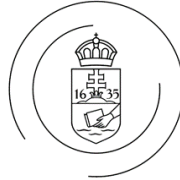


OTDK-dolgozat

2023

EÖTVÖS LÓRÁND TUDOMÁNYEGYETEM
GAZDASÁGTUDOMÁNYI KAR



SZIMULÁCIÓ ALAPÚ TÖBBTÉNYEZŐS DÖNTÉSI MODELL
GYAKORLATI ALKALMAZÁSA PROJEKTKÖRNYEZETBEN

APPLICATION OF SIMULATION BASED MULTI-CRITERIA
DECISION MODEL IN PROJECT ENVIRONMENT

Szenes Attila

Konzulens: Dr. Jónás Tamás

Kézirat lezárásának dátuma: 2023. január 11.

Tartalomjegyzék

ÁBRAJEGYZÉK.....	1
1. BEVEZETÉS.....	1
2. ELMÉLETI HÁTTÉR ÉS ELVI FELÉPÍTÉS	4
2.1. ÜGYNÖK ALAPÚ SZIMULÁCIÓS MÓDSZEREK (AGENT BASED SIMULATION, ABS).....	4
2.1.1. Az ABS általános bemutatása, felhasználása	4
2.1.2. Az ABS alkalmazása a dolgozatban.....	5
2.2. TÖBBTÉNYEZŐS DÖNTÉSI MODELLEK (MULTI-CRITERIA DECISION MODEL, MCDM)	11
2.2.1. Az MCDM általános bemutatása, felhasználása	11
2.2.2. Az MCDM alkalmazása a dolgozatban	16
3. DEMONSTRÁCIÓ.....	18
3.1. SZIMULÁCIÓ ÉS PARAMÉTEREI.....	18
3.2. KIMENETI NAPLÓK.....	18
3.3. FELHASZNÁLÓI INTERAKCIÓ.....	19
3.4. ELVÁRÁSOK.....	20
3.5. AGGREGÁCIÓ.....	20
3.6. GENERÁTORFÜGGVÉNY HATÁSA	21
3.7. GENERÁLT ÜTEMTERV	24
4. KONKLÚZIÓK ÉS TOVÁBBI KUTATÁSI LEHETŐSÉGEK	25
5. IRODALOMJEGYZÉK	27
6. MELLÉKLETEK.....	30

Ábrajegyzék

1. saját ábra, a NetLogo szoftver szimulációs felülete, balra a paraméterek és vezérlőgombok, középen az ügynökök tere, jobb oldalon pedig a szimuláció nyomonkövetésére készített diagramok láthatóak.....	5
2. saját ábra: A tevékenységek és erőforrások ábrázolása a Netlogo szoftverben.....	7
3. saját ábra: A projekt átfutási idejének és költségének ábrázolása hisztogramon, egy 350 iterációs szimuláció során	8
4. ábra: MCDM módszerek struktúrája, Forrás: <i>A Survey on Multi Criteria Decision Making Methods and Its Applications</i> (Aruldoss et al., 2013)	12
5. ábra, Saját szerkesztés: generátorfüggvények összehasonlítása	16
6. saját ábra, az elvárások megadására szolgáló felület.....	19
7. saját ábra, projektátfutás-költség diagram.....	19
8. saját ábra, $f(x)=-\ln(x)$ generátorfüggvény és az elvárásokkal szembenő eredmény.....	21
9. saját ábra, $f(x)=-\ln(1-x)$ generátorfüggvény alkalmazásával számolt döntés.....	22
10. saját ábra, $f(x)=x$ generátorfüggvény és a várakozásoknak megfelelő viselkedés.....	22
11. saját ábra, $f(x)=x^2$ generátorfüggvény és a várakozásoknak megfelelő viselkedés.....	23
12. saját ábra, a generátorfüggvények és az alkalmazásukkal számolt rangsorolás eredménye.....	23
13. saját ábra: A fenti elvárások alapján generált sávós ütemterv	24

1. Bevezetés

A bevezetőben áttekintjük azt az üzleti és gazdasági környezetet, amiben a problémát azonosítottuk és bemutatjuk az általunk javasolt szimulációs és döntési modell összekapcsolt alkalmazásával elérhető megoldást.

Napjainkban egyre gyakrabban hivatkozunk „projektként” a munkavégzésre. Ennek hátterében egyrészt a fogalom terjedése, bizonyos fokú túlhasználása is áll, másrészt az, hogy a szervezetek szükségszerűen és szívesen alkalmazzák a formátumot céljaik elérése érdekében (Bedzsula, 2021).

Projektként azonosítjuk az olyan tevékenységeket, amelyek valamilyen egyedi végeredmény (termék vagy szolgáltatás, szervezeten belüli átalakulás, stb) létrehozását célozzák meg (PMI, 2016). Erre jellemzően komplex környezetben történik az erőfeszítés, és bár az eredmény mérhető, az oda vezető út nem triviális, gyakori feladat az optimális megoldások keresése, a tervek optimalizálása.

A projektek sikerességét érthető módon kemény kritériumok között értelmezzük. Általában akkor nevezünk egy projektet sikeresnek, ha azt határidőre, tervezett költségkereten belül és a megállapodott képesség- és minőségcélok elérésével szállították (PMI, 2016). Várakozásunknak megfelelően sikertelennek tekinthető az a projekt, amelyiknél legalább egy tényező sérül.

Tanulmányok rámutatnak, hogy az esetek nagyobb hányadában ez be is következik. A KPMG és az Australian Institute of Project Management (2020) szerint az eredeti célok az esetek 51%-ában teljesülnek, a költségkereteket 42%-ban, míg a határidőt 40%-ban sikerül tartani. A PMI hasonló tanulmánya (2018) még ennél is lesújtóbb sikerrátával (rendre 31, 43, 48%) támasztja ezt alá. Továbbá azt állítja, hogy 20 másodpercenként 1 millió dollár kárt okoz világszerte a projekt menedzsment módszerek nem megfelelő alkalmazása. Kontextusként 2021-ben a teljes globális projekt portfólió méretét 4 milliárd dollár méretűre becsülik, ami várhatóan 7 milliárdra nő 2029-re (Fortune Business Insight, 2021).

A sikertelen teljesítések okait vizsgáló ranglistákat a szervezeti és kommunikációs problémák vezetik. Továbbá a PMI felmérése alapján a megkérdezettek szerint a kudarc legalább 20-20%-ban köthető a kapacitástervezéshez és -előrejelzéshez, változásmenedzsmenthez, költségbecsléshez köthető technikákhoz és a projektmenedzser felkészültségéhez (PMI, 2017).

Bátorkodunk levonni azt a következtetést, hogy magából a formátumból, a projektszerű felállás formai kötöttségéből következik, hogy ennyire kézenfekvő azonosítani a veszteségeket, a problémákat és azok okait.

Számos kutatás koncentrált a projektkudarcokra, az azokhoz vezető okokra. A listák élet legtöbbször csoportdinamikai, kommunikációs, szervezeti jellegű problémák állnak. Ezek mind olyan tényezők, amik túlmutatnak a dolgozat, illetve általában a kemény területekkel foglalkozó kutatások tárgyán. A Wellington által készített 2020-as riport (Wellington, 2022) azonban bemutatja, hogy az alul képzett projektmenedzserek a második, az alacsony minőségű kapacitásmenedzsment a harmadik, míg a tervezési- és a kockázatkezelési képességek hiánya a nyolcadik-kilencedik helyen állnak a legnagyobb kihívást jelentő akadályok között.

A Project.co (2021) felmérése alapján a megkérdezettek 10%-a szerint „ritkán” vagy „soha” nem teljesítik a projektet határidőre és alig 16%-uk mondja, hogy „mindig” sikerül a határidőt tartani.

A kudarcok mögött gyakran olyan tényezők állnak, amelyek negatív hatása megfelelő technológia támogatás segítségével jelentősen csökkenthető. Ahogy arra a mesterséges intelligencia és a projektmenedzsment kapcsolatát kutató tanulmányok rámutatnak, a kemény területeken bevont segítség olyan kapacitást szabadít fel, amit a projektmenedzserek a puha területekre fordíthatnak, így közvetlenül és közvetve is növelik a siker valószínűségét. (IPMA in association with PWC, 2020)

Számos tanulmány, „Artificial Intelligence impact in Project Management” (IPMA in association with PWC, 2020), valamint „AI Innovators: Cracking the Code on Project Performance” (PMI, 2019), továbbá „5 Implications of Artificial Intelligence for Project Management” (PMI, 2021) részletezi a közeljövőben várhatóan mesterséges intelligencia (MI) módszerek terjedését a projektmenedzsment területen (Taylor, 2022). Ezek a megoldások egyelőre nem terjedtek el széles körben, bár magas várakozás alakult ki feléjük, a menedzserek és a vállalatok több, mint fele úgy érzi, hogy készen áll valamilyen MI módszer bevezetésére (Holzmann et al., 2022), illetve (PWC, 2019).

A szakirodalom hat nagy MI technológiai területet említ (PMI, 2019):

- Tudás alapú rendszerek
- Gépi tanulás
- Döntésmenedzsment
- Szakértői rendszerek

- Mély tanulás
- Folyamat automatizálás

Fridgeirsson és társai (2021) által végzett felmérés szerint a MI hatása a következő 10 évben a PMI 49 projektmenedzsment folyamatát gyűjtő tíz tudásterület közül az ütemezésben, költségek és kockázatok kezelésében várható legerősebben, vagy ha a folyamatszoportokat tekintjük, a követési és felügyeleti folyamatok között.

