

Óbudai Egyetem

Doktori (PHD) értekezés



*Párosítás elméleti problémák megoldási lehetőségei egyetemi
környezetben*

Szikora Péter

Témavezető:

Kóczy Á. László

Biztonságtudományi Doktori Iskola

Budapest, 2016

Szigorlati bizottság:

Nyilvános védés teljes bizottsága:

Nyilvános védés időpontja:

TARTALOMJEGYZÉK

1.	Bevezetés	1
2.	Elméleti bevezetés	9
2.1.	Döntéseink háttere.....	9
2.1.1.	A döntés fogalma	11
2.1.2.	A döntési helyzet felismerése	13
2.1.3.	Döntés-előkészítés	14
2.1.4.	A döntéshozatal.....	28
2.1.5.	Végrehajtás és ellenőrzés.....	29
2.2.	Az operációkutatás	30
2.3.	Játékelmélet.....	35
2.3.1.	A játékelmélet típusai	36
2.3.2.	Kooperatív játékelmélet.....	37
2.4.	A párosításelmélet bemutatása	38
2.4.1.	A párosításelmélet és az operációkutatás kapcsolata.....	40
2.4.2.	Párosításelméleti alapfogalmak	41
2.4.3.	Párosítás elméleti modellek (problémák)	42
2.4.4.	Párosítás elméleti algoritmusok és azok összehasonlítása.....	44
3.	A szoftver tervezése és elkészítése	49
3.1.	A szoftver szükségessége	49
3.2.	A szoftver megtervezése	50
3.3.	Az adatbázis megtervezése	51
3.3.1.	Modell, modellezés, hasonlóság	51
3.3.2.	Adatmodellezés.....	52
3.3.3.	Tervezési szintek.....	53
3.3.4.	Hatékonyság.....	55

3.3.5.	A konkrét feladat megtervezése.....	55
3.4.	A szoftver bemutatása	58
3.4.1.	Az események meghatározása	59
3.4.2.	Az események felhasználókhöz tartozó preferenciája	60
3.4.3.	Az eredmények megtekintése	61
3.4.4.	A felhasználói oldal	62
4.	Létező párosítási helyzetek megvizsgálása.....	65
4.1.	Az Erasmusra való jelentkezés problémája	65
4.2.	KURZUS FELVÉTELI PROBLÉMA	72
4.2.1.	Gale-Shapley algoritmus.....	74
4.2.2.	Boston mechanizmus	77
4.2.3.	Konklúzió.....	79
4.3.	ÓRAI FELADATRA VALÓ JELENTKEZÉS PROBLÉMÁJA	80
4.3.1.	Statisztikai elemzés.....	83
4.3.2.	Konklúzió.....	85
4.4.	Hallgatók racionalitásának vizsgálata	86
4.5.	Szimulációs eredmények.....	88
4.5.1.	Eredmények	89
4.5.2.	Konklúzió.....	97
4.6.	Egyetemi felvételik elemzése.....	97
5.	Az elkészült szoftver alkalmazása	103
5.1.	Szervezeti magatartás órai feladatok kiosztása	103
5.1.1.	A probléma bemutatása	103
5.1.2.	A helyzet értékelése	105
5.1.3.	Hallgatói reakciók.....	106
5.2.	Kutatócsoport feladatainak szétosztása.....	107

5.2.1.	A helyzet bemutatása	107
5.2.2.	A helyzet értékelése	109
5.2.3.	Tagok elégedettsége.....	110
5.3.	További alkalmazási lehetőség	111
6.	Összefoglalás, tézisek	115
7.	Az eredmények hasznosítása, továbbfejlesztési lehetőségek	118
Irodalomjegyzék		I
Az értekezés témájához kapcsolódó tudományos közlemények		XV
Ábrajegyzék		XVII
Táblázatjegyzék		XIX
Mellékletek		XXI
Melléklet –SQL adatbázis létrehozó script		XXI
Melléklet – Kérdőív – elégedettség vizsgálat		XXIII
Melléklet – Kérdőív – Felsőoktatási jelentkezések		XXIV

1. BEVEZETÉS

A szervezetek célja önmaguk fenntartása; a tulajdonosok, és az érintettek (stakeholderek) érdekeinek a kiszolgálása. [1] Ennek megfelelően a szervezetek vezetői olyan döntéseket hoznak, amelyek lehetővé teszik a szervezet környezeti viszonyokhoz (pro-, pre-, reaktívan) való alkalmazkodását. [2] A szervezetek mindennapi életének velejáráói a kockázat és a bizonytalanság, amelyek alapjaiban befolyásolják a szervezet alkalmazkodásának sikerességét, szélsőséges a szervezet túlélését is. A turbulens változó piaci viszonyok és a technika, különösen az információtechnológia ugrásszerű fejlődése következtében a szervezet zavartalan működését egyre több biztonsági esemény fenyegeti, és ezek előfordulása is egyre gyakoribb. A szervezeti biztonsággal foglalkozók sokáig csak a szervezeten kívülről érkező fenyegetésekre, és az azokkal szembeni védelemre koncentráltak. Céljuk a szervezetek teljes körű, minden támadást kivédő (megelőző?) védelme volt. A biztonsági kockázatok minimalizálása, a veszélyforrások azonosítása, a lehetséges veszélyek - természeti katasztrófák, betörés/lopás, egyéb külső fenyegetések – valószínűségének és bekövetkezési kockázatának becslése, valamint a lehetséges károk elhárítása volt a szervezeti biztonság homlokterében. A biztonság így a védettség szinonimája volt, míg a biztonság megteremtése a védelem megszervezését és biztosítását jelentette. A védelmi stratégia ennek megfelelően a szervezetre, mint léte-sítményre, technikai és informatikai architektúrára tekintett. Nem vette figyelembe a szervezet komplex, szocio-technikai rendszerét. [3]

A holisztikus szervezetfelfogások térhódításával azonban egyre gyakoribbá vált a szervezeti biztonság belső fókuszának – a szervezeten belül működő csoportoknak, és az azokat alkotó egyéneknek – a hangsúlyozása. A döntéshozók azt is felismerték, hogy a stabil szervezeti környezetre és alacsony változékonyságú szervezeti struktúrára és folyamatokra jól adaptálható szabályzatok és SOP-k (Standard Operating Protocoll) nem képesek maradéktalanul megfelelni a modern kor biztonsági kihívásainak; szükségszerűvé vált tehát a társdiszciplínák felé történő nyitás. [4])

A szervezeti magatartás diszciplínája arra hívta fel a figyelmet, hogy a szervezeti tagok számára a biztonság már nem (csupán) a szervezet egy meghatározott állapota (a külső és belső fenyegetésektől való mentesség), hanem egy individuális jelenség is, melyet erősen befolyásolnak az egyén személyes jellemzőiből fakadó észlelési szűrők. A biz-

tonság iránti igény ugyanis egyetemes motivátor, [5] egyike az állatvilágban és az emberek között egyaránt megtalálható tényezőknél. Míg az állatvilágban ez a lét- és fajfenntartás szintjén jelenik meg, addig az embereknek ezen túlmenően nem csupán személyükkel, és hozzátartozóikkal kapcsolatban jelenik meg a biztonság igénye, de a tulajdonukkal, életterükkel, mi több a munkahelyükkel kapcsolatban is. [6] A biztonság iránti vágy az egyén azon törekvést fejezi ki, hogy szükségleteit a jövőben is tartósan kielégíthesse és körülményei viszonylag kiszámíthatók és jól előre jelezhetők legyenek. Ez az előre jelezhetőség, tervezhetőség azonban igen ritkán van jelen az egyén, vagy akár a szervezetek életében. A szervezetek jövőjét érintő döntések többnyire ad-hoc születnek, korlátozottan racionálisak, és ennek megfelelően szuboptimális eredményre vezetnek. A szervezeteknek, a szervezeti döntéshozóknak ezzel együtt az a célja, hogy lehetőség szerint jó döntéseket hozzanak.

A filozófia szempontrendszer szerint az a döntés tekinthető jónak, amely megfelel a döntéshozó céljának. [7] Az előíró irányzat szerint azonban a döntés, illetve a döntési alternatívák egy fix vonatkoztatási rendszer viszonylatában tekinthetők jónak, avagy rossznak. A preskriptív irányzat „jó”-ság fogalma tehát racionális döntéshozatali algoritmust, valamint teljes informáltságot feltételez. Míg a teljes informáltság nem érhető el, addig az adott információkat figyelembe vevő racionális döntéshozatal (korlátozott racionalitás), a szuboptimális megoldások kockázatát csökkentő döntések megvalósíthatóak mind egyéni, mind pedig szervezeti szinten. Ennek eszközéül szolgálhatnak a döntéseméleti, párosításeméleti algoritmusok.

A tervezhetőség, bejósolhatóság azonban nem csupán mint tartaloméleti építőelem jelenik meg a szervezeti tagok motivációjában, hanem közvetítő, eszközértékként is. Ahhoz, ugyanis, hogy egy egyén hatékonyan és eredményesen teljesítsen egy szervezetben, elengedhetetlen, hogy készletét érezzen a szervezeti céloknak megfelelő. Az elvárásélet szerint, amennyiben az egyén a munkavégzésének eredményére vonatkozó predikciói nem teljesülnek, az a motiváció csökkenéséhez, rosszabb színvonalú munkavégzéshez, hosszabb távon pedig munkával való elégedetlenséghez vezet. [8] Az egyén tehát törekszik a képességeinek leginkább megfelelő feladatok kiválasztására és azok megvalósítására, cserébe azonban jól tervezhető ellentételezést vár. Párosításeméleti algoritmusok segítségével megalkothatók olyan párosítások, ahol a szervezeti tagok úgy kapják a kompetenciáiknak leginkább megfelelő feladatot, hogy a motiváció (a ki fizetés), és az elégedettség szervezeti szinten maximális.

A motivációs folyamatelméletek másik fontos a szervezeti döntésekre vonatkozó indikációja a döntések végeredményeképpen kialakult helyzet észlelt igazságosságára vonatkozik. [9] A szervezeti tagok döntésük, munkájuk végeredménye képen legalább olyan jól szeretnének járni, legalább olyan hasznosságot szeretnének elérni, mint munkatársaik. Amennyiben ezt nem tudják elérni, a kialakult helyzetet méltánytalannak tartják, és motivációjuk csökken. Emellett a fennálló helyzet megváltoztatására törekedhetnek, amely a rendszer, és végső soron a vállalat stabilitásának csökkenéséhez vezethet. Számos párosítási algoritmus képes stabil párosítás létrehozására, amelyek jellemzője, hogy egyetlen szervezeti tag sem kaphat nagyobb kifizetést anélkül, hogy egy másik tag kifizetése ne csökkenne. Emellett az algoritmusok személyektől független 'racionális' működése megelőzheti a méltánytalanság észlelését [10].

A kultúra ugyanis nem más, mint egy módszer, annak módja, ahogy az emberek egy csoportja megoldja a problémáit és eldönti dilemmáit – majd ezek a problémák, amelyeket rendszeresen megoldanak, eltűnnek a tudatukból és alapvető feltételezéssé válnak, nyilvánvaló premisszává válnak. [11] Ezek az alapvető feltételezések azután értékeket, hiedelmeket generálnak, amelyeket a csoport tagjai egyöntetűen elfogadnak. Schein a kultúrát a tudatosság különböző mértékével rendelkező elemek rendszerének tekinti, amelynek mindhárom szintjén - alapvető feltevések, köznapi tudat, a tárgyi világ - más és más szabályok érvényesek. [12]

Az alapvető feltevések az axiomatikusan elfogadott alapvető hiedelmek, melyeknek az egyén legtöbbször nincs tudatában, éppen ezért nem reflektál rájuk. [13] Ezen a szinten minden individuum a saját kis világának foglya, tehát holisztikus megismerésre saját korlátai miatt képtelen. [14]

A köznapi tudat szintjén található a korábbiakban már említett morális iránytű, a jó és rossz megkülönböztetését elősegítő morális elvek; a világ működését, jelenségeit és ahhoz kapcsolódóan az emberi természetet magyarázó ideológiák, eszmék; a szervezeti tagok viselkedését alapvetően orientáló, a preferencia sorrendeket meghatározó értékek; valamint az egy adott helyzetben az észlelő érzelmi irányultságát meghatározó attitűdök.

A kultúra legszélesebb körben nyilvánvaló szintjén, a tárgyi világ szintjén jelenik meg a szervezeti biztonság fizikális eszköztára – a beléptető és védelmi rendszerek, az informatikai infrastruktúra valamint a biztonsági eseményekre vonatkozó szabályok, proto-

kollok és a vészhelyzeti kommunikáció is. Ez a szint a legtriviálisabb, és többnyire itt kezdődik a biztonsági/védelmi rendszerek kiépítése.

A biztonsági kultúra szervezeti biztonság felfogása szerint az eredményes szervezeti biztonsági rendszerhez szükséges, hogy a személyes (értékek, attitűdök, hiedelmek, észlelés és felfogó készség) és a viselkedési tényezők (megértés, bizalom, elkötelezettség, motiváció/szándék és az éberség) egyszerre irányuljanak a szervezeti biztonság megteremtésére, megtartására. [15] A biztonsági kultúrának tehát szükséges eleme a tudás és kompetenciák mellett a tudatosság és a szándékosság, azaz a megfelelő motiváció megléte. A szervezeti tagok szándékán múlik, hogy képesek és hajlandók-e a szervezeti biztonságot fenyegető események észlelésére és az adekvát döntések meghozatalára; magukénak tekintik-e a felmerülő problémákat, hogy azok megoldásán dolgozzanak; vállalnak-e felelősséget saját döntéseikért, közvetlen munkatársukért, vagy éppen az egész szervezetért, és annak zökkenőmentes működéséért?

A hatékony biztonsági kultúra képes a szervezeti tagok észlelését, éberségét és értékrendjét oly módon befolyásolni, hogy a szabályozások által nem érintett döntési helyzetekben is gyors, a szervezeti biztonságot erősítő döntések születhessenek.

A szervezeti kockázatok csökkentéséhez, és ez által a szervezeti biztonság növeléséhez a döntéelmélet tudománya is jelentősen hozzájárult. Az elmúlt években – a bizonytalan környezetben hozott döntéseket érintő kutatások és a vonatkozó döntéelméleti modellek fejlődésének köszönhetően – jelentősen megnőtt a döntéelmélet ipari felhasználásának gyakorisága. [16]

A munkahelyi folyamatok és feladatok mindegyikéhez számos szervezeti biztonságot fenyegető vészhelyzet kapcsolódik, vagy kapcsolódhat. A szervezetek összetettségének növekedésével nem várható el minden alkalmazottól, hogy ezeket teljes komplexitásában átlássa, megértse. A szervezeti biztonság szempontjából megfelelő munkavégzés ugyanis nem más, mint az adott feladatban rejlő kockázatok és az azokhoz kapcsolódó bizonytalanságok rendszerének észlelése, értékelése után a szervezet szempontjából optimális, illetve maximális hasznot hozó döntések meghozatala, és azok megvalósítása. [17]

A megértés és a tudatos döntéshozatal tehát lényegi része a szervezeti biztonságot szem előtt tartó mindennapi működésnek. [18] A rendszer, és az adott feladatok komplexitása

azonban sokszor lehetetlenné teszi a kockázatok teljes körének felmérését, így a programozható, illetve programozott (biztonsági irányelvekkel szabályozható) döntések köre is korlátozott.

Ennek megfelelően különösen fontos, hogy a megfelelő emberek kerüljenek a megfelelő pozíciókba [19], és még inkább az, hogy a megfelelőség ne (csak) szubjektív értékítélet legyen, hanem objektív szempontok is érvényesüljenek a feladatok emberek közötti szétosztásánál.

A döntéshozatali mechanizmusok objektív alapokra helyezése egyrészt legitimitást ad a döntésnek, és ez által a szervezeti folyamatok rendjének is [20] és [21], másrészt – pontosan a megnövekedett legitimitás miatt magasabb a döntések elfogadásának szintje, (kisebb, az azokkal együtt járó elégedettség, nagyobb a társas elfogadás). Ez a kognitív és pszicho-szociális rendezettség a rendszer stabilitását, hosszú távú fenntarthatóságát eredményezi, amely a biztonságosan működő szervezetek egyik elengedhetetlen feltétele.

Mint ahogyan a szervezeti magatartás és a döntéelméleti alapok szervezeti biztonsággal való kapcsolatát bemutató felvezetésből is világos, a szervezeti mindennapokat át meg át szövik a döntések. Jelen dolgozattal az a célom, hogy megmutassam, hogy bár számos probléma esetén használunk párosításokat, ezeket nagyon ritkán tesszük tudatosan, vagy nem hatékony algoritmusokat alkalmazunk; és ennek következménye képen a döntéseknek is nagy a biztonsági kockázata.

A döntéseink háttérében sokszor olyan problémák állnak, melyekben nem rendelkezünk elég információval, és ezért a döntéseink inkább csak ad-hoc jelleggel születnek. Információhiányos helyzetben általában szuboptimális döntések születnek. Ezekre a problémákra azonban a párosítás elmélet igen sok helyen kínál stabil, optimális megoldást.

Dolgozatom célja, hogy megvizsgálja a párosítás elméleti problémák megoldásának alternatíváit, a különböző algoritmusokat összehasonlítsa, továbbá olyan szempontok meghatározása, amelyek alapján az algoritmusok összemérhetőek, értékelhetőek. Teszem mindezt azért, hogy a megfelelő döntéshozatali mechanizmus(ok) szervezeti mindennapok részévé tétele által alternatívát kínáljak a döntéseket minden esetben programozott (programozható) relációkként kezelő biztonsági szabályozásokra. Célom volt továbbá egy szoftver megtervezése és elkészítése, amelynek segítségével ezeket a ma-

tematikai algoritmusokat egyszerűen, nagy létszámú csoportok esetében is alkalmazni lehet.

A piacon elérhető algoritmusok, bárki számára elérhető szoftverek, jelenleg nem képesek külső beavatkozás nélkül, - vagyis olyan ember segítségével nélkül, aki járatos a párosítás elméletben, - párosításokat alkotni, mert sokszor igen bonyolult, ha egyáltalán lehetséges, az adott problémához igazítani a kialakításukat, legyen szó a feladatok, vagy résztvevők számosságáról, a hozzárendelés szempontrendszeréről, vagy a párosításhoz alkalmazható algoritmusokról. Ezzel együtt ma is sok helyen alkalmaznak ilyen szoftvereket, de ezek a szoftverek nem nyilvánosak, a nagyközönség számára nem elérhetőek.

Ezért is tartom szükségyszerűnek és időszerűnek egy olyan program megalkotását, amely az informatikához, vagy az algoritmusok működéséhez nem hozzáértőknek – például: a vállalatvezetőknek, vagy akár a vállaltnál az emberi erőforrással foglalkozó alkalmazottaknak – is segítséget nyújt párosítási problémák megoldásában.

Egyetemi oktatóként az egyetemről, mint szervezetből indulok ki. Mindennapjaim során számos olyan problémával találkoztam, amelyek megoldhatók lennének párosítás elméleti algoritmusok segítségével, de jelen pillanatban inkább csak a véletlen számít ezeknek a párosításoknak a létrehozásánál, ritkább esetben pedig olyan algoritmust alkalmaznak a döntéshozók, amelynek a lépései hasonlítanak az úgynevezett Mohó algoritmusra.

Dolgozatomban megvizsgálom különböző probléma helyzeteket és az azokat feloldani képes algoritmusokat. A kutatásom során arra törekszem, hogy találjak olyan helyzeteket, amelyekben hatékonyan lehet párosításelméleti algoritmust alkalmazni. Ennek érdekében első lépésben megvizsgálom, hogy milyen kapcsolat van a problémák és az azokat megoldani hivatott döntések között; valamint hogy milyen algoritmusok segítenek a lehető legjobb eredmény elérésében.

Döntéseink esetén legtöbbször nem vagyunk képesek optimális, - számunkra, illetve a szervezet számára leginkább megfelelő - döntéseket meghozni, megvizsgálom ezért annak a hátterét, miért nem vagyunk képesek rá, illetve mire lenne szükség ahhoz, hogy hatékony döntésekkel növelhessük a szervezet hatékonyságát. Az egyik megoldás lehetne a nagyobb informáltság elérése. Mivel a tökéletes informáltság a dolgozatban később bemutatott indokok miatt nem elérhető, így figyelmemet a döntéshozatali algorit-

musok vizsgálatának szentelem. Igyeekszem megtalálni azt a döntéshozatali eljárást, amellyel optimális döntések, konkrétan párosítások születnek.

Az optimális döntés fogalmát többféle képen is meg lehet határozni. Megvizsgálom tehát, hogy mi a különbség a párosításelméleti algoritmusok által alkotott stabil párosítások és az operációkutatás céljaként kitűzött optimális párosítások között.

Egy párosítást akkor tekinthetünk optimálisnak, ha kiszámítható a párok értéke, és megtalálható a legmagasabb összesített hasznosságot. A párosítás ezen formájával foglalkozik az operációkutatás, azon belül is a szállítási és a hozzárendelési problémák témakörre. Természetesen nem minden probléma helyzetben oldható meg, hogy az optimalitás kritériumának megfelelő párokat hozzunk létre, mert az embereknel az egyéni hasznosság sokszor fontosabb, mint a közösség összesített hasznossága.

Ezzel szemben a stabil párosítások esetében nem az összesített hasznosság, hanem az egyének szempontjából számított egyéni hasznosságok a mérvadóak. Olyan párosításokat hoz létre, ahol az egyének csak azért nem kerülhetnek jobb párba, mert ott már egy náluk hasznosabb egyén van.

A vizsgálataim során a következő kérdésekre keresem a választ:

- Egyetemi környezetben melyek azok a jellemző helyzetek, ahol párosításokat kell létrehozni, de a döntéshozók nem alkalmaznak rá semmilyen algoritmust tudatosan, így a folyamat lassú és nem hatékony; nem ad optimális megoldást, nem generál megelégedettséget az érintettekben?
- Melyik algoritmust, vagy algoritmusokat érdemes a mindennapi párosítási helyzetekben alkalmazni?
- Alkalmazható(ak)-e az algoritmus/algoritmusok olyan párosítási helyzetekben is, amelyben nem ismert mindkét fél preferenciája?
- Lehet-e, és ha igen, akkor hogyan értékelni az alkalmazott algoritmusokat, a stabilitáson túlmenően valamilyen hasznosságot jelző számszerűsíthető érték segítségével?
- Racionálisan viselkednek-e a döntéshozók? Mennyire használják ki a rendelkezésükre álló lehetőségeket, olyan esetekben ahol nem egy konkrét párosítás létrehozása a cél, hanem az, hogy minden érintett részt tudjon venni a párosításban?

- Mennyire számít a preferencia lista hossza a különböző algoritmusoknál, ugyanolyan bemenő adatok esetén?
- Létrehozható-e olyan szoftver a párosítási helyzetekre, mint matematikai problémára, amely bárki számára könnyen kezelhető; amely az adatok bevitele után automatikusan megadja a párosításokat?
- Lehetséges-e ezt, ezeket az algoritmust/algoritmusokat számítógépes alkalmazás segítségével szélesebb körben is népszerűsíteni, alkalmazni?
- A döntésekben való részvétel növeli-e a résztvevők elégedettségét?

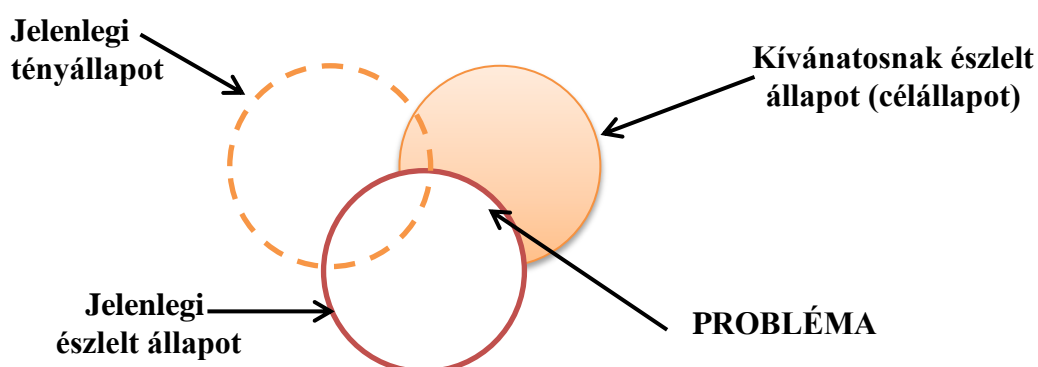
A kutatási kérdéseimnek megfelelően a dolgozatban a párosításelmélet neves algoritmusait alkalmazom, azonban törekszem az operációkutatás által célként definiált összes hasznosság mérésére és növelésére. A dolgozatban bemutatott szakirodalmi irányvonalak mentén különböző értékelő feltételeket határozok meg, és ezek alapján elkészíték egy egyszerűen használható szoftvert. Ez a szoftver először csupán „backend”-ként kerül tesztelésre, majd később könnyen használható internetes „frontend”-del kiegészülve lesz segítségemre abban, hogy megvizsgáljam, hogy melyik párosítás elméleti algoritmus a hatékonyabb az operációkutatási optimalitás szempontjából.

A „backend” program lényege az, hogy az érintett felek preferenciái segítségével különböző párosítás elméleti algoritmusokat tud szimulálni és segítségével összehasonlíthatóvá válnak azok eredményei. A program előnye a klasszikus papír alapú kidolgozáshoz képest, hogy valós problémák mellett jól kezel véletlen számok segítségével generált komplex helyzeteket, és a gyors futási idő miatt statisztikai módszerekkel elemezhető mennyiségű párosítás generálható a segítségével. A szoftver felhasználói felületének, az úgynevezett „frontend” elkészítésére azért került sor, mert a szervezeti magatartás elméleti alapvetései mentén, [22] [23] ha a döntés – jelen esetben a párosítás – úgy születik meg, hogy annak létrehozásában az érintettek tevékenyen részt vesznek, akkor az érintettek könnyebben elfogadják, magukénak tekintik a kapott eredményeket. Az így hozott döntések tehát azáltal, hogy az azok meghozatalában részt vevők helyzettel és a döntéssel való elégedettségét növelik, hozzájárulnak a rendszer/szervezet stabilitásához, hosszabb távú hatékony működéséhez.

2. ELMÉLETI BEVEZETÉS

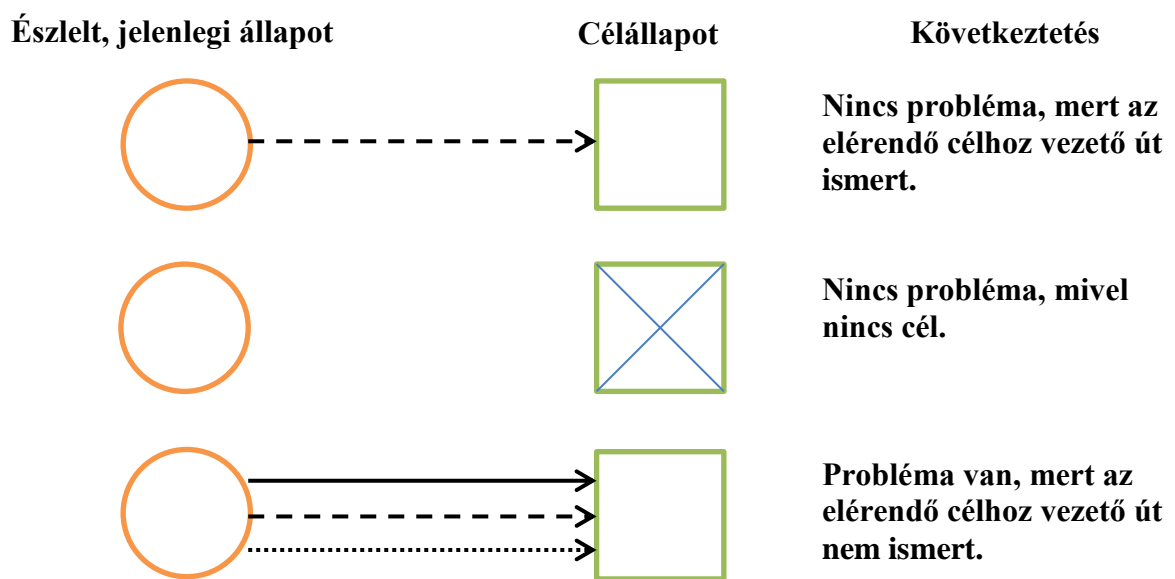
2.1. Döntéseink háttere

A szervezetek mind külső környezetük, mind pedig belső folyamataik, erőforrásaik elosztása miatt folyamatos döntéskényszerben vannak; és ezeknek a döntéseknek megszeméltő következményei lehetnek a szervezeti biztonságra, a szervezet gazdaságosságára és végső soron az életképességére is. A döntés igazából nem más, mint probléma megoldás. „Problémának nevezzük a legáltalánosabb értelemben azt a helyzetet, amelyben bizonyos célt el akarunk érni, de a cél elérésének az útja rejtve marad.” [24] A probléma jelen definíció szerint tehát a célállapot eléréséhez vezető út feltáratlansága, bizonytalansága. Létezik azonban a problémának, mint fogalomnak egy sokkal részletesebb definíciója is. A probléma: „egy észlelt idejű állapot megváltoztatását vagy fenntartását célzó kielégítetlen szükséglet, igény, ami egy kívánatosnak minősített állapot elérésére (vagy fenntartására) irányul. A megváltoztatni (vagy fenntartani) kívánt állapotot problémaállapotnak, a kívánatosnak minősített állapotot megoldási állapotnak, vagy célállapotnak nevezzük. A probléma megoldása akkor következik be, ha az észlelt jelenlegi állapotot és a kívánatos állapotot a döntéshozó azonosnak érzékeli. A problémamegoldás az a tevékenység, amely a problémaállapot megoldási állapotot (célállapotot) való alakításával kapcsolatos.” [25] Ahogy látható, a probléma nem más, mint amikor a kívánatos vagy célállapot eltér az általunk észlelt tényállapottól, és az elérésnek az útja számunka nem látható. A probléma meghatározása látható az 1.1. és a 1.2. ábrán.



1 ábra – A probléma meghatározása [26] alapján

Mint az 1. ábrán látható, probléma nem akkor adódik, amikor a jelenlegi tényállapot és a kívánatosnak ítélt állapot nem esik egybe, hanem akkor, amikor a jelenlegi észlelt állapot és a kívánatosnak észlelt állapot különbözik. Amíg nincs teljes informáltság, addig a tény- és az észlelt állapotok egymástól különböznek. Természetesen számunkra ez még kevés, hogy problémáról beszélhessünk. Ahogy a 2. ábrán látható, a probléma nem csupán az észlelt és kívánatos állapotok eltérése következtében alakulhat ki, de akkor is problémáról beszélünk, ha a kívánatos állapot többféleképpen érhető el, vagy ha nem ismerjük a jelenlegi állapotból a kívánatos állapotba vezető utat.

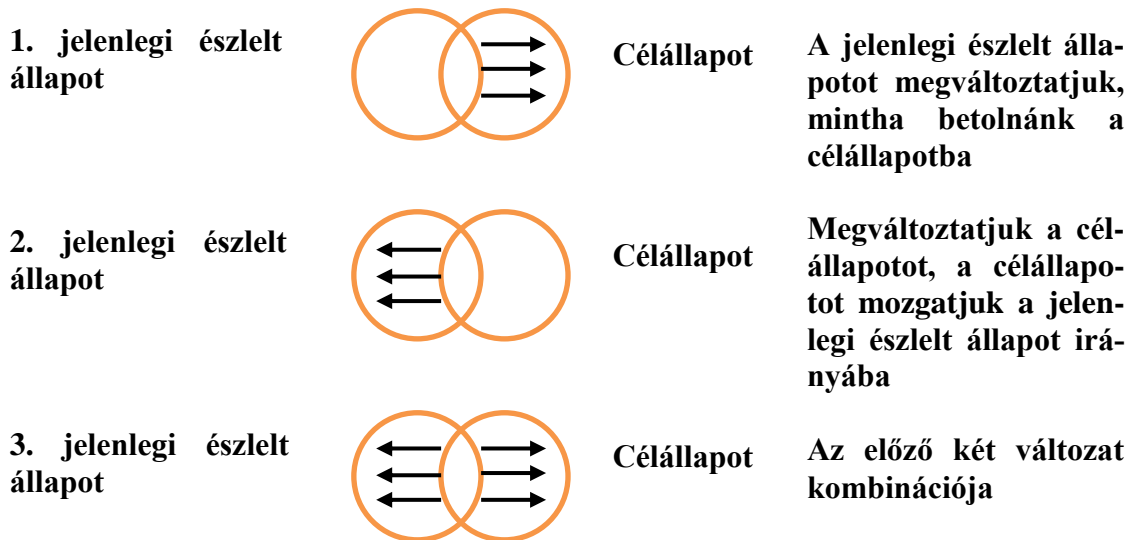


2 ábra – A probléma meghatározása [26] alapján

Tehát probléma megoldás esetén az a célunk, hogy ezt a két halmazt egymással fedésbe hozzuk. Erre több megoldási módot is kaphatunk.

1. Az észlelt jelen idejű állapotot a kívánatosnak észlelt állapottá alakítjuk.
2. A kívánatosnak észlelt állapotot alakítjuk az észlelt jelen idejű állapottá.
3. Az első két probléma megoldási változat kombinációját alkalmazzuk.

Probléma megoldás esetén általában a 3. megoldást alkalmazzuk leggyakrabban. [25]
Ezek a megoldási módok láthatóak a 3. ábrán.



3 ábra – probléma megoldásának módjai [26] alapján

2.1.1. A döntés fogalma

A döntés fogalma természetes módon összekapcsolódik a probléma fogalmával. „A döntés célirányos emberi választás adott környezetben cselekvési változatok között, ahol a cselekvési változatok a döntési folyamat, döntést megelőző szakaszában cselekvési lehetőségként vannak feltárva.” [27]

Mint ahogyan a probléma esetében is, a döntésnek is létezik egy jóval tágabb értelmű definíciója.

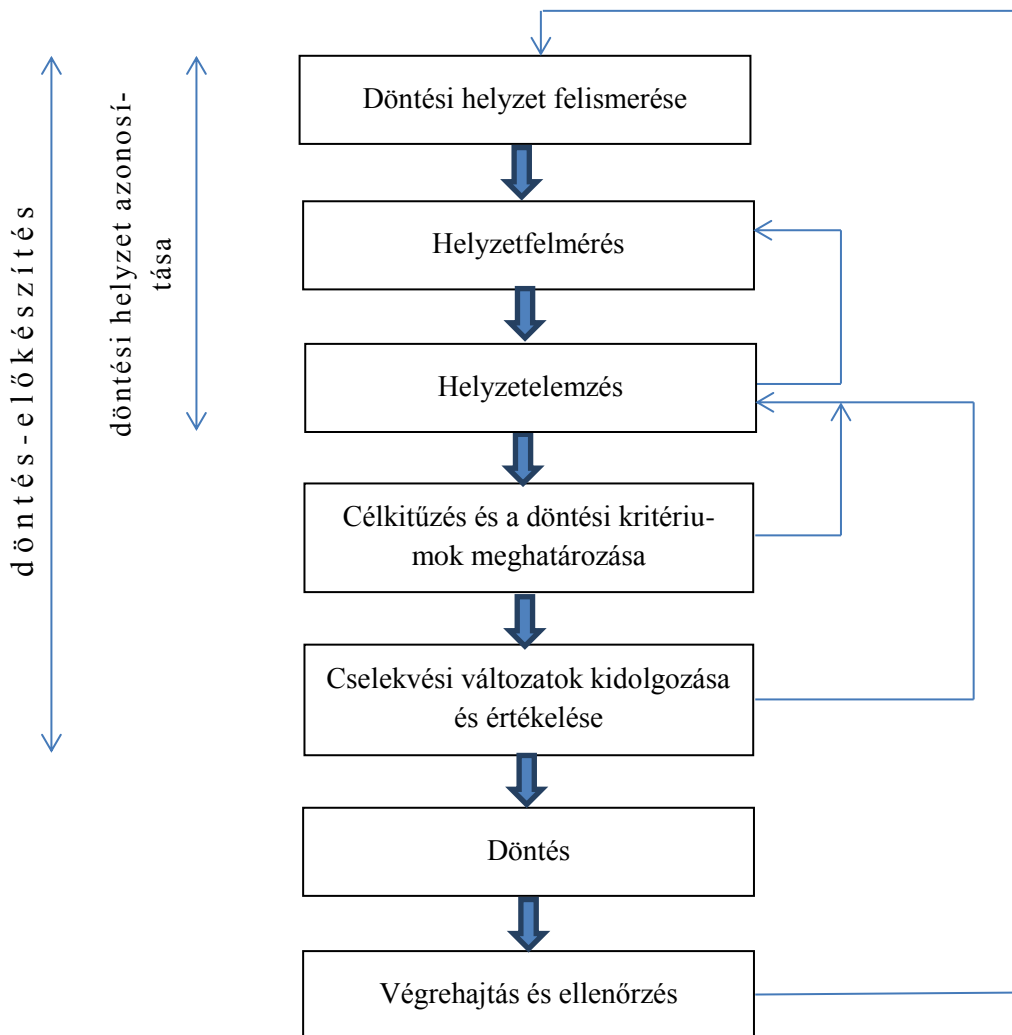
„A döntést felfoghatjuk úgy, mint átalakítások transzformációk sorozatát. Meghatározott adatokból (a memóriából átvett induló információkból és a döntés-előkészítés közben beérkező közlésekből) kiszámítanak egy eredményt: a döntést. A döntés a döntést hozó szervezet közlés outputjának egy speciális része. Funkciója: más egységek folyamatának a szabályozása” [28]

„A döntések tárgyalásában a döntést és a vezetést azonos fogalomként tekintjük és nem csupán a változatok közötti választás befejező aktusáról beszélünk, hanem inkább a döntéshozatal egész folyamatáról.

