősebb. Ennek a folyamatát látjuk leírva az 52. ábrán. A folyamat a "generál" jelzéssel a hajó beérkezés objektumból indul, ahol igény jelentkezik kirakásra váró hajó formájában. Ez a trigger indítja a további folyamatokat, ahol az egyes objektumokra állapotváltozásai jelölve vannak (lefoglal, várakozik, lerak). DES modellezésnél figyelembe kell vennünk az objektumok állapotát és az eseményeket, amik az állapotváltozáshoz vezetnek. Ezek jelölésére állapot és aktivitás diagramok használunk, amik a daruk, hajó, ingajáratok állapotát írják le.



52. ábra a konténermozgatás osztálydiagramja

Együttműködési diagram

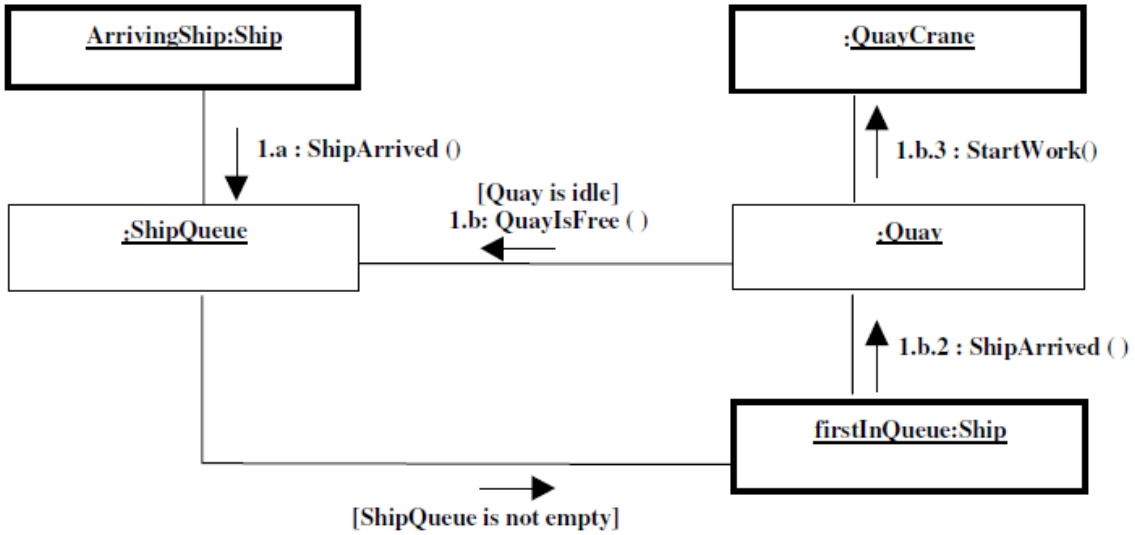
Az UML együttműködési diagramokkal a rendszerünk dinamikus jellemzőit írhatjuk le. A teljes rendszer leírásának nélkülözésével csak a hajó beérkezésének együttműködési diagramját mutatjuk be.



53. ábra hajó beérkezésének állapotdiagramja

Az ábrán a hajó beérkezése (Arrival) után ellenőrzi, hogy van-e szabad kikötési hely, illetve a hajó irányítását végző személy. Ezután a rakodás csak akkor kezdődhet meg, ha van

szabad daru (Crane availability - yes). Ezután a rakodás folyamata lineáris, míg a folyamat befejéseként a hajtó újra kifut a tengerre (Departure).



54. ábra együttműködési diagram a hajó megérkezéséről

Az esemény egy elem értesítés művelet előfordulásakor. Például amikor a hajó (ship) esemény küld a hajó megérkezéséről (Shiparrived) a "hajó várakozik"(ShipQueue) objektumnak a hajó megérkezésekkor.

Konténer terminál szimulációja

A folyamat szimuláció használatából származó előnyöket már bemutattuk a Szimulációs módszerek című leckében, ezért itt nem részletezzük. A terminál szimulációs modell esetében definiálni kell a rendszer működési költségének meghatározásához szükséges költségfüggvényeket, valamint a teljesítmény tényezőket. A modell kiértékeléséhez cél mérőszámokat kell meghatároznunk ilyen, lehet

- Globális termelékenység (daruval megmozgatott össz. konténerszám/ össz. hajó kiszolgálási idő)
- Nettó termelékenység (daruval megmozgatott össz. Konténerszám/ daru hajókiszolgálással töltött össz. Ideje)
- Kikötői daru kihasználtsága (darumozgás+ műveleti idő/ darumozgás+műveleti idő+várakozási idő)
- Ingajárat kihasználtsága (szállítás ideje/szállítás ideje+várakozó idő)
- Konténertároló kihasználtsága (konténer által elfoglalt hely/ teljes tároló kapacitás)

Felhasznált irodalom

[1] Dr. Kulcsár Béla: Ipari logisztika. LSI Oktatóközpont, 1998.

[2] Paláncz Béla - Soft Computing - 2011 - 10. Fuzzy modellezés. http://www.fmt.bme.hu/fmt/oktatas/feltoltesek/BMEEOFTMFT1/fejezet_10.pdf

[3] UML basics: An introduction to the Unified Modeling Language. http://www.ibm.com/

[4] Business Process Modeling, http://www.conceptdraw.com/

[5] http://www.conceptdraw.com/samples/resource/images/solutions//BUSINESS-PROCESS-DIAGRAMS-Flow-Charts-Trading-Process-Diagram-Sample.png

[6] Jurijs Tolujevs, Tatjana Zmanovska, Numerical modelling of processes in logistics flow systems. Proceedings of the 11th International Conference "Reliability and Statistics in Transportation and Communication" (RelStat'11), 19–22 October 2011, Riga, Latvia, p. 241-248. ISBN 978-9984-818-46-7

[7] Dr. Bohács Gábor: Termelési logisztika, elektronikus órai jegyzet

[8] Maurizio Bielli, Azedine Boulmakoul , Mohamed Rida: Object oriented model for container terminal distributed simulation. European Journal of Operational Research 175 (2006) 1731–1751