Korai felismerés, hogy a menedzsment területekre is hatással lesz (Simon & Newell, 1971). Napjaink kutatói, pl. Canals és Heukamp (10 Ways Artificial Intelligence Is Transforming Management, 2018), illetve Taylor (AI and the Project Manager, 2022) a munkaerőpiac (újbolí) átrendeződését is előrejelezték.

Az alkalmazási területeket vizsgálva a folyamatautomatizációk állnak az élen, ezt követik a chatbotok és kiterjesztett analitikák (IPMA in association with PWC, 2020). Egy másik, hasonló kutatás praktikusabb tevékenységeket emel ki: betekintés az összefüggésekbe, ütemtervezés készítése, tendenciák figyelése, kockázatok követése, adatok elemzése a portfólió kontextusában (PWC, 2019).

Elvart hozományként több, határidőre teljesülő projektet (61%) (PMI, 2019), produktivitásnövekedést (53%) (IPMA in association with PWC, 2020) várnak a megkérdezettek, de megjelenik a költségkezelésben, kapacitástervezésben elért fejlődés is.

Mindezekből nem meglepő, hogy az AI-ra elsősorban PM asszisztensként (52%) vagy PM tanácsadóként (44%) tekintünk és nem PM-et helyettesítő „vetélytársként”.

A dolgozat bemutat egy alacsony számításigényű szimulációs módszert, amivel projekt terv alternatívákat készíthetünk, javaslatokat téve a kapacitásra, a tevékenységek kezdési és befejezési idejére, valamint a terjedelmük optimalizálására. A technológiai adottságokat, a tevékenységek időigénye, szakmai tartalma és az azok közti függések adottak. Ügynök alapú szimuláció alkalmazásával másodpercek alatt több száz ütemterv alternatívát állítunk elő. A szimuláció körülményei, függő és független tulajdonságai az alternatívák attribútumait képzik.

Azt is látni fogjuk, hogy a túl kevés opcióhoz hasonlóan a túl sok opció is problémát jelent. Bár az eredmények halmaza várhatóan tartalmazza a számunkra ideális megoldást, azok között pusztán „átnézéssel” választani nem reális valós projektkörnyezetben. Kritikus elem a megfelelő döntéstámogató algoritmus bevezetése és beállítása. A tanulmány a Dombi és Jónás súlyozott aggregáló rendszerét alkalmazza. A rendszer döntéshozói elvárasi szintektől függő

eljárással kiértékeli az alternatívákat, ehhez pedig generátorfüggvénnyel operáló kvázi-aritmetikai közepet alkalmaz.

A módszer előnye, hogy a döntéshozó értékelési és súlyozási szempontjait egyetlen közös interakcióban egyesíti, amittől a döntés folyamata rugalmassá, flow-szerűvé válik

2. Elméleti háttér és elvi felépítés

2.1. Ügynök alapú szimulációs módszerek (Agent based simulation, ABS)

2.1.1. Az ABS általános bemutatása, felhasználása

A számítógépes szimulációs modellek terjedése a számítási kapacitás folyamatos növekedésével és az eszköz terjedésével egyre több (polgári és üzleti) területen válik meghatározóvá. Segítségükkel lehetőségünk adódik arra, hogy megértsük a folyamatok széles spektrumát csillagásztól, biológián át, a társadalmi, gazdasági területekig. A modellezés során valamilyen egyszerűsítés mellett rekonstruálunk egy folyamatot. A kimenetek egy részéről rendelkezünk empirikus adatokkal, ezt validálásra, a paraméterek finomhangolására alkalmazzuk. A másik része pedig lehetőséget teremt a témával kapcsolatos következtetések levonására. A teljes folyamat megfigyelése pedig elemzésre nyújt lehetőséget. (Davidsson et al., 2007)

Az ügynök alapú modellezés és szimuláció (agent based modelling/simulation, ABMS/ABS) a szimulációs technikák egy speciális irányzata. Felhasználása kiterjed számos szakterületre, epidemiológiai, illetve szociológia alkalmazása a leggyakoribb (Bankes, 2002).

Macal (2016, p. 145) tanulmányában rámutat, hogy az ABS „legfontosabb és legmegkülönböztetőbb jellemzője [más szimulációs irányzatokhoz képest] az ügynöki perspektíva”, folyamatok és tevékenységek helyett a modellalkotás során fundamentálisan az ügynökökre és azok viselkedésére fókuszálunk. Ez a megközelítés teszi alkalmassá az ABS-t a társadalmi, szervezeti és más csoportos viselkedések vizsgálatára.

Borgonovo és társai (2022) valamint Railsback és társai (2006) által végzett kutatások hangsúlyozzák az ABS a menedzsment területen növekvő jelentőségét. Az eredmények egyrészt a komplex folyamatok reprodukálására és megértésére irányulnak, másrészt elméletek fejlesztésében segítenek.

2.1.2. Az ABS alkalmazása a dolgozatban

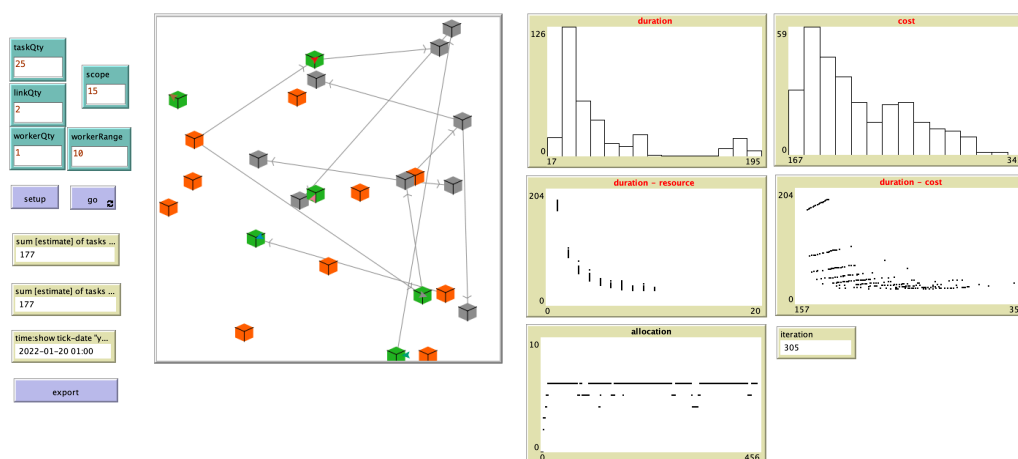
A munkánk során a NetLogo¹ szoftvert használtuk a folyamat modellezésére.

A modell építése során kettős célt tartottunk szem előtt. Egyrészt rövid idő alatt szerettünk volna nagy számú alternatívákat készíteni ugyanazon scenáriókra, másrészt a későbbi döntési modell számára részletes paramétereket kellett biztosítani.

A szimuláció során a projektmunkák célirányosan szűkített aspektusait vizsgáljuk. Bár tudomásul vesszük a puha tényezők szerepét, nem foglalkozunk velük, úgy tekintünk rájuk, hogy a tapasztalati, szakértői állásfoglalások úgy lettek megállapítva, hogy magukba foglalják azokat. Másképp megközelítve, az adatainkat leginkább egy projektmenedzsment szoftverben tárolt tényadatokhoz hasonlóan építettük fel.

A szimulációs térünk egy 32x32 egység méretű világ, amiben (lebegőpontos értékek mellett) keletkeznek és mozoghatnak az ügynökök. A vizuális megjelenítés segíti a megfigyelést, a térbeli bizonytalanság (tevékenységek és erőforrások távolsága, két tevékenység közötti mozgás) a tevékenységekhez való hozzáférhetőséget, információfeldolgozási tevékenységeket reprezentálja.

A szimuláció során két ügynöktípust határoztunk meg. Az ABS megközelítésében ügynökök a tevékenységek és az erőforrások is.



1. saját ábra, a NetLogo szoftver szimulációs felülete, balra a paraméterek és vezérlőgombok, középen az ügynökök tere, jobb oldalon pedig a szimuláció nyomonkövetésére készített diagramok láthatóak

¹ <https://ccl.northwestern.edu/netlogo/>

2.1.2.1. A projekt tevékenységei

A modell felépítése során tevékenység-csomópontú megközelítést alkalmazunk. A tevékenységek tehát olyan ügynökök, amelyek ugyanúgy szabályok szerint reagálnak a környezeti szabályokra és más ügynökök, az erőforrások és tevékenységek interakcióira.