A döntéshozatal a következő három fázisból áll:

1. döntési alkalom feltárása,
2. a lehetséges cselekvési változatok felkutatása,
3. választás a cselekvési változatok között”, [29]

Ezekből a definíciókból láthatóan a döntés nem más, mint egy cselekvés, ami által az addigra már feltárt és előkészített döntési lehetőségek, más néven cselekvési lehetőségek közül választunk. Ezáltal egy döntés sosem csak egy lépés, hanem mindig egy folyamat, amely mindig a probléma felismeréssel kezdődik, és a kiválasztott cselekvési alternatíva végrehajtásával folytatódik és ennek ellenőrzésével zárul. (4. ábra) A továbbiakban Simon által meghatározott definíciót használom. Bár minden lépésből van lehetőség a visszacsatolásra, az ábrán csak a gyakran előforduló visszacsatolások lettek jelölve.



4 ábra – Döntési folyamat [30] alapján

A következő fejezetekben bemutatom a Simon által megalkotott döntési folyamat lépéseit.

2.1.2. A döntési helyzet felismerése

De miről is szól a döntési helyzet felismerése? Egy ókori mondás szerint, „a megoldáshoz vezető út felét már megtetted, ha felismerted a problémát”. Ebben a helyzetben érzékeljük, hogy a célállapot eltér az észlelt tényállapottól. A szervezeti biztonságot fenyegető helyzetek észlelése elsődleges fontosságú a biztonsági rendszerek kialakítása, de még a szervezeti tagok tudatos és tudatalatti működését szabályozó biztonsági kultúra működése kapcsán is.

Milyen módokon lehet ezt megtenni? Enyedi szerint erre négy lehetőség van [26]:

1. Vannak úgynevezett *kényszerítően nyilvánvaló helyzetek*, amikor egyértelmű, hogy probléma van és dönteni kell. Ilyen lehet bármilyen természeti katasztrófa, vagy például egy vállalkozás számára például a fizetéseképtelenség vagy a csődhelyzet. Ekkor már mindenképpen dönteni kell, mert egyértelműen látható a számunkra, hogy probléma van.
2. Lehetőség van arra, hogy úgynevezett *figyelmeztető jeleket, jelrendszereket/jelzőrendszereket* használjon a döntéshozó, ilyen lehet például egy vállalkozás számára a számvitel, amelynek segítségével felismerheti a problémát, például az elmaradt bevételek, nem kalkulált veszteségek megjelenésével.
3. Természetesen ezeket a jeleket *külső forrásból* is meg lehet szerezni, ilyen lehet valaki olyan kívülálló, külső szakértő, aki úgy képes rávilágítani a hibákra, hogy őt nem vakítja el az üzemi vakság. Tehát akár egy könyvvizsgáló, vagy akár csak egy könyvelő alkalmazása.
4. Talán a leghatékonyabb módszer, ha elébe megyünk a problémának, és mi magunk keressük azokat. Ez a módszer a *problémakutatás*. Itt a vállalkozó folyamatosan figyelemmel kíséri a folyamatokat. Természetesen egyetemi életünk során is vannak hasonlóan használható helyzetek, például amikor egy hallgató számára, ha nem sikerül egy adott vizsga, az már egy *kényszerítően nyilvánvaló helyzetnek* kell lennie, arra, hogy valami hibádzik a tanulásban, és változtatnia kell. De érdekesebb már akkor megpróbálnia változtatni, amikor ez még csak mint *figyelmeztető jel* jelenik meg, vagyis példá-

ul nem sikerül valamilyen dolgozat, vagy egyszerűen csak nem érti az elhangzott anyagot, esetleg nem tudja megoldani az órai feladatokat sem. Használhat *külső forrást*, akár egy csoport társat, akár magát a tanárt, aki kérdés módszerével segíti a hallgatót a felismerésben. Persze ő maga is elébe mehet a problémának és megpróbálhat úgy készülni a vizsgára, hogy megnézi, hogy tisztában van-e minden területtel, amit tudnia kellene.

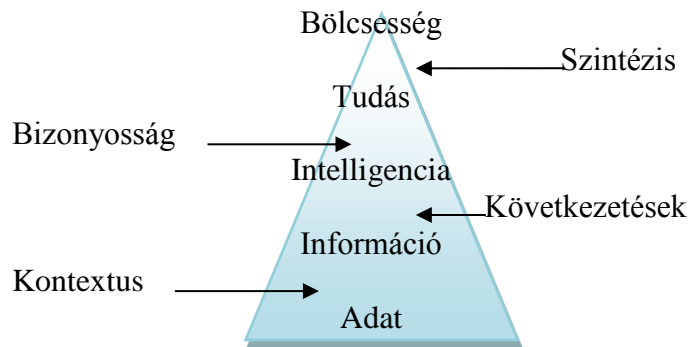
Ha felismertük, hogy probléma van, utána következik az információgyűjtés, valamint a döntés előkészítés szakasza. Jól dönteni csak akkor lehet, ha lehető a legtöbb információval rendelkezünk a problémáról, illetve minden lehetséges cselekvési változattal tisztában vagyunk. Döntéseméleti értelemben csak ebben az esetben tudunk optimális döntést hozni.

2.1.3. Döntés-előkészítés

A döntési folyamat (5. ábra) első szakaszát *döntés-előkészítésnek* nevezzük, ahol a probléma felismerése után jön az információk összegyűjtése és értékelése. A probléma felismerése után a döntéshozó már tisztában van azzal, hogy a rendszer nem működik megfelelően. A döntéshozó számára fontos minden információ, hogy tudja, a probléma hol jelentkezik, illetve hol kell a rendszerbe beavatkozni. A *helyzetfelmérés* szakaszában a döntéshozó csak összegyűjti az információkat, míg *elemzés* esetén az információk a helyükre is kerülnek. Az információkat lehet belső, illetve külső forrásból is megszerezni (primer), továbbá van lehetőség olyan, már meglévő információk felhasználására is, amelyeket már egy régebbi kutatásban megszereztünk (szekunder információk). Az információ azonban mindig összefügg a bizonytalansággal. Hiszen megfelelő információk megszerzésével mindig lehet csökkenteni a bizonytalanságot, és ez által a kockázatot is.

Adatból akkor lesz információ, ha valamilyen kontextusba tudjuk helyezni, ha egy adott információból következtetéseket tudunk levonni, akkor az számunka már mint intelligencia jelenik meg. Az intelligencia lényege, hogy képesek vagyunk egy váratlanul felmerülő probléma esetén nem tanultmegoldásokat is alkalmazni. Úgy lesz ebből tudás, hogy számunkra egy adott információ, mint bizonyosság jelenik meg, és ha sikeresen össze tudjuk kapcsolni ezeket az információkat, akkor ebből a tudásból bölcsesség lesz.

Az információ hierarchia modellje a szintek közötti kapcsolódásokkal látható a 5. ábrán.
[31]



5 ábra - Az információ hierarchiamodellje [31] alapján

Teljes informáltság (determinisztikus eset) az életben szinte sosem fordul elő, amelynek több oka lehet, például az ember nem képes minden információt akár idő, akár tárolási kapacitás miatt összegyűjteni és elemezni. Másik ok lehet, hogy lemondunk olyan információkról, amelyeket nehezebb elérni, vagy amelyeknek túl költséges lenne a beszerzése.

Az információk elemzése után a döntéshozó következő feladata (a döntési folyamat következő szakasza) a célok meghatározása és a cselekvési lehetőségek kialakítása, amelyekből majd a döntéshozónak döntenie kell. Probléma mindig akkor van, ha a célállapot és a jelenlegi észlelt tényállapot között nem csak egy, hanem több út van. Ezeknek az utaknak, cselekvési változatoknak a kialakítása a következő fontos lépés.

Kockázat

A kockázat fogalmát sokféleképpen szokták megközelíteni. [32] alapján a következő nézőpontokból érdemes vizsgálni:

- technikai;
- közgazdasági;
- pszichológiai;
- szociológiai és antropológiai;

Minden nézőpont a következő elemeket tartalmazza:

- a döntések káros vagy nem kívánatos következményei,
- ezeknek az események bekövetkezésének a valószínűsége,

- és a döntéseknek a valóságról alkotott feltételezése;

A különböző közelítésmódok között az a különbség, hogyan definiálják a negatív következményeket, illetve a bizonytalanságot.

A döntéshozó nézőpontjából vizsgálva a bizonytalanság kezelése, - bizonytalanság kerülés, bizonytalanságkerülés, - szorosan összefügg a kockázatvállalási hajlandósággal. Az a releváns, hogy a döntéshozó – jelen esetben akár a hallgató – mennyire tekinthető kockázatvállalónak, vagy inkább kockázat kerülő magatartást folytat-e.

A folyamatot befolyásoló tényezők a következők [26]:

- a döntéshozó motivációja;
- intelligenciája és más személyes jellemvonások;
- várakozások;
- elérhető információ;
- rendelkezésre álló idő;
- adott döntés komplexitása;

Ezek után, ha hatás szempontjából vizsgáljuk a kockázatot, akkor azt mondhatjuk, hogy három lehetséges kimenetele lehet: pozitív, ami egyben lehetőség, negatív, ami veszély, vagy ha a kockázat hatása megállapíthatatlan, akkor bizonytalanságról beszélünk.

Így megállapíthatjuk azt, hogy a gazdasági életben van egy, a kockázathoz szorosan kapcsolódó fogalom, ez a bizonytalanság. Ennek a két fogalomnak a pontos definíciója és elhatárolása nem annyira egyszerű, de a szakirodalom [33] [34] [35] [36] [37] különbséget tesz a kockázat és a bizonytalanság között, azáltal hogy mennyire ismerjük a következmények bekövetkezési valószínűségét. A kockázat alatt az olyan jelenségeket szokás általában érteni, melyekre alkalmazhatóak a valószínűség-számítás és a statisztika eszközei. Ezzel szemben a bizonytalanság esetében nincs információnk a valószínűségekről, tehát a valószínűség-számítás és a statisztika szokásos eszközeivel a gazdasági életben előforduló bizonytalanság nem kezelhető. Ennek az az egyszerű oka, hogy sem az adatok mennyisége, sem tartalma nem megfelelő. „Közgazdasági értelemben ez azt jelenti, hogy a gazdasági döntések bizonytalanság, és nem kockázat mellett születnek.” [38]

A kockázat és a bizonytalanság fogalmával elég régóta foglalkoznak kutatók, már a régi görögök is felismerték és elismerték létezését, de az 1800as évekig nem igazán tekintet-

ték fontos építőkőnek a tudomány területén. Leginkább a második világháború után kezdetek el mélyebben foglalkozni ennek az értelmezésével, napjainkban pedig gyakorlatilag az élet minden területén nagyon fontos a bizonytalanság és az ehhez ezer szállal kötődő kockázatkezelés. A döntések meghozatalakor a rendelkezésre álló információk alapján bizonyos mértékű bizonytalanság van jelen.

Az információkkal összefüggő bizonytalanság az alábbi fokozatokat érheti el a „bizonytalansági skálán” [39]

- nem strukturált bizonytalanság: ekkor a rendszer állapotai ismeretlenek minden, a jelenlegitől eltérő időpontban;
- strukturált bizonytalanság esetén a rendszer állapotai ismertek, de nem tudjuk előre megmondani a rendszer állapotát;
- véletlen esetén a rendszernek nemcsak az állapotai ismertek, hanem a valószínűségi törvényszerűségei is, amelyek a jelenlegitől eltérő időpontban jellemzőek, de az esemény konkrét kimenetelét nem ismerjük;
- bizonyosság esetén ismertek az állapotok, és azok bármely időpontra leírhatóak;

Bizonytalanság

„A bizonytalanság a döntéshozó szubjektív viszonya a környezetéhez, e külső világ állapotához, objektíve és állandóan létező attribútuma, velejárója az emberi létnek” [40]. A *bizonytalanság* tehát egy olyan állapot, amely a döntéshozó és annak környezete között alakul ki. A bizonytalanság mértéke a döntési folyamat során csökken, ahogy a döntéshozó egyre több információ birtokába jut, és egyre jobban megismeri az adott szituációt. A bizonytalanság soha nem szüntethető meg teljesen, csupán arra lehet törekedni, hogy a megfelelő információ-feldolgozással a minimálisra csökkenjen a jelenség.

Chikán, a bizonytalanság fogalmából kiindulva a döntési szituációkat a következő képen csoportosította [40]:

- determinisztikus eset;
- ismert (objektív vagy statisztikai) valószínűségek esete;
- meghatározható szubjektív valószínűségek esete;
- ismeretlen valószínűségek esete (Játék a „természettel”);
- értelmes ellenfél esete (Játékelmélet);

A determinisztikus eset

A döntéshozó ebben a helyzetben, a rendszer jelenlegi helyzetére és a jövőbeli állapotaira vonatkozó összes információval rendelkezik. Ilyen helyzet a valóságban nem fordul elő, mivel minden döntés velejárója a bizonytalanság-

A determinisztikus eset lényegi jegyei a következők:

- A döntés kritériuma a célfüggvény helyettesítési értéke.
- A döntéshozó ebben a szituációban a helyzetfelmérés és elemzés után az összes információval rendelkezik, amely a rendszer jelenlegi helyzetére és a jövőben szóba jöhető állapotaira vonatkozik.
- Minden döntés objektív velejárója a bizonytalanság.
- Ismert az indikátorvektor konkrét értéke.
- Ismert a célfüggvény helyettesítési értéke.
- Valóságban ilyen nem fordul elő.
- Minden információ ismert.

Példa: Ismerjük a terméket, a keresletet (S1, fog bekövetkezi), így a döntés kritériuma a célfüggvény helyettesítési értéke. Vagyis a választás: $C > D > B > A$

	S1	S2	S3
A	5	5	5
B	12	1	7
C	20	8	2
D	13	4	4

6 táblázat – példa a determinisztikus esetre

A klasszikus közgazdasági döntéseméleti irányzat megfelel ennek az esetnek, mert az csak a számszerűsíthető értékekkel foglalkozik. Taylora – klasszikus közgazdasági modell előfeltevései a következők: [7]

- Egy adott cselekvési változat választása esetében valamennyi eredmény valószínűsége egy;
- A cselekvési változatokra és az eredményekre vonatkozó információk teljesek;
- Az eredmények egy érték (hasznosság) skálán rendezhetőek;
- A döntéshozó azt a cselekvési változatot választja, amelyiknek a legnagyobb az értéke;

Ismert (objektív vagy statisztikai) valószínűségek esete

Ebben az esetben bár a döntéshozó nem rendelkezik teljes informáltsággal, és a cselekvési lehetőségeinek kimeneteit a véletlen befolyásolja, ennek a törvényszerűségei a döntéshozó számára ismertek. Fontos, hogy megbízható adatbázissal kell rendelkezni, hogy az eloszlást statisztikai úton meg lehessen határozni. Más néven ezt az esetet kockázati szituációnak is nevezik. [25]

Az ilyen döntési helyzet lényegi jellemzői a következők:

- A teljes bizonyosságtól eltérő esetekben a döntési szituációk sztochasztikusak.
- A döntéshozó ekkor ugyan nincs teljes információ birtokában, a cselekvési változatok kimeneteit a véletlen befolyásolja, de a véletlen törvényszerűségei a döntéshozó számára ismertek.
- Több kimenet is lehetséges, a szituáció kimenetelét a véletlen befolyásolja.
- Megbízható adatbázis kell hozzá; a valószínűségek matematikai – statisztikai úton határozhatók meg.
- A döntés kritériuma a célfüggvény várható értéke.

Példa: Ismerjük a terméket, de nem ismerjük a keresletet (csak a különböző esetek bekövetkeztének a valószínűségét), így a döntés kritériuma nem a célfüggvény helyettesítési értéke, hanem a célfüggvény várható értéke. (A kereslet adatait, a hozzájuk tartozó valószínűségekkel súlyozzuk.) Vagyis a választás: $C > D = B > A$

	S1	S2	S3	
A	5	5	5	5
B	12	1	7	7
C	20	8	2	9
D	13	4	4	7
	30%	20%	50%	

7 táblázat – példa az ismert valószínűségek esetére

Ennek az esetnek a modern közgazdasági irányzat felel meg, aminek az előfeltevései a következők [7]:

- Egy adott cselekvési változat választása esetében az eredmények bekövetkezésének valószínűsége egynél kisebb is lehet.
- A cselekvési változatok és eredmények bekövetkezési valószínűsége ismert
- A cselekvési változatok várhatóértékei egy hasznossági skálán rendezhetők
- A döntéshozó azt a változatot választja, amelyik a várható értéket maximalizálja

A két különböző közgazdasági irányzat makrogazdasági szinten működőképesnek bizonyult, makroszinten belül azonban, vállalati szinten már kevésbé érvényesek, egyén szintjén pedig nem érvényesek a gazdasági racionalitás elvei.

A hasznosság maximalizálásához bonyolult (matematikai) eljárásokat kell alkalmazni, elég ismerettel ehhez nem rendelkezik a döntéshozó, nem képes minden lehetséges cselekvési változat azonosítására. A legtöbb egyéni döntéshozónak nincs pontosan megfogalmazott, kitűzött célja.

Léteznek olyan nem számszerűsíthető értékek, amelyek a piacon a hasznosság maximalizálásában eltéréseket okozhatnak, és amelyek az embereket döntéseiknél „irracionális” viselkedésre készítetik.

Meghatározható szubjektív valószínűségek esete

Az életben előforduló helyzetek általában sztochasztikusak, de ritkán ismétlődőek azonos feltételek mellett, hogy a valószínűségeket, mint gyakoriság meg tudjuk határozni. Viszont hasonló helyzetekből a döntéshozó képes a valószínűségeket megbecsülni. Ilyen esetekben a döntéshozó információja még kevesebb: nem ismeri tökéletesen sem a keresletet, sem a különböző keresletekhez tartozó valószínűségeket. Ebben az esetben a döntéshozó saját maga határozza meg a keresletekhez tartozó valószínűségeket, például egy kísérleti árusítás alapján. Itt a valószínűségek egyértelműen szubjektívek, vagyis a döntéshozótól függenek. Ez az eset az egyik leggyakoribb az összes helyzet közül.

Tehát a lényegi jegyei ennek az esetnek a következők:

- Az információ még kevesebb.
- Döntéshozó tapasztalatai határozzák meg.
- Ez a leggyakoribb eset.
- Több kimenet is lehetséges, véletlen befolyásolja.
- A döntési kritérium egyértelmű, a döntéshozó a várhatóérték alapján választ.
- A kritérium nem objektív.

A bizonytalanság kezelésének módja tehát csupán abban különbözik a korábbi esettől, hogy a valószínűségeknek más a forrása.

Ehhez az esethez az adminisztratív modell áll a legközelebb, hiszen az adminisztratív modell a pszichológiából veszi át az eredmények bekövetkezési valószínűségének és

értékének a megközelítését, felfogását. Kutatások alapján azt állítja, hogy a valószínűségeknek nem a tényleges, hanem az észlelt nagysága a lényeges. [7]

Gyakorlati tapasztalatok:

- Cselekvési változatok az esetek többségében nem állnak rendelkezésre, a változatokat meg kell alkotni.
- Az információk erősen hiányosak, az információkat is keresni kell.
- Az információ pontatlan nem egyértelmű és ez az eredmények várható értékének bizonytalanságához vezet.
- Az információ beszerzése költséges.

A közelítésmód kognitív jellegű, azt nézi, hogy mi van a döntéshozó tudatában, hogyan gondolkodik a probléma kezelése esetén.

A modell az információkkal összefüggő bizonytalanságot a következőképpen kezeli:

- A számszerűsíthető értékekre irányítja a figyelmet.
- Hosszabb helyett inkább a rövidebb időtávon belül bekövetkező eredményekkel foglalkozik.

Információszerzés és értékelés költségeit a döntéshozó igyekszik csökkenteni:

- A döntéseknél egyszerű, könnyen elérhető információkkal és egyszerű számításokkal dolgozik.
- Nem optimalizálásra törekszik, hanem kielégítő döntéseket hoz.

Ismeretlen valószínűségek esete (Játék a „természettel”)

Ebben az esetben azt feltételezzük, hogy a cselekvési lehetőségek eredményének valószínűsége ismeretlen, és nem is lehet meghatározni. Ez az eset is nagyon ritkán fordul elő a valóságban, mivel itt a döntéshozóval szemben egy ismeretlen erő (természet) vagy egy másik (vagy több) döntéshozó áll vele szemben. Így ez a helyzet is felfogható már konfliktus helyzetként. Ilyen körülmények között nem adható meg egyértelmű döntési szabály, a döntéshozónak a döntési kritérium tekintetében is választania kell.

Néhány ilyen fontosabb döntési kritérium:

- Wald féle kritérium (minimax)
- Maximax kritérium
- Hurwitz kritérium ($0 \leq \lambda \leq 1$)
- Savage kritérium (minimális megbánás elve)
- Bayes – Laplace

Példa:

	S1	S2	S3	MiMa	MaMa	$\lambda=0,6$	SAVAGE	B-L
A	5	5	5	5	5	5	15	15
B	12	1	7	1	12	7,6	8	20
C	20	8	2	2	20	12,8	5	30
D	13	4	4	4	13	9,4	7	21

8 táblázat – példa az ismeretlen valószínűségek esetére

Minden helyzet esetén azt kell választani, ami a legjobb eredményt adja. (Kivétel a Savage, mert ott megbánásokat számolunk).

A normatív döntéelméleti irányzat egyik kvantitatív módszere a parametrikus döntés. Az ilyen döntések esetén a döntéshozó a természettel szemben igyekszik optimális döntést hozni (az egyes stratégiák kimenetei és azok bekövetkeztenek a valószínűségei a döntéshozótól függetlenek). A természet, mint ellenfél befolyásolja a hozott döntés kimenetelét, de nem tudatos a befolyásolás.

Értelmes ellenfél esete (Játékelmélet)

A döntési szituációk bizonytalanságtartalmuk szerinti osztályozásának utolsó esete a játékelmélet, Ennek azért kell külön fejezetet szentelnem ebben a dolgozatban, hiszen a helyzet alapjaiban különbözik a korábbiaktól abban, hogy az ellenfél is egy racionálisan cselekvő döntéshozó. Egy gondolkodó ellenfél tudatosan játszik a döntéshozó ellen.

Az ilyen típusú helyzeteket stratégiai játékoknak is nevezzük. Döntéshozó magatartását a stratégia határozza meg. A játékban résztvevők (elvbén) ismerik valamennyi döntéshozó stratégiáinak halmazát és a hozzátartozó kifizetéseket is. Stratégiákat egymástól függetlenül választják, azaz egyik fél sem tudja előre, hogy mit választ a másik. A felek által választott stratégiákban megjelenik a biztonságra való törekvés.

A normatív döntéseméleti irányzat egyik kvantitatív módszere a stratégiai döntés. Ebben az esetben a döntési helyzet folyamatosan változhat, a döntéshozó olyan gondolkodó, értelmes ellenféllel, áll szemben, aki saját érdekei szerint alakítja a döntési helyzet körülményeit. Az ellenfél tudatosan magáért és emiatt ellenünk is játszik. A konfliktushelyzetben történő döntéshozatal módszere a játékelmélet, ahol stratégiai játékok kimenetelére a döntéshozóknak a fennálló játékszabályok keretei között van közös befolyásuk. (Lásd bővebben: [2.3 fejezet](#))

Információ a döntési folyamatban

Az információ a döntésemélet meghatározó eleme, a döntési folyamatot végigkísérő tényező. Az előbb tárgyalt bizonytalansággal szoros kapcsolatban áll, hiszen az információ csökkenti a bizonytalansági állapotot. Nem minden esetben rendelkeznek azonban a döntéshozó a szükséges információkkal. Az információhiány alapvetően szubjektív és objektív okokra vezethető vissza. Objektív tényezőnek tekintjük azt, hogy a természet és az ember maga korlátokat állít fel a döntések során, lehetetlenné téve szinte a teljes informáltság állapotát. Szubjektív tényező pedig az, hogy sok esetben a döntéshozó lemond az információkról, általában az erőforrások - idő és pénz- szűkössége miatt.

Manapság megfigyelhető, hogy az információk szinte végeláthatatlan mennyisége veszik körül bennünket a mindennapokban, ezáltal sokszor megnehezítve a döntési folyamatot, hiszen ki kell szűrni az éppen releváns információkat a hatalmas adathalmazból. Ami a mennyiségi oldalnál még fontosabb az a minőség, azaz egy-egy információ értéke. Ezalatt azt értjük, hogy adott információ mennyire képes csökkenteni a döntési folyamat során felmerülő bizonytalanságot, és az azzal járó költségek mértékét. A modern társadalmakra jellemző az információk túlkínálata. A gazdaságban a „rossz” döntések számottevő része hibás vagy valótlan információknak köszönhető, amelyek miatt kialakult a versenyhelyzet az információszerzés és - szolgáltatás területén is. (Enyedi, 1997)

Információfeldolgozás korlátai

A teljes informáltság, mint olyan általában nem, vagy csak nehezen érhető el, ezt az előző fejezetekben már beláttuk. A kérdés az, hogy ennek milyen okai lehetnek? Az életünk sokkal bonyolultabb, minthogy képesek legyünk tökéletesen szimulálni, sosem lehetséges tökéletes modell létrehozása, nem szedhetünk össze minden információt, mert nincs elég időnk rá, illetve nagyon sok információ nem hozzáférhető, mert nincs rá

elég anyagi forrásunk. Mi több, még az általunk összeszedhető információk is olyan méretűek, hogy mi ezeket nem vagyunk képesek felfogni, és kezelni. Akár túl kevés rendelkezésre álló információ miatt, akár a feldolgozhatatlanul sok miatt történik, a döntési helyzetek velejárója az információhiány, és a vele járó bizonytalanság és kockázat.

Az Internet megjelenésével korlátlan mennyiségű információ eljuthat hozzánk, de ha napi 24 órát ülünk a számítógép előtt, akkor sem vagyunk képesek átnézni még a milliomod részét sem a felkerülő anyagoknak, nem hogy értelmezni.

Nézzük meg, hogy milyen korlátai vannak az információ kezelésének ezeken kívül. Az ember egyszerre max. 7 ± 2 egységnyi különböző információt képes felfogni. Ezt a számot hívják Miller-féle bűvös számnak [41]. A rövidtávú memóriánkból képesek vagyunk ezeket az információkat átpakolni, és elraktározni a hosszú távú memóriában. Ahhoz, hogy erre képesek legyünk, vagy egy már ott lévő információhoz kell kapcsolnunk, vagy létre kell hoznunk egy újabb bejegyzést (sémát).

„Bizonyított, hogy akinek jobb mentális modelljei, sémái vannak az éppen vizsgált dolgokról, az jobb következtetéseket képes levonni” [25]. Ez alapján az egyik nagy probléma az információfeldolgozás esetén a sémák hiánya, vagy képességbeli hiány adott sémák létrehozására.

Kognitív korlátok

Milyen egyéb problémái lehetnek még az információfeldolgozásnak? Ilyen problémák lehetnek az ember kognitív korlátai az információ befogadásakor, illetve az információk felhasználásakor.

March [42] szerint az információ befogadásakor a következő kognitív korlátok jelennek meg:

- A figyelem problémája – Az előző részben már említett korlátok, vagyis az ember nem képes minden információt összeszedni, mert a megfigyelésre fordított idő, képességek korlátozottak.
- A memória problémája – Bár rengeteg információ, inger ér minket, ezek nagy részét nem vagyunk képesek tárolni.
- A felfogás problémája – A megszerzett információkat még értelmezni is kell, és ez sokszor akadályokba ütközik, hibásan kapcsolunk össze információkat,

amennyiben pedig ellentmond az eddigi információinknak, akkor lehet, hogy nem is vesszük figyelembe őket.

- Kommunikációs problémák – nem minden ember képes az információk átadására, más és más sémákat használnak, és ezért másképpen kellene az információkat kommunikálni is.

Milyen kognitív korlátok létezhetnek az információk felhasználásakor? [25]

- A megszerzett információk között valamilyen rendszert, struktúrát keresünk, ami sokszor nem is létezik, vagy teljesen más, mint amit mi találunk.
- A különböző emberek keresési algoritmusai különbözőek, sőt még egy adott ember is váltogathatja a keresési módokat különböző időpontokban.
- Az emberek nehezen tolerálják a bizonytalanságokat, ezért sokszor rosszul kezelik a valószínűségeket is.
- Gyakori probléma, hogy rengeteg információval rendelkezünk, csak az az információ hiányzik, hogy mit is szeretnénk csinálni, mi a célunk.

A problémamegoldással és döntéshozattal foglalkozó irodalom megalkotta a *heurisztika (torzítási mechanizmus)* fogalmát, ami egy „hüvelykujj szabály bizonyos értékek megtervezéséhez vagy bizonyos problémák megoldásához” [42]. Amikor egy döntés előtt állunk, az információ sokasága miatt, gyakran az eredmények valószínűségének becslésére egyszerűsítéseket végzünk saját helyzetünk könnyítése végett. Így teszik az emberek saját maguk számára kezelhetővé a bonyolult szituációkat. Sok esetben a heurisztikák – egyszerűsítő jellegük miatt – segítséget nyújtanak a gyors és hatékony döntéshozatal során a komplex problémák megoldásában, viszont előfordulhat az is, hogy rossz irányba sodorják a döntéshozót. Jellemzően bizonytalan szituációkban kerülnek alkalmazásra a heurisztikák, így ha nem megfelelően alkalmazza azokat a döntéshozó, akkor problémák jelentkezhetnek.

Tversky és Kahneman [43] a következő torzítási mechanizmusokat fogalmazta meg:

- reprezentativitás: az alapgyakoriság mértékét figyelmen kívül hagyva a valószínűségeket aszerint értékeli a döntéshozó, hogy *A* milyen mértékben hasonlít *B*-re, azaz a döntéshozó a mintanagysággal szemben érzéketlen,
- hozzáférhetőség: az esemény bekövetkezésének nagyságát aszerint ítéli meg a döntéshozó, hogy mennyire könnyen tud példát felidézni arra vonatkozóan; azok

az események, amelyek könnyen jutnak eszébe a becslések során nagyobb gyakorisággal szerepelnek;

- kiigazítás és lehorgonyzás: a valószínűséget a döntéshozó egy kezdeti értékhez igazítja és ezen pont szerint irányítja a döntési folyamatot.

Ezt a három tényezőt tovább elemezve, Max Bazerman [44] a következő heurisztikákat írja le:

- könnyű felidézhetőség,
- elérhetőség,
- látszólagos korreláció,
- az előzetes valószínűségek mellőzése,
- a mintanagyságra való érzéketlenség,
- a véletlen félreértelmezése,
- az átlaghoz való visszatérés,
- az egybeesésből való téves következtetés,
- elégtelen kiigazítás,
- konjunktív és diszjunktív események,
- túl nagy önbizalom,
- a megerősítési csapda,
- az utólagos előrelátás és tudás átka.

Ezen heurisztikák alkalmazása veszélyeket tartogat a döntéshozók számára, amellyel tisztában kell lenniük annak érdekében, hogy meg tudják hozni a megfelelő kielégítő döntéseket. [45]

Cselekvési lehetőségek kialakítása

Miután összegyűjtöttük a lehetséges információkat, és azokat elemeztük is, ki kell alakítanunk a cselekvési változatokat.

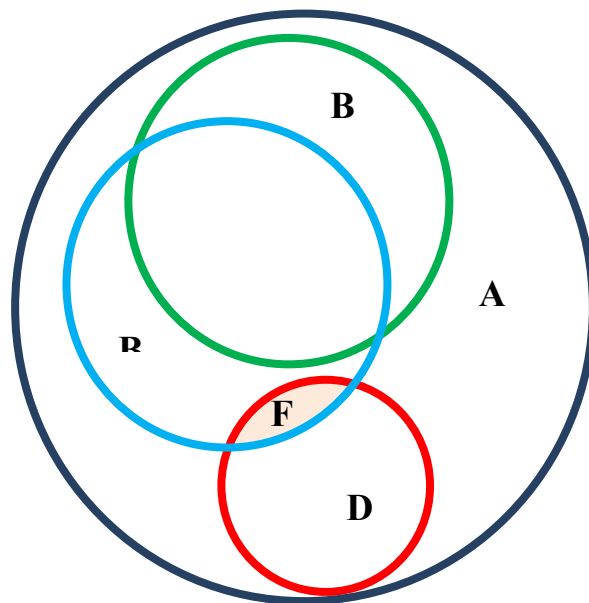
Enyedi [26] szerint a cselekvési változatok kialakításának több lépcsője van. Első a *lehetséges cselekvési változatok* (9. ábra „A” halmaz) kialakítása, ezek azok a lehetőségek, amelyeket a döntés előkészítés folyamán figyelembe lehet venni. Ezek között sok olyan is van, ami több okból kizárható, például nincs elég forrás rá.

A következő csoport a *végrehajtható cselekvési változatok halmaza*. (9. ábra „B” halmaz) Ez részhalmaza az előző csoportnak, de ez előzőből minden olyan változat kiesik, amelyet a döntéshozó nem tud elfogadni valamilyen okból kifolyólag, pl., anyagi, műszaki vagy fizikai korlátok miatt.

A harmadik csoport a *végrehajthatónak ítélt cselekvési változatok halmaza* (9. ábra „B” halmaz), amely olyan szintű szűkítése az eredeti halmaznak, amelyből a döntéshozó saját maga zárja ki számára nem végrehajthatónak tartott cselekvési változatokat. A végrehajtható és az annak ítélt cselekvési változatok között az a különbség, hogy a végrehajthatót a teljes informáltság esetén lehet kiválasztani és objektív halmaz, míg a végrehajthatónak ítélt halmaz a döntéshozó szubjektív véleményét tükrözi, ezáltal az ő általa észlelt világban lévő végrehajtható cselekvési változatokat tartalmazza.

A következő az *elfogadható cselekvési változatok halmaza* (9. ábra „D” halmaz), amelybe azok a cselekvési változatok tartoznak, amelyek megfelelnek a döntéshozó elvárásainak. Az utolsó halmaz, amelyből majd a döntéshozó választani fog a *számba jövő cselekvési változatok halmaza* (9. ábra „F” halmaz), ez a végrehajthatónak ítélt és a döntéshozó számára elfogadható cselekvési változatok metszete.

Természetesen - ha a döntéshozó számára ezek a cselekvési változatok rendelkezésre állnak -, akkor ezeket még értékelnie kell, majd ezek közül választania kell, vagyis döntést kell hoznia.



9 ábra – a különböző cselekvési lehetőségek halmaza, [26] alapján

2.1.4. A döntéshozatal

Miután meghatároztuk a számunkra elfogadható és megvalósítható cselekvési lehetőségeket, utána ki kell választani a legjobb, vagy esetleg ha nem tudjuk megtalálni az optimális megoldást, akkor a számunka még kielégítő döntést. Optimális döntés esetén, az összes megoldás közül általában egy, vagy ha több ugyanolyan létezik, akkor több megoldást lehet választani. Ezeknek a problémáknak a megoldásával foglalkozik az operációkutatás, ahol előfeltétel a teljes informáltság. Abban az esetben, ha ez - mint az előző fejezetben jól látható - nem, vagy túl sok befektetéssel (pénz, idő) lenne csak elérhető, akkor választjuk a kielégítő döntéseket. Kielégítő döntések esetén a döntéshozó meghatároz egy minimális igényszintet, amit mindképpen el kell érni a kiválasztott megoldásnak, de ha talál számára elfogadható megoldást, akkor már nem keres tovább, még akkor sem, ha esetleg lenne jobb, számára hasznosabb megoldása a problémának.

A döntések csoportosítása

A szakirodalomban többen is foglalkoztak azzal, hogy lehetne csoportosítani a döntéseket.

Simon [46] szerint a döntéseket lehet osztályozni az alapján, hogy a döntéshozók hogyan hozzák meg a döntéseiket. Ennek alapján vannak ismétlődő, rutin jellegű döntések (programozott döntések), illetve szabálytalanul jelentkező problémák, rosszul strukturált döntési helyzetekre vonatkozó (nem programozott döntések). Egyetemi életben például a tárgyfelvétel vagy akár az egyetemre való jelentkezést tekinthetjük nem programozott, míg mondjuk az órára készülést, mint probléma megoldást programozott döntésnek.

Ehhez nagyon hasonlóan Kornai [28] is osztályozza a döntéseket azok szokványosságára és alapvetősége szempontjából is. Szokványos döntésnek nevezi a szabály szerint ismétlődő, kevés számú és egyszerű lépésekből álló algoritmussal, csekély információ rendelkezésre állása mellett is megoldható döntéseket. Ezzel szemben az alapvető döntések nem szabály szerint ismétlődnek. Információ igényük nagy, megoldásuk sok lépésből áll, amelyek bonyolultak is lehetnek.

Kornai [28] a döntéseket a szabályozási és a reálszféra azonosságára alapján is két osztályba sorolja. Azon döntéseket, melyeknél a szabályozási szféra saját reálszférája szá-

mára hozza a döntéseket intern döntéseknek nevezi, míg ha a reálszféra a szabályozási szférától eltérő, akkor extern döntésekről beszélünk. Az extern döntésnél a szabályozási egység a „feladó”, míg a reálszféra a „címzett”.

Forrester [47] a döntéseket a döntéshozó szándékától, valamint a rendszer állapotától függően explicit és implicit döntési fajtákra osztja. Explicitnek azon döntéseket nevezi a vezetési folyamatban, melyeknél a döntéshozónak lehetősége van dönteni, s azt saját szándékából, tudatosan teszi. A hallgató azért választ, mert lehetősége van rá, ő dönt például, hogy milyen kurzusokat vesz fel vagy melyik tárgyból, mikor megy vizsgázni. Míg implicit döntési helyzet van akkor, amikor a rendszer állapota szükségessé teszi a döntést, vagyis a döntéshozónak döntenie kell. Ebben az esetben a hallgató már nem áll döntési lehetőség előtt, azért dönt, mert kívülről rákényszerítik. Érdekes a döntési folyamatban megismert probléma felismerést összehasonlítani ezzel a döntés tipológiával. Ha a két végletet vizsgáljuk, akkor látjuk, hogy a kényszerítően nyilvánvaló helyzetben való döntés megegyezik az implicit döntéssel, míg a problémakutatás és a hozzátartozó döntés megegyezik az explicit döntéssel.

A döntések ismétlődésük alapján is osztályozhatóak. Ha egy döntést egyszer kell csak meghozni, valamint az nem befolyásolja közvetlenül az utána következő döntéseket, akkor egy lépéses (egyfázisú) döntési problémáról beszélhetünk. Míg ha a döntés ismétlődik, több egymást követő lépés azonos problémára vonatkozik, akkor többlépéses vagy több fázisú döntésről beszélünk. Ha pedig a döntési helyzetek sorrendjét befolyásolják a megelőző döntések kimenetelei, akkor az egy dinamikus többlépéses probléma helyzet.

2.1.5. Végrehajtás és ellenőrzés

A döntéshozatali folyamat a választással még nem ér véget, fontos, hogy végre is kell hajtani, majd utána ezt ellenőrizni is kell. Az ellenőrzés nem arról szól, hogy megvizsgáljuk, hogy a döntésünk jó volt vagy esetleg rossz, hanem, hogy a döntést jól hajtottuk-e végre. A döntési folyamatban folyamatosan van visszacsatolásra lehetőség,

2.2. Az operációkutatás

Ahogy a döntési folyamatban látható volt az előző fejezetben, az információk szerepe nagyon fontos. Ha teljes informáltsággal rendelkezünk, akkor van lehetőség optimalizálásra. Az optimalizálás esetén az operációkutatás ad nekünk ebben segítséget.

Az élet különböző területein, különösen pedig a gazdasági tevékenységeknél olyan döntések meghozatalára törekednek, amelyek valamilyen szempontból optimálisak. (Optimálisnak nevezzük például azt a döntést, amely lehetővé teszi, hogy a kívánt célt, vagy célokat a legkisebb ráfordítással, vagy pedig a legnagyobb haszonnal érjük el.) Bonyolult problémák esetében optimális döntést azonban csak akkor tudnak megvalósítani, ha a tervvariánsok kidolgozásakor tudományos módszereket használnak.

Az operációkutatás az a tudomány, amely az optimális döntések előkészítésében matematikai módszereket használ fel. Kialakulását a II. világháborútól számíthatjuk, amikor harcászati jellegű problémák megoldására használták ezeket a módszereket. A világháborút követő időszakban aztán egyre inkább előtérbe került az operációkutatás gazdasági alkalmazása. Ma már az operációkutatást egyre inkább felhasználják mind a modern ipargazdaságban (készletgazdálkodási, sorbanállási problémák), mind pedig a konkrét vállalati gyakorlatban.

Meg kell jegyeznünk, hogy az operációkutatás csak a döntés-előkészítés eszköze, nem egyenlő magával a döntéssel, így az embert nem iktathatjuk ki a döntési folyamatból. A legtökéletesebb operációkutatási módszer sem elegendő egymagában valamely döntési probléma megoldására, hiszen a figyelembe vett tényezőkön kívül sok egyéb, általában nem számszerűsíthető tényező is hat a döntési folyamatra. Ha pedig már megtaláltuk a „legjobb döntési variánst”, annak gyakorlati megvalósítása során felléphetnek olyan problémák, amelyek miatt a várt eredmény nem realizálható.

Az operációkutatási módszerek egyik csoportjába tartoznak azok, amelyek széleskörűen alkalmazhatók különböző, egymástól lényegesen eltérő, de bizonyos követelményeknek eleget tevő döntési és ellenőrzési probléma típusok matematikai, közgazdasági, statisztikai leírására, modellezésére elemzésére (matematikai programozás, hálózati folyamatok, digitális szimuláció). A másik csoportot azok a módszerek alkotják, amelyek speciális problémákból fejlődtek ki, adott típusú problémák vizsgálatára alkalmasak (sorbanállási és készletgazdálkodási problémák).

Ahhoz, hogy optimális döntéseket tudjunk hozni, a következőkre van szükség:

- ismerni kell az összes cselekvési lehetőséget;
- ismerni kell a cselekvési változatok eredményét;
- és ismerni kell az eredmények preferencia sorrendjét is;

Az első kettő a már említett teljes informáltságot jelenti, míg az utolsó azt, hogy a döntéshozó csak egy cél alapján akar dönteni, vagy képes a céljait különböző kritérium súlyok segítségével egységesíteni. Ha ezek a rendelkezésünkre állnak, akkor képesek vagyunk optimális döntést hozni. Simon szerint az ember nem keres optimális megoldást, nem maximalizálja a hasznosságot, hanem mindig kielégítő döntést akar hozni. Ennek okai, hogy sosem rendelkezik minden információval, és nagyon ritkán rendelkezik egy céllal. [46]

Az operációkutatás lényeges jegyei:

- döntéselőkészítő eszköz;
- a döntéseket valamilyen szempont szerint lehet optimalizálni;
- a döntés-előkészítéshez matematikai módszer alkalmazható;

Operációkutatás segítségével tehát minden olyan probléma megoldható, amely matematikai modellben leírható és analitikailag optimalizálható.

Az élet nagy részében a döntéseink esetében nincs lehetőség optimalizálni, ezekben az esetekben leginkább kielégítő döntéseket hozunk. Az információ hiány együtt jár a bizonytalansággal.

Az operációkutatás ismertebb elméletei/problémái:

- Szimulációk;
- Lineáris programozás;
- Szállítási feladatok;
- Hozzárendelési feladatok;
- Sorbaállási feladatok;
- Hálótervezési feladatok;



10 ábra - Az operációkutatás és annak válogatott problémái

Ezek a feladatok gyakran gráfelméleti eszközökkel modellezhetők, és egy legrövidebb út megtalálásával oldhatók meg. A gyakorlatban a sorrendiség meghatározásának, és a szállítási feladatnak van nagyobb szerepe például a projektvezérlésben.

Szállítási feladatok

Ez egy speciális lineáris programozási feladat.

Legyen adott m telephely, amelyeken bizonyos fajta, tetszés szerint osztható termékből a_1, a_2, \dots, a_m mennyiséget tárolnak. Adott továbbá n felvevőhely, amelyek b_1, b_2, \dots, b_n mennyiséget igényelnek ebből a termékből.

Egységnyi terméknek az i -edik telephelyről a j -edik felvevőhelyre való szállítási költsége c_{ij} -vel legyen jelölve. Jelölje továbbá x_{ij} az i -edik telephelyről a j -edik felvevőhelyre szállítandó – egyelőre ismeretlen – mennyiséget. $i = 1, 2, \dots, m, j = 1, 2, \dots, n$

Feltesszük, hogy

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

azaz, hogy a tárolt áru összmennyisége megegyezik az igényelt áru összmennyiségével. Ez nem jelenti az általánosság megszorítását, hiszen vagy fiktív telephely, vagy fiktív felvevőhely beiktatásával mindig elérhető az előbbi egyenlőség. Olyan szállítást kell megvalósítanunk, amelynek során minden telephelyről minden árut elszállítanak, az

egy felvevőhelyek igényeit kielégítik, és ezt mind úgy teszik, hogy az össz-szállítási költség minimális.

A szállítási problémát matematikailag a következőképpen fogalmazhatjuk meg: Legyen adott egy

$$C = \begin{pmatrix} c_{11} & c_{1j} & c_{1n} \\ c_{i1} & c_{ij} & c_{in} \\ c_{m1} & c_{mj} & c_{mn} \end{pmatrix}$$

$m \times n$ -es mátrix, a költségmátrix. Legyenek továbbá adva az

$$a_1 \geq 0, \dots, a_m \geq 0 \text{ (tárolt mennyiségek)}$$

illetve

$$b_1 \geq 0, \dots, b_n \geq 0 \text{ (igényelt mennyiségek)}$$

melyekre a

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

teljesül. Meghatározandók az olyan x_{ij} mennyiségek, amelyek eleget tesznek a

$$\sum_{j=1}^n x_{ij} = a_i, i = 1, 2, \dots, m$$

$$\sum_{i=1}^m x_{ij} = b_j, j = 1, 2, \dots, n$$

$$x_{ij} \geq 0, i = 1, 2, \dots, m, j = 1, 2, \dots, n$$

feltételeknek, s amelyekkel a

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

költségfüggvény felveszi a minimumát.

A szállítási probléma egy minimum lineáris programozási feladat.

Hozzárendelési feladat

Ez egy speciális szállítási feladat, ahol az „elszállítandó mennyiség” mindenhol egyformán 1 egység. Vagyis létezik n alkalmazott, m feladat és $m = n$. A feladatokhoz tartozó költségmátrix:

$$C = \begin{pmatrix} c_{11} & c_{1j} & c_{1n} \\ c_{i1} & c_{ij} & c_{in} \\ c_{m1} & c_{mj} & c_{mn} \end{pmatrix}$$

Célunk a feladatok olyan kiosztása az alkalmazottaknak, hogy minden alkalmazott egy feladatot kapjon, és minden feladat el legyen látva, úgy hogy ez összességében a legkisebb költséggel teljesíthető legyen.

Egyszerű König-feladat (házasság feladat)

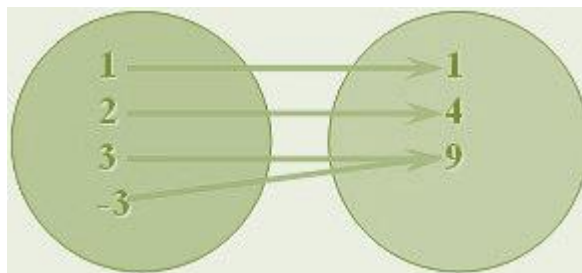
Legyenek adottak $I_1, I_2, \dots, I_i, \dots, I_m$ személyek és a $J_1, J_2, \dots, J_i, \dots, J_m$ munkák.

Azt, hogy melyik személy melyik munkához ért (melyik munkára van kvalifikálva) célszerűen az úgynevezett kvalifikációs mátrixba foglalhatjuk össze: az $m \times n$ méretű mátrix (i,j) –edik cellájában * álljon, ha az I_i személy a J_j munkát el tudja látni [48], [49]

A feladat annak eldöntése, hogy hozzárendelhető-e minden személy olyan munkához, amihez ért, feltéve, ha egy munkát csak egy munkás láthat el. (Kölcsönösen egyértelmű hozzárendelés megengedett csak.)

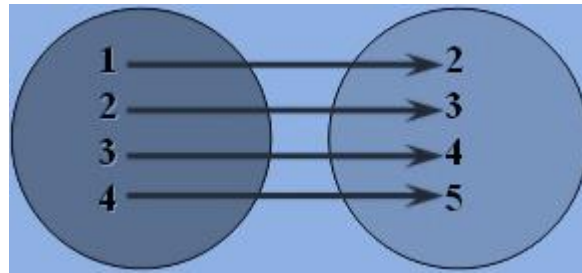
Hozzárendelések fajtái a következők lehetnek:

Egyértelmű hozzárendelés: olyan hozzárendelés, ahol az egyik halmaz eleméhez a másik halmazból csak egy elem tartozik.



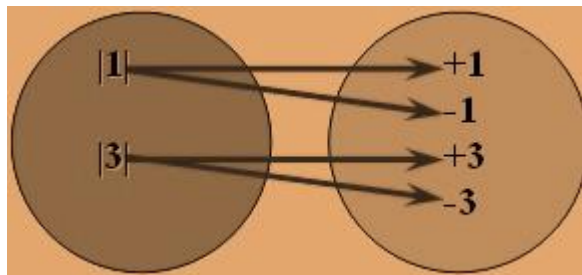
11 ábra – Az egyértelmű hozzárendelés

Kölcsönösen egyértelmű hozzárendelés: olyan egyértelmű hozzárendelés, ahol az egyik halmaz minden eleméhez hozzárendeljük a másik halmaz egy-egy (különböző) elemét.



12 ábra – A kölcsönösen egyértelmű hozzárendelés

Többértelmű hozzárendelés: olyan hozzárendelés, ahol az egyik halmaz eleméhez a másik halmazból több elem is tartozhat.



13 ábra – A többértelmű hozzárendelés

2.3. Játékelmélet

Az élet nagy részében a döntéseink esetében nincs lehetőség optimalizálni, ezekben az esetekben leginkább kielégítő döntéseket hozunk. Az információ hiány együtt jár a bizonytalansággal (lásd: [2.1.3. fejezet](#))

„A játékelmélet többszereplős döntési problémákat tanulmányoz, amelyek gyakran felmerülnek a közgazdaságtanban” [50].

Azt a tudományt, ami az információhiányos döntésekkel foglalkozik, ahol a döntéseink eredményét befolyásolják a többiek lehetséges választása, játékelméletnek nevezzük.

„A játékelmélet olyan helyzetekkel foglalkozik, amelyekben legalább két döntéshozó (például egyén, család, vállalat, intézmény, ország, stb.) próbálja saját hasznosságfüggvényét maximalizálni.” [51]

Mint ezekből a definíciókból is jól látszik, a játékelmélet egy olyan matematikai tudományterület, amit nagyrészt a közgazdaságtan hasznosít, és mindig legalább két különböző szereplője van, akik szeretnék az egyéni hasznosságukat maximalizálni.

A játékelmélet alapjait Neumann János fektette le, az által, hogy igazolta a minimax tételt. [52]

Neumann Jánosról közismert, hogy szeretett pókerezni, és kezdetektől fogva érdekelte, hogyan lehet a játékban a különböző stratégiákat alkalmazni. Ha az osztást nem lehet befolyásolni, akkor csak arra van lehetőség, hogy a stratégiáinkat miként alkalmazzuk. [53] A probléma felírásához matematikai nyelvezetet használt, de a fő érdeme sokkal inkább az elméletnek a játékokon messze túlmenő általánosítása volt. A minimax tételt bizonyító első publikációjában a már a napjainkban is használt normális alakot (normal form) használta a játékok leírására. [52] Oskar Morgensternnel közösen írt könyvében már a játékelmélet széleskörű felhasználhatóságát bizonyította. [54]

2.3.1. A játékelmélet típusai

Minden játék három részből áll: játékosokból, játékszabályokból és az eredmények értékeléséből. „A játék célja a minél kedvezőbb kifizetés elérése, s egy játékos ezt a célt szem előtt tartva, választja lépését vagy lépéseit – természetesen a játékszabályok figyelembevételével. Függetlenül attól, hogy hányszor vagy mikor kerül döntéshelyzetbe, stratégiának nevezzük azt a döntéssorozat tervet, amely a játék minden lehetséges döntéshelyzetére és az ebben tapasztalható minden lehetséges állapotára előír egy konkrét döntést. Bár a játékban előálló helyzetek függenek a játékosok lépéseitől, a játékos stratégiája nem, legfeljebb más-más válaszlépést ír elő. Így, ha a játékosok lépései függenek is egymástól, a stratégiáik nem. A játék kifizetését az egyes játékosok választott stratégiái döntenek el.” [55]

A kifizetés függvény az a függvény, amely a játékosok által választott cselekvések alapján meghatározza a játékosok kifizetését.

Alapvetően kétféle szempontból tekinthetünk egy játékra. Az egyikben, amit nevezhetünk úgy is, hogy „alulnézetből” nézzük a játékot, azonosítjuk magunkat az egyik játékosal és azt vizsgáljuk, hogy mi ezen játékos optimális viselkedése. A másik a „madár-

távlat” szemlélet. Ekkor mintegy felülről nézve azt vizsgáljuk, hogy a játékosok együttes cselekvéseként kialakuló helyzet milyen, elsősorban azt, hogy mennyire „stabil”.

A játékelméleti modelleket a következő módokon lehet osztályozni [56]

- A játékosok száma szerint (kettő, véges, végtelen).
- A játékosok számára rendelkezésre álló lehetőségek száma (véges, végtelen).
- A szembenállás foka (antagonisztikus, nem antagonisztikus).
- A megengedett kooperáció foka (kooperatív, nem kooperatív).
- A játék információs struktúrája (teljes, nem teljes, tökéletes, nem tökéletes).
- Az idő szerepe (statikus, dinamikus).
- A véletlen szerepe (determinisztikus, sztochasztikus).
- A matematikai megfogalmazás specialitása (normál forma, extenzív forma, karakterisztikus függvény forma).

2.3.2. Kooperatív játékelmélet

Természetesen a játékok nem csak olyanok lehetnek, ahol a játékosok csak, mint önálló individuumok léteznek, hanem össze is foghatnak egy kedvezőbb eredmény érdekében.

A nonkooperatív játékelmélet esetén olyan többszereplős játékokat, problémákat vizsgálunk, ahol a résztvevők különböző stratégiák alapján cselekedhetnek. Ezek a stratégiák egymástól eltérőek lehetnek, de mindig hatással vannak egymásra. Így ha egy játékos talál olyan stratégiát, ahol a többi játékos bárhogy is dönt, ő nem járhat rosszabbul, akkor ez a stratégia Nash egyensúly.

Kooperatív játékelmélet esetén a játékosok egymással szövetkezhetnek, vagyis koalíciókat alkothatnak. A koalíciók segítségével akár nagyobb hasznosságot is érhetnek el a játékosok, mintha egyedül maradnának, egyénileg cselekednének. Érdekes kérdés ebben az esetben, hogy miként is alakulnak meg ezek a koalíciók, illetve, hogy miként osztják el a koalíció által megszerzett hasznot.

A játékelmélet, mint látható a matematika és a közgazdaságtan egyik közös területe, amelyben több nagy kutató is kapott Nobel-díjat. Többek között Harsányi János 1984-ben, és Roth és Shapley 2012-ben. Shapley és társa a játékelmélet egy speciális részé-

ben, vagyis a párosítás elméletben elért eredményeik alapján kapták meg ezt az elismerést.

Kooperatív játékelmélet esetén is létezik olyan megoldási mód, mint a nem kooperatív esetben volt a Nash-egyensúly, ilyen lehet például a mag. Sokan foglalkoztak a mag vizsgálatával, már Neumann is érdekes gondolatnak tartotta a magot, az általa vizsgált zérusösszegű játékokban a mag mindig üres, így a definíció Gillies [57] és Shapley [58] nevéhez kötődik. A mag üressége [59] a kezdetektől foglalkoztatta a kutatókat. Bondareva [60] és Shapley [58] egymástól függetlenül állították fel a nem üres mag feltételeit. Ezzel párhuzamosan elindult a kutatás egy hasonló, de nemüres megoldás felé.

Dinamikus megoldások. A mai napig nincs olyan megoldás koncepció, amely minden kívánságnak eleget tenne. Zhou [61] foglalta három pontba a követelményeket. Egy megoldás sohasem üres, nem definiáljuk a játékosoknak sem egy előre megadott, sem az összes lehetséges partíciójára. A Neumann–Morgenstern-megoldás, a mag és még sokan mások az elsők, az alkuhalmaz például a második feltételen bukik el.

Eredményt hozhatnak az olyan dinamikus megközelítések, amelyek egy játék ergodikus halmazát tekintik megoldásnak. Lényegében ez történik Shenoy [62] dinamikus, Packer [63] sztochasztikus megoldása, Sengupta–Sengupta [64] életképes javaslatai (*viable proposals*) és a legkisebb domináns halmaz esetében [65]. Ezek a megoldások általában már definíciójukból adódóan nem lehetnek üresek. Utóbbi kettő külön érdekessége, hogy egybeesnek a nemüres maggal. Agastya [66] bemutatta, hogy a sztochasztikusan stabil csoportok részhalmazai a magnak, illetve Yang [67] bizonyít egy alacsony lépésszámot, amivel a mag elérhető.

2.4. A párosításelmélet bemutatása

A párosításelmélet lényege, hogy halmazok elemeit akarjuk egymással párosítani. Ha két egymástól eltérő, és semmilyen szinten nem keveredő halmazok elemeit akarjuk egymással összerendezni, akkor kétoldali párosításról beszélünk (pl.: egyetemi felvételi probléma), ha csak egy halmaz elemeit akarjuk egymáshoz rendelni, akkor egyoldali párosításról beszélünk (pl.: szobatárs probléma). [68] [69] [70]

Az első igazán jelentős párosításelméleti cikk is kétoldali párosításról szól, ebben Shapley és Gale közösen, házastárs keresési algoritmus kapcsán mutatták be a párosítást, mint problémát és adtak rá megoldást. [71]

Ha halmazokat szeretnénk egymással párosítani, akkor annak többféle megoldása lehetséges. (Lásd [2.2](#) fejezet.)

Léteznek úgynevezett *egy az egyhez* párosítások, ahol az egyik halmaz minden eleméhez a másik halmazból csak egy-egy (de ezek egymástól különböző), elemét rendeljük. Ezt hívják kölcsönös egyértelmű hozzárendelésnek. Abban az esetben, ha az egyik halmaz elemeihez a másik halmazból több elem is választható, akkor többértelmű hozzárendelésről beszélünk, ezek az úgynevezett *egy a többhöz* hozzárendelések. Létezhetnek még olyan hozzárendelések, ahol nincs korlát egyik halmazban sem, hanem bármelyik elemhez bárhány elemet lehet hozzárendelni a másik halmazból.

Az párosításelmélet alapproblémájának a lényege, hogy van két egymástól független halmazunk, és szeretnénk a két halmaz elemeinek egymáshoz rendelésével párokat alkotni. Egy egyszerű algoritmus segítségével megmutatták, miként található meg stabil párosítás az ezen halmazok elemei között, ha mindkét félnek léteznek preferenciái. Lényeges, hogy nem csak létrehoztak párokat, hanem olyan párokat alkottak meg, amelyek stabilnak mondhatók [72] [73], mivel nincsen olyan blokkoló pár, ahol mindkét fél jobban járna, ha együtt kötne házasságot elhagyva az előző párját [74]

A házassági modellben például 1 fiúhoz mindig 1 lányt választottak [75] [76] [77], így az *egy az egyhez* modellről beszélünk, de a Gale-Shapley algoritmus alkalmazható *sok az egyhez* és *sok a sokhoz* probléma esetén is. Ilyen például a gyakornokok elhelyezése kórházakban. [78], [79], [80] Ez egy *sok az egyhez* probléma, vagyis egy kórházban több gyakornok is elhelyezhető, míg egy gyakornok csak egy kórházban fog dolgozni. [81] [82] Roth [83] matematikailag is bizonyította, hogy az amerikai orvosi rezidensek elhelyezkedését támogató program a Gale és Shapley [71] által publikált algoritmussal megegyező algoritmust használja. [84], [85] Az eredeti algoritmus tehát tökéletesen alkalmazható, annyi különbséggel, hogy ha megtaláltunk egy párosítást, az nem jelenti a probléma megoldását, hanem addig fut az algoritmus, amíg fel nem tölti a teljes kapacitás esetén elérhető helyeket, vagy el nem fogynak a jelentkezők. Az algoritmus szerint csak akkor nem fog bekerülni egy gyakornok az adott kórházba, vagy jelentkező az

adott egyetemre, ha kapacitás-hiány miatt nincs lehetősége beférni, vagy már felvették egy általa előbbre rangsorolt helyre.

Az algoritmust eredetileg a kórházak irányából futatták (vagyis a kórházak preferenciáit vizsgálták első lépésben), [83] [86] [87] de ezt mára megfordították és a jelentkezők oldaláról fut le az algoritmus, amivel a jelentkezők szempontjából jobb eredményt lehet kapni, bár Roth és Peranson [86] elemzése szerint ez a különbség nem szignifikáns.

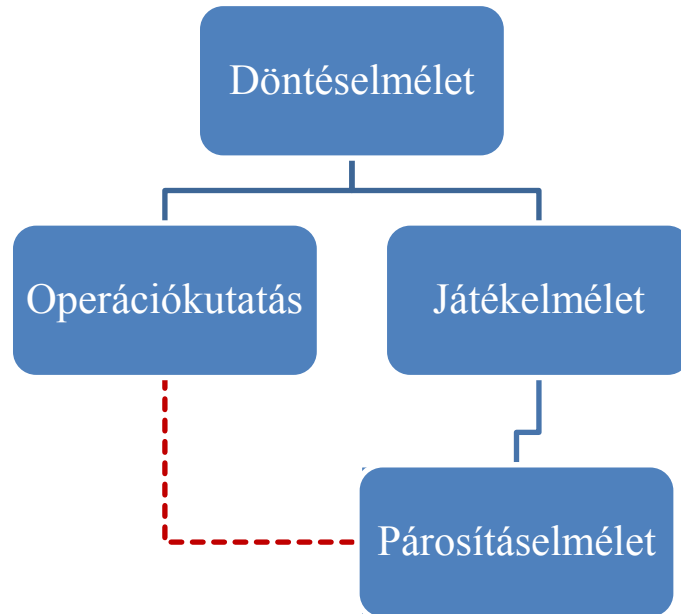
A Gale-Shapeley algoritmus *sok a sokhoz modellekre* is alkalmazható [88] [89], mint amilyen például a hallgatók tárgyfelvetele, ahol egy hallgató több tárgyat is felvehet, míg egy tárgyra több hallgató is jelentkezhet.

Jelen dolgozatnak a célja a párosítási elmélet alkalmazási lehetőségeinek bemutatása. Emellett olyan szoftver megalkotása, amivel egyszerűbben akár matematikai tudás nélkül is lehessen párokat alkotni. A következő alfejezetekben megvizsgálom a különböző modelleket, és a rájuk adott válaszként született algoritmusokat, összehasonlítom őket, és értékelem.

2.4.1. A párosításelmélet és az operációkutatás kapcsolata

Mint a 14. ábrán jól látható a döntéselméletben bár elég sok részhalmoz van, de van 2 egymástól független terület. Ezek a játékelmélet és az operációkutatás, bár a játékelmélet nagyon sok esetben alkalmaz operációkutatási megoldásokat, mint például lineáris programozást. Az egyik esetében optimalizálni akarunk, és minden információ a rendelkezésünkre áll, míg a másik esetében általában valamilyen információ hiányos helyzet van, ahol általában több szereplő „játszik” egymással, és előfordulhat, hogy össze is fognak egymással. Az operációkutatásra egy egyszerű példa bármilyen szállítási feladat, míg a játékelméletnél gyakorlatilag bármilyen olyan példa található, ahol emberek egymással vagy egymás ellen játszanak, ilyesmi lehet például a parlamenti választás.

A lényege, hogy két diszjunkt halmazban szereplő elemeket az általuk meghatározott preferencia sorrendnek megfelelően párosítunk egymással. [90] A párosítási problémák felírhatók gráfelméleti nyelvezettel is. Ebben az esetben a játékosok a gráf csúcspontjai, míg a közöttük lévő kapcsolat a gráf élei. Párosításnak akkor nevezzük ezt az élhalmazt, ha semelyik két élnek sincs közös pontja, vagyis az élhalmaz *független*.



14 ábra – A párosítás- és a döntésemélet kapcsolata

A párosításeméleti problémák nagyon sok részben hasonlítanak a már ismertetett operációkutatási problémákhoz. A nagy eltérés az, hogy amíg az operációkutatásban valamilyen szempont alapján (például legkisebb szállítási költséggel, vagy legrövidebb szállítási idő, tehát valamilyen könnyen számolható szélsőérték segítségével) próbálunk meghatározni optimális párokat, addig a párosításeméletben az a fontos, hogy ha nem tudjuk a szállítási költségeket, de tudjuk, hogy milyen sorrendben szeretnék a rendelési helyekről a célállomásokra anyagokat szállítani, akkor lehetőleg minden esetben a számunkra (és itt jelenik meg az egyéni hasznosság kérdése, az összesített hasznossággal szemben) jobb helyre szállíthassunk előbb.

2.4.2. Párosításeméleti alapfogalmak

A fejezet további részeinek megértéséhez ebben az alfejezetben definiálom a főbb fogalmakat.

Párosításon különböző elemek egymáshoz való hozzárendelését értjük. Létezik egyoldali és kétoldali párosítás is. *Egyoldali* párosítás esetén egy adott halmaz elemeit rendeljük hozzá ugyanezen halmaz valahány eleméhez. Ilyen probléma lehet az úgynevezett szobatárs probléma, ahol embereket akarunk egymáshoz rendelni. [91] [92] [93] [94]

Kétoldali párosítás esetén két egymástól független, különböző halmaz elemeit akarjuk egymáshoz rendelni, ennek klasszikus példája a felvételi eljárás, ahol egyetemekhez akarunk hallgatókat hozzárendelni. [95]

Más fogalom tartozik az operációkutatás és a párosításelméletben a párosítás fogalmához. Operációkutatás esetén a párosítás, például egy szállítási probléma vagy hozzárendelési probléma esetén mindig együtt jár a sorrendek meghatározása valamilyen érték hozzárendelésével is. Például szállítási költség egyik helyről a másik helyre, és ebben az esetben összesített hasznosság optimalizálásáról, vagyis például költség minimalizálásról beszélünk. Párosítás elmélet esetében alapértelmezetten ez az érték nem mindig velejáró, és nem összesített, hanem egyéni hasznosság alapján keresünk megoldásokat.

Stabil párosítás esetén nincsen olyan szereplő vagy szereplők, akiknek lenne lehetősége arra, hogy egy új együttműködés létrehozásával mindannyian jobban járnának. [96] [97] [98] Házassági probléma esetén nem létezik úgynevezett blokkoló pár, melyben mindkét félnek megérné egy új házasságot kötnie egymással, és így elhagyva a jelenlegi házastársaikat.

2.4.3. Párosítás elméleti modellek (problémák)

Házassági probléma

Létezik két, egymással semmilyen szinten nem keveredő halmaz, nevezzük őket F-nek és N-nek. F jelölje a férfiak, N a nők halmazát. Az elemeket a halmazokban jelezzük f és n karakterekkel. Akkor elmondhatjuk, hogy $F \cap N = \emptyset$ és $F = \{f_1, f_2, \dots, f_m\}$, illetve $N = \{n_1, n_2, \dots, n_p\}$, ha a férfiakból m , míg a nőkből p darab van. Meg kell még határozni a férfiak és a nők preferencia-sorrendjét is. A férfiak sorrendje $P(f_1) = n_1, n_3, n_2, \dots$, míg a nőké $P(n_1) = f_3, f_1, f_2, \dots$. Ezáltal a párosítás lehetőségeit a (F, N, P) hármassal lehet leírni. [99]

Ennél a problémánál kölcsönös egyértelmű hozzárendelés van, mivel minden férfihez csak egy nő rendelhető, és ez igaz fordítva is. [100]

Iskolai felvételi eljárás

Az egyetemi felvételi eljárás a korábbiakban leírtaknak megfelelően abban különbözik házassági problémától, hogy nem egy az egyhez, hanem több az egyhez párosítási algoritmust alkalmaz. [101] [102] [71] A sok az egyhez párosítások mind visszavezethetők

az egy az egyes megoldásokra, ebben az esetben az egyetemi szakokat kisebb egységekből (egy hallgatót tartalmazó) álló részekre bontjuk. [103] [104] [105]

Itt E jelölje az egyetemek, H a jelentkezők halmazát. Az elemeket a halmazokban jelezzük e és h karakterekkel. Akkor elmondhatjuk, hogy $E \cap H = \emptyset$ és $E = \{e_1, e_2, \dots, e_m\}$, illetve $H = \{h_1, h_2, \dots, h_p\}$, ha az egyetemekből m , míg a leendő hallgatókból p darab van. Meg kell még határozni az egyetemek és a hallgatók preferencia sorrendjét is. $P(e_1) = h_1, h_3, h_2, \dots$ az egyetemek sorrendje, míg a hallgatóké $P(h_1) = e_3, e_1, e_2, \dots$. Ezáltal a párosítás lehetőségeit a (E, H, P) hármassal lehet leírni. Ebben a modellben - főleg a magyarországi egyetemi felvételik szabálya alapján - az egyetemek által meghatározott hallgató rangsort nehéz meghatározni.

Gyakornok elhelyezési probléma

Észak-Amerikában a gyakornok kórházaknál való elhelyezésére már az '50-es évek óta használják azt az algoritmust, amit a Gale és Shapley 1962-ben publikált. Ezt a felfedezést Roth [83] mutatta meg. Az Egyesült Királyságban több különböző párosító modellt használnak, ezeket hasonlítja össze Roth [106] A Skóciában bevezetett mechanizmust Irving és Manlove [107] mutatta be.

Ez a probléma nagyon sokban hasonlít az iskolai felvételik problémájára. R jelölje a gyakornokokat, H pedig a kórházak halmazát. Az elemeket a halmazokban jelezzük r és h karakterekkel. Akkor elmondhatjuk, hogy $R \cap H = \emptyset$ és $R = \{r_1, r_2, \dots, r_m\}$, illetve $H = \{h_1, h_2, \dots, h_p\}$, ha a rezidensekből m , míg a kórházakból p darab van. Meg kell még határozni a gyakornok és a kórházak preferencia sorrendjét is. $P(r_1) = h_1, h_3, h_2, \dots$ a kórházak sorrendje, míg a gyakornokoké $P(h_1) = r_3, r_1, r_2, \dots$. Ezáltal a párosítás lehetőségeit ugyancsak a (R, H, P) hármassal lehet leírni.

Több a többhöz probléma

Létezhetnek még olyan problémák, ahol nem *egy a többhöz*, hanem *több a többhöz* a párosítás alapja. [108] [109] ilyen lehet például a hallgatók tárgyfelvétele az egyetemeken, ahol egy hallgató több tárgyat is felvehet, de egy tárgyat több hallgató is felvehet. Nincs igazán nagy változás az előző problémához képest, hiszen ahogy tudunk az *egy az egyes* problémákból *egy a többes* problémákat alkotni itt is ugyanaz a lépés megtehető.

2.4.4. Párosítás elméleti algoritmusok és azok összehasonlítása¹

Az előző fejezetben bemutatott problémák mindegyike megoldható a következőekben felsorolt módszerekkel, de elég nagy különbség lehet ezek felhasználásában. A hétköznapi életünkben nagyrészt az úgynevezett Mohó algoritmust alkalmazzuk. Párosítás elmélet esetén általában csak a saját preferenciánkkal vagyunk tisztába a választási lehetőségek ismeretén túl, és nem ismerjük a többiek választásait.

Mohó algoritmus

A véletlen vagy sorozatos diktatúra alkalmazásakor a hallgatókat (véletlen, azaz sorossal kialakított) sorba rendezik, és a soron következő hallgató mintegy diktátorként választhat a megmaradt opciók közül. Így az algoritmus nem veszi figyelembe a másik fél, felvételi probléma esetében pl. az egyetemek preferenciáját, nem venné figyelembe, valóságban természetesen erre a problémára nem alkalmazzák ezt az algoritmust.

1. Vizsgáljuk az egyéneket egyenként
2. Minden egyén esetében megvizsgáljuk a hozzá tartozó preferencia sorrendeket. Ha találunk olyan alternatívát, amely szerepel az egyén preferencia listáján és még szabad, akkor kínáljuk fel azt neki – innentől az alternatíva és az egyén párt alkotnak
3. Ha nem találunk az egyénnek megfelelő még szabad alternatívát, akkor az egyén nem kerül párosításra

Gale-Shapley algoritmus (GS)

Amellett, hogy a Mohó algoritmus nem egy stabil párosító mechanizmus, a beiskolázási algoritmusok esetében egy másik probléma is felmerül, nevezetesen, hogy a különböző iskolák más és más prioritási sorrendbe rendezik a hallgatókat. Tehát a beiskolázási mechanizmusnak figyelembe és tudomásul kell vennie az iskolák ilyen jellegű preferenciáit. Balinski–Sönmez [80] ; Abdulkadiroğlu–Sönmez [110] rávilágítanak, hogy a Gale–Shapley algoritmus nemcsak hogy megfelel ezeknek az igényeknek, de még olyan további szempontok figyelembevételére is alkalmas, mint az úgynevezett szabályozott választás, ahol bizonyos korlátokat alkalmaznak a nemi, faji vagy etnikai alapú szegregáció csökkentésére

1. Az egyének és az alternatívák egyaránt rendelkeznek saját preferenciákkal.
2. Minden egyén a legmagasabb preferenciával rendelkező alternatívát választja

¹ Publikálva [154]

3. Ha többen választják egyszerre ugyanazt az alternatívát, az alternatívák “kiválasztják” a preferencia sorrendjük alapján számukra legkedvezőbb egyént, és a többit visszautasítják
4. Azok az egyének akik nem kerültek kiválasztásra (mert voltak náluk jobban preferált egyének), a soron következő legmagasabb preferenciájú alternatívát választják – és megismétlődik a második lépés.
5. A harmadik lépés addig ismétlődik, míg minden személy megtalálja a számára megfelelő alternatívát, vagy saját preferencia listájának végére nem ér

Bostoni mehanizmus

Ezt a mechanizmust Bostonban (1999 és 2005 között) és még számos más városban (*Ergin–Sönmez* [111] , *Abdulkadiroğlu és szerzőtársai* [112]) használták. Az algoritmus a következő ([110] alapján):

1. Minden egyén a legmagasabb preferenciával rendelkező alternatívát választja
2. Ha többen választják egyszerre ugyanazt az alternatívát, az alternatívák “kiválasztják” a számukra legkedvezőbb egyént, és a többit visszautasítják
3. Azok az egyének, akik az első körben nem kerültek kiválasztásra, a soron következő legmagasabb preferenciájú alternatívát választják – és megismétlődik a második lépés
4. A harmadik lépés addig ismétlődik, míg minden személy megtalálja a számára megfelelő alternatívát, vagy saját preferencia listájának végére nem ér

Legnagyobb problémája, hogy a jelentkezőknek taktikázniuk kell. Nagyon kockázatos dolog ugyanis olyan iskolát első helyen megjelölni, ahova nagy a túljelentkezés, hiszen ha ide nem sikerül bejutni, könnyen lehet, hogy a második, harmadik, stb. helyen megjelölt iskolák is betelnek (*Glazerman és Meyer* [113]).