A tevékenységek szimulációhoz szükséges tulajdonságai:

- Név: erre a későbbi azonosításhoz, hivatkozáshoz lesz szükségünk, automatikusan generálódik
- Felmérés (estimation): a tevékenység elvégzéséhez megállapított, becsült idő
- Függőség (dependency): bemeneti kapcsolat olyan tevékenység felől, aminek teljesülése szükséges a tevékenység elkezdéséhez
- Státusz: dinamikus változó
 - Szürke, ha még nem kezdhető a tevékenység, pl. függőség miatt
 - Kék, ha kezdhető a tevékenység
 - Zöld, ha folyamatban van a tevékenység elvégzése egy erőforrás által
 - Narancs, ha a tevékenység elkészült
- Kijelölt erőforrás: dinamikus változó, induláskor üres
- Aktuális készültség: dinamikus változó, induláskor 0%
- Kezdési idő: induláskor üres, a szimuláció során kap értéket
- Befejezési idő: induláskor üres, a szimuláció során kap értéket
- Tényleges munka ideje: induláskor üres, a szimuláció során kap értéket

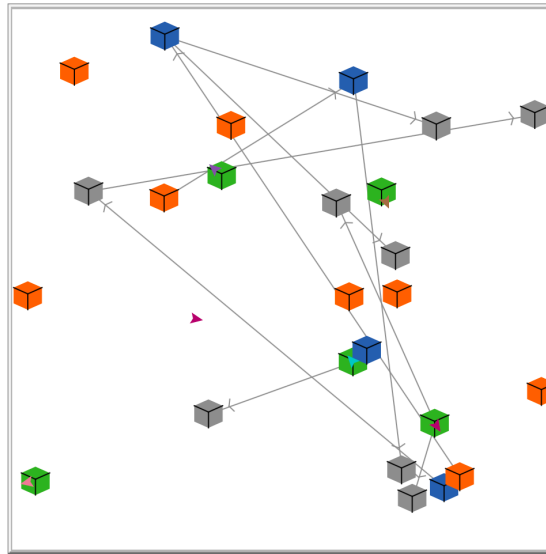
2.1.2.2. A projekt erőforrásai

Az erőforrás ügynökök tulajdonságai:

- Név, ID: a beazonosításhoz és a tevékenységek kiválasztásához
- Céltevékenység: az erőforrás elmenti magának a következő illetve aktuális tevékenység azonosítóját

A lenti ábrán a NetLogo szoftverben megjelenő ügynökök láthatóak a szimuláció során. A színes nyilak az erőforrásokat reprezentálják, a dobozok a tevékenységet az aktuális

státuszoknak megfelelő színek szerint. A tevékenységek közötti irányított szakaszok a dependenciákat jelképezik.



2. saját ábra: A tevékenységek és erőforrások ábrázolása a Netlogo szoftverben

2.1.2.3. A szimulációs téren belüli döntések

Az ABS rendszerek sajátossága, hogy az ügynökök saját „aggyal”, önálló döntési algoritmussal rendelkeznek. Illetve az algoritmusokat típusonként (tehát tevékenység és erőforrás) előre meghatározzuk, de a szimuláció során minden ügynök a saját egyéni és az érzékelt környezeti változók alapján hozza a döntését.

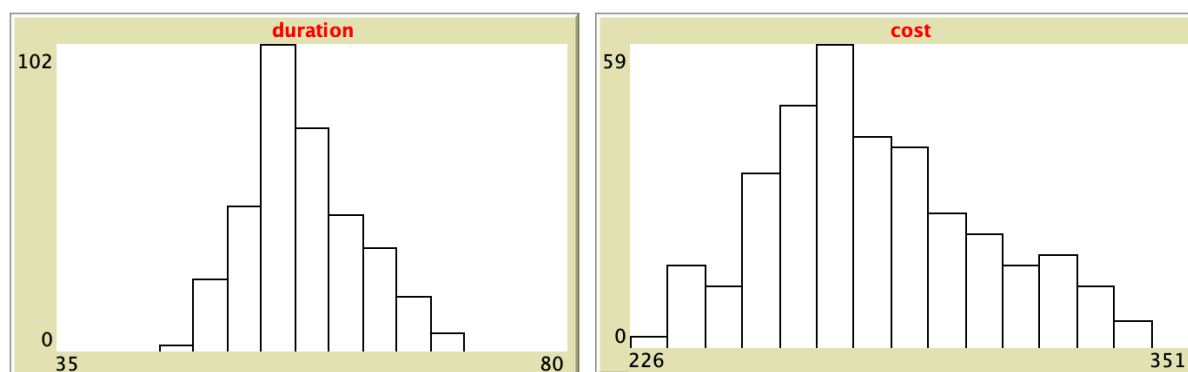
Esetünkben ilyen döntési algoritmus a tevékenységválasztás kérdése. Néhány megkötés mellett minden erőforrás-ügynök véletlenszerűen választ magának tevékenységet. A tevékenység státusza kék kell legyen, ezzel irányítjuk, hogy csak valóban elkezdhető tevékenységen dolgozzanak. Továbbá azok szabadok kell, hogy legyenek, más erőforrás még nem foglalta le magának, ezzel elkerülhető, hogy az ügynökök fölöslegesen vándoroljanak a tevékenységek között.

Érdemes itt megemlíteni, hogy bár a szimulációk során megengedjük az erőforrás-ügynököknek, hogy szabadon, véletlenszerűen válasszanak maguknak tevékenységet, azt csak a szimuláció heurisztikus jellege miatt tehetjük meg. A végrehajtás során szigorú tervekötés mellett tarthatóak a kiválasztott paraméterek.

Mind az erőforrások, mind a tevékenységek rendelkeznek interakciós tulajdonságokkal, ezek a döntési algoritmusokhoz hasonlóan előre meghatározott eljárások:

- Tevékenység kiválasztása: az erőforrás az elérhető (kék) tevékenységek közül véletlenszerűen kiválaszt egyet, majd megközelíti azt, az összekapcsolás a tevékenység és az erőforrás ügynökön is rögzítésre kerül
- Feldolgozás: a tevékenység státusza zöldre vált és az erőforrás a tevékenységnél „parkol”, a készültségi értéket folyamatosan számolja a tevékenység az eltelt idő és a becsült felmérési idő alapján
- Elkészülés: amint a feldolgozás eléri a 100%-ot a tevékenység státusza narancsra vált, a kezdési és befejezési időt is eltárolja és a követő tevékenységnek jelzi, hogy ha minden más előzmény tevékenysége kész, váltszon szürkéről kékre, az erőforrást pedig, hogy keressen új tevékenységet
- a környezet pedig figyeli, hogy minden tevékenység elkészült-e, ekkor újraindítja a szimulációt, minden visszaáll alapállapotba

Már önmagában az, hogy ugyanannyi erőforrás ugyanazon programterv tevékenységeit véletlenszerű sorrendben végezte el is látványosan szóródást eredményezett az átfutási idő tekintetében. A középértékhez képest akár 20-30%-os eltérések is kirajzolódtak.



3. saját ábra: A projekt átfutási idejének és költségének ábrázolása hisztogramon, egy 350 iterációs szimuláció során

Projektmenedzsment és gazdaságossági oldalról értelmezve pedig ez azt jelenti, hogy rögzített technológiai terv mellett is érdemes a tevékenységek elvégzésének optimális, összehangolt sorrendjét keresni. Ezzel többleterőforrások bevonása nélkül tehetünk versenyelőnyre.

A továbbiakban bevezetünk néhány szimulációs paramétert.

2.1.2.4. Erőforrások száma

Eddig rögzített erőforrással készítettük a szimulációkat. Adódhat viszont olyan helyzet, amikor pont az ideális méretű erőforrás meghatározása a tevékenység. A paraméter megadásával lehetőséget adunk a szimulációk során egy tartományon belül véletlenszerű erőforrás számmal számolni.

2.1.2.5. Terjedelem optimalizálása

A projekt jellegű munkák egyik kulcstulajdonsága, hogy az eredményük egyedi, így bár a tervezés során hagyatkozunk a szakértői becslésekre, még a prediktív szituációkban is keletkezik némi bizonytalanság azok pontosságát illetően. Ezzel a paraméterrel, lehetőségünk van meghatározni egy maximum értéket a terjedeleme csökkentésére, a szimuláció során egy tevékenység legfeljebb ekkora mértékben csökkentett terjedelemmel készül.

Már kellően alacsony érték (1-5%) alkalmazása is drasztikus változásokat idéz, szemben a statikus, kritikus út számítással.

Az eredmény nagyon hasonló következtetésekre vezet, mint a korai Monte Carlo módszerrel készített PERT hálózat vagy CPM elemzések. Rögzített felmérések esetén egy nagyon egyszerű algoritmussal számolható a projekt átfutási ideje, ami így szintén rögzített. Apró ingadozások viszont drasztikus hatással lehetnek a várható ütemezésre (Van Slyke, 1963).

Későbbiekben ezekre az ingadozásokra fogunk építeni.

2.1.2.6. A szimuláció kimenetei

Összefoglalva, egy olyan rendszert építettünk, aminek rögzített bemenetét adja egy szakértők által meghatározott tevékenységlista, beleértve az azok elvégzéséhez szükséges munkaidőbecslést és függőségi kapcsolataikat. Paraméteres bemenetként szabályozhatjuk a maximális erőforrásszámot és a terjedeleme optimalizálását. A rendszeren belül pedig az ügynökök véletlenszerű sorrendben és véletlenszerű terjedeleme csökkentés mellett végrehajtják a projektet. Betartják azokat az interakciós szabályokat, amik biztosítják az életszerű lefutást, mint a tevékenységek rákövetkezése vagy a hozzáférhetőség.

Természetesen a szimuláció sebessége függ az ügynökök számától, de akár több száz tevékenységet és több tíz erőforrás ügynököt tartalmazó szimulációk is másodpercek alatt lefutnak, legfeljebb néhány perc alatt pedig sok száz iterációt kapunk.