A Gale-Shapeley algoritmus esetén a Boston mechanizmussal ellentétben a párosítások csak az algoritmus végén kerülnek véglegesítésre, tehát egy adott alternatívára bekerülhet olyan egyén is, aki az alternatívát nem első helyen jelölte meg, de az alternatíva preferencia listáján magasabb helyen van, mint az a személy, aki azt első helyen jelölte meg.

A columbusi algoritmus

A columbusi algoritmus sokban hasonlít a rezidensek központosítás előtti felvételi rendszeréhez, azzal a lényegi eltéréssel, hogy mivel az elfogadás után a jelentkező kikerül a rendszerből, nem kap utólag kedvezőbb ajánlatot, ami esetleg destabilizálhatná a rendszert, illetve nem merül fel a jogos irigység problémája sem.

A Columbus Cityben alkalmazott algoritmus a következő (*Abdulkadiroğlu–Sönmez* [110] alapján).

1. Minden egyén legfeljebb három alternatívát jelölhet meg.
2. Bizonyos alternatívák egyértelműen preferálnak bizonyos tulajdonságokat és az azzal rendelkező személyeket. Egyébként a jelentkezők rangsorát sorsolással határozzák meg.
3. A (még) szabad helyeket a fenti preferenciák figyelembevételével ajánlják meg a jelentkezőknek. Az ajánlatra három napon belül kell válaszolni. Elfogadás esetén a jelentkező kikerül a rendszerből, az elfogadott ajánlat alapján hozzárendelődik az alternatívához. Ahogy egyes ajánlatok elutasításra kerülnek, ezek a helyek megnyílnak a korábban várólistás személyeknek.

A legjobb csere-körök módszere (Top trading Cycle)

Az *Abdulkadiroğlu–Sönmez* [110] által javasolt algoritmus lényege, hogy az iskolák által legjobbnak tartott hallgatók egymás között elcserélhetik a megszerzett helyüket. Előnye, hogy a hallgatóknak érdeke a valós preferenciák felfedése, tehát az algoritmus orvosolja az előbbi algoritmusok esetében felmerült problémák nagy részét.

1. Minden hallgató és iskola megnevezi, hogy mit/kit rangsorol az első helyre. Jelentse s_n a párosításban részt vevő n -edik hallgatót ($n=1, \dots, k$), míg C_m a párosításban résztvevő m -edik főiskolát (college) ($m=1, \dots, k$). Mivel a résztvevők száma véges, létezik olyan $s_1, C_1, s_2, \dots, C_k$ kör, hogy s_i a C_i -t preferálja, aki viszont s_{i+1} -t, továbbá C_k az s_1 -t preferálja. Minden hallgató és minden iskola legfeljebb egy-egy körhöz tartozik. Minden olyan hallgatót, aki egy ilyen körhöz tartozik, felveszi az általa megnevezett iskola. Ezzel a hallgató kikerül a rendszerből, az iskolának pedig egy-egyel kevesebb szabad helye marad. Ha minden hely elfogyott, akkor az iskola is kikerül a rendszerből, így a továbbiakban a hallgatók már nem nevezhetik meg, mint kedvencüket.
2. Minden további lépésben a megmaradt hallgatók és a megmaradt iskolák vesznek részt, ettől eltekintve a lépés lefolyása ugyanaz, tehát a résztvevők megnevezik a preferenciájukat, majd a körökhöz tartozó hallgatókat az általuk megnevezett iskola veszi fel.
3. Az algoritmus akkor ér véget, ha a hallgatók elfogynak. Mivel minden lépésben legalább egy hallgató felvételt nyer, a szükséges lépések száma nem több mint a hallgatók száma.

A különböző algoritmusok összehasonlítása látható a következő táblázatban.

Algoritmusok	van értelme taktikázni	aki egyszer bekerült egy helyre, az bent is marad	a legmeghatározóbb karakterisztika	figyelembe veszi a referenciákat
Mohó	nincs	nem	leginkább preferált	nem mindig
Gale-Shapley	nincs	nem	preferenciák, bármely választás	igen

Boston	van	igen	leginkább rált	prefe-	nem mindig
Columbus	van	nem	leginkább rált	prefe-	nem mindig
Top Cycles	Trading nincs	igen	leginkább ráltak, csere	prefe-	igen

15 táblázat – A különböző párosítás elmélet algoritmusok összehasonlítása [114]

Ahogy az 15. táblázatból is látható, vannak olyan algoritmusok, amelyek támogatják a jelentkezőket, hogy megadják a valós preferenciájukat, míg egyes algoritmusoknál inkább taktikázni kell. Nem véletlen, hogy azért a különféle felvételiknél, akár közép- vagy felsőoktatás, akár a rezidens képzés esetében a Gale-Shapley által meghatározott algoritmust alkalmazzák.

Ismert alkalmazási területek

Az előzőek alapján tehát nagyon sok különféle algoritmus létezik, és a különböző helyzetekben más és más algoritmust alkalmaznak. Magyarországon a felsőoktatásban a bekerülésnél lévő felvételi kivételével szinte sehol sem alkalmaznak kifejezetten párosításelméleti algoritmusokat, leginkább ezekben az esetekben a Mohó algoritmusnak megfelelő algoritmus segítségével alakulnak ki a párok. Ennek - ahogy az előző pontban látható volt -, nem csak az a problémája, hogy nem ad stabil megoldást, hanem a hallgatókat sem lehet vele rangsorolni. Így valójában csak a véletlenül múlik, hogy ki hova kerül be. Nemzetközi szinten is csak nagyon kevés olyan helyzetet ismerünk, ahol használnának stabil párosítási algoritmusokat, ezeket lehet látni a 16. táblázatban.

Sok tanulmány született, amik bővebben foglalkoznak ezekkel, az iskolai felvételikről ([112], [115], [111], [116]). A munkaerőpiacban ([117], [118], [119], [120], [121]), a vesecseré programban használt algoritmusokról, ([122], [123], [124], [125]) illetve számos egyéb helyen használt párosítási algoritmusok. (rezidensek: [126], [127], [128], ösztöndíj: [129], [130])

Ország	Iskolai felvételik	Munkaerőpiac	Vesecseré program	egyéb
Franciaország		tanárok elhelyezése		
Németország	felsőoktatás			
Magyarország	közép és felsőoktatás			
Izrael				kollégiumi elhelyezés

Hollandia			The Dutch program	
Spanyolország	felsőoktatás		The Spanish Program	
Törökország	felsőoktatás			
Egyesült Királyság		Scottish Foundation Allocation Scheme (SPA-SFAS), Teacher Induction Scheme (TIS)		

16 táblázat – A párosítási algoritmusok használata Európában, [114]

A tapasztalatok szerint nagyon kevés helyen használnak párosítási algoritmusokat, és általában párosítások esetén a Mohó algoritmust használják. Sokat lehetne javítani a párosítások létrejöttének hatékonyságán, vagyis, hogy mennyi idő és energia befektetéssel határozzuk meg a párokat - és ezzel párhuzamosan, nem mellékesen a megelégedettségén, vagyis a halmazokban résztvevők boldogságán, ha egy nekik sokkal inkább megfelelő párt találunk -, ha valamelyik párosítási algoritmust használnák.

3. A SZOFTVER TERVEZÉSE ÉS ELKÉSZÍTÉSE

Mint látható nagyon sok olyan helyzet és probléma létezik, ahol szükség lenne egy hatékony, és stabil párosítást adó algoritmus használatára. Véleményem szerint ahhoz, hogy hatékony párosítást tudjanak alkalmazni az előttük álló problémára, ahhoz nincs is szükségük arra, hogy részletesen ismerjék az algoritmusokat, elég, ha van egy szoftver, amely megoldja helyettük a problémát. A szoftvertervezés első és talán legfontosabb lépése, hogy mit is szeretnénk megoldani vele. Ahogy az előző fejezetben is bemutatam kétféle párosítás elméleti adatbevittet alkalmaztam. Az első esetben (Erasmus) szükség volt mind a hallgatók, mind a fogadó fél preferenciáinak a meghatározására, míg a többi esetben a hallgatók preferenciáinak a súlyozása alapján határoztam meg a másik fél preferenciáit. A program tervezésénél mind a két megoldást figyelembe kell venni. A következő fontos lépés, hogy milyen algoritmusok segítségével szeretném meghatározni a párosításokat. Ahogy a 2. fejezetben megmutattam, különböző helyzetekben más és más algoritmus alkalmazása a logikus, így a szoftver első verziójában a következő algoritmusok lesznek elérhetőek: Mohó és a Gale-Shapley algoritmus, illetve a Boston mechanizmus.

Fontos egy szoftvertervezésnél még az ergonómia, a könnyen kezelhetőség, illetve, hogy a szoftver ténylegesen azt tudja, amire kitalálták, nincs szükség benne felesleges design elemekre.

A következő alfejezetekben a szoftvertervezés lépéseit, az adatbázis megtervezését, illetve a szoftver elkészítését, majd a kész szoftvert mutatom be.

3.1. A szoftver szükségessége

Ahogy a kutatásaimban is bemutatam nagyon sok olyan helyzet létezik, ahol szükség lenne párosításelméleti algoritmusok használatára. Miért nem tesszük mégsem? Ennek egyik oka, hogy nem ismerjük őket, a másik oka, hogy nehézkes a használatuk, magasabb szintű matematikai tudásra, és bonyolult, nagy hibalehetőséggel rendelkező lépésekre van szükség a probléma megoldására. Természetesen, mint minden olyan matematikai probléma, ami jól algoritmizálható, a párosításelméleti problémák is könnyen megoldhatók számítástechnikai segítséggel. Ahogy az [2.4.4.](#) fejezetben bemutatam több helyen használnak párosításelméleti algoritmusokra épülő szoftvereket, ilyen pél-

dául Magyarországon a felsőoktatási felvételi helyzete. Ami probléma mégis, hogy ezek a szoftverek nem elérhetők a nagyközönség számára, ha egy vállalkozás, vagy akár egy egyetemi oktató szeretne ilyen problémákat megoldani akkor három lehetősége van. Az első, hogy nem alkalmaz szoftvert, hanem matematikai módszerek segítségével kiszámolja a párokat, ez minél nagyobb halmazokról beszélünk annál nehezebb. A következő lehetőség, hogy ő maga elkészít egy szoftvert, ez természetesen a matematikai tudásán kívül még alapvető informatikai, szoftver tervezési és fejlesztési tudást is megkövetel. A harmadik megoldási lehetőség, hogy piacon megpróbál ilyen szoftvert beszerezni, akár fizetős akár szabad szoftverről beszélünk.

Ezért mielőtt nekikezdtém volna a szoftver fejlesztésének, én is megvizsgáltam, hogy milyen olyan szoftverek érhetőek el a piacon, amikkel ezek a problémák megoldhatóak.

Több olyan szoftver is elérhető, ahol csak maga a forrás érhető el, ilyenek vannak például python [131] vagy Java [132] programozási nyelven. Ezek nagyjából nem tudnak többet, mint egy előre megadott, a szoftverbe beágyazott adatok alapján kiszámolják a párosításokat. Emellett vannak olyan szoftverek is, amiknél van frontend is, ilyen szabad szoftver [133], ami bárki számára hozzáférhető képes megoldani házassági problémát. A két halmaz maximálisan 10-10 elemet tartalmazhat, és csak az *egy az egyhez* hozzárendeléseket képes kezelni. A szoftver kezelése nagyon egyszerű, bár a tudása is hasonlóan alapszinten mozog. Amire használható, hogy bemutassuk vele például oktatás során, hogy miként is működik maga az algoritmus.

3.2. A szoftver megtervezése

A szoftver elkészítése előtt dönteni kellett a programozási nyelvről, illetve a környezetről. A döntés végül internetes weboldalra esett, mivel internet lassan mindenhol elérhető (de ha nem, akkor is lehet egyszerűen az intranetre is telepíteni futtató környezetet), és közben teljesen platform független, így bármilyen operációs rendszeren futtatható. Így a php nyelvre esett a választás, amivel mySQL adatbázis szervert választottam.

A következő lépés a tervezésben az úgynevezett „Specifikáció”, vagyis hogy melyik algoritmusokat akarom alkalmazni, milyen bemenő és kimenő adatokat várok el. Első lépésben ezt leegyszerűsítettem. Mivel a dolgozatban is bemutatott *Gale-Shapley algo-*

ritmus, stabil megoldást ad, így ennek mindenképpen érdemes benne szerepelnie, emellett az összehasonlítás érdekében az első verzióban szerepel még a *Mohó algoritmus*, és a *Bostoni mechanizmus*. Természetesen a szoftver későbbiekben tovább bővíthető, és ha adódnak olyan helyzetek, amikor más, vagy másfajta algoritmusra van szükség, akkor a szoftverbe ez könnyen implementálható.

A következő lépés az adatbázis megtervezése volt, hogy le tudjuk tárolni az adatokat a későbbi felhasználás miatt, illetve, hogy a felhasználók adatait külön-külön és akár más-más időben is fel lehessen tölteni. Minden szoftverfejlesztés egyik legfontosabb része, hogy miként és milyen struktúrában akarjuk tárolni az adatokat, ezért én itt erre kiemelt figyelmet fordítottam.

Ezek után jött a program elkészítése, majd tesztelés, és ahol szükség volt, ott javítás.

3.3. Az adatbázis megtervezése

3.3.1. Modell, modellezés, hasonlóság

Az általános és természettudományos értelemben vett hasonlóság és modell fogalmát Szücs Ervin [134] [135] [136] alapján foglalom össze.

A világ bármely része (gyakorlatilag) végtelen sok tulajdonsággal bír. Ezzel ellentétben az emberi megismerő képesség viszont erősen véges: tudásunk, időnk, gazdasági-pénzügyi-műszaki erőforrásaink korlátozott mivolta okán. Ezzel a véges megismerő képességünkkel kellene a végtelent megragadni és leírni. Mivel ez lehetetlen, ezért szükségszerűen egyszerűsíteniünk kell: a fontos, vagy annak vélt tulajdonságokat kiragadjuk, a lényegteleneket pedig elhanyagoljuk.

A (való) világ bármely részének ezt a végtelen sok tulajdonságát együtt állapotnak nevezzük. Mivel véges lehetőségeinkkel többnyire nem tudjuk, nem lehet a kívánt eredményt elérni, vagy nem lehet gazdaságosan elérni, ezért szükséges az eredeti tárgyat, technikai rendszert valahogyan helyettesíteni. Amikor vizsgálódásunkat a fontosnak kijelentett jellemzők vizsgálatára szűkítjük, a többi jellemző elhanyagolásával, állapotjellemezésről beszélünk.

Az állapot maga – nyilvánvalóan – objektíven adott. Az állapotjellemezés pedig – ugyancsak nyilvánvalóan – szubjektív: függ a rendelkezésünkre álló erőforrásoktól,

meglévő tudásunktól, sőt a vizsgálat céljától is (geocentrikus és heliocentrikus világmodell, a légellenállás vizsgálatához a szélcsatornába betehetünk egy geometriailag az eredetivel egybevágó hungarocellt, de ugyanez alkalmatlan lesz a töréstartományra).

A modell „jósa” attól függ, hogy mennyire felel meg az egyértelműség és a reprodukálhatóság követelményeinek. A vizsgálat célja határozza meg, hogy mely állapotokat lehet azonosnak venni, és mely állapotokat szükséges megkülönböztetni. Pl. esetenként száz-kétszáz hallgató teljesítményét értékeljük mindösszesen öt csoportba sorolva (elégtelen, ..., jeles), holott jó eséllyel nincs közöttük két egyforma zárthelyi.

Az egyértelmű állapotjellemzéshez *szükséges* tulajdonságok közül egyet sem lehet elhagyni, különben hamis eredményeket kapunk. Az egyértelmű állapotjellemzéshez szükséges tulajdonságok minimális halmaza jelenti a *szükséges és elégséges* jellemzőket. A pontszerű test, mint a fizikában alkalmazott modell kiválóan alkalmas jó néhány természeti törvény vizsgálatára. A továbblépéshez azonban bonyolultabb modelleket kell választani, a merev testen át egészen a végeselem módszerig.

Összefoglalva tehát a modell olyan helyettesítő eszköz, amely meghatározott célra, és/vagy meghatározott időre helyettesíti az eredetit. A modell és az eredeti hasonlósága tehát funkcionális jellegű és célfüggő.

3.3.2. Adatmodellezés

A világ egyes részeit – adott esetben – különféle adatokkal is le tudjuk írni („90-60-90” – utalhat az ideálisnak tartott női alkatra és a sötétlila színre egyaránt), azaz adatokkal is lehet modellezni. Halassy ([137] 44 oldal) úgy határozza meg az adatmodell fogalmát, hogy „Az adatmodell véges számú egyedtípusnak, azok egyenként is véges számú tulajdonság- és kapcsolattípusának a szervezett együttese”, azaz meghatározzuk a lényeges dolgokat, azok fontos tulajdonságait, majd adott szabályok szerint leírjuk összefüggéseiket és tartalmukat.

A modellalkotás általában, ezen belül az adatmodellezés, alkotó szellemi tevékenység, kicsit hasonlít a művészi alkotás létrehozásának folyamatára. Abban, hogy mit (és hogyan) tart fontosnak az adatmodell létrehozása során a tervező, abban legalább egy kicsit benne van a tervező egyénisége, világlátása is.

Számos tervezési és fejlesztési módszer létezik, Raffai Mária három tucatot ismertet *Információrendszerek fejlesztése és menedzselése* c. munkájában ([138]. 346. oldal), messze nem a teljesség igényével. A különféle, tervezést és fejlesztést segítő számítógépes eszközök száma ennél jóval nagyobb. Ezek elsősorban a „nagy” feladatok projektszemléletű megvalósítását könnyítik meg. Figyelembe véve azonban azt, hogy egy ilyen módszer önmagában is egyfajta modellezése a valóságnak, nem szabad elfelejtenünk figyelembe venni a határait és korlátait.

Ugyancsak szem előtt kell tartani, hogy az adatmodellezés (bármiféle modellezés) alkotó szellemi tevékenység. Az ennek segítésére rendelkezésünkre álló mindenkori eszközkészlet és technika (ide értve a modellezési-tervezési-fejlesztési-tesztelési módszereket és eszközöket is) azonban bár fontos, sőt szükséges feltétele a megfelelő minőségű eredmény elérésének, de önmagában még nem elégséges. A legjobb eszköz sem ér önmagában semmit, ha nincs ott a modellező személy, aki az éppen a rendelkezésére álló technikát értelmesen, alkotó módon használja.

Az adatmodell jóságát a fentiek értelmében és Halassy ([137]) nyomán úgy határozhatjuk meg, hogy a jó adatmodell a) valósághű, b) teljes, c) minimális. Ehhez két előfeltételt is kiköthetünk: érthetőnek és egyértelműnek kell lennie (különben a három alapkritériumot nem tudjuk vizsgálni). A minimalitás két különböző dolgot is jelenthet. Egyrészt: az adatbázis legyen mentes a redundanciától, másrészt pedig a cél szempontjából nem elsőrendűen fontos funkciókat ne tervezzünk.

A redundancia – *főlöszleges* adatismétlődés – okozta problémák egyrészt a megnövekedett tárhelyfoglalás és a megnövekedett feldolgozási idők. Ezeket lehet valamilyen mértékben jelentősebb hardver-erőforrások felhasználásával ellensúlyozni. Másrészt azonban a redundancia sokkal súlyosabb következménye, hogy a redundáns adatok előbbutóbb ellentmondásosakká válnak, s ezt semmilyen hardvereszközzel, semmilyen költséggel nem lehet ellensúlyozni.

3.3.3. Tervezési szintek

Halassy ([137], [139]) részletesen tárgyalja és példákkal támasztja alá, hogy a korrekt és eredményes munka alapja a tervezés három szintjének világos elkülönítése.

„Fogalminak nevezzük a jelenségeket, azok sajátosságait és viszonyait a valóságnak megfelelően és természetes fogalmakban tükröző adatszerkezetet.” ([137]. p. 51.) Azaz: az adatmodellezés során *nem* foglalkozunk semmilyen egyéb szemponttal, csak a valóság (hű, teljes és minimális) leírására törekszünk. Például, ha élelmiszercikkek nyilvánítását tervezzük, a minőségmegőrzési határidő nyilvánvalóan fontos tulajdonság, és modellezni kell, ellenben nem foglalkozhatunk – még – avval, hogy a majdani kezelőben van-e beépített dátum adattípus. A „valóság leírása” nem csupán az egyed- tulajdonság- és kapcsolattípusainak meghatározását jelenti, hanem az ezek között létező összefüggések és korlátok feltérképezését is. Pl. egy kérdőíves felmérés adatbázisában a válaszadó életkora – nyilván – nem lehet negatív, és 120-nál nagyobb.

A fogalmi modell alapján lehetséges megkezdeni a logikai szintű tervezést. Ennek során vesszük figyelembe szükség szerint a technikai, hatékonysági és hozzáférési követelményeket, illetve korlátokat. Azonosításra, és - a kompatibilitás és szabványosság követelményének szem előtt tartása mellett - beépítésre kerülnek a kezelő korlátai és az általa elvárt extra szolgáltatások. Ugyancsak itt lesz szempont az adatvédelem és a hatékonyság is. Az előző példát tovább vizsgálva a logikai tervezés feladata annak meghatározása, hogy adott lekérdezés esetén ezt a lejárat dátumot milyen formában szükséges megjeleníteni.

A fizikai szintű tervezés a logikai tervre épül, ennek során határozzuk meg az adatoknak a tárolón való elhelyezkedésének, hozzáférésének és ábrázolásának (típusának és méreteinek) a rendjét. Példa: A fizikai tervezés feladata azt meghatározni, hogy a fentebb említett lejárat dátumot hogyan és milyen módon tároljuk: mondjuk fix 8 karakter hosszúságú szöveggént, 'ééééhhnn' formában, vagy UNIX-időbélyegként, vagy a kezelő beépített dátum típusát használjuk.

A valóság előbb-utóbb változni fog, s ez érinti eredeti adatmodellünket. Figyelembe véve, hogy a Halassy által megkülönböztetett három szint egymásra épül, a legszembe-tűnőbb előnye a változások hatásának kezelhetősége. A fizikai szintet érintő változások nem érintik a logikai szintű tervet, és még kevésbé a fogalmi modellt, ennél fogva esetenként jelentős munka takarítható meg.

Ha példabeli élelmiszercikkeinket jövőre külföldi piacon is értékesítjük, szükséges lesz a lejárat időnek a nemzetközi (fordított) formában való nyomtatása. Nyilvánvaló, hogy ez a körülmény *nem* érinti a fogalmi modellt, hiszen a lejárat időt, mint lényeges tulaj-

donságot így is, úgy is tartalmazza a modell. Ellenben az adott körülmények közötti megjelenítés formája lehet eltérő.

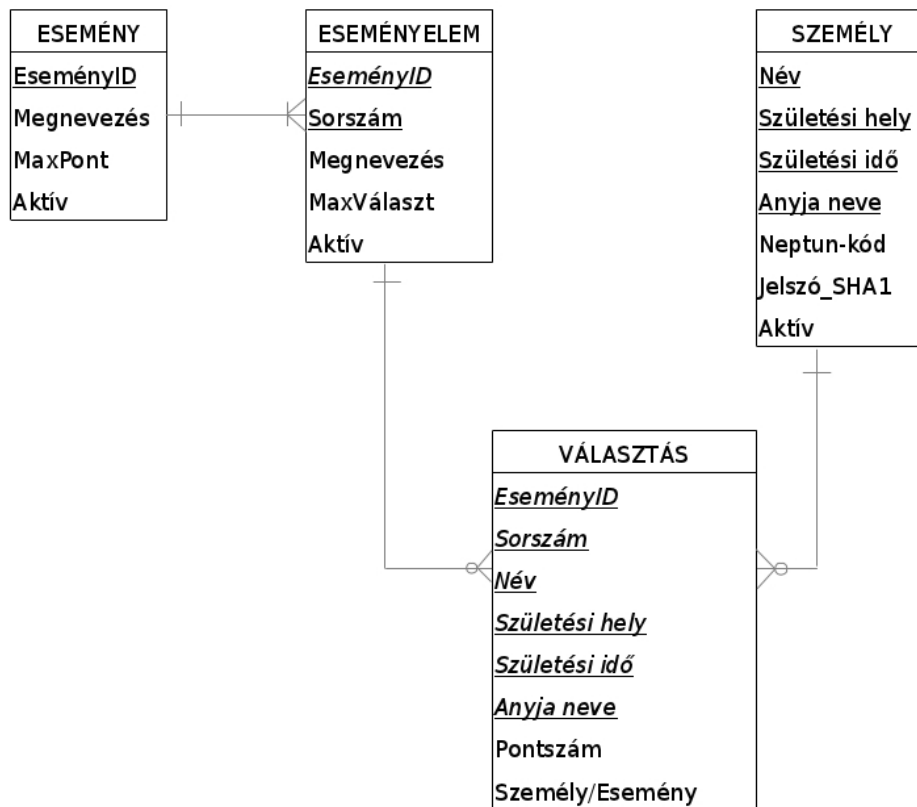
3.3.4. Hatékonyság

A hatékonyság (sebesség, illetve terhelhetőség) szempontját a fentiek szerint tehát a logikai szintű tervezésnél lehet, illetve kell érvényesíteni. A hatékonyság legelső biztosítója azonban a jó fogalmi modell, különös tekintettel a minimalitás (redundancia mentesség) követelményének teljesülésére. A redundancia ugyanis a feldolgozási idők igen jelentős mértékű növekedését okozhatja, vagy fordítva: a jó fogalmi szintű modellnek jelentős szerepe van a hatékonyság javulásában a redundáns változathoz képest. [140] [141] [142]

Egy adatbázis hatékonyságán, illetve terhelhetőségén – némiképp leegyszerűsített módon – két dolgot is érthetünk: egyrészt a tipikus lekérdezések válaszidőit, másfelől pedig az időegység alatt kiszolgált lekérdezések mennyiségét. A kettő nyilván nem független egymástól. A terhelhetőség, illetve a teljesítmény mérését, a modell jóságának alapvetően meghatározó szerepét vizsgálta Keszthelyi [143], [144] bemutatandó, hogy a modell jósága (valóság-hű, teljes, minimális) alapvetően, sőt elsődlegesen befolyásolja a hatékonyságot. Ezen mérések során pusztán az adatmodell jóságát (minimalitását) szem előtt tartó adatmodellre épülő teszt-adatbázison az akkori Neptunhoz képest a hatékonyság minőségi javulását lehetett kimutatni. Becslésem szerint a logikai szintű eszközök és módszerek alkalmazása csak kismértékű további javulást eredményezett volna, de ezen mérések elvégzésére nem került sor.

3.3.5. A konkrét feladat megtervezése

A konkrét feladatomhoz szükséges adatbázis ugyan kisméretű, a disszertációm elkészítéséhez szükséges igénybevétele sem jelentős, ennek ellenére a fentebb vázolt eljárást, a háromszintű tervezést alkalmazom Halassy nyomán, továbbá az általa következetesen megalapozott terminológiát használom. A fogalmi szintű modellben mindössze négy egyedtípus van: ESEMÉNY, ESEMÉNYELEM, SZEMÉLY, VÁLASZTÁS. Ezek és a kapcsolataik láthatóak a 17. ábrán.



17 ábra – A fogalmi szintű modell ábrája

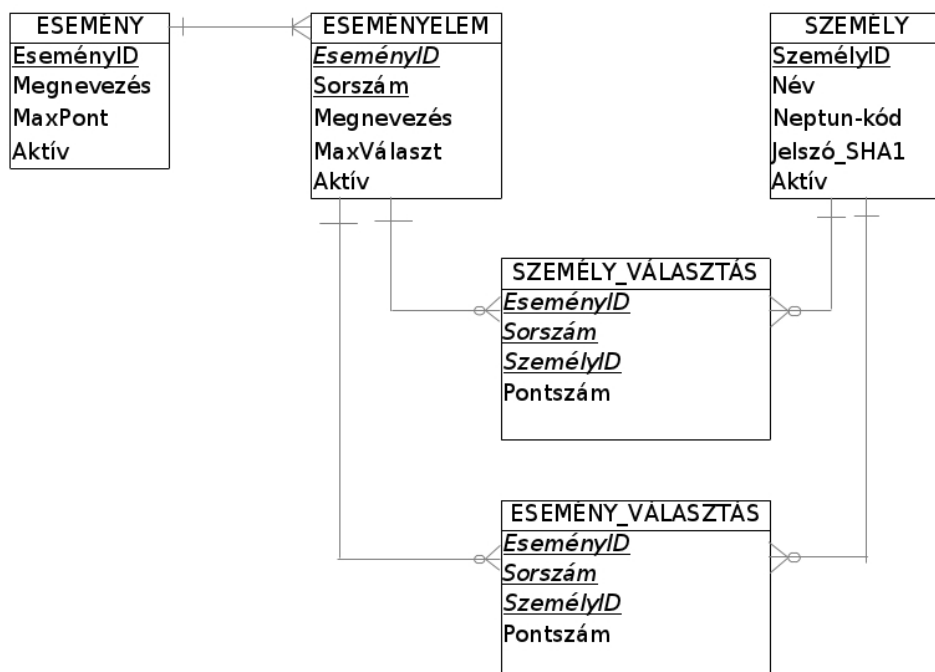
Az ESEMÉNY egyedtípus írja le a lehetséges kiválasztási/párosítási feladatokat, pl. Gazdasági informatika kurzusválasztás. A MaxPont a rangsorolásnál szétosztható maximális pontszámot tartalmazza. Az ESEMÉNYELEM az egyes eseményeken belüli ténylegesen választandó konkrét lehetőségeket tartalmazza (Hétfő 9.50, Hétfő 11.40, Kedd 13.30 stb.). A MaxVálaszt határozza meg, hogy az egyes lehetőségeket összesen és maximálisan hányan választhatják. A SZEMÉLY – értelemszerűen – a rangsorolási/párosítási műveletben részt vevő személyek alapadatait írja le. A VÁLASZTÁS pedig az egyes személyek által felállított eseményelem-rangsort írja le, illetve azt, hogy ez a személy rangsorolása, vagy pedig az eseményé. Elképzelhető ugyanis, hogy – fenti példánknál maradva – nemcsak a hallgatók adhatják meg az egyes kurzusokra vonatkozó preferenciájukat, hanem a kurzusok is rangsorolják valamilyen módon a hallgatókat (pl. korábbi tanulmányi eredmények alapján) (Személy/Esemény tulajdonságtípus).

Az „Aktív” tulajdonságtípus mutatja, hogy az adott elem megjelenik-e választhatóként, avagy sem. A Max.pont az adott elemre megítható legnagyobb pontszám.

A SZEMÉLY esetében van, történelmileg hagyományosan, természetes azonosító. Az ESEMÉNY esetében ilyen nincs, ezért ott mesterséges azonosítót alkalmazunk, generált sorszám formájában.

A kapcsolatok számossága és kötelezősége magától értetődő.

A logikai szintű tervezésnél (18. ábra) a fogalmi modellből indulunk ki. Technikai követelményekről és korlátokról nem beszélhetünk, a számunkra itt szükséges elemeket minden számításba jöhető adatbázis-kezelő ismeri és biztosítja. A felhasználás jellege okán különleges hatékonysági szempontokat sem szükséges figyelembe vennünk. A hozzáférési követelmények, illetve ezen belül adatbiztonsági szempontok miatt a VÁLASZTÁS-t felosztjuk SZEMÉLY_VÁLASZTÁS és ESEMÉNY_VÁLASZTÁS táblákra. A személyes adatok védelmének okán pedig a személy természetes azonosítója helyett egy ugyancsak általunk generált sorszámot fogunk használni, így nem lesz szükséges tárolnunk – a vizsgálat szempontjából fölöslegesen – az érzékenyebb személyes adatokat. A Neptun-kód a felhasználói név szerepét tölti be. A hallgatók amúgy is ismerik saját Neptun-kódjukat, így nincs szükség külön felhasználói nevek létrehozására, ami lényegesen egyszerűbbé teszi a munkafolyamatot.

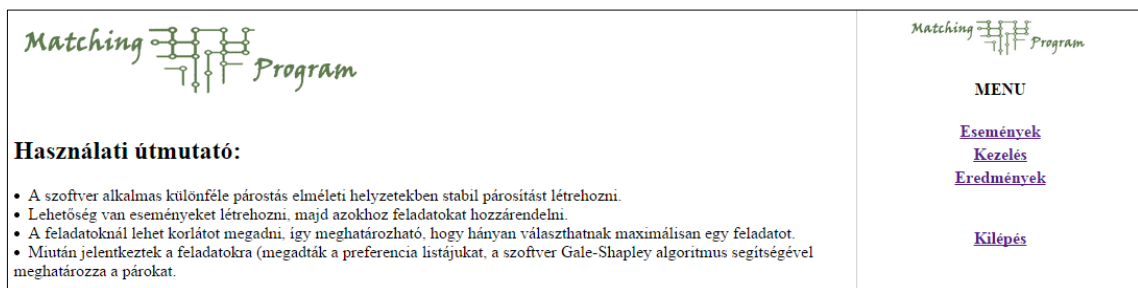


18 ábra – A logikai tervezés ábrája

A fizikai szintű tervezés során nehézség nem merül föl. Megjegyzendő, hogy a klasszikus logikai (boolean) adattípust az általam használt MySQL 5.1 változat nem ismeri. Ehelyett az egy bájtos előjel nélküli numerikus (tinyint) adattípust használom, 0 és 1 értékekkel, ami a gazdanyelvi környezet (PHP) szempontjából is előnyös.

3.4. A szoftver bemutatása

Az elkészült szoftver interneten keresztül érhető el, így gyakorlatilag bárhol használható, és nagyon könnyen testre szabható. A szoftvernek 2 fő része van. Az egyik főalkotó elem az úgynevezett adminisztrációs felület. Ez az a felület ahol be lehet állítani, hogy milyen párosításokat akarunk futtatni. Az adminisztrációs oldal kezdő képernyője a 19. ábrán látható:



19 ábra – az adminisztrációs oldal kezdő képernyője


Az ábra két részből áll, van egy menüpontrendszer, és van egy használati utasítás. A menüsor az végig látható, míg a fő ablakban a használati utasítás helyére kerülnek az általunk választott funkcióknak a kezelő felülete.

A menüpontok alapján 3 lehetőségünk van:

- 1), az első, hogy eseményeket adunk meg a rendszernek,
- 2) a második, hogy a létrehozott preferenciákhoz adunk meg másik oldali preferenciákat, vagy
- 3) használjuk a súlyozás útján megadottakat. Illetve meg tudjuk nézni a végeredményeket. A kezdőoldalon látható, hogy melyik részben mit tudunk tenni.

3.4.1. Az események meghatározása

A szoftverben van lehetőség úgynevezett események definiálására, illetve azokhoz válasszlehetőségek meghatározására, illetve ezek módosítására, törlésére. Csak az aktív események jelennek meg a felhasználói oldalon, ahol a felhasználók akár sorba rendezhetik, akár súlyozhatják az eseményekhez tartozó lehetőségeket. Ennek a szerkesztő oldalán látszik a 20. ábrán.

ESEMÉNYEK SZERKESZTÉSE				Matching  Program
Új esemény hozzáadása				
Controlling	Szerkesztés	Kezelés	Engedélyezés	MENU Események Kezelés Eredmények Kilépés
Gazdasági Informatika	Szerkesztés	Kezelés	Engedélyezés	
Erasmus 2013	Szerkesztés	Kezelés	Engedélyezés	
Vállalkozásgazdaságtan	Szerkesztés	Kezelés	Engedélyezés	
Szállítási feladat	Szerkesztés	Kezelés	Engedélyezés	
Organisational Behaviour	Szerkesztés	Kezelés	Tiltás	
tesztzt	Szerkesztés	Kezelés	Engedélyezés	
Hozzárendelési feladat	Szerkesztés	Kezelés	Engedélyezés	
Tig feladatok	Szerkesztés	Kezelés	Tiltás	

20 ábra – Az események szerkesztésének oldala

Ha szeretnénk új eseményt létrehozni, akkor ehhez tartozó űrlapon kell megadnunk a következő adatokat, az eseménynek a neve, illetve azt, hogy elérhető-e felhasználók számára. Ennek a kezelőfelülete látható a 21. ábrán:

ÚJ ESEMÉNY HOZZÁADÁSA	
NÉV:	<input type="text"/>
AKTÍV:	<input type="checkbox"/> igen <input type="checkbox"/> nem
<input type="button" value="ÚJ ESEMÉNY HOZZÁADÁSA"/>	

21 ábra - Új esemény hozzáadásának az űrlapja

Természetesen előfordulhat, hogy egy már létrehozott eseményt is át kell nevezni, ebben az esetben lehetőség van a szerkesztésére. A szerkesztő űrlap hasonló a létrehozó felülethez, itt is lehetőség van az esemény elérhetőségének a láthatóvá tételére. Ennek a kezelőfelülete látható a 22. ábrán:

ESEMÉNY MÓDOSÍTÁSA

NÉV:	<input type="text" value="Controlling"/>
LÁTHATÓ:	<input type="checkbox"/> Nem <input type="checkbox"/> Igen
<input type="button" value="ESEMÉNY MÓDOSÍTÁSA"/>	

22 ábra - Már létező esemény módosításának az űrlapja

Eseményt törölni nem lehet a rendszerből, de mint látható van lehetőség arra, hogy beálítsuk, hogy látható-e a felhasználók számára vagy sem. Ha letiltjuk, akkor a felhasználók nem fogják tudni kitölteni az adott eseményhez tartozó adatokat. Később természetesen bármikor tudjuk engedélyezni.

Természetesen minden eseményhez lehet meghatározni lehetőségeket, egy ilyen már létező eseményhez tartozó lehetőségeket láthatunk a 23 ábrán:

LEHETŐSÉGEK MENEDZSELÉSE

Típus feladatok

ÚJ LEHETŐSÉG HOZZÁADÁSA

Honlap fejlesztés	1	SZERKESZT	TILT
Honlap fordítás angol	1	SZERKESZT	TILT
Honlap fordítás német	1	SZERKESZT	ENGEDÉLYEZ
Honlap frissítés - aktualitásokkal való feltöltés	1	SZERKESZT	TILT
Cikk nyelvi lektorálás (angol)	2	SZERKESZT	TILT
Cikk nyelvi lektorálás (magyar)	2	SZERKESZT	TILT
Pályázat figyelés	1	SZERKESZT	TILT
Konferencia figyelés	1	SZERKESZT	TILT
TIG könyvtár feltöltése, levelezés kiadókkal	1	SZERKESZT	TILT
Konferencia regisztrációk kezelése	1	SZERKESZT	TILT
Adminisztráció	1	SZERKESZT	ENGEDÉLYEZ
Hajcsár-határidők figyelése, betartatása	1	SZERKESZT	TILT
Jegyzőkönyvvezető	1	SZERKESZT	TILT
Rendezvényszervező	1	SZERKESZT	TILT
Pályázat író	1	SZERKESZT	TILT
Fund raising	1	SZERKESZT	TILT

Matching Program

MENU

Események

Kezelés

Eredmények

Kilépés

23 ábra – Egy már létező esemény lehetőségei

3.4.2. Az események felhasználókhöz tartozó preferenciája

A rendszerben lehetőség van megtekinteni a létrehozott preferenciákat, és azokat módosítani, ha nem megfelelő számunkra, hogy a preferenciákat a felhasználók által megadott preferenciák súlyozása alapján számoljuk. Ezt tudjuk a következő pontban megtenni.

A 24. ábrán láthatjuk, hogy milyen a felhasználók, illetve az ehhez kapcsoló események preferenciája, illetve a felhasználók által megadott súlyok láthatóak a zárójelekben. A feladatok preferencia sorrendjét a felhasználók által megadott preferencia sorrendek alapján adjuk meg, természetesen van lehetőség ezen változtatni, ezért is lett az adatmodellezésben a választás egyedtípus szét lett szedve két külön egyedtípusra, az egyikben a felhasználók által meghatározott sorrend található, a másikban az ebből generált a feladatokhoz tartozó sorrend, és ha módosítjuk, akkor azt már újra nem lehet előállítani az eredeti adatokból. Hogy miért került az ugyanolyan adatokat tartalmazó 2 egyedtípus

külön, erre a választ az adatmodellezés bemutatásánál már megadtam (Lásd 3.3.5. fejezet)

A felhasználók preferencia sorrendje:

S1	C7(50)	C9(30)	C6(10)	C5(10)					
S2	C3(30)	C7(20)	C5(10)	C4(10)	C1(10)	C2(10)	C8(5)	C9(4)	C6(1)
S3	C5(50)	C9(50)	C7(0)	C8(0)	C6(0)	C3(0)	C1(0)	C2(0)	C4(0)
S4	C4(60)	C8(20)	C6(20)						
S5	C4(60)	C7(20)	C5(20)						
S6	C4(50)	C5(25)	C2(20)	C7(5)					
S7	C4(90)	C7(5)	C6(5)						
S8	C8(30)	C9(20)	C5(20)	C2(20)	C7(10)				
S9	C4(30)	C7(20)	C6(20)	C8(8)	C9(5)	C5(5)	C1(5)	C2(5)	C3(2)
S10	C3(60)	C8(10)	C2(10)						
S11	C8(60)	C7(20)	C4(20)						
S12	C4(16)	C3(15)	C2(15)	C5(11)	C9(10)	C7(10)	C1(8)	C6(8)	C8(7)
S13	C4(70)	C7(20)	C9(10)						
S14	C1(20)	C6(20)	C5(20)	C9(10)	C8(10)	C4(10)	C7(10)		
S15	C1(30)	C7(30)	C9(20)	C4(20)					

A feladatokpreferencia sorrendje:

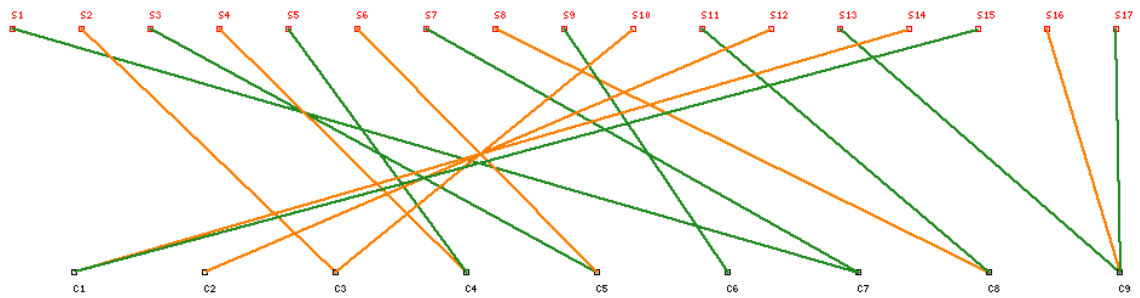
C7	2	S1(50)	S15(30)	S11(20)	S13(20)	S16(20)	S17(20)	S9(20)	S5(20)	S2(20)	S14(10)	S12(10)	S8(10)
S6	(5)	S7(5)	S3(0)										
C9	3	S3(50)	S16(30)	S1(30)	S17(25)	S15(20)	S8(20)	S14(10)	S12(10)	S13(10)	S9(5)	S2(4)	
C6	2	S9(20)	S14(20)	S4(20)	S17(10)	S1(10)	S16(10)	S12(8)	S7(5)	S2(1)	S3(0)		
C5	2	S3(50)	S6(25)	S14(20)	S8(20)	S5(20)	S12(11)	S17(10)	S2(10)	S1(10)	S16(5)	S9(5)	
C3	2	S10(60)	S2(30)	S12(15)	S17(5)	S16(5)	S9(2)	S3(0)					
C4	2	S7(90)	S13(70)	S5(60)	S4(60)	S6(50)	S9(30)	S15(20)	S11(20)	S12(16)	S16(10)	S2(10)	S17(10)
S14	(10)	S3(0)											
C1	3	S15(30)	S14(20)	S2(10)	S12(8)	S16(5)	S9(5)	S17(5)	S3(0)				
C2	2	S6(20)	S8(20)	S12(15)	S10(10)	S2(10)	S16(10)	S17(5)	S9(5)	S3(0)			
C8	3	S11(60)	S8(30)	S4(20)	S14(10)	S17(10)	S10(10)	S9(8)	S12(7)	S16(5)	S2(5)	S3(0)	

24 ábra – Az események felhasználókhöz és a feladatokhoz tartozó preferencia sorrendje

3.4.3. Az eredmények megtekintése

Miután a felhasználók megadták a saját preferencia sorrendjüket, eldönthetjük, hogy a rendszer a résztvevőket, illetve preferenciáikat tetszőleges sorrendben vizsgálja, vagy adminisztrátorként megadjuk a résztvevők vizsgálati sorrendjét. A sorba rendezés azért fontos, mert két teljesen egyforma preferenciával rendelkező individuum közül az először vizsgált (párosított) jobb (magasabb preferencia értékkel rendelkező) helyre fog kerülni, mint az algoritmus által másodjára párosított egyén. A résztvevők sorrendjének beállítása után megtekinthetők a párosítások.

A következő gráfon egy adott eseményhez tartozó eredményeket tudjuk megtekinteni.



25 ábra - Egy adott eseményhez tartozó párosítás grafikus ábrája

A felső sorban láthatóak a felhasználók, jelen esetben S1–S17, természetesen nagyon egyszerűen megjelenhet itt akár az azonosításra alkalmazott kód (egyetemi környezetben Neptun azonosító), vagy akár a felhasználók teljes neve is, ha szeretnénk látványosan megmutatni a párosítások végeredményét. Az alsó sorban pedig a feladatok láthatóak (C1 – C9) névvel.

Vállalati környezetben például fel lehet tenni a párosítás eredményét az intranetre, vagy a vállalati hirdetőre, így a munkavállalók mindig tisztában lehetnek azzal, hogy adott feladatok kinek a felelősségi körébe tartoznak – probléma esetén kihez kell fordulni. A másik módszer, amivel szemléltetni lehet az eredményeket az egy táblázatos módszer, ahol ugyanúgy C1–C9 jelöli a 9 feladatot, míg a hallgatókat az S1 – S17. Itt is lehetőség van a sorszámok helyett az azonosító vagy a teljes nevek megjelenítésére.

Gale-Shapley's algorithm (625)							
C1(50)	C2(15)	C3(90)	C4(160)	C5(75)	C7(70)	C8(110)	C9(55)
S14(1)	S12(3)	S2(1)	S7(1)	S3(1)	S1(1)	S4(2)	S16(1)
S15(1)		S10(1)	S13(1)	S6(2)	S9(2)	S8(1)	S17(1)
						S11(1)	

26 ábra – Egy adott eseményhez tartozó párosítás táblázatos ábrázolása

3.4.4. A felhasználói oldal

A felhasználói oldal ehhez képest sokkal egyszerűbb. A felhasználónak meg kell adnia egy azonosításra alkalmas adatot, (mint az egyetemen a Neptun azonosító, de bármilyen egyéb egyedi azonosító is megfelelő lehet rá). A szoftver egyelőre más adatot nem kér, későbbiekben lehet ezt másféle azonosítási móddal is megvalósítani. Ezek után, ha sikeresen bejelentkezett, akkor az aktív események közül választhat. A kiválasztott ese-

ménynél pedig vagy sorba rendezheti az aktuális lehetőségeket, vagy súlyozhatja is őket. A 27. ábrán látható a preferencia súlyozásának lehetősége.

Opera
matching theory
phd.syxthus.hu/mt/

Belépve: syxtus

Rendezzük sorba a feladatokat!

Válassz!

Rendezd sorba ezeket a feladatokat!

Rendezd sorba a számodra megfelelő feladatokat, és súlyozd őket, a súlyozott értékek összege nem lehet 100nál több. Nem kell az összes feladatot kiválasztanod, csak a számodra szimpatikusakat!

Leírás	Össz. hely	Súly
Honlap fejlesztés	[1]	100
Honlap fordítás angol	[1]	
Honlap fordítás német	[1]	
Honlap frissítés - aktualitásokkal való feltöltés	[1]	
Cikk nyelvi lektorálás (angol)	[2]	
Cikk nyelvi lektorálás (magyar)	[2]	
Pályázat figyelés	[1]	
Konferencia figyelés	[1]	
TIG könyvtár feltöltése, levelezés kiadókkal	[1]	
Konferencia regisztráció kezelése	[1]	
Adminisztráció	[1]	
Hajcsár-határidők figyelése, betartatása	[1]	
Jegyzőkönyvvezető	[1]	
Rendezvény szervező	[1]	
Pályázat író	[1]	
Fund raising	[1]	
<input type="button" value="Mentés"/>	Összesen[18]	Összesen: [100]

MENU
[Sorbarendeáz](#)
[Kilépés](#)

27 ábra – Egy adott eseményhez tartozó lehetőségek

A felhasználóknak nem kötelező minden lehetőséget értékelni, de ahogy az előző fejezetekben látható volt, minél több helyet ad meg a felhasználó, annál több az esélye arra, hogy bekerüljön valahova. Nagyon fontos előre tudatni, hogy a felhasználók által meghatározott súlyok alapján számoljuk a feladatokhoz tartozó értékeket, vagy az adminisztrációs oldalon az eseményt kezelő adja meg őket, mert más és más taktikára van szükség a felhasználók részéről. Ha a feladatokhoz tartozó preferencia sorrend nem kapcsolódik a felhasználók által meghatározott súlyokhoz, akkor a felhasználónak érdemes minél több helyet megadni, a súlyok értéke igazából mindegy, a lényeg csak a sorrenden van ebben az esetben. Ha viszont a preferencia sorrendeket a felhasználók által megadott súlyokból számolja a szoftver, akkor fontos, hogy milyen súlyokat adunk meg egy adott feladathoz, hiszen ha mások oda magasabbat adnak meg, akkor ők kerülnek előnyösebb helyzetbe. Ilyenkor szükség van a taktikázásra is.

A következő fejezetekben az elméleti részben bemutatott algoritmusok működését, működtethetőségét fogom vizsgálni először szoftveres támogatás nélkül, majd az általam

készített szoftver segítségével. Célom, hogy megmutassam, egyszerű döntési (párosítási) helyzetekben is szignifikánsan magasabb hasznosság érhető el a párosításelméleti algoritmusok segítségével. A bemutatott algoritmusok további pozitívuma, hogy az eredményükként létrejövő párosítások stabilak, tehát nem hozható létre olyan új megoldás, melyben akármelyik résztvevő is preferenciáinak jobban megfelelő párosításban venne részt anélkül, hogy valaki más érdekei sérülnének a változás következtében. A stabil párosítások tehát racionálisan indokolható rendszer szintű döntéseket eredményez, melyek növelik a szervezet, mint holisztikus egység stabilitását.

4. LÉTEZŐ PÁROSÍTÁSI HELYZETEK MEGVIZSGÁLÁSA

Egyetemi oktatóként számtalan olyan helyzettel találkoztam, ahol egy valójában párosítási feladatként is felírható problémát lényegében a véletlenre, illetve a gyorsaságra bízunk az érintettek preferenciáit figyelmen kívül hagyva, ezáltal a résztvevők megelégedettsége sem lehet igazán magas. Ezért számomra a legkézenfekvőbb megoldás az volt, hogy megvizsgáljam a környezetemben lévő különböző helyzeteket, ahol alkalmazhatónak tűntek a különböző párosítás elméleti algoritmusok. Ilyen helyzet lehet például az Erasmusra való jelentkezés, kurzus felvétel, órai feladatok kiosztása hallgatók számára. Ezek a mindennapi problémák olyanok, hogy megfelelnek a párosítás elméleti problémáknak és eddig nem alkalmaztunk rá semmilyen hatékony algoritmust, általában csak valamilyen olyan megoldási mód volt, ami minden helyzetben változott. Ebben a fejezetben ezeket a problémákat mutatom be, megvizsgálva az eddig alkalmazott módszereket, és bemutatva, hogy ha alkalmazunk rájuk párosítás elméleti algoritmusokat, akkor sokkal hatékonyabb megoldásokat kaphatunk.

Ezek a problémák, amiket itt bemutatok mind az *egy a többhöz* problémák, tehát Az első halmaz eleme a második halmazban csak egy elemhez tartozhat, de a második halmaz elemeihez több elem is tartozhat az első halmazból. A problémák által összegyűjtött adatokat, az általam fejlesztett „Backend” segítségével elemzem. Ennek a szoftvernek a lényeg, hogy szimpla szöveges file beolvasása utána képes megadni a párosításokat különböző algoritmusok segítségével. A szoftver felhasználhatóságát csökkenti, hogy a szöveges file előállítására nagyobb elemszám esetén bonyolult, ezért szükség van egy „Frontend” elkészítésére is majd.

4.1. Az Erasmusra való jelentkezés problémája ²

A probléma lényege a következő. Adottak a külföldi fogadó intézmények és adottak a jelentkező hallgatók. Minden jelentkező hallgatónak lehetősége van maximum 3 különböző helyet megadni, és ezeket sorba rendezni. (Ezt a korlátot az Erasmus iroda adja meg.) A hallgatókat az eddigi tanulmányi eredményük, az adott nyelvi tudásuk (ez min-

² publikálva: [152]

dig a fogadó intézménytől függ, de lehet például angol vagy német), illetve egyéb eredményeik alapján (ilyen lehet például a TDK eredmény) sorba rendezik. Így kapunk minden hallgatóhoz, illetve minden fogadó intézményhez egy külön preferencia sorrendet. A most alkalmazott algoritmus a már bemutatott Mohó algoritmushoz hasonló, anynyi különbséggel, hogy ha valaki mégsem kerül be a számára megfelelő helyre, akkor megkapja a lehetőséget, hogy egy még nem választott helyre mehessen. A cél, hogy minden jelentkezett hallgató kikerüljön külföldre (természetesen, ha elér egy olyan szintet, amely már elegendő lehet arra, hogy ő képviselhesse az egyetemünket egy külföldi egyetemen), és amíg több hely van, mint jelentkező addig számunkra is fontos, hogy minden lehetséges helyet betöltsünk.

A kutatásom célja itt az volt, hogy megvizsgáljam, hogy egy általam ismert esetben ahol tudom, hogy a párok meghatározása nem túl egyszerű, lehetőség lenne-e egy sokkal jobb megoldásra, esetleg a megoldás lehetne jóval egyszerűbb. Amíg néhány fő esetén könnyű fejben is meghatározni a párosításokat, itt közel 30 hallgatónál (és ez csak egy adott karhoz tartozó hallgatók) már ez sokkal nehezebben megy.

Azt vizsgáltam, hogyha a hallgatók egy szoftver segítségével online tudnak jelentkezni, akkor miután mindenki megadta a preferencia sorrendjét, egyből lenne lehetőség a kész párosítás megnézésére, illetve értékelésére, nem lenne szükség továbbiakban semmilyen egyéb matematikai műveletre.

A következő táblázatban láthatóak a hallgatók (az anonimitás biztosításáért H1-H28) és a fogadó intézmények (E1-E26) preferencia sorrendjei.

Hallgatói preferenciák:

H1:	E15	E16	E12	H15:	E1	E7	E23
H2:	E12	E17	E10	H16:	E7	E2	E3
H3:	E11	E15	E12	H17:	E16	E7	E10
H4:	E7	E14	E9	H18:	E23	E25	E22
H5:	E24	E23	E20	H19:	E23	E25	E22
H6:	E25	E23	E8	H20:	E17	E26	E12
H7:	E1	E6	E7	H21:	E7	E23	E1
H8:	E4	E5	E26	H22:	E12	E16	E19
H9:	E21	E20		H23:	E7	E1	E2
H10:	E24	E20	E25	H24:	E11	E19	E17
H11:	E7	E3	E8	H25:	E18	E13	
H12:	E1	E21	E7	H26:	E18	E13	
H13:	E11	E19	E17	H27:	E18	E13	

H14: E11 E19 E17

H28: E11 E9 E7

Egyetemi preferenciák:

E1 (2):	H7	H21	H15	H23	H12					
E2 (2):	H16	H23								
E3 (2):	H16	H11								
E4 (2):	H8									
E5 (3):	H8									
E6 (2):	H7									
E7 (3):	H17	H7	H16	H21	H11	H4	H15	H28	H23	H12
E8 (2):	H6	H11								
E9 (2):	H4	H28								
E10 (1):	H17	H2								
E11 (2):	H3	H14	H24	H13	H28					
E12 (3):	H3	H20	H22	H1	H2					
E13 (4):	H27	H26	H25							
E14 (5):	H4									
E15 (2):	H3	H1								
E16 (2):	H17	H22	H1							
E17 (6):	H20	H14	H24	H13	H2					
E18 (2):	H27	H26	H25							
E19 (4):	H22	H14	H24	H13						
E20 (2):	H10	H9	H5							
E21 (2):	H9	H12								
E22 (3):	H18	H19								
E23 (2):	H6	H18	H19	H21	H15	H5				
E24 (2):	H10	H5								
E25 (2):	H10	H6	H18	H19						
E26 (2):	H8	H20								

28 táblázat - Hallgatói és fogadó intézményi preferencia sorrendek

Tehát a hivatalosan is alkalmazott algoritmus lényege, hogy nem veszi figyelembe az egyetemek preferenciáját, és ha egy hallgató egyszer hozzá lett rendelve az intézményhez (párt alkotott vele), akkor az a párosítás már nem változtatható, végleges. Így hiába kerülne oda egy egyetem számára kedvezőbb hallgató a következő lépésben, annak a hallgatónak már másik egyetemet kell választania. Ha az algoritmusból néhány lépést kiemelünk, akkor talán jobban megérthető a futása. Első lépésben, amíg van elég hely az egyetemeken, addig a hallgatók az általuk első helyen megadott egyetemre kerülnek be. Ez látható H1-H13 hallgatók esetén. Az első problémás hallgató az H14, aki az E11 egyetemet jelölte meg elsőként. Az E11 egyetemre maximálisan 2 hallgató juthat ki, ez pedig már a H3, és a H13, így a H14-nek a preferenciája alapján a következő egyetemre kell mennie, ez lesz az E19. Látható a táblázatban, hogy ha a preferenciákat is figyelembe vennénk, akkor a H14 hallgatót kellene választania az E11 egyetemnek a H13 helyett.

A következő érdekes helyzet a H21-es hallgatóé. Nála a három, általa megadott egyetem közül már egyik sem rendelkezik üres hellyel, így ez a hallgató már nem juthat ki külföldre. Ilyenkor lehetőség van utólagos javításra, abban az esetben, ha van még szabad egyetemi hely. A táblázatban pirossal jelöltem a párosított egyetem/hallgató párosokat, húztam azokat, akiket nem lehet választani, és azokat az egyetemeket, hallgatókat, akikre nem került sor az algoritmus futása során, mert már megkaptuk a párosítást, azok maradtak változatlanok.

A 29. táblázatban látható a most alkalmazott algoritmus segítségével milyen párosítások jöttek létre.

Hallgatói preferenciák:

H1:	E15	E16	E12	H15:	E4	E7	E23
H2:	E12	E17	E10	H16:	E7	E2	E3
H3:	E11	E15	E12	H17:	E16	E7	E10
H4:	E7	E14	E9	H18:	E23	E25	E22
H5:	E24	E23	E20	H19:	E23	E25	E22
H6:	E25	E23	E8	H20:	E17	E26	E12
H7:	E1	E6	E7	H21:	E7	E23	E4
H8:	E4	E5	E26	H22:	E12	E16	E19
H9:	E21	E20		H23:	E7	E4	E2
H10:	E24	E20	E25	H24:	E44	E19	E17
H11:	E7	E3	E8	H25:	E18	E13	
H12:	E1	E21	E7	H26:	E18	E13	
H13:	E11	E19	E17	H27:	E48	E13	
H14:	E44	E19	E17	H28:	E11	E9	E7

Egyetemi preferenciák:

E1 (2):	H7	H21	H15	H23	H12					
E2 (2):	H16	H23								
E3 (2):	H16	H11								
E4 (2):	H8									
E5 (3):	H8									
E6 (2):	H7									
E7 (3):	H17	H7	H16	H21	H11	H4	H15	H28	H23	H12
E8 (2):	H6	H11								
E9 (2):	H4	H28								
E10 (1):	H17	H2								
E11 (2):	H3	H14	H24	H13	H28					
E12 (3):	H3	H20	H22	H1	H2					
E13 (4):	H27	H26	H25							
E14 (5):	H4									
E15 (2):	H3	H1								
E16 (2):	H17	H22	H1							
E17 (6):	H20	H14	H24	H13	H2					
E18 (2):	H27	H26	H25							
E19 (4):	H22	H14	H24	H13						

E20 (2):	H10	H9	H5			
E21 (2):	H9	H12				
E22 (3):	H18	H19				
E23 (2):	H6	H18	H19	H21	H15	H5
E24 (2):	H10	H5				
E25 (2):	H10	H6	H18	H19		
E26 (2):	H8	H20				

29 táblázat- A jelenleg alkalmazott párosítás eredménye

Természetesen a szakirodalom szerint létezik jobb (hatékonyabb) és stabil párosítást adó megoldás is, erre használható például a Gale-Shapley algoritmus is.

Ennek az algoritmusnak fontos része, hogy már figyelembe veszi az egyetemek (hallgatók, ha egyetemek irányából futtatjuk) rangsorát is, így, ha egy adott egyetemre jobb hallgató jelentkezik a későbbiekben, akkor az eredetileg ott lévő hallgató új egyetem után nézhet. A lépések során így nem végleges, hanem csak ideiglenes párosítások jönnek létre, amelyek csak akkor kerülnek végleges elfogadásra, ha az algoritmus befejezte a futását. Nézzünk itt is néhány érdekesebb lépést. Az első tizenhárom hallgató esetében a lépések megegyeznek az előző részben vázolt lépésekkel, minden hallgató a saját listájának élen szereplő egyetemet választja, egészen addig, amíg van ott elég hely. Az első problémás hallgató a H14-es, aki az E11-re adja be a jelentkezését, előző példában láthattuk, hogy mivel ott nem vettük figyelembe a preferenciákat, és a hallgatói helyek száma megtelt az adott egyetemen, ezért a H14-es hallgatónak új hely iránt kellett néznie, ebben az algoritmusban az egyetem dönthet és döntenie is kell (hasznosságmaximalizálás), így a számára értékesebb hallgatót választja. Így a H13-as hallgatónak új hely után kell néznie, vagyis a listáján következő helyre fogja beadni.

A táblázatban itt is pirossal jelöltem a párosított egyetem/hallgató párosokat, át húztam azt, akit nem lehet választani, és azok az egyetemek, hallgatók, akikre nem került sor az algoritmus futása során, mert már megkaptuk a párosítást, azok maradtak változatlanok. Azok a párosítások, amelyek ideiglenesen léteztek csak, és nem lettek véglegesítve, azokat pirossal és áthúzva jelölöm.

A Gale-Shapley algoritmus hallgatóbarát eredménye látható a 30. táblázatban:

Hallgatói preferenciák:

H1:	E15	E16	E12	H15:	E1	E7	E23
H2:	E12	E17	E10	H16:	E7	E2	E3
H3:	E11	E15	E12	H17:	E16	E7	E10
H4:	E7	E14	E9	H18:	E23	E25	E22
H5:	E24	E23	E20	H19:	E23	E25	E22

H6:	E25	E23	E8	H20:	E17	E26	E12
H7:	E1	E6	E7	H21:	E7	E23	E1
H8:	E4	E5	E26	H22:	E12	E16	E19
H9:	E21	E20		H23:	E7	E4	E2
H10:	E24	E20	E25	H24:	E44	E19	E17
H11:	E7	E3	E8	H25:	E18	E13	
H12:	E4	E21	E7	H26:	E18	E13	
H13:	E44	E19	E17	H27:	E18	E13	
H14:	E11	E19	E17	H28:	E44	E9	E7

Egyetemi preferenciák

E1 (2):	H7	H21	H15	H23	H12				
E2 (2):	H16	H23							
E3 (2):	H16	H11							
E4 (2):	H8								
E5 (3):	H8								
E6 (2):	H7								
E7 (3):	H17	H7	H16	H21	H11	H4	H15	H28	H23 H12
E8 (2):	H6	H11							
E9 (2):	H4	H28							
E10 (1):	H17	H2							
E11 (2):	H3	H14	H24	H13	H28				
E12 (3):	H3	H20	H22	H1	H2				
E13 (4):	H27	H26	H25						
E14 (5):	H4								
E15 (2):	H3	H1							
E16 (2):	H17	H22	H1						
E17 (6):	H20	H14	H24	H13	H2				
E18 (2):	H27	H26	H25						
E19 (4):	H22	H14	H24	H13					
E20 (2):	H10	H9	H5						
E21 (2):	H9	H12							
E22 (3):	H18	H19							
E23 (2):	H6	H18	H19	H21	H15	H5			
E24 (2):	H10	H5							
E25 (2):	H10	H6	H18	H19					
E26 (2):	H8	H20							

30 táblázat – A hallgatóbarát Gale-Shapley algoritmus eredménye

Illetve a Gale-Shapley algoritmus egyetembarát eredménye látható a következő táblázatban:

Hallgatói preferenciák:

H1:	E15	E16	E12	H15:	E1	E7	E23
H2:	E12	E17	E10	H16:	E7	E2	E3
H3:	E11	E15	E12	H17:	E16	E7	E10
H4:	E7	E14	E9	H18:	E23	E25	E22
H5:	E24	E23	E20	H19:	E23	E25	E22
H6:	E25	E23	E8	H20:	E17	E26	E12
H7:	E1	E6	E7	H21:	E7	E23	E4
H8:	E4	E5	E26	H22:	E12	E16	E19
H9:	E21	E20		H23:	E7	E1	E2

H10:	E24	E20	E25	H24:	E11	E19	E17
H11:	E7	E3	E8	H25:	E18	E13	
H12:	E1	E21	E7	H26:	E18	E13	
H13:	E11	E19	E17	H27:	E18	E13	
H14:	E11	E19	E17	H28:	E11	E9	E7

Egyetemi preferenciák

E1 (2):	H7	H21	H15	H23	H12					
E2 (2):	H16	H23								
E3 (2):	H16	H11								
E4 (2):	H8									
E5 (3):	H8									
E6 (2):	H7									
E7 (3):	H17	H7	H16	H21	H11	H4	H15	H28	H23	H12
E8 (2):	H6	H11								
E9 (2):	H4	H28								
E10 (1):	H17	H2								
E11 (2):	H3	H14	H24	H13	H28					
E12 (3):	H3	H20	H22	H1	H2					
E13 (4):	H27	H26	H25							
E14 (5):	H4									
E15 (2):	H3	H1								
E16 (2):	H17	H22	H1							
E17 (6):	H20	H14	H24	H13	H2					
E18 (2):	H27	H26	H25							
E19 (4):	H22	H14	H24	H13						
E20 (2):	H10	H9	H5							
E21 (2):	H9	H12								
E22 (3):	H18	H19								
E23 (2):	H6	H18	H19	H21	H15	H5				
E24 (2):	H10	H5								
E25 (2):	H10	H6	H18	H19						
E26 (2):	H8	H20								

31 táblázat – Az egyetembarát Gale-Shapley algoritmus eredménye

Látható, hogy akár az egyetemek, akár a hallgatók irányából futtatjuk az algoritmust, ugyanazt a megoldást kapjuk, ahogy Roth és Peranson [86] is bemutatta, sokkal nagyobb játékos szám esetén is csak kis különbség várható.

Konklúzió

Amikor a hallgatók szeretnék egyetemi tanulmányaik alatt külföldre menni, akkor legtöbbször nem az egyetem az elsődleges választási szempont, hanem, hogy milyen nyelvterületre mennének, és hogy mindenképpen szeretnék kijutni. Az eddigi többlépcsős, részben Mohó algoritmusra épülő párosító módszert érdemes lecserélni bármelyik más párosító módszerre, hiszen a Gale és Shapley által meghatározott algoritmus képes stabil párosítást adni, és ha egy adott hallgató mégsem jutna ki az általa meghatározott helyre, mindaddig, amíg több hely van, mint jelentkező, van esélye, hogy találjon

magának másik, számára megfelelő helyet. Természetesen azt a problémát, hogy valaki nem talál magának helyet, mert nem tud elég hosszú preferencialistát megadni, meg lehetne előzni, ha felemelnénk esetleg eltörölnénk a hallgatók jelentkezési limitjét. Nagy előny lenne egy ilyen helyzetben, ha hallgatók a választásukat interneten egy regisztrációs űrlapon keresztül tehetnék meg, aminek a célja nem csak az lenne, hogy az adatok digitálisan a rendelkezésünkre állnának, hanem az is, hogy pár másodperccel az utolsó megadott preferencia sorrend után lehet párokat alkotni. Szoftveresen a párok létrehozása gyakorlatilag automatikusan egyből elkészül, míg a mostani megoldás esetén nagyon sok idő megy az az úgynevezett „sakkozással”.

A megkapott eredményeken már utána lehet javítani, ha esetleg van olyan, ahol szükség lenne rá, de így is rengeteg munkát, és ez által időt lehet megtakarítani.

4.2. KURZUS FELVÉTELI PROBLÉMA ³

A következő probléma esetén már nem adtam meg külön preferenciákat a hallgatókra, illetve a kurzusokra, hanem a kurzusok preferenciáját az alapján határoztam meg, hogy a hallgatók mennyit hajlandóak rá áldozni.

Ebben a kutatásban a céloom már hármas volt. Először is azt akartam megvizsgálni, hogy ha egy nagyobb elemszámú halmazokban alkotok párokat, akkor milyen eltérések várhatóak az eredményben az előző kutatáshoz képest. A második cél az volt, hogy olyan kutatást keressek, ami jobban modellezi a valóságot, mert nagyon sok párosításméleti probléma esetén nem igazán létezik mindkét félnek preferenciája. A harmadik céloom az volt, hogy megvizsgáljam, hogy miként is lehetne értékelni, és ezáltal összehasonlítani a különböző algoritmusokat, ennek kiemelt szerepe van abban, hogy elkészült szoftverben melyik algoritmust érdemes alkalmazni.

A kutatás folyamán a hallgatók által megadott adatokat már nem kézzel vizsgáltam, hanem egy egyszerű elkészített szoftver segítségével, ami képes volt egy egyszerű szöveges formában (csv) megadott adatokat értelmezni, majd belőlük eredményeket számolni. Természetesen ez a későbbiekben elkészülő, már felhasználók által is alkalmazott szoftvernek az úgynevezett „backend”-je lesz.

³ publikálva [155]

Számszerűsítve: van 260 hallgató, akik 9 kurzus valamelyikére szeretnének jelentkezni. Minden kurzusra maximálisan 30 hallgató kerülhet be, így 270 hely van a hallgatóknak. Ebben az esetben minden hallgatónak lehetősége van felvenni az egy kurzust, maximum nem arra a kurzusra kerül, ahova szeretne menni.

Egyetemünkön a hallgatók minden félévben online, egy hallgatói információs rendszer segítségével veszik fel a tárgyaikat, a jelentkezés sorrendjében kerülnek be a kurzusokra. A tárgyak felvételéhez annyi kurzust hirdetünk meg, hogy minden hallgató beférjen. Így a kurzusválasztás csak arról szól, hogy ki, mikor jut számítógéphez, és hogyan tud előbb bejelentkezni egy adott időpontra. Ez a megoldás nem optimális, sokkal inkább csak arról szól, hogy kinek van szerencséje, és sokan elfoglalhatnak olyan helyeket másoktól, amelyek számukra kevésbé fontos lenne, míg a másik számára az lenne az egyetlen lehetőség. Az 1. ábrán bemutatok egy kurzus felvételt, ahol 9 kurzusra jelentkezett a 260 hallgató. Minden kurzus alából 30 fős, de az oktató engedélyezte néhány kurzus 31-32 főre való kibővítését.

Az első órákon a hallgatók megadhattak saját preferencia sorrendet a különböző kurzusokra, meghatározva, hogy melyik kurzusra szeretnének inkább járni, ezek az értékek láthatóak a hallgatók sorszama után zárójelben. A preferencia sorrend megadása közben azt kértem a hallgatóktól, hogy ne csak sorrendet állítsanak fel, hanem értékeljék is ezeket a kurzusokat, így lehetőségük volt 1000 pontot szétosztani a 9 kurzus között. Nem volt kötelező mindegyik kurzusra pontot adni, de a szétosztott pontok összegének 1000-nek kellett lennie. A kurzusok sorszama melletti összeg a hallgatók által az adott helyre megadott pontszámok összege. Így az alap szétosztás alapján a pontok összege, amelyet a kurzusok megkaptak, **127.927** volt. Az eredeti párosítás látható az 29. táblázatban. A hallgatók neve után látható zárójelben, hogy a saját preferencia sorrendjében hányadik helyen van az aktuális kurzus.