Kiegészítő megjegyzés: A kísérletek egyszerű lefutása érdekében a rendszert kiegészítettük egy projektgenerátor funkcióval is, amivel életszerű tevékenységláncokat készíthetünk a szimulációinkhoz.

Tekintsük át a kimeneti naplókat:

- Tevékenységek: a szimuláció bemenetét képező technikai alapadatok (esetünkben kimenet, a projektgenerátor alkalmazása miatt). 1. melléklet
 - Tevékenység neve, egyedi azonosítója
 - Tevékenység becsült munkaigénye
 - Kapcsolódások, dependencia
- Szimuláció: a teljes eseménynapló, ami a szimulációk releváns eseményeit tartalmazza, így a döntés eredményeként választott alternatívából képesek leszünk rekonstruálni az ütemtervet egy Gantt diagramon. 2. melléklet
 - Iteráció azonosító száma
 - A tevékenység neve
 - Kezdet
 - A Valódi (optimalizált) hossza
 - A végrehajtó erőforrás neve
- Összesítés: számunkra a legértékesebb adat, ami a döntési modellben feldolgozandó tényezőket tartalmazza. 3. melléklet
 - Iteráció azonosító száma
 - Átfutási idő
 - Felhasznált erőforrás (erőforrások) száma
 - Költség: származtatott adat az idő és erőforrás szorzata
 - Terjedelem csökkentés mértéke
 - Várakozási/tartalékidő: azon idők összessége, amikor az erőforrások várakoztak új, kezdhető tevékenységre

2.2. Többtényezős döntési modellek (Multi-criteria Decision Model, MCDM)

2.2.1. Az MCDM általános bemutatása, felhasználása

Döntés² során az emberek a lehetséges opciók közül választanak. Azonosítják az értékelési szempontjaikat, azok szerint pedig értékelik a lehetőségeiket. Végül valamilyen intuíció segítségével kiválasztják számukra a legelőnyösebbet (Zargini, 2020).

Számíthatunk arra, hogy nem találunk olyan lehetőséget, ami minden értékelési szempont szerint a legkiválóbb a döntéshozó számára, tehát a döntés során valamilyen kompromisszum alkalmazására van szükség. Továbbá nem minden szempontot vesz figyelembe azonos súllyal (Zargini, 2020).

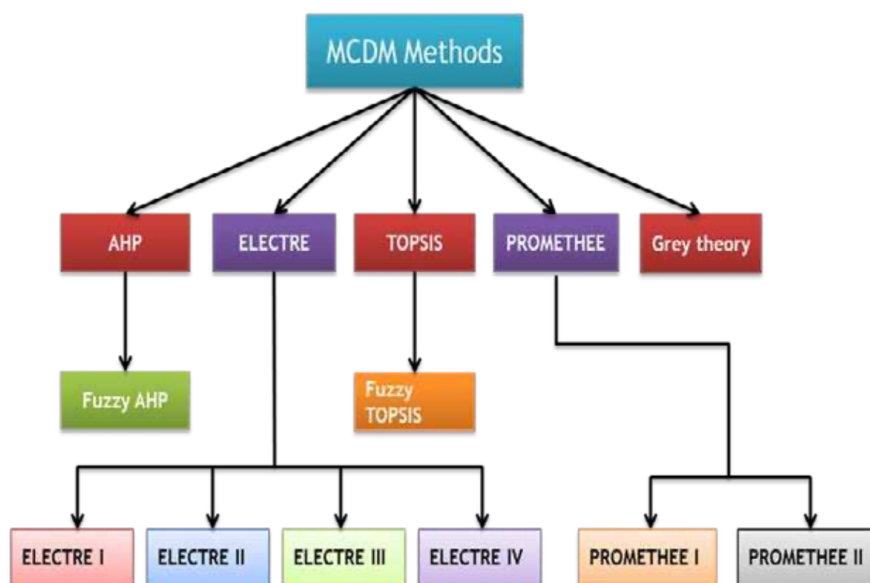
Az emberi döntést támogató rendszerek ezt a folyamatot modellezik (Frini, 2017). A fenti leírás alapján kiemelendő, hogy a döntés nem választható el a döntéshozó szubjektív értékítéletétől, ami megjelenhet a kritériumok azonosításában, azok értékelésében, a súlyok meghatározásában és végül a kompromisszum alkalmazásában.

A többtényezős döntési modellek széles skálája áll rendelkezésre, a teljesség igénye nélkül bemutatunk néhányat. (Aruldoss et al., 2013)

- AHP: szakértő ismeretek visszaadására törekszik, fuzzy elméletekből kölcsönzött intervallum becslést és hierarchikus felépítést használ
- TOPSIS: minden szempontra jellemzően monoton növekvő vagy csökkenő hasznosság-tendenciaként tekint és összehasonlítások sorozatával állapítja meg az alternatívák sorrendjét
- ELECTRE: a legjobb választást úgy adja meg, hogy keresi a legelőnyösebb és a legkevésbé konfliktusos megoldást a szempontok között
- Gyakori továbbá a metódusok fuzzy változatai is

A modellek hierarchikus ábrázolását a 4. ábra mutatja be.

² A kontextusból adódóan a dolgozatban a döntésre racionális döntésként gondolunk.



4. ábra: MCDM módszerek struktúrája, Forrás: *A Survey on Multi Criteria Decision Making Methods and Its Applications* (Aruldoss et al., 2013)

Formálisan, jelöljük az alternatívák attribútumait $a_1, a_2, \dots, a_n$ alakban, aminek változó értékei $x_1, x_2, \dots, x_n$, illetve a hozzájuk tartozó súlyok $w_1, w_2, \dots, w_n$ lesznek. A továbbiakban tekintsünk úgy az x_i változó értékre és a w_i súlyértékre, mint rendre a döntéshozó preferenciarendszerében alkalmazott i -edik hasznosság és fontosság értékekre (Dombi & Jónás, 2022).

2.2.1.1. Dombi és Jónás féle súlyozott aggregáló rendszer

A dolgozatban Dombi József és Jónás Tamás által készített súlyozott aggregáló rendszert alkalmazzuk. A módszer újdonsága és értéke, hogy szemben a hagyományos döntési eljárásokkal, ahol a súlyértékek megadása és az attribútumok értékelése kétlépésben, egymástól függetlenül történik, alkalmazásával az elvárési szintből származtathatjuk a súlyokat is.

A szimulációnk során tehát projekt terveket mint döntési alternatívákat készítünk. Ezen projektterveket felruházzunk néhány, a döntéshozó szempontjából releváns attribútummal.

A rendszer a döntéshozó elvárési szintjeit bemenetként fogadja. Ezeket az elvárásokat hasznosság szerint súlyszámokká alakítja és egy kvázi-aritmetikai középszámítás segítségével egyetlen értékben vonja össze. Ezt rangsorolva születik a döntési kimenet (outranking) (Xianliang & Shijing, 2019).

A továbbiakban tekintsük át a fenti módszer elméleti hátterét.

2.2.1.2. Aggregáló függvények és a kvázi-aritmetikai közepek

A kiterjedt információ összegzése, aggregálása, számhalmazok reprezentálása egyetlen számértékben, egy egyszerű és érthető koncepció. A számtani közép, de a minimum és a maximum értékek számítása is egyfajta aggregáció. (Grabisch et al., 2011)

Belátható, hogy a művelet során, az aggregálás mellett, el is veszítünk számos értékes, az eredeti halmazhoz kapcsolódó információt, ezért fontos az alkalmazásnak megfelelő aggregáló függvény kiválasztása (Zolfani et al., 2018). Mára ezen függvények tulajdonságainak és felhasználási területeinek kutatása a matematika önálló ágává vált. Az általánosan megfogalmazott tulajdonságoknak számos függvény, függvénycsalád megfelel (Grabisch et al., 2011).

A továbbiakban a súlyozott kvázi-aritmetikai közepeket fogjuk aggregálási célra alkalmazni (Grabisch et al., 2011), (Daróczy & Páles, 2003), (Dombi & Jónás, 2022) a kiterjesztett nemnegatív valós számegyenesen, amelyet a továbbiakban $\bar{\mathbb{R}}_+$ jelöl, azaz $\bar{\mathbb{R}}_+^n = [0, \infty]$.

Legyen $n \in \mathbb{N}, n \geq 1$. Akkor mondhatjuk, hogy az $M_{f,w}: \bar{\mathbb{R}}_+^n \rightarrow \bar{\mathbb{R}}_+$ függvény egy súlyozott kvázi-aritmetikai közép az $\bar{\mathbb{R}}_+$ halmazon, ha létezik olyan folytonos és szigorúan monoton $f: \bar{\mathbb{R}}_+ \rightarrow \bar{\mathbb{R}}$ függvény, és egy valós $w = (w_1, w_2, \dots, w_n) \in (0,1)$ vektor, amire igaz, hogy $\sum_{i=1}^n w_i = 1$ és

$$M_{f,w}(x) := f^{-1} \left(\sum_{i=1}^n w_i f(x_i) \right)$$

teljesül bármely $x_1, x_2, \dots, x_n \in \bar{\mathbb{R}}_+$ esetén.