C1 (19380)	C2 (6325)	C3 (3270)	C4 (13900)	C5 (20600)	C6 (18935)	C7 (23600)	C8 (18834)	C9 (30830)
S13 (1)	S1 (2)	S12 (2)	S5 (1)	S3 (-)	S4 (1)	S2 (1)	S14 (1)	S69 (1)
S29 (1)	S6 (3)	S24 (-)	S10 (1)	S17 (1)	S9 (1)	S7 (1)	S18 (1)	S76 (-)
S37 (1)	S8 (7)	S26 (2)	S15 (1)	S22 (-)	S11 (1)	S16 (1)	S21 (1)	S81 (5)
S50 (1)	S20 (9)	S27 (2)	S25 (-)	S36 (1)	S41 (1)	S19 (1)	S34 (2)	S98 (4)
S57 (1)	S42 (1)	S28 (-)	S52 (1)	S39 (-)	S45 (1)	S23 (1)	S38 (1)	S101 (3)
S60 (8)	S54 (3)	S32 (7)	S61 (1)	S44 (1)	S49 (1)	S30 (1)	S51 (1)	S155 (-)
S77 (2)	S63 (-)	S33 (-)	S70 (-)	S47 (1)	S62 (-)	S31 (1)	S56 (1)	S168 (1)
S84 (9)	S78 (2)	S43 (1)	S71 (1)	S48 (1)	S64 (-)	S35 (1)	S72 (1)	S203 (3)

S90 (-)	S79 (1)	S46 (2)	S74 (8)	S55 (1)	S67 (3)	S40 (1)	S83 (1)	S216 (-)
S106 (1)	S80 (1)	S82 (-)	S96 (8)	S58 (5)	S68 (1)	S53 (2)	S89 (1)	S222 (-)
S112 (1)	S91 (2)	S86 (-)	S108 (-)	S65 (2)	S73 (1)	S59 (1)	S92 (2)	S223 (1)
S116 (1)	S95 (4)	S88 (7)	S110 (-)	S102 (1)	S85 (1)	S66 (2)	S97 (-)	S226 (2)
S127 (1)	S114 (7)	S115 (2)	S120 (2)	S129 (1)	S93 (1)	S75 (1)	S107 (-)	S259 (1)
S135 (1)	S117 (2)	S121 (1)	S131 (2)	S139 (1)	S99 (2)	S87 (1)	S111 (-)	
S146 (1)	S118 (7)	S123 (-)	S132 (1)	S154 (-)	S103 (9)	S94 (1)	S119 (1)	
S165 (-)	S124 (8)	S130 (-)	S143 (-)	S158 (1)	S109 (1)	S100 (-)	S122 (3)	
S166 (1)	S126 (-)	S144 (-)	S145 (1)	S161 (1)	S125 (1)	S104 (1)	S128 (1)	
S176 (1)	S133 (1)	S147 (3)	S148 (1)	S174 (1)	S134 (-)	S105 (-)	S136 (1)	
S177 (1)	S138 (-)	S149 (-)	S153 (1)	S181 (1)	S137 (-)	S113 (1)	S140 (1)	
S179 (1)	S141 (1)	S173 (2)	S156 (1)	S183 (1)	S142 (1)	S160 (-)	S157 (1)	
S189 (1)	S150 (-)	S182 (-)	S167 (-)	S186 (1)	S164 (-)	S162 (1)	S159 (1)	
S192 (1)	S151 (1)	S197 (-)	S180 (1)	S188 (2)	S178 (1)	S172 (1)	S163 (1)	
S209 (1)	S152 (6)	S198 (-)	S187 (1)	S195 (1)	S233 (-)	S184 (1)	S169 (1)	
S215 (1)	S185 (2)	S207 (-)	S202 (-)	S201 (1)	S234 (1)	S191 (1)	S170 (2)	
S220 (1)	S194 (-)	S211 (-)	S204 (1)	S205 (1)	S237 (-)	S193 (1)	S171 (1)	
S221 (1)	S196 (-)	S241 (-)	S213 (1)	S210 (1)	S242 (1)	S199 (-)	S175 (1)	
S225 (1)	S218 (-)	S243 (-)	S214 (1)	S212 (-)	S245 (1)	S200 (-)	S190 (2)	
S236 (1)	S229 (2)	S249 (-)	S217 (1)	S219 (1)	S246 (1)	S232 (1)	S206 (3)	
S244 (1)	S230 (2)	S254 (3)	S224 (1)	S227 (-)	S248 (1)	S240 (1)	S208 (-)	
S252 (1)	S231 (2)	S256 (-)	S238 (3)	S228 (1)	S253 (1)	S250 (2)	S235 (-)	
S257 (-)	S239 (6)			S251 (1)	S260 (1)	S255 (1)	S247 (-)	
						S258 (1)		

32 táblázat – Az eredeti jelentkezés eredménye

Az 32. táblázatban látható, hogy több olyan hallgató is van, aki csak a sokadik helyen megadott kurzusra kerülhetett be, mivel valószínűleg lassabb volt a jelentkezéskor, mint a többiek. Ezért több módszert is kipróbáltam, amelyek a hallgatók szempontjából optimális (vagy legalább elfogadhatóbb) megoldást adhatnak. Sajnos a hallgatók egy része nem tudott bejárni órára, így 260 hallgatóból végül csak 197-en adtak meg preferencia sorrendet. Első lépésben a hallgatói preferencia sorrendek segítségével meghatároztam a kurzusok sorrendjeit is. Aki többet ígért egy adott kurzusra, az többet ér a kurzus számára, így a kurzusok szempontjából törekedtem a haszon maximalizálásra, vagyis, hogy olyan hallgatókat válogatni a kurzusokra, akik többet „fizetnének” érte.

4.2.1. Gale-Shapley algoritmus

Ez az algoritmus csak annyiban tér el az eredeti Gale-Shapley algoritmustól, hogy a kurzusok nem csak egy hallgatót fogadhatnak, hanem van egy 1-nél nagyobb keretszámuk. Nézzük meg az adott feladatot ezzel az algoritmussal:

Ha csak a 197 hallgató adatait használjuk és az általuk megadott preferencia sorrenddel, és az eredetileg meghatározott 30 fős kurzusokkal számolok, akkor a következő párosítást kapom, ami a 33. táblázatban látható. Az táblázatokban félkövérrel jelölöm azokat a hallgatókat, akik arra helyre kerültek be, ahol eredetileg is tanultak, és aláhúzással azokat a hallgatókat, akik adtak meg preferencia sorrendet.

C1 (25080)	C2 (8350)	C3 (3883)	C4 (16100)	C5 (26099)	C6 (22518)	C7 (27800)	C8 (20509)	C9 (6550)
<u>S1 (1)</u>	<u>S12 (1)</u>	<u>S29 (2)</u>	S5 (1)	S17 (1)	S4 (1)	S7 (1)	<u>S2 (2)</u>	<u>S13 (2)</u>
<u>S6 (1)</u>	<u>S26 (1)</u>	S81 (2)	S10 (1)	<u>S20 (1)</u>	S9 (1)	S16 (1)	S14 (1)	S69 (1)
<u>S8 (1)</u>	<u>S27 (1)</u>	<u>S96 (1)</u>	S15 (1)	S36 (1)	S11 (1)	S19 (1)	S18 (1)	<u>S115 (1)</u>
S37 (1)	S42 (1)	<u>S103 (2)</u>	S52 (1)	S44 (1)	S41 (1)	S23 (1)	S21 (1)	<u>S147 (1)</u>
S50 (1)	<u>S74 (1)</u>	<u>S120 (1)</u>	<u>S60 (1)</u>	S47 (1)	S45 (1)	S30 (1)	S38 (1)	S155 (3)
<u>S54 (1)</u>	<u>S77 (1)</u>	S121 (1)	S61 (1)	S48 (1)	S49 (1)	S31 (1)	S51 (1)	S168 (1)
S57 (1)	S79 (1)	<u>S209 (2)</u>	S71 (1)	S55 (1)	<u>S53 (1)</u>	<u>S34 (1)</u>	S56 (1)	<u>S206 (1)</u>
<u>S78 (1)</u>	S80 (1)		<u>S84 (1)</u>	<u>S67 (1)</u>	<u>S65 (1)</u>	S35 (1)	<u>S58 (1)</u>	S223 (1)
<u>S91 (1)</u>	S133 (1)		S131 (2)	<u>S81 (1)</u>	S68 (1)	S40 (1)	<u>S66 (1)</u>	<u>S238 (2)</u>
<u>S95 (1)</u>	S141 (1)		S132 (1)	<u>S98 (1)</u>	S73 (1)	<u>S46 (1)</u>	S72 (1)	<u>S250 (1)</u>
S106 (1)	S151 (1)		S145 (1)	<u>S99 (1)</u>	S85 (1)	S59 (1)	S83 (1)	S259 (1)
S112 (1)	<u>S215 (2)</u>		S148 (1)	S102 (1)	<u>S88 (1)</u>	S75 (1)	S89 (1)	
<u>S114 (1)</u>	<u>S225 (2)</u>		S153 (1)	<u>S122 (1)</u>	S93 (1)	S87 (1)	S119 (1)	
S116 (1)	<u>S254 (1)</u>		S156 (1)	<u>S124 (1)</u>	S109 (1)	<u>S92 (1)</u>	S128 (1)	
<u>S117 (1)</u>			<u>S173 (1)</u>	S129 (1)	<u>S118 (2)</u>	S94 (1)	S136 (1)	
S127 (1)			S180 (1)	S139 (1)	S125 (1)	<u>S101 (1)</u>	S140 (1)	
S135 (1)			S187 (1)	S158 (1)	S142 (1)	S104 (1)	<u>S152 (2)</u>	
S146 (1)			S204 (1)	S161 (1)	<u>S177 (3)</u>	S113 (1)	S157 (1)	
S166 (1)			S213 (1)	S174 (1)	S178 (1)	S162 (1)	S159 (1)	
S176 (1)			S214 (1)	S181 (1)	<u>S203 (1)</u>	<u>S170 (1)</u>	S163 (1)	
S179 (1)			S217 (1)	S183 (1)	<u>S226 (1)</u>	S172 (1)	S169 (1)	
<u>S185 (1)</u>			S224 (1)	S186 (1)	S234 (1)	S184 (1)	S171 (1)	
S189 (1)			<u>S230 (1)</u>	S188 (2)	S242 (1)	S191 (1)	S175 (1)	
S192 (1)				S195 (1)	S245 (1)	S193 (1)	S190 (2)	
S220 (1)				S201 (1)	S246 (1)	<u>S231 (1)</u>		
S221 (1)				S205 (1)	S248 (1)	S232 (1)		
<u>S229 (1)</u>				S210 (1)	S253 (1)	<u>S239 (1)</u>		
S236 (1)				S219 (1)	S260 (1)	S240 (1)		
S244 (1)				S228 (1)		S255 (1)		
S252 (1)				S251 (1)		S258 (1)		

33 táblázat – A párosítás eredménye csak a sorrendet megadó hallgatók alapján

Minden hallgató bekerült valamelyik kurzusra, és nem volt olyan hallgató, aki ne került volna be legalább az általa második helyre sorolt kurzusra. De mivel félév elején még nem lehet tudni, hogy kik nem járnak, ezért azokat a hallgatókat is szét lehet osztani a kurzusok üres helyein. (Nekik én határoztam meg egy véletlen preferencia sorrendet a

kurzusokra (1 és 10 közötti véletlen számok segítségével), Természetesen lehetőség lett volna őket szimplán csak az üres helyekre is betenni. Az így kapott plusz preferenciákkal kapott sorrend látható a 34. táblázatban. Míg az eredeti szétosztás alapján a 197 hallgató összesen **127.927** pontot költött volna a kurzus választásra, ebben az esetben már a megoldás **159.972** pont lenne, ami nagyjából 25%-os haszon növekedés a kurzus számára.

C1 (25080)	C2 (8350)	C3 (3883)	C4 (16100)	C5 (26099)	C6 (22518)	C7 (27800)	C8 (20509)	C9 (6550)
<u>S1 (1)</u>	S3 (6)	S22 (7)	S5 (1)	S17 (1)	S4 (1)	S7 (1)	<u>S2 (2)</u>	<u>S13 (2)</u>
<u>S6 (1)</u>	<u>S12 (1)</u>	S24 (7)	S10 (1)	<u>S20 (1)</u>	S9 (1)	S16 (1)	S14 (1)	S69 (1)
<u>S8 (1)</u>	<u>S26 (1)</u>	S25 (7)	S15 (1)	S36 (1)	S11 (1)	S19 (1)	S18 (1)	<u>S115 (1)</u>
S37 (1)	<u>S27 (1)</u>	S28 (7)	S52 (1)	S44 (1)	S41 (1)	S23 (1)	S21 (1)	S143 (3)
S50 (1)	S33 (6)	<u>S29 (2)</u>	<u>S60 (1)</u>	S47 (1)	S45 (1)	S30 (1)	<u>S32 (3)</u>	S144 (3)
<u>S54 (1)</u>	S39 (6)	S43 (1)	S61 (1)	S48 (1)	S49 (1)	S31 (1)	S38 (1)	<u>S147 (1)</u>
S57 (1)	S42 (1)	S76 (7)	S71 (1)	S55 (1)	<u>S53 (1)</u>	<u>S34 (1)</u>	S51 (1)	S149 (3)
<u>S78 (1)</u>	S62 (6)	S82 (7)	<u>S84 (1)</u>	<u>S67 (1)</u>	<u>S65 (1)</u>	S35 (1)	S56 (1)	S150 (3)
<u>S91 (1)</u>	S63 (6)	S86 (7)	S131 (2)	<u>S81 (1)</u>	S68 (1)	S40 (1)	<u>S58 (1)</u>	S154 (3)
<u>S95 (1)</u>	S64 (6)	S90 (7)	S132 (1)	<u>S98 (1)</u>	S73 (1)	<u>S46 (1)</u>	<u>S66 (1)</u>	S155 (3)
S106 (1)	S70 (6)	<u>S96 (1)</u>	S145 (1)	<u>S99 (1)</u>	S85 (1)	S59 (1)	S72 (1)	S168 (1)
S112 (1)	<u>S74 (1)</u>	S97 (7)	S148 (1)	S102 (1)	<u>S88 (1)</u>	S75 (1)	S83 (1)	S182 (3)
<u>S114 (1)</u>	<u>S77 (1)</u>	S100 (7)	S153 (1)	<u>S122 (1)</u>	S93 (1)	S87 (1)	S89 (1)	S199 (3)
S116 (1)	S79 (1)	<u>S103 (2)</u>	S156 (1)	<u>S124 (1)</u>	S109 (1)	<u>S92 (1)</u>	S119 (1)	S202 (3)
<u>S117 (1)</u>	S80 (1)	S105 (7)	<u>S173 (1)</u>	S129 (1)	<u>S118 (2)</u>	S94 (1)	S128 (1)	<u>S206 (1)</u>
S127 (1)	S130 (6)	S107 (7)	S180 (1)	S139 (1)	S125 (1)	<u>S101 (1)</u>	S136 (1)	S222 (3)
S135 (1)	S133 (1)	S108 (7)	S187 (1)	S158 (1)	S142 (1)	S104 (1)	S140 (1)	S223 (1)
S146 (1)	S141 (1)	S110 (7)	S204 (1)	S161 (1)	<u>S177 (3)</u>	S113 (1)	<u>S152 (2)</u>	S227 (3)
S166 (1)	S151 (1)	S111 (7)	S213 (1)	S174 (1)	S178 (1)	S162 (1)	S157 (1)	S233 (3)
S176 (1)	S196 (6)	<u>S120 (1)</u>	S214 (1)	S181 (1)	S198 (4)	<u>S170 (1)</u>	S159 (1)	S235 (3)
S179 (1)	S197 (6)	S121 (1)	S217 (1)	S183 (1)	S200 (4)	S172 (1)	S160 (2)	S237 (3)
<u>S185 (1)</u>	S207 (6)	S123 (7)	S224 (1)	S186 (1)	<u>S203 (1)</u>	S184 (1)	S163 (1)	<u>S238 (2)</u>
S189 (1)	S208 (6)	S126 (7)	<u>S230 (1)</u>	S188 (2)	<u>S226 (1)</u>	S191 (1)	S164 (2)	S241 (3)
S192 (1)	S211 (6)	S134 (7)		S195 (1)	S234 (1)	S193 (1)	S165 (2)	S243 (3)
S220 (1)	S212 (6)	S137 (7)		S201 (1)	S242 (1)	<u>S231 (1)</u>	S167 (2)	S247 (3)
S221 (1)	<u>S215 (2)</u>	S138 (7)		S205 (1)	S245 (1)	S232 (1)	S169 (1)	S249 (3)
<u>S229 (1)</u>	S216 (6)	<u>S209 (2)</u>		S210 (1)	S246 (1)	<u>S239 (1)</u>	S171 (1)	<u>S250 (1)</u>
S236 (1)	S218 (6)			S219 (1)	S248 (1)	S240 (1)	S175 (1)	S256 (3)
S244 (1)	<u>S225 (2)</u>			S228 (1)	S253 (1)	S255 (1)	S190 (2)	S257 (3)
S252 (1)	<u>S254 (1)</u>			S251 (1)	S260 (1)	S258 (1)	S194 (2)	S259 (1)

34 táblázat – A párosítás eredménye minden hallgató alapján

Mind a két táblázatban (33., 34.) aláhúzással jelöltem azokat a hallgatókat, akik megadták a preferencia sorrendjüket. Látható, hogy azon hallgatók számára, akik megadták a preferenciájukat, nincs változás a 34 táblázatban, csak bekerült a többi hallgató is a kur-

zusokra. A táblázatban félkövérrel jelöltem azokat a hallgatókat, akik ugyanazt a csoportot kapják, mint amit eredetileg megadtak, a többi új helyre került. Azoknál, akiknek nem volt előre meghatározott sorrendjük ott nem igazán számít, hogy hova is kerül. Azoknál, akik alá vannak húzva, de nem félkövérek (ők azok, akik adtak meg elvárást előre, de máshol vannak, mint ahova hivatalosan felvették), ott egyértelműen azért van eltérés, mert a preferencia-sorrendjük eltért az előzetes tárgyfelvételtől. Számukra ez jobb párosítást adott.

4.2.2. Boston mechanizmus

Azáltal, hogy a hallgatók nem csak preferencia sorrendet adnak meg az egyes kurzusokra, hanem megadják azt is, hogy mennyire fontos számukra, van lehetőség egy másik módszert is alkalmazni.

Ebben a módszerben először a hallgatókat az első helyes jelentkezésük alapján rakjuk be egy kurzusra. Attól hogy valaki bekerült már egy kurzusra, a helye még nem végleges, mert kieshet, ha nála magasabb preferencia pontszámú hallgató jelenik meg. A kurzusok irányából itt is a haszon maximalizálása a cél. Első lépésben itt is csak azokkal a hallgatókkal foglalkozok, akik adtak meg preferencia sorrendet (35 táblázat), majd utána azokat a hallgatókat is felveszem a kurzusra, akik nem határoztak meg sorrendet (36. táblázat).

Jól látható az 35. táblázatban, hogy az összes hallgató itt is bekerül vagy az első vagy az általa megadott második kurzusra. Nincs szignifikáns eltérés a Gale-Shapley algoritmushoz képest, alig néhány hallgató került csak másik kurzusra. Ilyen például az S81-es hallgató, aki így bekerült az első helyre, pedig volt olyan hallgató, akinek a második helyes pontszáma magasabb volt nála.

Az összebevétel, mivel kardinális preferenciákkal dolgozunk, a kurzusok szempontjából pedig 156.489, ami alig tér el a Gale-Shapley algoritmus által meghatározott párosítás eredményétől.

Természetesen - amennyiben a hallgatók előre tisztában vannak az algoritmusok működésével -, akkor előfordulhat, hogy máshogy osztják ki a pontokat, mert például a Boston algoritmus esetén érdemes a leginkább vágyott helynek adni a maximális értéket.

C1	C2	C3	C4	C5	C6	C7	C8	C9
----	----	----	----	----	----	----	----	----

(25080)	(8350)	(3583)	(16100)	(25999)	(22518)	(27800)	(20509)	(6550)
<u>S1 (1)</u>	<u>S12 (1)</u>	<u>S29 (2)</u>	<u>S5 (1)</u>	<u>S17 (1)</u>	<u>S4 (1)</u>	<u>S7 (1)</u>	<u>S2 (2)</u>	<u>S13 (2)</u>
<u>S6 (1)</u>	<u>S26 (1)</u>	<u>S43 (1)</u>	<u>S10 (1)</u>	<u>S20 (1)</u>	<u>S9 (1)</u>	<u>S16 (1)</u>	<u>S14 (1)</u>	<u>S69 (1)</u>
<u>S8 (1)</u>	<u>S27 (1)</u>	<u>S96 (1)</u>	<u>S15 (1)</u>	<u>S32 (1)</u>	<u>S11 (1)</u>	<u>S19 (1)</u>	<u>S18 (1)</u>	<u>S115 (1)</u>
<u>S37 (1)</u>	<u>S42 (1)</u>	<u>S103 (2)</u>	<u>S52 (1)</u>	<u>S36 (1)</u>	<u>S41 (1)</u>	<u>S23 (1)</u>	<u>S21 (1)</u>	<u>S147 (1)</u>
<u>S50 (1)</u>	<u>S74 (1)</u>	<u>S120 (1)</u>	<u>S60 (1)</u>	<u>S44 (1)</u>	<u>S45 (1)</u>	<u>S30 (1)</u>	<u>S38 (1)</u>	<u>S168 (1)</u>
<u>S54 (1)</u>	<u>S77 (1)</u>	<u>S121 (1)</u>	<u>S61 (1)</u>	<u>S47 (1)</u>	<u>S49 (1)</u>	<u>S31 (1)</u>	<u>S51 (1)</u>	<u>S206 (1)</u>
<u>S57 (1)</u>	<u>S79 (1)</u>	<u>S209 (2)</u>	<u>S71 (1)</u>	<u>S48 (1)</u>	<u>S53 (1)</u>	<u>S34 (1)</u>	<u>S56 (1)</u>	<u>S223 (1)</u>
<u>S78 (1)</u>	<u>S80 (1)</u>		<u>S84 (1)</u>	<u>S55 (1)</u>	<u>S65 (1)</u>	<u>S35 (1)</u>	<u>S58 (1)</u>	<u>S238 (2)</u>
<u>S91 (1)</u>	<u>S133 (1)</u>		<u>S131 (2)</u>	<u>S67 (1)</u>	<u>S68 (1)</u>	<u>S40 (1)</u>	<u>S66 (1)</u>	<u>S250 (1)</u>
<u>S95 (1)</u>	<u>S141 (1)</u>		<u>S132 (1)</u>	<u>S81 (1)</u>	<u>S73 (1)</u>	<u>S46 (1)</u>	<u>S72 (1)</u>	<u>S259 (1)</u>
<u>S106 (1)</u>	<u>S151 (1)</u>		<u>S145 (1)</u>	<u>S98 (1)</u>	<u>S85 (1)</u>	<u>S59 (1)</u>	<u>S83 (1)</u>	
<u>S112 (1)</u>	<u>S215 (2)</u>		<u>S148 (1)</u>	<u>S99 (1)</u>	<u>S88 (1)</u>	<u>S75 (1)</u>	<u>S89 (1)</u>	
<u>S114 (1)</u>	<u>S225 (2)</u>		<u>S153 (1)</u>	<u>S102 (1)</u>	<u>S93 (1)</u>	<u>S87 (1)</u>	<u>S119 (1)</u>	
<u>S116 (1)</u>	<u>S254 (1)</u>		<u>S156 (1)</u>	<u>S122 (1)</u>	<u>S109 (1)</u>	<u>S92 (1)</u>	<u>S128 (1)</u>	
<u>S117 (1)</u>			<u>S173 (1)</u>	<u>S124 (1)</u>	<u>S118 (2)</u>	<u>S94 (1)</u>	<u>S136 (1)</u>	
<u>S127 (1)</u>			<u>S180 (1)</u>	<u>S129 (1)</u>	<u>S125 (1)</u>	<u>S101 (1)</u>	<u>S140 (1)</u>	
<u>S135 (1)</u>			<u>S187 (1)</u>	<u>S139 (1)</u>	<u>S142 (1)</u>	<u>S104 (1)</u>	<u>S152 (2)</u>	
<u>S146 (1)</u>			<u>S204 (1)</u>	<u>S158 (1)</u>	<u>S177 (3)</u>	<u>S113 (1)</u>	<u>S157 (1)</u>	
<u>S166 (1)</u>			<u>S213 (1)</u>	<u>S161 (1)</u>	<u>S178 (1)</u>	<u>S162 (1)</u>	<u>S159 (1)</u>	
<u>S176 (1)</u>			<u>S214 (1)</u>	<u>S174 (1)</u>	<u>S203 (1)</u>	<u>S170 (1)</u>	<u>S163 (1)</u>	
<u>S179 (1)</u>			<u>S217 (1)</u>	<u>S181 (1)</u>	<u>S226 (1)</u>	<u>S172 (1)</u>	<u>S169 (1)</u>	
<u>S185 (1)</u>			<u>S224 (1)</u>	<u>S183 (1)</u>	<u>S234 (1)</u>	<u>S184 (1)</u>	<u>S171 (1)</u>	
<u>S189 (1)</u>			<u>S230 (1)</u>	<u>S186 (1)</u>	<u>S242 (1)</u>	<u>S191 (1)</u>	<u>S175 (1)</u>	
<u>S192 (1)</u>				<u>S195 (1)</u>	<u>S245 (1)</u>	<u>S193 (1)</u>	<u>S190 (2)</u>	
<u>S220 (1)</u>				<u>S201 (1)</u>	<u>S246 (1)</u>	<u>S231 (1)</u>		
<u>S221 (1)</u>				<u>S205 (1)</u>	<u>S248 (1)</u>	<u>S232 (1)</u>		
<u>S229 (1)</u>				<u>S210 (1)</u>	<u>S253 (1)</u>	<u>S239 (1)</u>		
<u>S236 (1)</u>				<u>S219 (1)</u>	<u>S260 (1)</u>	<u>S240 (1)</u>		
<u>S244 (1)</u>				<u>S228 (1)</u>		<u>S255 (1)</u>		
<u>S252 (1)</u>				<u>S251 (1)</u>		<u>S258 (1)</u>		

35 táblázat – Párosítás csak a sorrendet megadó hallgatók alapján

Természetesen itt is megtörtént a többi hallgató hozzárendelése is maradék helyekre, amely lényegében nem változtatja meg az értékeket.(36. táblázat)

C1 (25080)	C2 (8350)	C3 (3583)	C4 (16100)	C5 (25999)	C6 (22518)	C7 (27800)	C8 (20509)	C9 (6550)
<u>S1 (1)</u>	S3 (6)	S22 (7)	<u>S5 (1)</u>	<u>S17 (1)</u>	<u>S4 (1)</u>	<u>S7 (1)</u>	<u>S2 (2)</u>	<u>S13 (2)</u>
<u>S6 (1)</u>	<u>S12 (1)</u>	<u>S24 (7)</u>	<u>S10 (1)</u>	<u>S20 (1)</u>	<u>S9 (1)</u>	<u>S16 (1)</u>	<u>S14 (1)</u>	<u>S69 (1)</u>
<u>S8 (1)</u>	<u>S26 (1)</u>	S25 (7)	<u>S15 (1)</u>	<u>S32 (1)</u>	<u>S11 (1)</u>	<u>S19 (1)</u>	<u>S18 (1)</u>	<u>S115 (1)</u>
<u>S37 (1)</u>	<u>S27 (1)</u>	<u>S28 (7)</u>	<u>S52 (1)</u>	<u>S36 (1)</u>	<u>S41 (1)</u>	<u>S23 (1)</u>	<u>S21 (1)</u>	<u>S143 (3)</u>
<u>S50 (1)</u>	S33 (6)	<u>S29 (2)</u>	<u>S60 (1)</u>	<u>S44 (1)</u>	<u>S45 (1)</u>	<u>S30 (1)</u>	<u>S38 (1)</u>	<u>S144 (3)</u>
<u>S54 (1)</u>	S39 (6)	<u>S43 (1)</u>	<u>S61 (1)</u>	<u>S47 (1)</u>	<u>S49 (1)</u>	<u>S31 (1)</u>	<u>S51 (1)</u>	<u>S147 (1)</u>
<u>S57 (1)</u>	<u>S42 (1)</u>	S76 (7)	<u>S71 (1)</u>	<u>S48 (1)</u>	<u>S53 (1)</u>	<u>S34 (1)</u>	<u>S56 (1)</u>	<u>S149 (3)</u>

<u>S78 (1)</u>	S62 (6)	S82 (7)	<u>S84 (1)</u>	S55 (1)	<u>S65 (1)</u>	S35 (1)	<u>S58 (1)</u>	S150 (3)
<u>S91 (1)</u>	S63 (6)	S86 (7)	S131 (2)	<u>S67 (1)</u>	S68 (1)	S40 (1)	<u>S66 (1)</u>	S154 (3)
<u>S95 (1)</u>	S64 (6)	S90 (7)	S132 (1)	<u>S81 (1)</u>	S73 (1)	<u>S46 (1)</u>	S72 (1)	S155 (3)
S106 (1)	S70 (6)	<u>S96 (1)</u>	S145 (1)	<u>S98 (1)</u>	S85 (1)	S59 (1)	S83 (1)	S168 (1)
S112 (1)	<u>S74 (1)</u>	S97 (7)	S148 (1)	<u>S99 (1)</u>	<u>S88 (1)</u>	S75 (1)	S89 (1)	S182 (3)
<u>S114 (1)</u>	<u>S77 (1)</u>	S100 (7)	S153 (1)	S102 (1)	S93 (1)	S87 (1)	S119 (1)	S199 (3)
S116 (1)	S79 (1)	<u>S103 (2)</u>	S156 (1)	<u>S122 (1)</u>	S109 (1)	<u>S92 (1)</u>	S128 (1)	S202 (3)
<u>S117 (1)</u>	S80 (1)	S105 (7)	<u>S173 (1)</u>	<u>S124 (1)</u>	<u>S118 (2)</u>	S94 (1)	S136 (1)	<u>S206 (1)</u>
S127 (1)	S130 (6)	S107 (7)	S180 (1)	S129 (1)	S125 (1)	<u>S101 (1)</u>	S140 (1)	S222 (3)
S135 (1)	S133 (1)	S108 (7)	S187 (1)	S139 (1)	S142 (1)	S104 (1)	<u>S152 (2)</u>	S223 (1)
S146 (1)	S141 (1)	S110 (7)	S204 (1)	S158 (1)	<u>S177 (3)</u>	S113 (1)	S157 (1)	S227 (3)
S166 (1)	S151 (1)	S111 (7)	S213 (1)	S161 (1)	S178 (1)	S162 (1)	S159 (1)	S233 (3)
S176 (1)	<u>S188 (5)</u>	<u>S120 (1)</u>	S214 (1)	S174 (1)	S198 (4)	<u>S170 (1)</u>	S160 (2)	S235 (3)
S179 (1)	S197 (6)	S121 (1)	S217 (1)	S181 (1)	S200 (4)	S172 (1)	S163 (1)	S237 (3)
<u>S185 (1)</u>	S207 (6)	S123 (7)	S224 (1)	S183 (1)	<u>S203 (1)</u>	S184 (1)	S164 (2)	<u>S238 (2)</u>
S189 (1)	S208 (6)	S126 (7)	<u>S230 (1)</u>	S186 (1)	<u>S226 (1)</u>	S191 (1)	S165 (2)	S241 (3)
S192 (1)	S211 (6)	S134 (7)		S195 (1)	S234 (1)	S193 (1)	S167 (2)	S243 (3)
S220 (1)	S212 (6)	S137 (7)		S201 (1)	S242 (1)	<u>S231 (1)</u>	S169 (1)	S247 (3)
S221 (1)	<u>S215 (2)</u>	S138 (7)		S205 (1)	S245 (1)	S232 (1)	S171 (1)	S249 (3)
<u>S229 (1)</u>	S216 (6)	<u>S209 (2)</u>		S210 (1)	S246 (1)	<u>S239 (1)</u>	S175 (1)	<u>S250 (1)</u>
S236 (1)	S218 (6)			S219 (1)	S248 (1)	S240 (1)	S190 (2)	S256 (3)
S244 (1)	<u>S225 (2)</u>			S228 (1)	S253 (1)	S255 (1)	S194 (2)	S257 (3)
S252 (1)	<u>S254 (1)</u>			S251 (1)	S260 (1)	S258 (1)	S196 (2)	S259 (1)

36 táblázat – A párosítás eredménye minden hallgatóra Boston mechanizmus segítségével

Egy hatékonyabb párosítás elérése érdekében megoldási lehetőség lenne, ha hallgatók jelentkezésük esetén több kurzust is választhatnának (preferencia sorrenddel), illetve bekerülne a kurzusfelvételhez várólista. Ebben az esetben minden hallgató ténylegesen arra a kurzusra kerülhetne be, amely számára a legmegfelelőbb lenne.

4.2.3. Konklúzió

Amikor a hallgatók szeretnének kurzusokat felvenni, akkor legtöbbször a szerencsén és a gyorsaságon múlik csak, hogy bekerülnek-e az általuk választott kurzusra. Nincs lehetőségük egy tárgy kurzusai esetén sorrendet megadni, így ha nem jut be oda, ahova először szeretett volna, akkor lehet, hogy csak a sokadik hely marad számára. Ezért az eddig használt párosító módszert érdemes lecserélni bármelyik más párosító módszerre, hiszen a dolgozatban bemutatott *Gale és Shapley* által meghatározott algoritmus vagy akár a *Bostoni mechanizmus* is képes stabil párosítást adni, és ha egy adott hallgató mégsem jutna be az általa először választott helyre, még mindig lenne lehetősége, hogy a saját rangsora alapján bekerüljön egy általa preferált helyre. Mint látható ebben a

problémában, a kurzusokra nem volt külön meghatározva a hallgatókra preferencia sorrend, azokat a hallgatók által megadott súlyokból hoztuk létre, mivel az eredeti tárgyfelvétel esetén is csak annyi a feltétel egy hallgató felé, hogy legyen meg a tárgy előkövetelménye. Ebben az esetben azt vizsgáltam, hogy a különböző módszerek alapján milyen összesített hasznosságot kap a kurzus.

Ennek a helyzetnek az egyik problémája, hogy nehéz lenne a létező egységesített tanulmányi rendszerbe való beillesztése. A másik probléma, hogy a hallgatók egyszerre nem csak egy tárgyat tanulnak, így olyan feltételeket is meg kell adni, hogy ha adott hallgató egyik tárgy esetén bekerül egy adott kurzusra, akkor a többi tárgy esetén hogyan változik a többi tárgy esetében a preferencia sorrendje. Sajnos, ez alapján gyakorlatilag várhatóan egy végtelen körbe kerülnénk, ha folyamatosan változnának a különböző tárgyakon a preferencia sorrendek. Ezáltal ebben az esetben nincs stabil megoldás. Mindemellett a kutatás eredménye az volt, hogy ha találunk független eseményeket, akkor nagyon hatékony megoldásokat lehet alkotni.

A kutatásban összehasonlítottam a különböző algoritmusokat, és a három vizsgált algoritmus esetében igazából nagy eltérés csak az Mohó algoritmus és a másik kettő között volt. Ez az eredmény is azt bizonyította, hogy érdemesebb mindképpen valamilyen, más párosításelméleti algoritmust alkalmazni a Mohó algoritmus helyett.

4.3. ÓRAI FELADATRA VALÓ JELENTKEZÉS

PROBLÉMÁJA ⁴

Az előző kutatásokból megállapítható volt, hogy a léteznek olyan problémák az egyetemeken, ahol sokkal hatékonyabban lehetne alkalmazni a különböző párosítási algoritmusokat, mint ahogy most próbálunk párokat létrehozni, így továbbvizsgáltam, hogy ezeket a párokat miként lehet még hatékonyabban létrehozni. Megvizsgáltam, hogy milyen különbségek adódnak, ha a hallgatók minden lehetőséget sorba rendezhetnek, vagy csak egy adott darabszámú lehetőséget választhatnak, illetve, ha van két párhuzamos kurzus, amely ugyanazt kínálja, akkor azok eredményeit hogyan lehet egymással cserélni, és ebből különböző új adatokat meghatározni. Ebben a kutatásban már nem csak az volt a célom, hogy adott helyzetben megvizsgáljam az algoritmusokat, és bizo-

⁴ publikálva [157], [153], [158]

nyítsam, hogy hatékonyabb megoldásokat is lehetne alkalmazni, hanem az is, hogy ha van két párhuzamos kurzusom, akkor van-e lehetőség közöttük hallgatókat cserélni, és ezáltal jobb eredményeket kapunk-e. A másik cél az volt, hogy megvizsgáljam, ha a hallgatók számára van korlát, hogy hány feladatot választhatnak, akkor az mennyiben befolyásolja a végső párosítási eredményeket. A hipotézisem az volt, hogy ha lehetőséget adunk a hallgatóknak, akkor ők több helyet fognak kihasználni, és ez által jobb megoldásokat kaphatunk. Megvizsgáltam a hallgatók racionalitását is, mivel a hallgatók legfontosabb célja, hogy teljesítsék a tárgyat, ehhez pedig követelmény a feladatválasztás, így fontos lenne részükről, hogy biztosan kapjanak feladatot. Minden félévben előfordul, hogy a hallgatók kapnak, választhatnak valamilyen feladatot, amelyet beleszámítunk a féléves eredménybe. Ilyen lehet például kiselőadás tartása egy adott témából, vagy valamilyen órai feladat elvégzése, beadandó választás, és számos egyéb feladat. Ezek kiosztása általában kétféle módon történik: vagy az oktató önhatalmúlag kiosztja őket, vagy a hallgatók választanak, ez pedig gyakorlatilag egy Mohó algoritmusnak felel meg, ahol mindenki csak abból választ, ami megmaradt számára a többiek után. Aki az elején tud választani, annak több lehetősége van, aki a végén, annak kevesebb. Az Óbudai Egyetemen az aktuális félévben egy adott tárgyból 2 kurzuson is oktatok. A hallgatók száma mindkét kurzuson 20-20 fő. A félév folyamán minden hallgatónak kell csinálnia egy feladatot, amit az egyik órán be kell mutatnia. Mivel a félévben nagyjából 10 olyan hét van, ahol a hallgatók prezentálhatnak, így minden héten 2-2 hallgató adhat elő. A félév első óráján van a hallgatóknak lehetőségük jelentkezni a féléves feladatokra. Különböző módszerek segítségével jelentkezhetnek. Az alapszabvány esetén minden hallgató kimegy a táblához, érkezési sorrendben, és valamelyik héthez felírhatja a nevét, ahol még van üres hely. Ebben az esetben a hallgatók egymás után választhatnak a szabad hetek között, így aki a végére marad, annak már nincs lehetősége választani. Ezért is találtam ki egy másik módszert. Minden hallgató értékelheti a heteket, preferencia sorrend megadásával, úgy, hogy még súlyoznia is lehet őket. A súlyokat úgy kell elosztaniuk a hetek között, hogy az összegük maximálisan 100 lehet. Így van lehetőség a különböző párosításelméleti algoritmusok futtatására. A hallgatók nem ismerik előre a párosítás elméleti módszereket, főleg, hogy minden módszer esetén más és más stratégiát lenne érdemes alkalmazni. Mindössze annyit tudnak, hogy érdemes minél több helyet megadni, mert aki nem kerül be egyik hétre sem, az nem tudja majd teljesíteni azt a feladatot. A hallgatói preferencia sorrend az 37 táblázatban látható (S1-20 jelöli a hallgatókat, C0-9 a feladatokat)

	1	2	3	4	5	6	7	8	9	10
S1	C4	C7	C8	C6						
S2	C6	C7	C2	C8	C5	C4				
S3	C9	C8	C7							
S4	C4	C9	C5	C8						
S5	C9	C5	C4							
S6	C1	C7	C3	C5						
S7	C7	C4								
S8	C7	C8								
S9	C5	C3	C0							
S10	C7	C9	C6							
S11	C6	C5	C3	C8						
S12	C4	C1	C3	C2	C8	C6				
S13	C8	C2	C1	C0						
S14	C5	C4	C7	C8	C6	C1	C3	C2		
S15	C2	C5	C6	C8	C4	C1	C0	C3		
S16	C7	C3	C2							
S17	C1	C7	C8	C9	C6	C4	C2	C3	C0	
S18	C6	C7	C8	C9						
S19	C9	C3								
S20	C8	C6	C7	C5	C9	C4	C1	C2	C3	C0

37 táblázat – A hallgatók preferencia sorrendje, ha minden feladatot sorba rendezhetnek

A 38. táblázatban látható a hallgatók táblás jelentkezése. Zárójelben az az érték látható, hogy végül a saját preferencia sorrendje alapján hányadik helyre kerül be. Akinél a zárójelben nem látható szám, ő olyan hetet választott, vagy választhatott a táblánál, ami a saját preferenciái között nem szerepelt. Az alap táblás jelentkezésnél ez az összeg $R=345$ egység. Látható, hogy a legtöbb hallgató olyan feladatot kapott csak, amit soka-dik helyen választott, vagy nem választott, így ez a módszer nagyon messze van az optimálistól. Az U értéke $45 \cdot 20 / 15 = 60$.

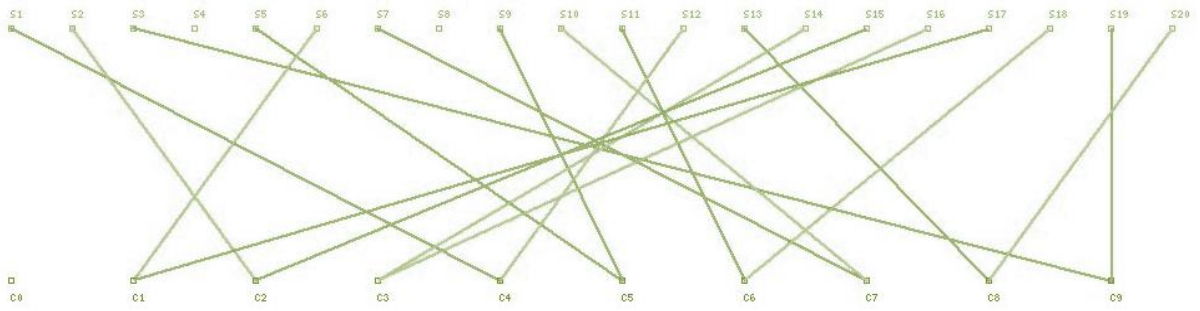
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
S9 (3)	S17 (1)	S16 (3)	S6 (3)	S12 (1)	S1 (-)	S8 (3)	S2 (2)	S7 (-)	S3 (1)
S20 (10)	S18 (-)	S19 (-)	S15 (8)	S14 (2)	S5 (2)	S10 (3)	S11 (-)	S13 (1)	S4 (2)

38 táblázat – Hallgatók által a táblánál történő feladatválasztás ($R=345$, $U=60$)

Miután megkaptam a preferenciákat, a megadott súlyok segítségével a hallgatókat sorba rendeztem az adott hetekhez. Így már volt lehetőségem a különböző algoritmusokat lefuttatni. A Gale-Shapley algoritmus futásának eredménye a 39. táblázatban és az 40. ábrán látható.

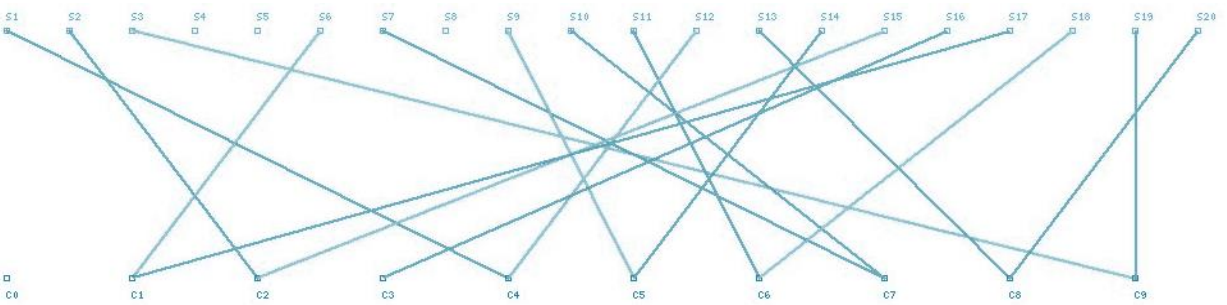
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
	S6 (1)	S2 (3)	S14 (7)	S1 (1)	S5 (2)	S11 (1)	S7 (1)	S13 (1)	S3 (1)
	S17 (1)	S15 (1)	S16 (2)	S12 (1)	S9 (1)	S18 (1)	S10 (1)	S20 (1)	S19 (1)

39 táblázat – Párosítás Gale-Shapley algoritmussal ($R=934$, $U=31,11$)



40 ábra – Párosítás Gale-Shapley algoritmus segítségével (R=934, U=31,11)

Hasonló eredményt kapunk akkor is, ha a Bostoni módszert alkalmazzuk, ennek eredménye látható a 41. táblázatban és a 42. ábrán.



41 ábra – Párosítás Boston mechanizmussal (R=804, U=23,53)

C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
	S6 (1)	S2 (3)	S16 (2)	S1 (1)	S9 (1)	S11 (1)	S7 (1)	S13 (1)	S3 (1)
	S17 (1)	S15 (1)		S12 (1)	S14 (1)	S18 (1)	S10 (1)	S20 (1)	S19 (1)

42 táblázat – Párosítás Boston mechanizmussal (R=804, U=23,53)

Mint látható, akár az U, akár az R mutatót nézzük, látványosan jobb eredményt kapunk, ha a Gale-Shapley vagy a Bostoni algoritmust alkalmazzuk, mintha a mostani algoritmus szerint párosítanánk.

4.3.1. Statisztikai elemzés

A következőben [145] alapján készítettem egy rekombinálást. Minden futtatás esetén egy hallgatót kicseréltem a másik kurzusból egy olyan hallgatóval, aki ugyanarra hétre jelentkezett a táblánál, és így kaptam 40 újabb eredményt. Ezek láthatóak a 43. táblázatban. A táblázatokban az *Eredeti* párosítás eredményeit hasonlítom össze a *Mohó*, *G-S* (*Gale-Shapley*), és a *Bostoni* algoritmus eredményeivel.

hallgatók	eredmények	hallgatók	eredmények
-----------	------------	-----------	------------

eredeti	csere	Eredeti	Mohó	G-S	Boston	eredeti	csere	Eredeti	Mohó	G-S	Boston
eredeti			832	934	804	eredeti		345	832	934	804
S1	S29	375	822	924	824	S11	S30	345	787	894	794
S1	S32	375	792	909	809	S11	S31	355	737	894	764
S2	S30	325	797	874	814	S12	S25	285	812	884	844
S2	S31	335	757	914	784	S12	S26	285	812	884	844
S3	S36	290	737	869	789	S13	S33	295	767	914	824
S3	S37	335	762	814	794	S13	S35	285	797	899	794
S4	S36	330	777	909	809	S14	S25	325	812	884	844
S4	S37	375	822	894	804	S14	S26	325	812	884	844
S5	S29	335	822	924	824	S15	S21	355	782	914	804
S5	S32	335	812	909	809	S15	S22	365	782	914	814
S6	S21	355	742	874	764	S16	S34	345	812	864	794
S6	S22	365	742	874	774	S16	S40	325	812	914	794
S7	S33	355	809	904	804	S17	S38	335	812	904	804
S7	S35	345	839	879	774	S17	S39	325	732	914	794
S8	S27	375	859	909	804	S18	S38	355	822	909	819
S8	S28	346	839	894	851	S18	S39	345	752	909	789
S9	S23	344	771	878	848	S19	S34	365	802	824	784
S9	S24	344	781	883	803	S19	S40	345	802	874	784
S10	S27	372	832	872	767	S20	S23	344	751	878	828
S10	S28	343	812	857	814	S20	S24	344	766	903	783

43 táblázat – A hallgatói cserék alapján elért „R” pontszámok a különböző algoritmusokkal

	Skewness		Kurtosis	
	Statistic	Std. Error	Statistic	Std. Error
Eredeti	-0,846	0,369	0,502	0,724
Mohó	-0,291	0,369	-0,784	0,724
Gale-Shapley	-1,093	0,369	1,891	0,724
Boston	0,281	0,369	-0,491	0,724

44 táblázat - Az adatok normális eloszlásának vizsgálata

A ferdeségi mutatók alapján megállapítható, hogy míg a Bostoni algoritmus eredményei jobbra, addig a többi algoritmusé balra ferde eloszlást mutatnak. Mivel egy sokaság a -2 és 2 közé eső csúcosság értékek esetében fogadható el normális eloszlásúként, így a táblázatban bemutatott adatokból megállapíthatjuk, hogy mind 4 vizsgált algoritmus normális eloszlású eredményeket generál. A következő táblázat adatai szerint a Mohó által generált adatok szórása szignifikánsan eltér a többi adattól, míg a más algoritmusok által generált párosítási értékek szórása hibahatáron belüli eltérést mutat.

		Levene's Test for Equality of Variances		t-test for Equality of Means						
--	--	---	--	------------------------------	--	--	--	--	--	--

		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
G-S-Boston	Equal variances assumed	0,119	0,731	16,210	80	0,000	86,488	5,336	75,870	97,106
G-S-Mohó	Equal variances not assumed	4,911	0,030	15,407	74,934	0,000	98,415	6,388	85,690	111,140
Mohó-Boston	Equal variances not assumed	6,767	0,011	-1,908	72,753	0,060	-11,927	6,251	-24,386	,532
Eredeti-Boston	Equal variances assumed	0,002	0,963	-87,732	80	0,000	-464,951	5,300	-475,498	-454,405
Eredeti-G-S	Equal variances assumed	0,136	0,714	-100,998	80	0,000	-551,439	5,460	-562,305	-540,573
Eredeti-Mohó	Equal variances not assumed	6,231	0,015	-71,257	74,492	0,000	-453,024	6,358	-465,691	-440,358

45 táblázat – A statisztikai elemzés eredménye

A 45. táblázatban azt mutattam be, hogy különböző algoritmusok által generált adathalmazok között páronként milyen kapcsolat van. Látható, hogy az Eredeti és a Mohó algoritmusok eredményét mind a Boston, mind a Gale-Shapley dominálja. A táblázat eredményeinek tükrében felrajzolható egy többszereplős tranzitív reláció, melyben

$$Eredeti < Mohó < Boston < Gale - Shapley$$

Ahogy a táblázatból látszik, az egyetlen nem szignifikáns különbség a Mohó és a Boston között van, ahol bár az átlagok közötti eltérés 11,927, ez az átlagok tükrében nem tekinthető szignifikáns eltérésnek.

4.3.2. Konklúzió

Az eredmények alapján megállapítható, hogy a legkevésbé hasznos az eredeti párosítás volt, melyet hasznosság tekintetében a Mohó algoritmus követ. Az átlagok eltérése megmutatja, hogy a Gale-Shapley eredmény hasznossága szignifikánsan magasabb volt a Bostoni algoritmus által meghatározott párokénál.

4.4. Hallgatók racionalitásának vizsgálata

A hallgatók egyik csoportban sem használták ki a rendelkezésükre álló lehetőségeket, bár az első lépésnél, ahol 3 lehetőséget lehetett megadni, ott a hallgatók 75%-a adott meg három, a maradék 25% pedig két helyet. Második esetben a hallgatók 5-5%-a adott meg 1, 2 illetve 3 helyet, 10 % 4 helyet, a többiek pedig 5 helyet. Az utolsó kutatásban a hallgatók 15 %-a adott meg 2, illetve 9 helyet, negyede adott meg 3, illetve 4 helyet, 10-10 % adott meg 6, illetve 8 helyet. Így a hallgatók választásának átlaga kevesebb, mint 5 hely volt.

	Lehetőségek száma		
	3 hely	5 hely	10 hely
átlag	2,75	3,9	4,8

46 táblázat – A jelentkezések átlaga

A döntéshozók korlátozott racionalitás axiomatikus. Érdekes azonban, hogy jelen helyzetben nem az információk hiánya, vagy az információs túlterheltség volt a korlátozott racionalitás oka, hanem az a döntéshozók szándékából származott. A hallgatók a választható cselekvési lehetőségek mindegyikéről minden fontos információval rendelkeztek, így a Neumann-Morgenstern féle axióma rendszer [26] szerint lehetőségük volt a racionális döntésre. Mivel ez nem egy egyszerű egyéni döntés, hanem egy egyértelmű konfliktus helyzet is volt, így a hallgatók ismerték a többiek választási lehetőségeit is.

Konklúzió

Az egyetemi életben rengeteg olyan probléma fordulhat elő, ahol két egymástól különböző halmazt kell összepárosítani. Jelenleg ezekre legtöbbször még semmilyen algoritmust nem alkalmaznak, legtöbbször a szerencsén múlik, például a vizsgára való jelentkezésnél a hallgató arra a vizsgára kerül-e be, amelyre szeretne. Ebben a fejezetben bemutatam egy ilyen problémát, részletesen elemezve és bizonyítva, hogy az ismert algoritmusok alkalmazása mind az egyetem, mind a hallgatók számára hasznos megoldást nyújtanának, mivel ha elégedettebb a hallgató, akkor nagyobb valószínűséggel fog jobb eredménnyel is végezni.

Természetesen fontos lépés a döntéseink vizsgálata is. Az életünkben fontos szerepe van a döntéseknek. Minden lépésünk döntésekből áll. A döntések, mint egy tanulási folyamat jelennek meg. Minden döntésből tanulni kell, minden problémára meg kell próbálni előre felkészülni. Fontos, hogy a problémákat minél hamarabb felismerjük és mindig a problémának megfelelő döntéshozatali módot válasszunk, akár egyéni akár csoportos döntésről beszélünk. Csoportos döntések esetén a csoport tagjainak megválasztása, a csoport méretének megválasztása, valamint a problémára specializált tagok beválasztása a csoportba egyaránt fontos.

Tehát a döntések jó megválasztása minden helyzetben, a döntési folyamat véghezvitele, továbbá az információk megszerzése mind-mind fontos.

A racionalitás szerepe a döntéseikben központi szerepet tölt be. Sajnos különböző korlátok alapján az ember nem képes a környezete, és ezáltal a döntési helyzet tökéletes meg-

ismerésére, de ilyen információhiányos helyzetben is törekszik racionális, vagyis szabályoknak megfelelő döntés meghozatalára. A kutatók sokáig csak azt tartották racionálisnak, ami a hasznosságot maximalizálja, későbbiekben a képet sikerült árnyalni olyan kritériumokban a felhasználásával, amelyeket nem lehet számszerűsíteni. A döntéseméleti vizsgálatok elején csak azt vizsgálták, hogyan lehet jó döntést hozni, későbbiekben bővebben foglalkoztak magával a döntéshozóval is, és azt vizsgálták, hogy miként hozunk döntéseket.

4.5. Szimulációs eredmények

Az előző kutatásokból megállapítható volt, hogy a léteznek olyan problémák az egyetemeken, ahol sokkal hatékonyabban lehet alkalmazni a különböző párosítási algoritmusokat. Az is kiderült, hogy a hallgatók többnyire nem használják ki a rendelkezésükre álló lehetőségeket, illetve nem tudják, hogy ezzel saját pozíciójukat gyengítik. A racionális magatartás feltárása, megértése érdekében egy szimulált probléma helyzetre 1024 különböző futtatást vizsgáltam. Minden futtatásban 20-20 hallgatói preferenciát határoztam meg, illetve ehhez 10-10 feladatot. A hallgatói preferenciákat itt is súlyozva határoztam meg, ami alapján számolhattam a feladatok preferencialistáját. Majd ezt a 1024 futtatást összesen 30 alkalom megismételtem a következők szerint. Ha egy hallgató 10 helyre jelentkezhet, akkor vizsgáljuk meg, hogy mi van akkor, ha csak 1 helyet adhat meg a preferencia listáján, mi van akkor, ha 2, ha 3, és így tovább, ha korlátlan, vagyis 10 helyet is megadhat. Ez alapján 10 különféle futtatást kapok. Majd ezeket még megvizsgáltam akkor, ha az adott feladatokat csak 2 hallgató láthatja el, ha 3, és ha 4,- vagyis, összesen 3 fajta futtatás, - ebből jön ki az $1024 \cdot 3 \cdot 10$ futtatás, vagyis a kicsivel több, mint 30ezer különböző eredmény.

Minden futtatásban 3-3 különböző módszert alkalmaztam a párosításokra, ezek a következők voltak: Mohó algoritmus, Gale-Shapley algoritmus, valamint a Bostoni mechanizmus.

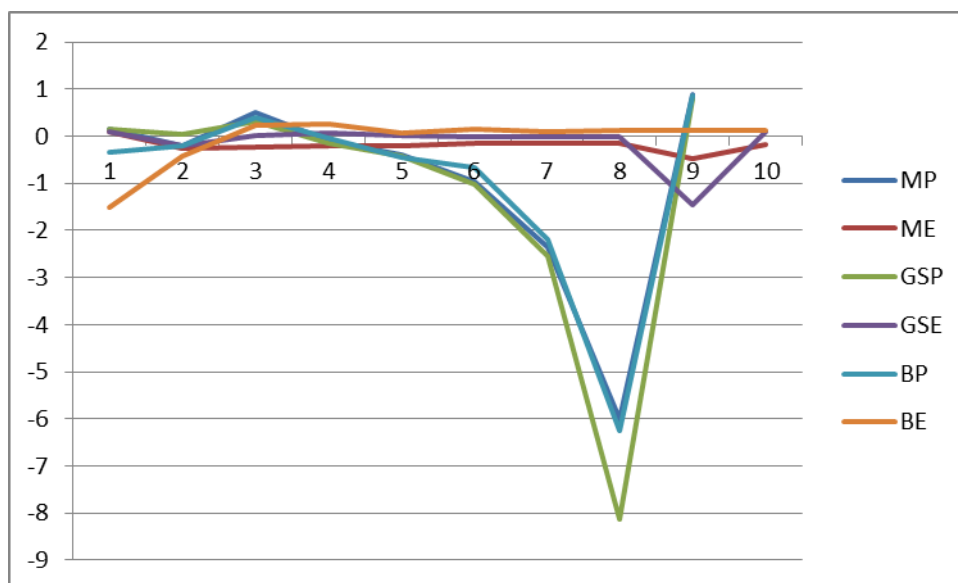
A nagy mennyiségű kutatási adat:

- lehetővé tette a különböző algoritmusok értékének egymáshoz viszonyított mértékének vizsgálatát,

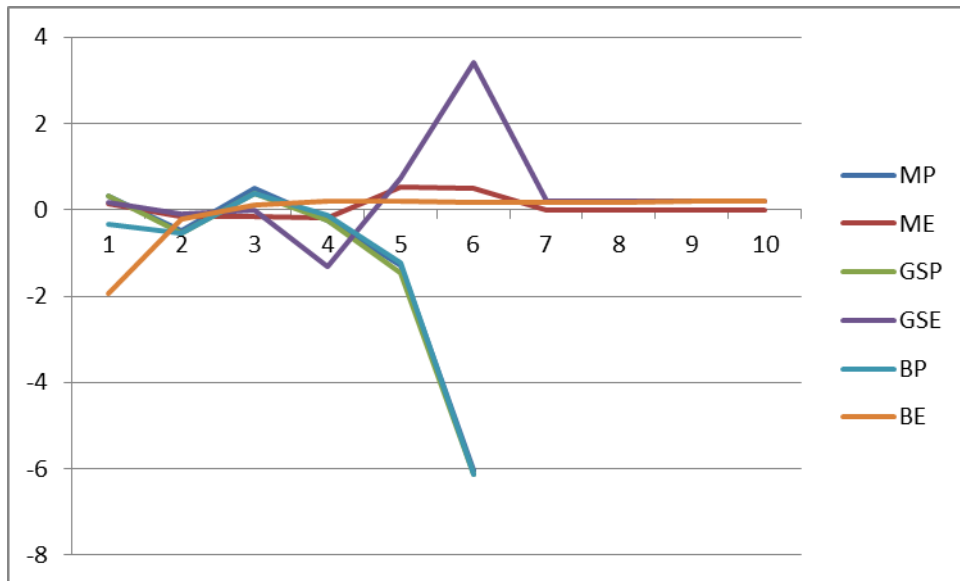
- statisztikai eredményekkel támasztotta alá a hallgatók irracionális viselkedésére vonatkozó korábbi állításomat, mely szerint a választási lehetőségeket ki nem használók saját esélyeiket csökkentik

4.5.1. Eredmények

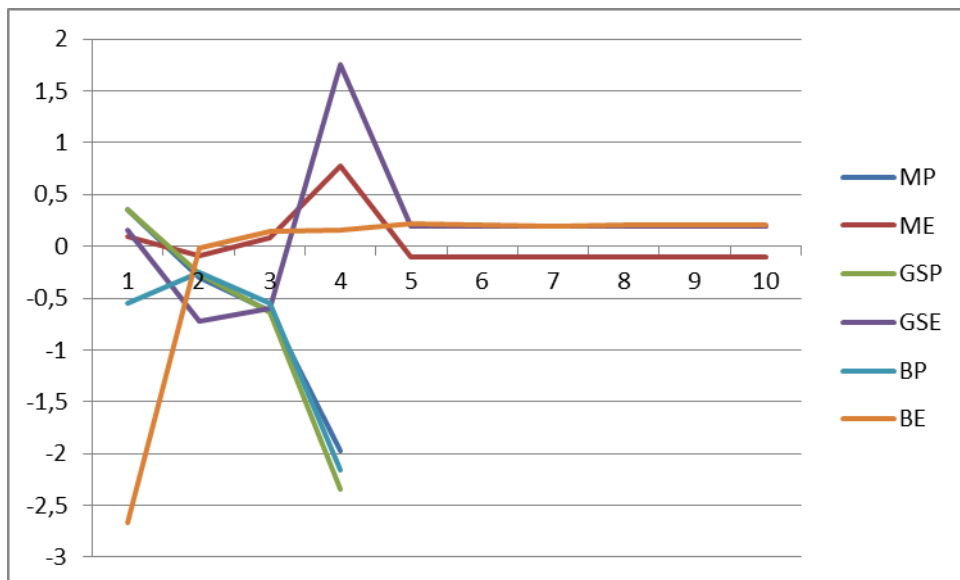
A random generált értékekből készült futtatások adatainak normalitását vizsgálva megállapítottam, hogy az értékek eloszlásának ferdesége a különböző algoritmusok által generált párosítások száma (MP, GSP, BP, a Mohó, Gale-Shapley, és Bostoni algoritmus által generált párosításokban a párok száma) esetében nem volt tendenciózus, de az értékek (ME, GSE, BE, a Mohó, Gale-Shapley, és Bostoni algoritmus által generált párosításokban a számított értékek eredménye) eloszlása mindig közelebb volt a normál eloszláshoz. A grafikonokon ábrázolt függvények megszakadása jelzi, hogy a sokaság adatainak szórása nulla, azaz minden hallgató párosításra került, mint például a két helyet hirdető kurzusoknál a 10 alternatíva felkínálásakor (47. ábra).



47 ábra – 2 hely van a kurzuson (az x tengely a kiválasztható helyek száma)

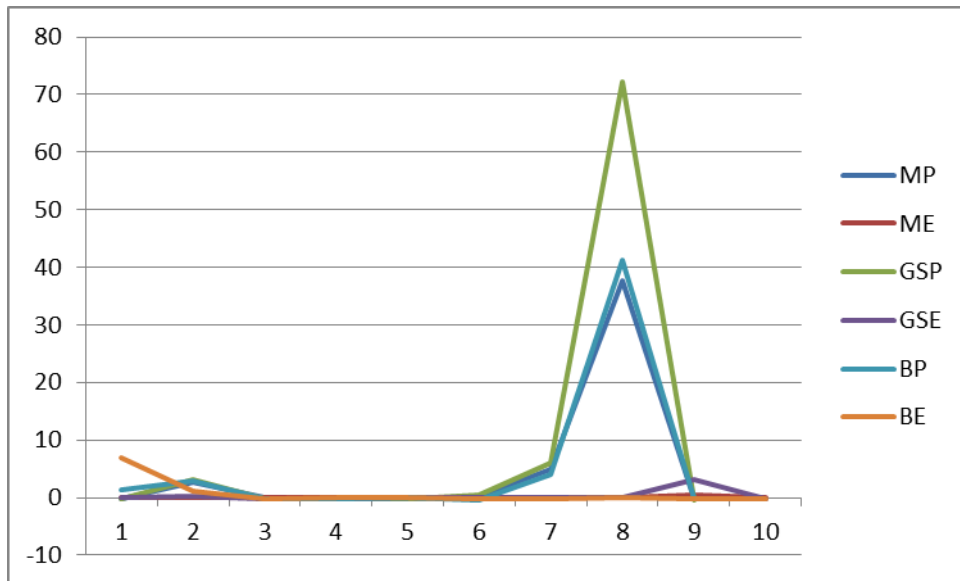


48 ábra – 3 hely van a kurzuson (az x tengely a kiválasztható helyek száma)

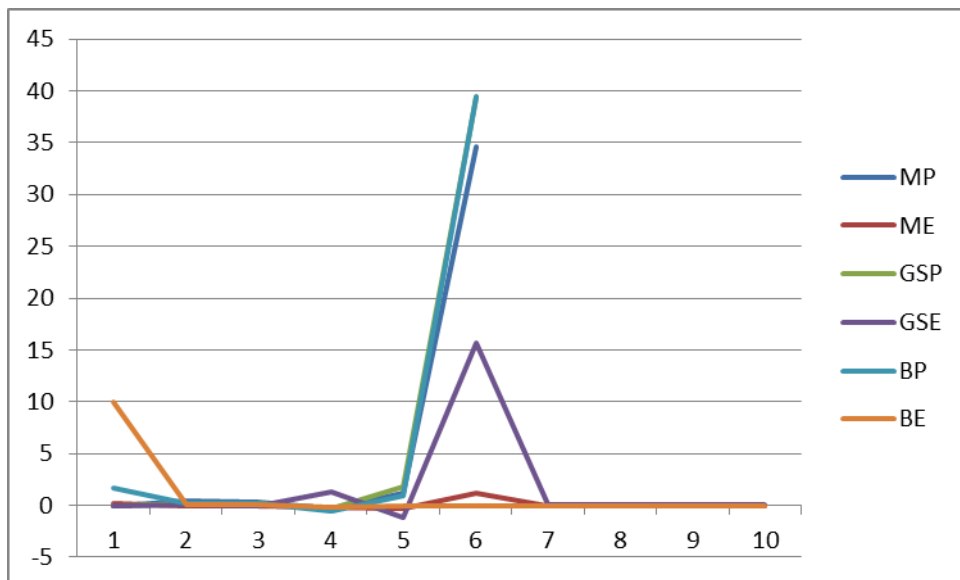


49 ábra – 4 hely van a kurzuson (az x tengely a kiválasztható helyek száma)

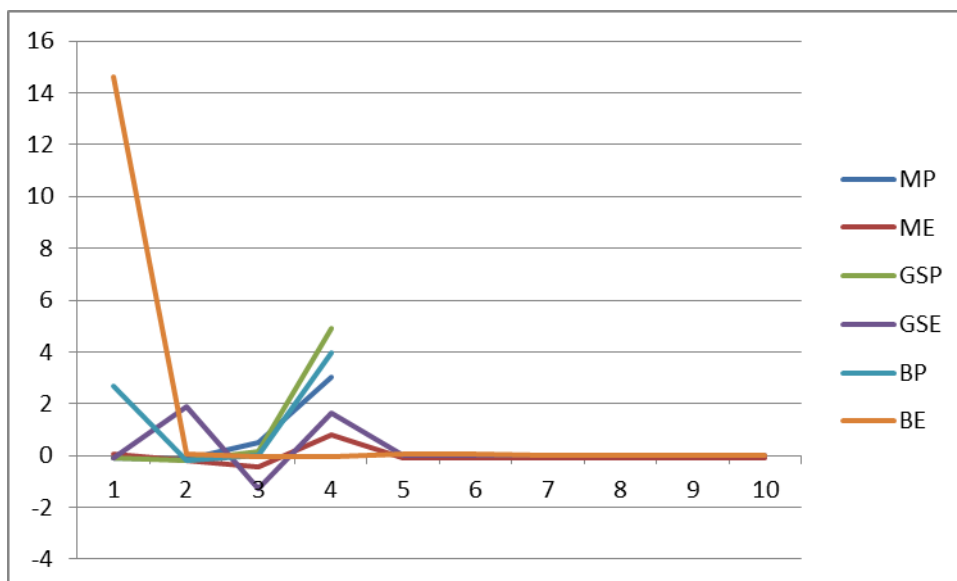
A random generált értékekből készült futtatások adatainak normalitását vizsgálva megállapítottam, hogy az értékek eloszlásának csúcsossága a különböző algoritmusok által generált párosítások száma esetében nem tendenciózus, de az értékek eloszlása mindig közelebb van a normál eloszláshoz.



50 ábra – 2 hely van a kurzuson (az x tengely a kiválasztható helyek száma)



51 ábra – 3 hely van a kurzuson (az x tengely a kiválasztható helyek száma)



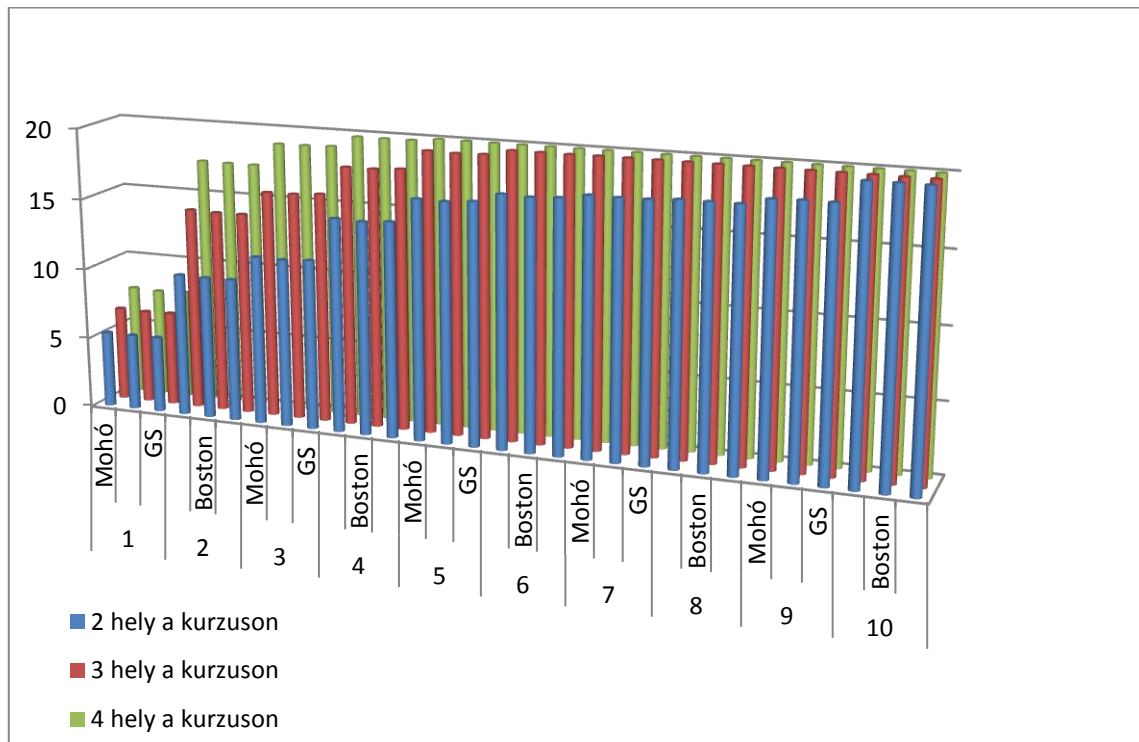
52 ábra – 4 hely van a kurzuson (az x tengely a kiválasztható helyek száma)

A különböző algoritmusok által generált párosítások számát (53 táblázat), és az általuk elért hasznosságértékeket (55 táblázat) a következő két táblázat tartalmazza. Jól látszik, hogy a különböző algoritmusokkal alkotott párosítások átlagértéke közt a legnagyobb eltérés is csupán 0,17, ami elhanyagolható, 5%os hibahatáron belüli érték. Az is jól ábrázolódik, hogy mind a jelentkező által megadott helyek száma, mind pedig az adott kurzuson meghirdetett helyek száma pozitívan hatott a kialakult párok számára. Az is jól látható, hogy a Gale-Shapley vezet az adott körülmények közötti legmagasabb számú párhoz.

a jelentkező által megadott helyek száma		2 hely a kurzuson	3 hely a kurzuson	4 hely a kurzuson
1	Mohó	5,34	6,60	7,65
	Boston	5,31	6,55	7,58
	G-S	5,34	6,60	7,65
2	Mohó	10,03	14,24	17,34
	Boston	9,99	14,19	17,31
	G-S	10,03	14,22	17,32
3	Mohó	11,79	15,91	18,93
	Boston	11,76	15,93	18,95
	G-S	11,88	16,07	19,02
4	Mohó	14,94	18,07	19,80
	Boston	14,87	18,08	19,80
	G-S	15,02	18,21	19,82
5	Mohó	16,71	19,58	20,00
	Boston	16,68	19,53	20,00
	G-S	16,83	19,59	20,00
6	Mohó	17,46	19,97	20,00

	Boston	17,40	19,97	20,00
	G-S	17,51	19,97	20,00
7	Mohó	17,82	20,00	20,00
	Boston	17,80	20,00	20,00
	G-S	17,84	20,00	20,00
8	Mohó	17,97	20,00	20,00
	Boston	17,97	20,00	20,00
	G-S	17,98	20,00	20,00
9	Mohó	18,43	20,00	20,00
	Boston	18,50	20,00	20,00
	G-S	18,50	20,00	20,00
10	Mohó	20,00	20,00	20,00
	Boston	20,00	20,00	20,00
	G-S	20,00	20,00	20,00

53 táblázat - A különböző párosítási algoritmusok által generált párosítások száma a választható helyek függvényében

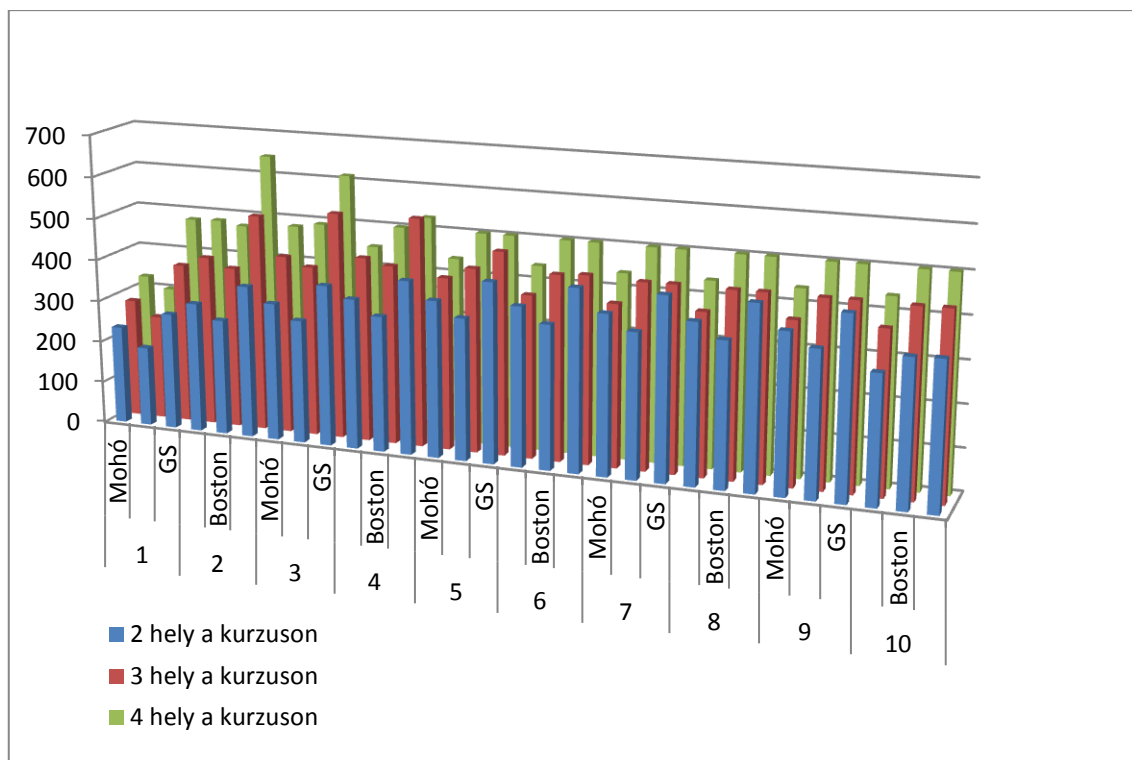


54 ábra - A különböző párosítási algoritmusok által generált párosítások száma a választható helyek függvényében

A táblázatos adatokat grafikonon ábrázolva jól vizualizálható, hogy minél több hely van egy-egy kurzuson, annál több pár hozható létre - annál jobb eredmény érhető el a párosítással. Az is egyértelműen leolvasható, hogy a betöltött helyek számának növekedése nem a választott párosítási algoritmus függvénye, mindhárom algoritmus esetében bekövetkezik.

a jelentkező által megadott helyek száma		2 hely a kurzuson	3 hely a kurzuson	4 hely a kurzuson
1	Mohó	234,14	283,28	328,24
	Boston	189,58	248,99	302,26
	G-S	278,08	381,74	479,70
2	Mohó	309,72	405,73	482,20
	Boston	275,39	385,55	473,72
	G-S	363,01	516,37	645,70
3	Mohó	327,77	424,72	482,90
	Boston	293,33	405,14	493,03
	G-S	382,52	537,44	613,33
4	Mohó	356,26	438,10	449,97
	Boston	321,71	424,60	501,06
	G-S	411,22	541,32	529,12
5	Mohó	370,94	407,79	437,84
	Boston	336,05	435,69	502,62
	G-S	426,08	480,59	502,93
6	Mohó	376,07	385,13	437,84
	Boston	340,50	438,37	502,64
	G-S	430,26	443,08	502,93
7	Mohó	377,71	383,30	437,84
	Boston	342,41	438,49	502,71
	G-S	431,71	439,44	502,93
8	Mohó	378,16	383,30	437,84
	Boston	342,95	438,49	502,71
	G-S	432,14	439,44	502,93
9	Mohó	376,03	383,30	437,84
	Boston	343,53	438,46	502,69
	G-S	427,72	439,44	502,93
10	Mohó	303,24	383,30	437,84
	Boston	345,03	438,46	502,69
	G-S	347,47	439,44	502,93

55 táblázat – A különböző párosítási algoritmusok által generált hasznosság értékek a választható helyek függvényében



56 ábra - A különböző párosítási algoritmusok által generált hasznosság értékek a választható helyek függvényében

A párosítások számával szemben az algoritmusok által generált párosítások értékeinél szignifikáns eltérések mutatkoznak: a Mohó tendenciózusan alacsonyabb hasznosságot generált, mint a GS, és a különbség egyre nőtt, ahogy nőtt a kurzuson levő helyek száma, de csökkent a hallgatók által megjelölhető helyek számának növekedésével.