Ezeket a közepeket azért nevezzük kvázi-aritmetikaiaknak, mert $f(x) = x$ esetén az aritmetikai közepet kapjuk (Grabisch et al., 2011). Az f függvényt a szóban forgó (súlyozott) kvázi-aritmetikai közép generátorfüggvényének nevezzük.

Az 1. táblázat bemutat néhány neves generátorfüggvény és kvázi-aritmetikai közép párosát.

1. táblázat: Példák kvázi-aritmetikai közepekre, Forrás: Saját szerkesztés (Grabisch et al., 2011)

$f(x)$	$M_f(x)$	Függvény neve
x	$\frac{1}{n} \sum_{i=1}^n x_i$	számtani közép
x^2	$\left(\frac{1}{n} \sum_{i=1}^n x_i^2 \right)^{1/2}$	négyzetes közép
$\log x$	$\left(\prod_{i=1}^n x_i \right)^{1/n}$	geometria közép
x^{-1}	$\frac{1}{\frac{1}{n} \sum_{i=1}^n \frac{1}{x_i}}$	harmonikus közép

2.2.1.3. Elvárási szintek, súlyok és a súlyozott kvázi-aritmetikai közép

Tekintsünk egy $f: [0,1] \rightarrow \overline{\mathbb{R}}_+$ folytonos és szigorúan monoton függvényt. Egy döntéshozó $x_1, x_2, \dots, x_n$ értékekhez tartozó elvárásai legyenek rendre $v_1, v_2, \dots, v_n \in (0,1)$. Itt v_i azt fejezi ki, hogy a döntéshozó számára mennyire fontos az a_i attribútum, azaz minél nagyobb v_i , annál fontosabb számára az a_i attribútum.

A súlyérték meghatározáshoz egy olyan $t_f: (0,1) \rightarrow (0,1)$ függvényt keresünk, amire teljesül az alábbi F1 és F3 feltétel.

F1:

$$\sum_{i=1}^n t_f(v_i) = 1$$

F2:

- Ha az f függvény szigorúan csökkenő, akkor bármely $v_i > v_j$ esetén $t_f(v_i) < t_f(v_j)$
- Ha az f függvény szigorúan növekvő, akkor bármely $v_i > v_j$ esetén $t_f(v_i) > t_f(v_j)$

Megmutatható, hogy az alábbi t_f függvény eleget tesz az F1 és F2 feltételeknek:

$$t_f(v_i) = \frac{f(v_i)}{\sum_{j=1}^n f(v_j)}$$

A w_i súly a döntési kritériumokból, tehát költség, erőforrás, átfutási idő stb. várakozási értékéből származik. A w_i súlyértékek nem negatívak és összegük 1. Az aggregáló rendszer felhasználási ereje ebben a lépésben rejlik. A döntéshozó meghatározza az elvárásait, amit azon logika mentén fordítunk át súlyokká, hogy feltételezzük, minél fontosabb számára egy szempont, annál magasabb értéket vár el. Így az i -edik súly értéke

$$w_i = \frac{f(v_i)}{\sum_{j=1}^n f(v_j)}$$

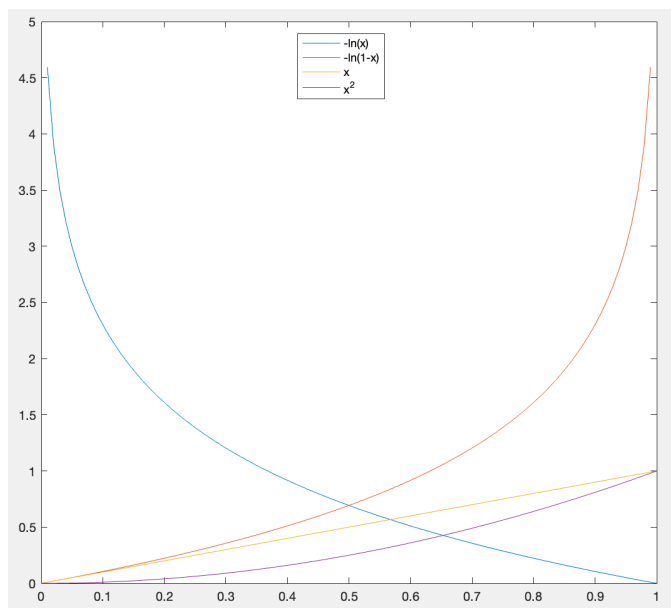
ahol vi_i az i -edik elvárási szintnek a $(0,1)$ intervallumon normalizált értéke, $i = 1, 2, \dots, n$. Továbbá x_i az aggregációban szereplő szimulált költségek, erőforrások, átfutási idők stb. minimum-maximum normalizált értékeit jelöli, azaz $x_i \in (0,1)$, $i = 1, 2, \dots, n$ (Dombi & Jónás, 2022).

2.2.1.4. A választott generátorfüggvény hatása

Projektkörnyezetben jellemzően kedvezőbbnek tekintjük azokat a megoldásokat, amelyek rövidebb idő alatt, kevesebb költségből érik el ugyanazt az eredményt. Az $f(x) = -\ln(x)$ generátorfüggvény használatával az elvárásnak megfelelő alternatívák viszont távol estek az origótól. Emiatt megvizsgáltunk más javasolt generátorfüggvényeket, nevezetesen az $f(x) = -\ln(1-x)$, $f(x) = x$, valamint az $f(x) = x^2$ függvényeket. Ezek alkalmazásával már az empirikus eredményekkel harmonizáló döntéseket állítunk elő.

A dolgozatban a $f(x) = x^2$ függvényt alkalmaztuk generátorfüggvényként.

Ahogy a 5. ábra mutatja, a jelenség hátterében a függvények monoton csökkenő, illetve növekvő jellege áll, ami logikai diszjunkcióval, illetve konjunkcióval köthető össze.



5. ábra, Saját szerkesztés: generátorfüggvények összehasonlítása

2.2.2. Az MCDM alkalmazása a dolgozatban

Az MDCM segítségével dolgozzuk fel az ABS során készített verziók összesítő kimeneti naplóját. Ahogy a fentiekben levezettük a Dombi és Jónás féle kvázi-aritmetikai közepet és többféle generátorfüggvényt alkalmazunk.

2.2.2.1. A hasznosság és elvárások

Ahhoz, hogy a Dombi és Jónás féle rendszert alkalmazni tudjuk, az összesítő eredményeket tartalmazó kimeneti napló elmeire a sikeres projektteljesítés szempontjából megítélt hasznosságként kell tekintenünk. Ez a gyakorlatban azt jelenti, hogy a nagyobb érték nagyobb hasznosságot hordoz.

A legtöbb szempontunk esetén viszont a kisebb értéknek tekintünk nagyobb hasznossággal. Az átfutási idő, a költségek, a felhasznált erőforrások száma és a terjedelem optimalizáció is kisebb értékek esetén tekinthetjük kedvezőbbnek. Ennek kezelésére a minimum-maximum normalizálást követően kivonjuk a kapott értéket 1-ből. Így a legrövidebb átfutású szimulációt tekintjük a leghasznosabbnak, míg a legtovább tartót a legkevésbé hasznosnak. A várakozás/tartalékidő ilyen konverziójára nincs szükség, ott csak a minimum-maximum normalizálás elvégzése szükséges.

Következő lépésként meghatározzuk (a döntéshozó nevében) az elvárásainkat, azokat a szempontonkénti értékeket, amely számunkra ideális eredményt adna. A gyakorlatban manuális ütemezés során ilyen eredményre törekednénk, pl.:

- Átfutási idő legfeljebb 30 nap
- Csapat mérete legfeljebb 4 fő
- A teljes költség maximum 121 embernap
- A terjedelem optimalizáció maximum 2%
- Az ütemterv tartalék pedig legalább 7 embernap

2.2.2.2. A generátorfüggvény és a súlyozott aggregáció

A 2.2.1.3 és 2.2.1.4 fejezetekben taglaltak szerint elkészítjük a szimulációs esetekre az aggregált értékeket. Egyrészt természetesen a legmagasabb ranggal rendelkező alternatíva felel meg leginkább a döntéshozó elvárásainak, másrészt az összes olyan szimuláció, ami magasabb eredményt ér el, mint az elvárásból számolt érték valójában szintén kielégíti a várakozást. Fontosnak tartom megjeleníteni mindkét értéket. Alacsony, illetve magas küszöb-érték feletti iterációs darabszám hasznos visszajelzést adhat a döntéshozó számára abban, hogy mennyire kiélezettek az elvárásai.

3. Demonstráció

A dolgozat készítése során a NetLogo szoftver mellett a coda.io³ online szoftvert alkalmaztuk az interaktív tervező felületet felépítésére. Az elvárások egy grafikus felületen történő paraméterezése alapján valós időben kirajzolódik a megfelelő ütemterv egy Gantt diagramon, illetve egy pontdiagramon nyomon követhetjük a választásunk projektátfutás-költség diagramon való elhelyezkedését.

3.1. Szimuláció és paraméterei

A valós idejű, interaktív felület csak a korábbi szimulációk kimenetei között képes választani, az értékes adat a NetLogo szoftverben keletkezik, a következő bemeneti paraméterek mellett:

- Task Quantity: a projektgenerátorban előállított tevékenységek száma
- Link Quantity: a projektgenerátorban előállított tevékenységek között kapcsolatok maximális száma
- Scope Reduction: a terjedelem optimalizálás legmagasabb értéke
- Worker Quantity és Range: a szimulációk során véletlenszerűen létrehozott erőforrások számának alsó és felső határa

3.2. Kimeneti naplók

A mellékletben találhatóak a kimeneti naplók, illetve azok részletei, amik a nyers feldolgozatlan, az MDCM rendszer számára készített adatokat tartalmazzák

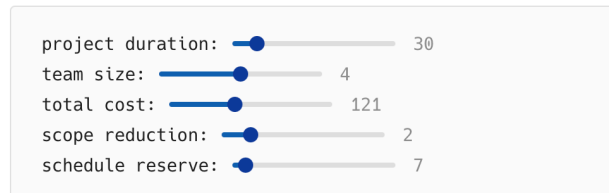
- 1. melléklet: Tevékenységek, kimeneti napló: 15 sor
- 2. melléklet: Szimuláció (részlet), kimeneti napló: 3755 sor
- 3. melléklet: Összesítés (részlet), kimeneti napló: 250 sor

³ <http://coda.io/>

3.3. Felhasználói interakció

A lenti ábrán a felhasználói interakciót biztosító vezérlőpanel látható. A vezérlők minimum, illetve maximum értékei az „Összesítés” adatból épülnek fel dinamikusan. Az interakció során van lehetőségünk megadni az elvárási szinteket az attribútumok szerint.

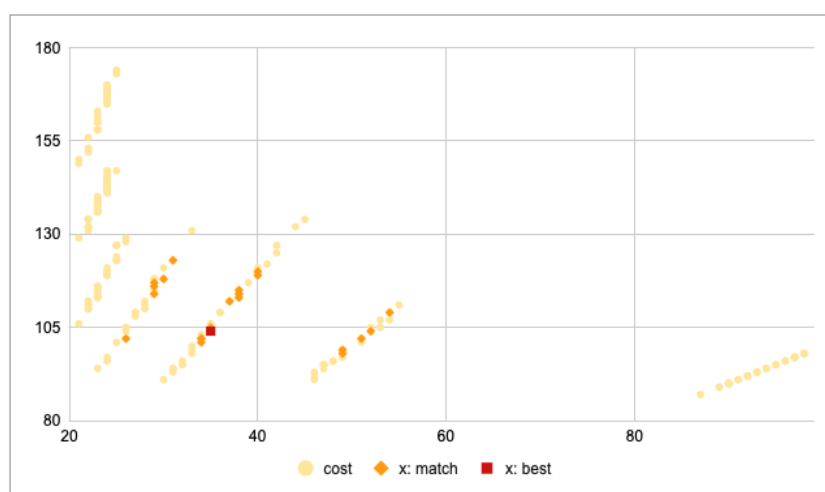
▼ expectations



6. saját ábra, az elvárások megadására szolgáló felület

Az lenti ábrán a projektátfutás-költség összefüggést ábrázoltuk, vízszintes tengelyen a projektalternatívák átfutásának napokban vett ideje, a függőleges tengelyen az alternatívák költsége jelenik meg. Minden pont egy alternatívát ábrázol. A diszkrét rendeződés az erőforrások nem folytonos jellegéből adódik. Az elvárási szintnek megfelelő (match) és legjobb (best) értékek valós időben adnak visszajelzést a felhasználó számára.

Tehát a vörös négyzet egy olyan szimulációs alternatívát ábrázol, amelyhez kapcsolódó projektterv alternatíva a leghasznosabb a döntéshozó számára, a narancs rombuszok pedig mind megfelelnek a megadott elvárási szintnek. A sárga körök pedig az elvárás alattiak.



7. saját ábra, projektátfutás-költség diagram

3.4. Elvárások

Tekintsük a demonstráció további részében, hogy döntéshozóként az alábbi paramétereket állítottuk be:

- Átfutási idő legfeljebb 30 nap
- Csapat mérete legfeljebb 4 fő
- A teljes költség maximum 121 embernap
- A terjedelem optimalizáció maximum 2%
- Az ütemterv tartalék pedig legalább 7 embernap

3.5. Aggregáció

A fenti elvárási értékek megadásával a lenti táblázat értékeit számoltuk ki

- Normalized: az elvárások a szimuláció minimum és maximum értékei segítségével előállított normalizált értékei, szükség szerint a hasznosság felől korrigálva
- $f(nu_i)$: a generátorfüggvény segítségével előállított értéke
- w_i : súlyozott aritmetikai középkel képzett súlyszám
- $w_expected$: az elvárásokból meghatározott referenciaérték, ami felett minden szimuláció elfogadható a döntéshozó számára

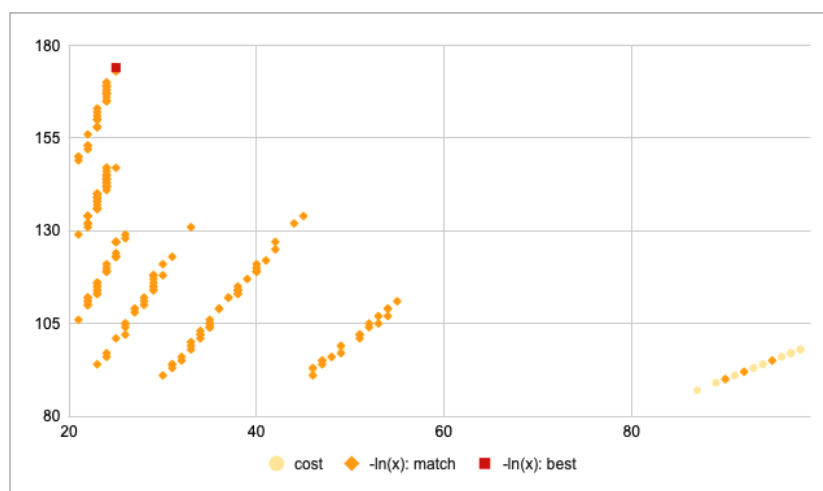
2. saját táblázat: Az elvárásokból számított értékek, normalizálás, generátor függvény alkalmazása, súlyértékek meghatározása

Name	min	max	expected	normalized	$f(nu_i)$	w_i	$w_expected$
duration	18,90	107,80	30	0,8751	0,7659	0,37	0,5696
resource	0,00	8,00	4	0,5000	0,2500	0,12	0,2460
cost	78,30	191,40	121	0,6225	0,3875	0,19	0,3417
scope	-1,00	15,00	2	0,8125	0,6602	0,32	0,5095
reserve	3,00	82,00	7	0,0506	0,0026	0,00	0,0079

3.6. Generátorfüggvény hatása

Az alábbi ábrákon hasonlítjuk össze a generátorfüggvényekkel előállított döntések eredményét. Kihangsúlyozandó, hogy a megfelelő generátorfüggvény választása (és paraméterezése) kritikus a döntés eredményességének szempontjából.

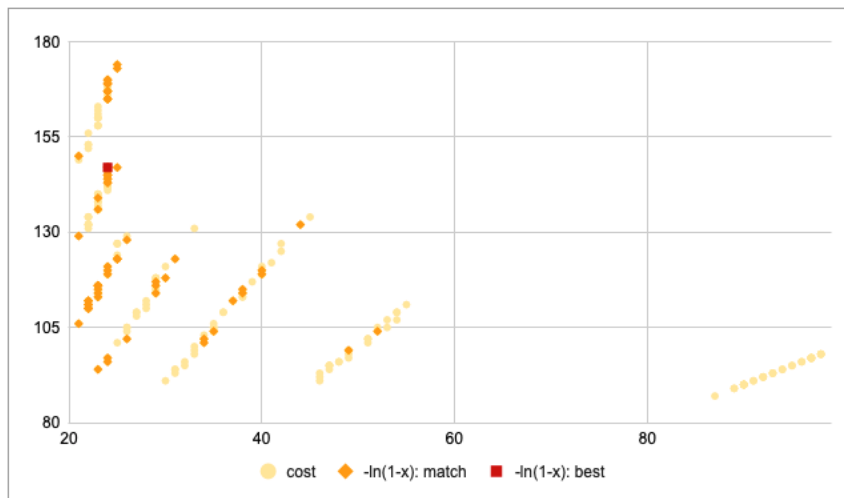
A 8. saját ábra diagramján a megfelelők (narancssárga rombuszok) a fenti szigorú elvárási szintek után is magában foglalja a szimulációk jelentős részét, és a döntés eredménye is több ponton szembe megy a várakozással. Ahogy fent megállapítottuk, $-\ln(x)$ monoton csökkenő jellegéből következik ez.



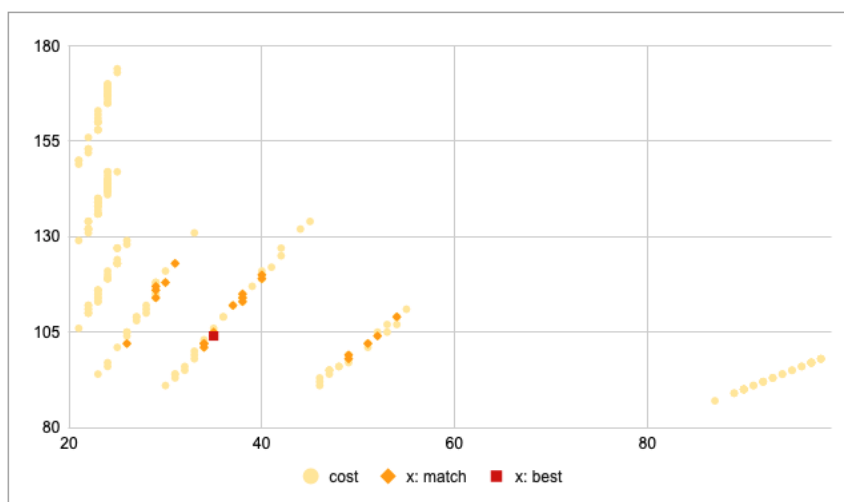
8. saját ábra, $f(x)=-\ln(x)$ generátorfüggvény és az elvárásokkal szembemenő eredmény

Vizsgáljunk meg további, monoton növekvő generátorfüggvényeket. A lenti ábrákon (9. és 10. ábra) a $-\ln(1-x)$ illetve az x generátorfüggvények döntései láthatóak, elvárásainknak

inkább megfelelő eredmény rajzolódik ki, szűkített számú megfeleléssel, de nem elég jól közelítő döntéssel. Indokolatlanul magas költséget generálnak.



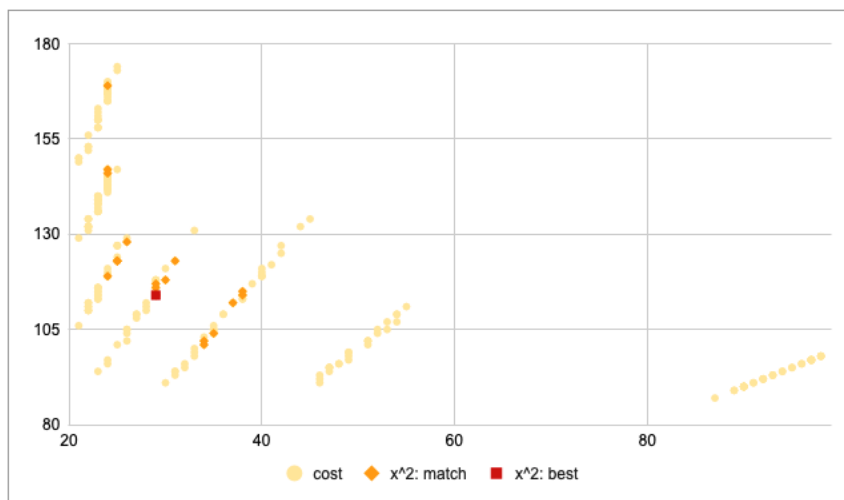
9. saját ábra, $f(x)=-\ln(1-x)$ generátorfüggvény alkalmazásával számolt döntés



10. saját ábra, $f(x)=x$ generátorfüggvény és a várakozásoknak megfelelő viselkedés

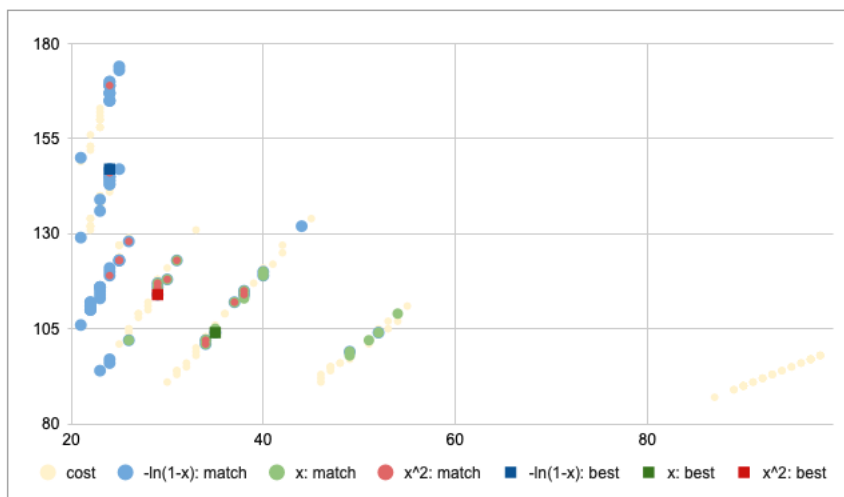
Végül vizsgáljuk az x^2 függvényt, a 11. saját ábra diagramján a várakozásunknak megfelelő eredmény rajzolódik ki, a megfelelő projekttervek száma kellően szűk, illetve a

döntés eredménye is harmonizál az elvárásként meghatározott értékekkel: az átfutás, erőforrás, költség, terjedelem, tartalék rendre 30,4,121,2,7, míg az eredményünk rendre 29,4,114,0,21.



11. saját ábra, $f(x)=x^2$ generátorfüggvény és a várakozásoknak megfelelő viselkedés

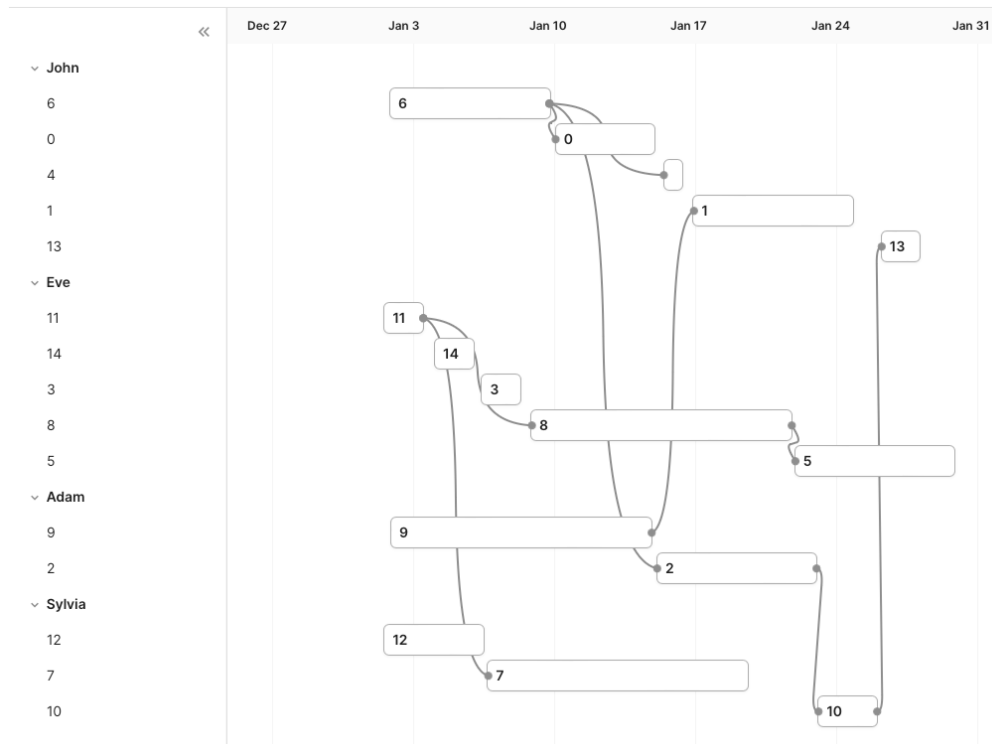
A lenti ábrán a három kedvezőbb kimenetet adó generátorfüggvény alkalmazásával számolt rangsorolás és döntés eredménye jelennek meg. Mind a megfelelő, mind a legjobb eredmények jelentősen más projektterveket eredményeznek.



12. saját ábra, a generátorfüggvények és az alkalmazásukkal számolt rangsorolás eredménye

3.7. Generált ütemterv

Végül a lenti ábrán a *coda.io* szoftverben felépített sávos ütemterv látható. A 3.3 fejezetben bemutatott projektátfutás-költség diagramhoz hasonlóan az elvárások megadásával valós időben visszkapjuk az iterációnak megfelelő ütemtervet.



13. saját ábra: A fenti elvárások alapján generált sávos ütemterv

Figyeljük meg a nem intuitív (emberi) döntéseket! Bár a tevékenységek között több belső függőségi kapcsolat is található, az első mégis egy független tevékenység (12), valamint a (11) és (8) közé beékelte (14) és (3) tevékenységeket.

4. Konklúziók és további kutatási lehetőségek

Évtizedek óta „készülünk” a mesterséges intelligencia térhódítására. Az utóbbi években sikerült némileg el is használni a kifejezést. A cél valójában egészen egyszerű és egyértelmű: készítsünk és használjunk olyan eszközöket, amelyek kiterjesztik az emberi intuíciót, tapasztalatot és intelligenciát.

A dolgozatban bemutatott módszer is ezzel a céllal született. A különféle terv variációk elkészítésére, az egymásnak gyakran ellentmondó megfelelési kritériumok elemzésére fordított idő felszabadítható a projektmenedzserek számára, akik ezt olyan, értékes feladatokra tudják fordítani, ahol még valós előnyünk van a gépekkel szemben.

Az ügynök szemlélet alkalmazása a projektkörnyezethez kiválóan illeszkedő szimulációs megközelítés. Segítségével leképezhetünk egyedi összefüggéseket, szabályokat. Megadhatjuk, hogy bizonyos kompetenciacsoporthoz tartozó erőforrások csak bizonyos tevékenységeket válasszák, vagy azok nehézségi szintjét a saját képességeikhez igazítsák. Az ügynökök tulajdonságait akár a konkrét projekthelyzethez is igazíthatjuk. A szimuláció alacsony erőforrásigénye kiemelt alkalmazási előnyt hordoz.

Az elvárási szint alapú döntési modell segítségével intuitív módon, egyetlen lépésben választhatunk a szimulált alternatívák közül, anélkül, hogy az attribútumok súlyáról nyilatkoznunk (egyáltalán elgondolkodnunk) kellene. Mivel az eredmény valós időben megjelenik, a paraméterek állításával a döntésünk finomhangolására is lehetőség adódik.

Egy integrált rendszerben az is elképzelhető, hogy az elvárási szint meghatározásával egy időben fut a szimuláció, és arra reagálva, valós időben generáljuk a döntéshozó számára a terveket.

Bár a dolgozatban a leginkább kedvezőnek értékelt $f(x) = x^2$ generátorfüggvény az empirikus várakozásoknak megfelelő eredményt produkál, további finomhangolással a döntéshozó személyére szabható. Egy tartós kísérlet keretében végigkísérhetünk projekteket és a különböző stádiumok, esetleg korrekciók után kialakított ütemezést hozzávethetjük a szimulációban levőkhöz. Megvizsgálhatjuk, hogy a végső tényadatokhoz legjobban illeszkedő terv milyen aggregáció mentén kapott volna legmagasabb értéket.

Szintén érdekes irány a szimulációk és paramétereinek optimalizálása. Belátható (sőt ismert tény), hogy az erőforrások növelése csak egy bizonyos szintig eredményez javulást az átfutási időben, így az azon túl futtatott szimulációk fölöslegesek. Sejtésünk (és egyszerű

kísérletek) szerint csökkenő logisztikus függvény mentén változik az erőforrás-átfutás grafikon. A szigmoid két paraméterével pedig közelíteni tudnánk a legnagyobb erőforrás mértékét, amivel még érdemes futtatni a szimulációt.

5. Irodalomjegyzék

- Dombi, J., & Jónás, T. (2022). Weighted aggregation systems and an expectation level-based weighting and scoring procedure. *European Journal of Operational Research*, 299, 580-588.
- Aruldoss, M., Lakshmi, T. M., & Venkatesan, V. P. (2013). A Survey on Multi Criteria Decision Making Methods and Its Applications.
- Davidsson, P., Holmgren, J., Kyhlbäck, H., Mengistu, D., & Persson, M. (2007). Applications of Agent Based Simulation. In L. Antunes, & K. Takadama, *Multi-Agent-Based Simulation VII. MABS. Lecture Notes in Computer Science* (Vol. 4442, pp. 5--27). Berlin: Springer.
- AIPM and KPMG. (2020). *Project Delivery Performance in Australia*.
- Railsback, S. F., Lytinen, S. L., & Jackson, S. K. (206, September). Agent-based Simulation Platforms: Review and Development Recommendations. *SIMULATION*, 82, 609-623.
- Wellington. (2022). *ANNUAL REPORT THE STATE OF PROJECT MANAGEMENT 2020*. Wellington.
- PMI. (2018, February). *\$1 Million Wasted Every 20 Seconds By Organizations Around the World*. Project Management Institute: <https://www.pmi.org/about/press-media/press-releases/2018-pulse-of-the-profession-survey>
- Fortune Business Insight. (2021). *Project Portfolio Management (PPM) Market Size, Share & COVID-19 Impact Analysis*. Fortune Business Insight: <https://www.fortunebusinessinsights.com/project-portfolio-management-ppm-market-104257>
- Finance Online. (2022). *101 Essential Project Management Software Statistics: 2022 Market Share & Data Analysis*. Finance Online: <https://financesonline.com/project-management-software-statistics/>
- Van Slyke, R. M. (1963). Monte Carlo Methods and the PERT Problem. *Operations Research*, 11(5), 839-860.
- Grabisch, M., Marichal, J.-L., Mesiar, R., & Pap, E. (2011). Aggregation Functions. Means. *Information Sciences*, 181(1), 1-22.

- IPMA in association with PWC. (2020). *Artificial Intelligence impact in Project Management*. IPMA and PWC.
- PMI. (2019). *AI Innovators: Cracking the Code on Project Performance*. PMI.
- PMI. (2021). *5 Implications of Artificial Intelligence for Project Management*. <https://www.pmi.org/learning/publications/pm-network/digital-exclusives/implications-of-ai>
- Holzmann, V., Zitter, D., & Peshkess, S. (2022). The Expectations of Project Managers from Artificial Intelligence: A Delphi Study. *Project Management Journal*, 55(5), 438-455.
- PMI. (2016). *PMBOK® Guide*. PMI.
- Taylor, P. (2022). *AI and the Project Manager*. Routledge.
- PMI. (2017). *Success Rates Rise*. PMI.
- PWC. (2019). *A Virtual Partnership?* PWC.
- Zargini, B. (2020). Multicriteria Decision Making Problems Using Variable Weights of Criteria Based on Alternative Preferences. *American Scientific Research Journal for Engineering, Technology, and Sciences*, 74(1), 1-14.
- Frini, A. (2017). A Multicriteria Intelligence Aid Methodology Using MCDA, Artificial Intelligence, and Fuzzy Sets Theory. *Mathematical Problems in Engineering*, 2017.
- Xianliang, L., & Shijping, W. (2019). A method to calculate the ranges of criteria weights in ELECTRE I and II methods. *Computers & Industrial Engineering*, 137.
- Zolfani, S. H., Yazdani, M., & Zavadskas, E. K. (2018). An extended stepwise weight assessment ratio analysis (SWARA) method for improving criteria prioritization process. *Soft Computing*.
- Bedzsula, B. (2021). *Projektmenedzsment Alapok, oktatási segédanyag a Minőség- és projektmenedzsment tantárgyhoz mesterszakos hallgatók részére*.
- PMI. (2016). *PMBOK® Guide, 6th Edition*. PMI.
- Project.co. (2021). *State of Project Management 2021*. Project.co.
- Fridgeirsson, T. V., Ingason, H. T., Jonasson, H. I., & Jonsdottir, H. (2021). An Authoritative Study on the Near Future Effect of Artificial Intelligence on Project Management Knowledge Areas. *Sustainability*, 13(4).

- Banks, S. C. (2002). Agent-based modeling: A revolution? *Proceedings of the National Academy of Sciences*, 99(3), 7199-7200.
- Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, 10, 144–156.
- Borgonovo, E., Pangallo, M., Rivkin, J., Rizzo, L., & Siggelkow, N. (2022). Sensitivity analysis of agent-based models: a new protocol. *Computational and Mathematical Organization Theory*, 28, 52-94.
- Railsback, S. F., Lytinen, S. L., & Jackson, S. K. (2006, September). Agent-based Simulation Platforms: Review and Development Recommendations. *Simulation*, 82(9), 609-623.
- Daróczy, Z., & Páles, Z. (2003). The Matkowski–Suto” problem for weighted quasi- arithmetic means. *Acta Mathematica Hungarica*, 100(3), 237–243.
- Simon, H. A., & Newell, A. (1971). Human problem solving: The state of the theory in 1970. *American psychologist*, 26(2), 145.
- Canals, J., & Heukamp, F. (2018. April 24). *10 Ways Artificial Intelligence Is Transforming Management*. Retrieved 2022. November, from IESE: <https://www.iese.edu/stories/10-ways-artificial-intelligence-is-transforming-management/>

6. Mellékletek

1. melléklet: Tevékenységek, kimeneti napló

name	estimate	link
9	13	
12	5	
5	8	8,
2	8	6,
1	8	9,
11	2	
10	3	2,
8	13	11,
6	8	
7	13	11,
3	2	
0	5	6,
13	2	10,
4	1	6,
14	2	

2. melléklet: Szimuláció (részlet), kimeneti napló

task	start	wip	worker	take	link	duration
3	1/1/2022, 1:00	48	John	1		2 days
12	1/1/2022, 6:00	112	Eve	1		4 days 16 hrs
9	1/1/2022, 7:00	306	Adam	1		12days 18 hrs
14	1/3/2022, 7:00	45	John	1		1 day 21 hrs
6	1/6/2022, 1:00	184	John	1		7 days 16 hrs

task	start	wip	worker	take	link	duration
11	1/6/2022, 9:00	47	Eve	1		1 day 23 hrs
7	1/9/2022, 3:00	290	Eve	1	11	12 days 1 hr
0	1/13/2022, 7:00	112	John	1	6	4 days 16 hrs
4	1/14/2022, 9:00	23	Adam	1	6	23 hrs
8	1/15/2022, 10:00	312	Adam	1	11	13 days
1	1/19/2022, 12:00	188	John	1	9	7 days 19 hrs
...	...	...	...	...	...	...

3. melléklet: Összesítés (részlet), kimeneti napló

take	duration	resource	cost	scope	wait
1	35	3	105	8	17
2	90	1	90	13	7
3	23	5	115	11	31
4	90	1	90	11	4
5	90	1	90	13	7
6	49	2	99	4	8
7	24	5	119	3	28
8	24	5	120	4	29
9	39	3	117	7	29
10	22	6	132	14	50
11	35	3	105	4	14
12	23	5	116	8	29
...	...	...	...	...	...