Meglepő eredmény, hogy a Boston algoritmus segítségével elért párosítás összhassznossága abban az esetben, amikor csupán két hely van a kurzusokon, valamint amikor a hallgatók csupán kevés alternatívát rangsorolnak alacsonyabb a Mohó algoritmus által kialakított párosításénál.

Ahogy a választható helyek száma és a kurzuson meghirdetett helyek száma nő, a Boston algoritmus által kialakított párosítás egyre "jobbá válik" (magasabb abszolút és a Mohó algoritmusához viszonyított relatív hasznosságot generál) - ami természetes az algoritmus belső logikája ismeretében. Az adatok tehát alátámasztják az elméleti részben bemutatottakat. Ami a kutatás szempontjából igazán fontos, hogy ahogy nő a választható helyek száma, úgy csökken a Gale-Shapley előnye a Bostonhoz képest, de az előny sose fordul át negatívba.

Ahogy a Pearson korreláció is mutatja, a különböző algoritmusok által generált értékek között szignifikáns kapcsolat van (Szig.: 0,000). A következő táblázat a párosítások által generált értékek egymáshoz viszonyított kapcsolatát mutatja. Látszik, hogy a három algoritmus által generált értékek páronként összefüggnek, azonban az is leolvasható, hogy a kapcsolat erőssége változó.

	M#GS			GS#B			M#B		
	2 hely	3 hely	4 hely	2 hely	3 hely	4 hely	2 hely	3 hely	4 hely
1	0,747	0,818	0,836	0,924	0,887	0,838	0,595	0,592	0,563
2	0,573	0,684	0,575	0,596	0,755	0,799	0,255	0,301	0,273
3	0,596	0,645	0,217	0,673	0,51	0,516	0,311	0,163	0,12
4	0,592	0,3	0,16	0,607	0,466	0,222	0,309	0,119	0,214
5	0,587	0,268	0,375	0,38	0,271		0,194	0,133	0,369
6	0,603	0,25	0,375	0,216	0,087		0,179	0,354	0,372
7	0,612	0,392	0,375	0,138			0,179	0,395	0,374
8	0,617	0,392	0,375				0,178	0,393	0,371
9	0,403	0,392	0,375				0,155	0,394	0,376
10	0,431	0,392	0,375				0,443	0,394	0,376

57 táblázat – A vizsgált algoritmusok által generált értékek viszonya

A 57. táblázatban azt vizsgáltam, hogy a különböző eljárások, által generált értékek átlagai valóban szignifikánsan különbözőek-e. Érdekes eredmény, hogy míg a Gale-Shapley-Boston és a Mohó-Gale-Shapley páronként szignifikánsan különböző, addig a Mohó- Boston nem, ahogy ez az átlagok különbségéből, és a korábbi átlagokat bemutató táblából is jól látszott. A Mohónál és a Bostonnál is szignifikánsan magasabb értékeket generál a Gale-Shapley

Kérdés azonban, hogy a kapott különbség szignifikánsnak tekinthető-e. Ennek eldöntése érdekében számos peremfeltételt teljesítenie kell a vizsgált változóknak (a vizsgált csoportokban az elemszám közel azonos, normál eloszlású, a függő változó szórása azonos a különböző csoportokban). A korábban bemutatott grafikonok eredményei alapján már csak a szórásra vonatkozó premisszát kell megvizsgálnom.

Levene-próba (a varianciák eltérésének vizsgálatára)		Független mintás t-próba (a csoportátlagok eltérésnek vizsgálatára)							
F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference		
						Lower	Upper		

Mohó-G-S	1,849	,179	- 4,228	58	,000	-73,800	17,453	- 108,737	-38,863
Mohó-Boston	9,005	,004	-,606	58	,547	-11,733	19,352	-50,470	27,003
G-S-Boston	2,052	,157	2,917	58	,005	62,067	21,279	19,473	104,661

58 táblázat – A Levene-próba eredménye

Levene-próba eredménye azt mutatja, hogy a szórások egyenlőségének feltétele csak a Mohó és a Boston algoritmusok eredményei esetében valósult meg. Ennek megfelelően a t-próba eredménye csak ebben az esetben megfelelő biztonságú.

4.5.2. Konklúzió

A fejezetben bemutatott eredmények tükrében elmondható, hogy a párosítások számát tekintve egyik algoritmus sem volt szignifikánsan jobb, mint a másik. Ezzel szemben az algoritmusok által generált párosítások hasznossági értékei eltérést mutatkoznak: a Mohó tendenciózusan alacsonyabb hasznosságot generált, mint a Gale-Shapley, és a különbség egyre nőtt, ahogy nőtt a kurzuson levő helyek száma, de csökkent a hallgatók által megjelölhető helyek számának növekedésével. Megfigyelhető volt továbbá, hogy a Boston először rosszabb eredményt hoz a Mohó algoritmusnál, majd a választható helyek és a kurzuson meghirdetett helyek számának növekedésével a generált hasznosság értékek egyre jobbra váltak.

4.6. Egyetemi felvételik elemzése

Az egyetemi felvételik - bár számos nemzetközi publikáció foglalkozik azzal, hogy a párosításelméleti algoritmusok hatékonyabbá tehetnék – még mindig szuboptimálisan működnek.

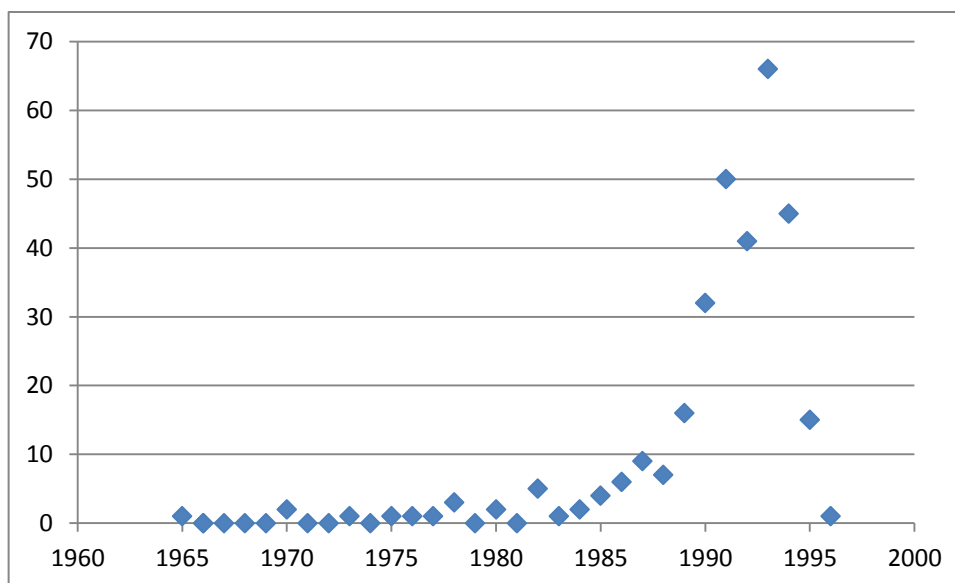
Magyarországon a középiskolai [146] és az egyetemi felvételi rendszer [146], [147] is dominánsan a Gale-Shapley algoritmust használja, bár kisebb módosításokat hajtottak végre rajta. Alapvető különbség a rezidens allokációs rendszerhez képest [148] [149], hogy hazánkban még mindig az egyetemi szakok irányából futtatják, ami a diákok szempontjából mindig a lehető legrosszabb eredményt adja. [114] A világ sok részén nem használják még ezt a rendszert, ilyen például az Amerikai Egyesült Államok, ahol nincs központi felvételi rendszer és az elbírálás helyi szinten történik. Abdulkadiroğlu és Sönmez [110] rávilágított arra, hogy sok egyesült államokbeli iskolában tisztázatlan a

felvételi rendje, a döntések gyakran önkényesek, és a legritkább esetben felelnek meg a tudományban már évtizedek óta ismert alapvető elvárásoknak. Míg az Egyesült Királyságban csak regionálisan csináltak központosított párosítási mechanizmusokat, de ezek közül a legtöbb nem adott stabil megoldást.

Kóczy [150], [151] bemutatja a magyarországi felvételi rendszer sajátosságait vizsgálva arra a következtetésre jutott, hogy a felvételizők által megadható maximális egyetemi sorrend rontja a hallgatók felvételi esélyét, ezzel növelve azt, hogy több betöltetlen hely, és sok olyan felvételiző marad, aki nem kerül be a felsőoktatásba. Ezzel a hallgatók nagy része nincs tisztában, mert nagy részük még a 3 helyet sem használja ki, lemondva esetleg olyan helyekről is, ami számukra jobbnak tűnt, de mégsem merték megpróbálni.

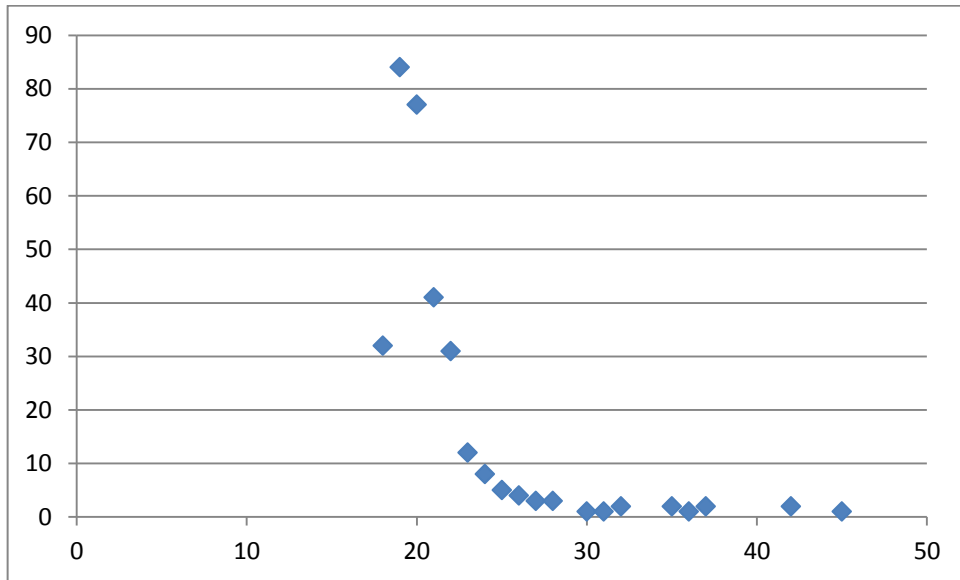
Én a problémát nem az intézményi, hanem az egyéni szinten vizsgálva arra kerestem a választ, hogy milyen stratégiát alkalmaztak/nak a hallgatók, a felvételi jelentkezési lapok kitöltésénél? Mennyire tudtak racionális döntést hozni és korlátozva érezték-e magukat azáltal, hogy csak bizonyos (korlátozott) számú helyre adhatják bejelentkezésüket?

A vizsgálatban (kérdőív: 0 mellékelt) 312 Óbudai Egyetemen tanuló hallgató vett részt. 274 nappali tagozatos, 29 levelezős és 9 távoktatásos. Nemek szerinti megoszlásuk (163 férfi, 149 nő) tükrözi az Óbudai Egyetem hallgatóinak nemek szerinti megoszlását. Életkor szerinti megoszlásukat a 59. grafikon ábrázolja



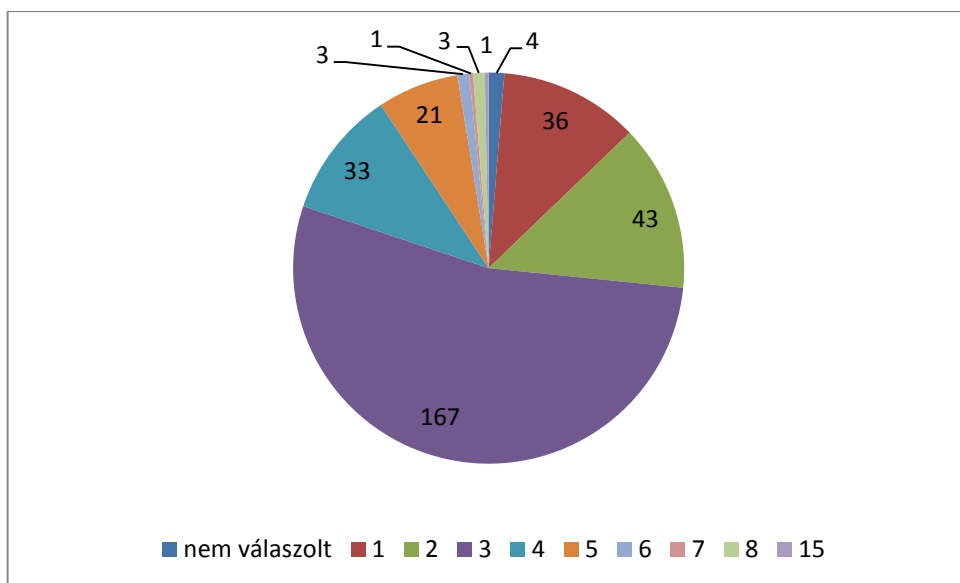
59. ábra – A kitöltők születési évük szerinti megoszlása

A válaszadók többsége közvetlenül középiskolai tanulmányai után jelentkezett egyetemünkre, de voltak akik – életkorukból következően – már több éves munkatapasztalattal rendelkeztek, és talán jobban ráláttak döntéseik következményeire. A válaszadók felvételi évében való életkorát a 60. grafikon mutatja.



60 ábra – A kitöltők felvételük évében való életkoruk szerinti megoszlása

Voltak olyan válaszadók, akik (az éppen aktuális szabályozásnak köszönhetően) igen sok helyre adták be jelentkezésüket, azonban jelentős számban voltak olyanok is, akik még a lehetséges három, illetve öt alternatívát sem használták ki. Ezzel együtt 90% (281 fő) elégedett volt a számára rendelkezésre álló alternatív döntési lehetőségek számával. Azt, hogy hány helyre adták be jelentkezésüket, a következő grafikon mutatja



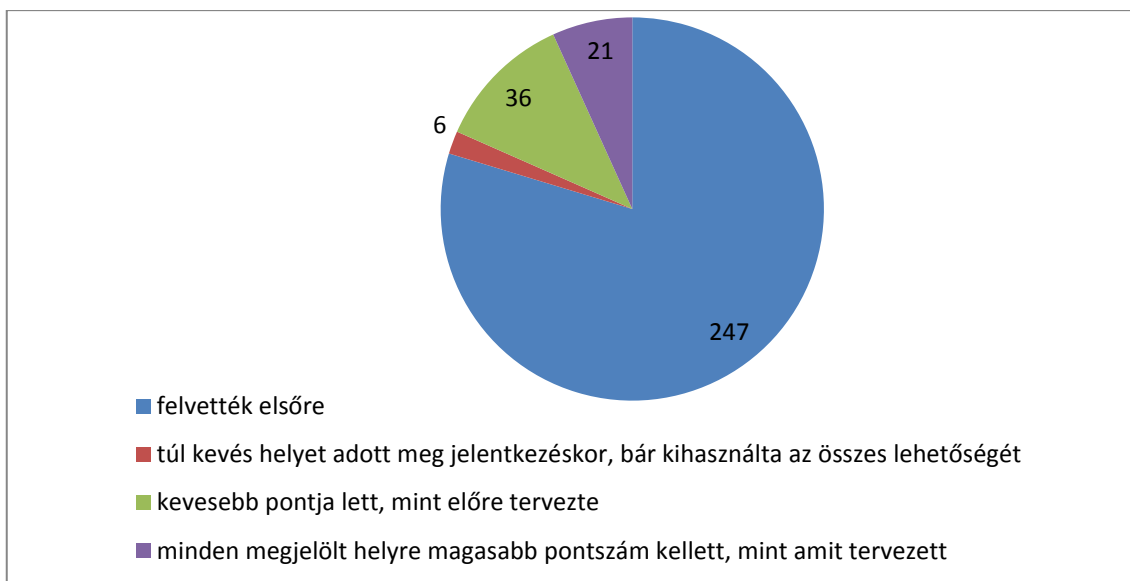
61 ábra – A hallgatók megoszlása az alapján, hogy hány helyre adták be a jelentkezésüket

Az eredmények nem erősítik meg Kóczy [151] hipotéziseit, mely szerint „... *aligha van olyan hallgató, aki a képzések, szakok teljes listáját rangsorolni kívánja, valószínű, hogy sokak számára korlát a három jelentkezés, hiszen ennél több elérhető és elfogadható szakra is jelentkeznének. Ennek némileg ellentmond, hogy a hallgatók közel egyharmada még a három lehetőséget sem tölti ki.*”

A jelentkezési stratégiákkal kapcsolatban kiderült, hogy 201 fő azért jelentkezett azokra a helyekre, ahova beadta jelentkezési lapját, és azért olyan sorrendben jelölte be a célintézményeket, hogy „Biztosan legyen olyan hely, ahova felvesznek”, 170 válaszadó nyilatkozott úgy, hogy a döntésénél az volt a szempont, hogy „Legyen olyan is a listán, ahova nagyon szeretnék bekerülni”. 25 fő egyéb stratégiai szempontokat is megnevezett, mint például a megcélzott intézmény lakhelytől való távolsága, a tandíj nagysága, vagy ismerősök ajánlása, esetleg a diploma piaci értéke, vagy az, hogy milyen nemzetközi kapcsolatai vannak az intézménynek.

A hallgatók közel 47%-a (145 fő) csak államilag támogatott képzéseket jelölt meg, 10%-uk (33 fő) csak önköltséges helyeket, míg 43 százalékuk vegyesen – a költségvonzattól függetlenül – válogatott szakokat jelentkezéskor.

A megkérdezettek 20%-ának (65 fő) nem sikerült az első felvételi időszakban semmilyen általa megjelölt intézménybe felvételt nyernie. Ők tehát biztosan nem jól döntöttek a jelentkezési lapjuk kitöltésekor. 6 fő szerint az volt a hiba, hogy túl kevés helyet adott meg jelentkezéskor, bár kihasználta az összes lehetőségét. 36 fő szerint a baj az volt, hogy kevesebb pontja lett, mint előre tervezte. 21 fő pedig azért nem került be egyetlen általa megjelölt helyre sem az első felvételi alkalmával, mert minden megjelölt helyre magasabb pontszám kellett, mint amit tervezett. A képet tovább árnyalja, hogy 30 fő úgy gondolja, hogy ha több helyre jelentkezhetett volna, akkor biztosan felvették volna valahova.



62 ábra – A hallgatók eredménye a felvételin

Az első körben felvételt nem nyerők több mint 70%-a (47 fő) változtatta meg korábbi stratégiáját, és nyert második körben felvételt, míg csupán 30% került be sikeresen felsőoktatási intézménybe a korábbi stratégiája konzekvens alkalmazásával.

A válaszadók 31%-a (98 fő), gondolja azt, hogy ha lett volna még több választási alternatívája, akkor a korábbi listája végére tett volna még olyan szakot, ahova biztosan bekerül. A válaszadók több mint fele (171 fő) azonban a listája elejére tett volna még olyan szakot, ahova kicsi volt az esélye, hogy bekerül, de nagyon örült volna, ha sikerül. Retrospektív 13 %-uk (41 fő) érzi úgy, hogy nem döntene ma sem másképp, mint a jelentkezési lap kitöltésekor.

A válaszadók jellemzői közötti kapcsolatot vizsgálva érdekes összefüggés (Pearson korrel: 0,346, Szig.: 0,000), hogy saját jelentkezési (preferencia) listája minél többedik helyére vették fel az adott hallgatót, annál több elemű volt egyébként is az a lista. Ebből következtethetünk arra, hogy az a hallgató, aki bizonytalan volt jelentkezésének sikerességét illetően inkább több helyet is megjelölt, hogy ezzel is növelje esélyeit a bekerülésre.

A másik érdekes összefüggés, ami a hallgatói stratégiák vizsgálatokor felszínre került, hogy azok, akik a jelentkezési lapjuk kitöltésekor úgy igyekeztek intézményeket/szakokat választani, hogy legyen olyan is a listájukon, ahova nagyon szeretnék bekerülni, azok kevésbé figyeltek arra, hogy biztosan legyen olyan hely ahova felveszik őket (Pearson korrel: -0,451, Szig.: 0,000).

Konklúzió

Bár az előző kutatásaim eredménye alapján látható volt, hogy a racionális döntés az lenne, ha a felvételik esetében a leendő hallgatók a lehetséges összes helyet rangsorolnák, ez a mostani szabályozás szerint 5 helyet jelent. A hallgatók vagy túlságosan optimisták a saját eredményükben bízva, vagy nem fontos számukra, hogy minden áron tovább tanuljanak, inkább hajlandóak kihagyni fél-egy évet, hogy olyan egyetemre (szakra) kerüljenek be, ahol igazán szeretnének tanulni. Ezt bizonyítja az is, hogy a hallgatók nem figyeltek arra, hogy biztosan legyen a listájukon olyan hely, ahova gond nélkül felveszik őket. Mindemellett én azon az állásponton vagyok, hogy érdemes lenne eltörölni a jelentkezési korlátot, mint látható a legtöbb hallgató esetében ez nem jelenthetne többletmunkát, hiszen a rendelkezésre álló kereteket sem használják fel, de van sok olyan jelentkező is, akinél ez segítséget jelenthetne. Mellette ez nem jelentene a felvételi rendszerre sem többletmunkát, hiszen a szoftveresen ez a számosság nem jelent igazából különbséget. Amikor még nem csak interneten keresztül lehetett jelentkezni, hanem kötelező volt postán keresztül papíron is leadni, akkor egyértelműen nehezebb lett volna a több lehetőség elbírálása, ez ma már az informatika korában nem lehet korlát.

5. AZ ELKÉSZÜLT SZOFTVER ALKALMAZÁSA

Miután sikeresen elkészült a szoftverhez az úgynevezett Frontend, (vagyis egy olyan felület, ahol bárki mindenféle matematikai ismert nélkül is képes használni a szoftvert) az eddigi feladatokkal is sikeresen teszteltem, így lehetőségem volt élesben is kipróbálni. A szoftver célja, hogy élőben, „on the fly” lehessen különböző problémákat megoldani. Ahogy a dolgozatomban bemutattam, sok olyan probléma létezik akár csak az egyetemi életben, de különböző munkahelyeken is ahol alkalmazható lenne a párosítás elmélet, de mégsem tesszük, és sokszor csak a véletlenül múlik, hogy milyen párok jönnek létre. Ezek okai a következők: Nem létezik olyan szoftver, amely képes lenne kezelni a különböző preferenciákat, és azokból stabil párosításokat alkotni. Az emberek többsége nem ismeri ezeket az algoritmusokat, és ha ismernék is, szoftver segítségével ezek alkalmazása hosszadalmas, és sok hibával teli. Tehát a szoftver elkészítése után már csak problémákat kellett keresnem, és megoldást találni rájuk a szoftver segítségével. Ilyen problémák például órai feladatok szétosztása hallgatók között, egy adott kutatócsoport feladatainak a tagokhoz való rendelése, illetve akár különböző csoportok beosztása.

5.1. Szervezeti magatartás órai feladatok kiosztása

5.1.1. A probléma bemutatása

A probléma lényege a következő: Van egy kurzuson 17 hallgató, és van összesen 9 különböző feladat. Úgy kell a hallgatókhoz hozzárendelni a különböző feladatokat, hogy a párosítások stabilak legyenek, és a hallgatók általa megadott súlyok segítségével határozzuk meg a feladatokhoz tartozó preferenciákat, vagyis, egy adott feladatnál az a hallgató lesz a leginkább preferált, aki a legmagasabb értéket adta meg. Hallgató a belépés után a 63. ábrán látható űrlapot látja maga előtt. Az űrlap segítségével hasznossági értéket kell rendelnie a 9 felkínált alternatívához oly módon, hogy a hasznosságok összege 100 pont legyen. Megtehetik, hogy csupán egy alternatívát értékelnek hasznosként (és azt 100 ponttal ’jutalmazzák’), de úgy is dönthetnek, hogy mind a kilenc alternatívához hozzárendelnek egy nullától különböző pontértéket (melyek összege 100 lesz). Ez a választás, illetve preferencia megadás az első órán megtörténik, miután a hallgatóknak ismerttettem a különböző feladat tartalmát, illetve, hogy melyik feladatnál mit is kell

csinálni. A tárgy külföldieknek meghirdetett ERASMUS tárgy, ezért is látható angolul minden szöveg az ábrán.

Logged in: S1

ADDING ORDER TO EVENTS

EVENTS:

PLEASE ORDER THESE ITEMS WITH SUM OF 100

Please add you prefer

Description	Max space	Number
Individual perspective	[3]	<input type="text" value=""/>
Decisions	[2]	<input type="text" value=""/>
Game Theory	[2]	<input type="text" value=""/>
Motivation	[2]	<input type="text" value=""/>
Group	[2]	<input type="text" value=""/>
Culture	[2]	<input type="text" value=""/>
Leadership	[2]	<input type="text" value=""/>
Conflicts and negotiation	[3]	<input type="text" value=""/>
Communication	[3]	<input type="text" value=""/>
<input type="button" value="SAVE"/>		SUM[21] SUM: [0]

63 ábra – A szervezeti magatartás tárgyhoz tartozó űrlap

A hallgatók választása látható a 64 táblázatban. A táblázat első oszlopa mutatja a hallgatókat (S1-S17), míg a sorok a hallgatók választását, és a hozzájuk adott értékeket mutatja.

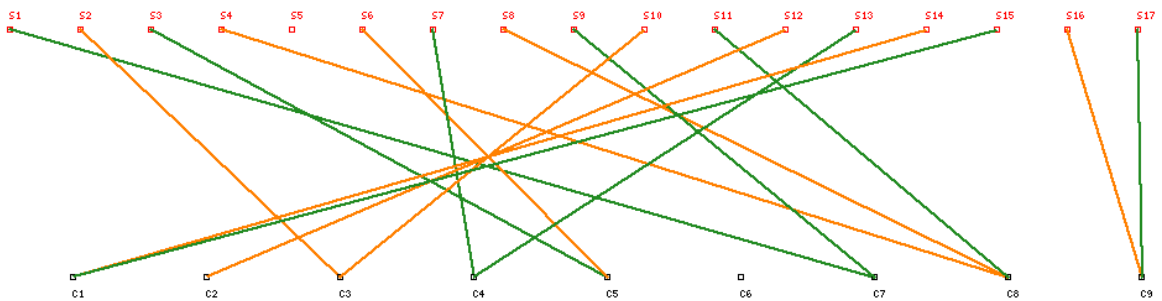
S1	C7(50)	C9(30)	C6(10)	C5(10)					
S2	C3(30)	C7(20)	C5(10)	C4(10)	C1(10)	C2(10)	C8(5)	C9(4)	C6(1)
S3	C5(50)	C9(50)	C7(0)	C8(0)	C6(0)	C3(0)	C1(0)	C2(0)	C4(0)
S4	C4(60)	C8(20)	C6(20)						
S5	C4(60)	C7(20)	C5(20)						
S6	C4(50)	C5(25)	C2(20)	C7(5)					
S7	C4(90)	C7(5)	C6(5)						
S8	C8(30)	C9(20)	C5(20)	C2(20)	C7(10)				
S9	C4(30)	C7(20)	C6(20)	C8(8)	C9(5)	C5(5)	C1(5)	C2(5)	C3(2)
S10	C3(60)	C8(10)	C2(10)						
S11	C8(60)	C7(20)	C4(20)						
S12	C4(16)	C3(15)	C2(15)	C5(11)	C9(10)	C7(10)	C1(8)	C6(8)	C8(7)
S13	C4(70)	C7(20)	C9(10)						
S14	C1(20)	C6(20)	C5(20)	C9(10)	C8(10)	C4(10)	C7(10)		
S15	C1(30)	C7(30)	C9(20)	C4(20)					
S16	C9(30)	C7(20)	C4(10)	C6(10)	C2(10)	C8(5)	C5(5)	C3(5)	C1(5)
S17	C9(25)	C7(20)	C8(10)	C6(10)	C5(10)	C4(10)	C3(5)	C1(5)	C2(5)

64 táblázat – A hallgatók választása döntési súlyokkal

Mint a táblázatból jól látszik, a hallgatók legalább három alternatívának adtak hasznossági értéket, de több olyan hallgató is volt, aki minden lehetséges alternatívához hozzárendelt hasznossági értéket (ezáltal rangsorolva azokat).

5.1.2. A helyzet értékelése

Miután a hallgatók meghatározták a preferenciákat, az adminisztrációs oldalon látható a párosítás eredménye, amelynek az ábrázolása látható a 65. ábrán, illetve a 66. táblázatban.



65 ábra – A szervezeti magatartás tárgy feladat kiosztásának eredménye

A 66. táblázatban a hallgatók neve mögött látható, hogy ki hányadik helyen került be, míg a táblázat első sorában látható feladatoknál a zárójelben az látható, hogy hány szabad hely volt összesen az adott feladatnál. Mint látható, a feladatban nem lett minden feladat ellátva, sőt nem is kapott minden hallgató feladatot, hiszen ez nem is következik a párosítás definíciójából.

C1(3)	C2(2)	C3(2)	C4(2)	C5(2)	C7(2)	C8(3)	C9(3)
S14 (1)	S12 (3)	S2 (1)	S7 (1)	S3 (1)	S1 (1)	S4 (2)	S16 (1)
S15 (1)		S10 (1)	S13 (1)	S6 (2)	S9 (2)	S8 (1)	S17 (1)
						S11 (1)	

66 táblázat – A párosítás eredménye

Ha egy hallgató nem kapott feladatot, akkor ő nem adott meg elég helyet a saját preferencia sorrendjében. Létezhetnek olyan esetek, amikor azt mondja, hogy ha ő nem kapott olyan feladatot, ami számára szimpatikus lett volna, akkor nincs is szüksége feladatra, ilyen lehet például a már említett egyetemi felvételi, ahol - ha egy hallgató nem kerül be a számára preferált helyek egyikére sem -, akkor inkább vár egy fél évet és újra felvételizik, mint hogy olyan helyen tanuljon, ami számára nem megfelelő. Ebben az

esetben viszont a tárgy teljesítésének a feltétele, hogy valahova bekerüljön. Mint a grafikus ábrázoláson látható, ez a hallgató az S5 jelzésű hallgató volt. Az ő preferencia sorrendje a C4(60), C7(20), C5(20) volt. A C4 feladatra csak a 3. választási lehetőség volt a feladat preferencia sorrendjében (S7(90), S13(70), S5(60)), a C7 feladat esetén is csak a 3. helyen volt (S1(50), S15(30), S5(20)), és a C5 esetén is csak a 3. helyre került. (S3(50), S6(25), S5(20))).

Hogyan tudta volna javítani a bejutási esélyeit az S5 hallgató? Az egyik megoldás, hogy több feladatot rendez sorba. A másik megoldás, hogy magasabb értékeket ad meg valamelyik választásához, vagy más sorrendben határozza meg őket. Mivel nem ismeri a többiek preferencia sorrendjét, így az első lehetőség az, amire inkább van esélye. Mint ahogy említettem, a tárgy teljesítésének feltétele, hogy mindenki válasszon valamilyen feladatot, így az adott hallgatónak a maradék helyek közül volt lehetősége választani.

5.1.3. Hallgatói reakciók

A hallgatók nagy érdeklődéssel és lelkesedéssel láttak neki a feladatok értékelésének, meglepődtek, hogy nem én osztom ki önkényesen a feladatot, hanem lehetőségük van nekik is részt venni a döntésben. Az ilyen feladat szétosztásnál lehetőség lenne az előző fejezetekben már említett, úgynevezett Mohó algoritmus alkalmazására, ami a legjobban úgy szimulálható, hogy körbeadok egy papírt és mindenki választ magának egy szabad helyet. Így ebben az esetben az első hallgató az összes lehetőség közül választhat, míg az utána következők folyamatosan egyre kevesebb hely közül. Ezt a verziót ebben a kutatásban nem valósítottam meg, mivel az előző kutatások már megmutatták, hogy nem ad jó párosítást. Miután megvolt a párosítás és minden hallgató megkapta a feladatát egy szóbeli felmérést csináltam az elégedettségről. Ennek eredménye a következő volt. A hallgatóknak különböző típusú kérdésekre kellett válaszolnia, az egyik csoportban a kapott párosítás eredményéről kellett véleményt nyilvánítania, míg a másik csoportban az volt a fő kérdés, hogy az oldal kinézete, kezelhetősége mennyire volt számára elfogadható.

Az első csoportba tartozó kérdések a következők voltak: „Mennyire vagy elégedett a kapott feladattal?” itt 1-től 5-ig lehetett értékelni, az 5 jelentette azt, hogy teljesen elégedett a kapott feladattal, míg az egyes, azt, hogy teljesen elégedetlen. Ezeknek az értékeknek az átlaga 4,41 volt, ami azt jelzi, hogy a hallgatók kifejezetten elégedettek vol-

tak a kapott feladattal. A relatív szórása 18%, így látható, hogy a hallgatók nagyrészt egymáshoz közeli értékeket adtak meg. A következő kérdés arra vonatkozott, hogy „A feladat, amit kaptál, hányadik volt a preferencia listádban?”, itt az volt a lényeg, hogy emlékszik-e a saját preferencia sorrendjére, és hogy abban hányadik helyen volt végül a végső eredmény. A hallgatók 2 kivételével eltalálták, hogy hányadik helyen adták meg a végül győztes eredményt, a két kivétel közül az egyik felfelé a másik lefelé tért el, az eredetitől. Ennek a kérdésnek az volt a lényeg, hogy a hallgatók preferencia sorrendje változik-e, illetve, hogy mennyire tartják észben ezeket a válaszokat. A következő kérdés csoportban az oldal használatáról kérdeztem a hallgatókat, itt volt olyan kérdés, hogy „Mennyire volt használható az oldal”. Ennél a kérdésnél is 1-től 5-ig lehetett értékelni a kérdést, hasonlóan az első kérdéshez. A hallgatók értékelése alapján ennek az eredménye a következő lett. A válaszok átlaga 4,64, a szórás értéke 0,49, míg a relatív szóródási értéke 11% lett. Ez alapján látható, hogy a hallgatók a program használatával meg voltak elégedve. A következő kérdés arra vonatkozott, hogy mennyire volt átlátható az oldal, mindent könnyen meg lehetett találni, vagy mennyire ergonomikus a kezelőfelületek kialakítása. Az értékelés itt is 1-től 5-ig tartó skálán volt, ahol 1 volt a legrosszabb, 5 a legjobb érték. A válaszok átlaga 4,11 volt, a szórás 0,85, míg a relatív szórás értéke pedig 21%.

Az értékelő beszélgetésből kiderült, hogy a hallgatók többnyire (több mint 80%-uk) meg voltak elégedve a szoftverrel. Örültek annak, hogy nem a véletlenül múlik, hogy ki milyen feladatot kap, hanem van lehetőségük megadni valamilyen sorrendet.

5.2. Kutatócsoport feladatainak szétosztása

5.2.1. A helyzet bemutatása

Ebben a kutatásban a Társadalom, Informatika és Gazdaság kutatócsoport egyik problémáját oldom meg. A probléma lényege a következő: Több olyan feladat van, amiket egyelőre nem lát el senki sem. Természetesen erővel hozzá lehetne mindenkinek rendelni egy-egy feladatot, de akkor ha az emberek nem azt a munkát látják el amit igazán szeretnének, akkor az nem lesz hatékony. Lehetne lehetőséget adni arra, hogy mindenki válasszon magának egy feladatot, például jelentkezés sorrendjében, de ekkor ebben az esetben egy szimpla Mohó algoritmus lenne, ami nem lenne stabil. Ezért ennél a feladtnál is a szoftver alkalmazása a jó megoldás.

Mivel a szoftver elérhető interneten keresztül, így nincs szükség mindenkit egy helyre összehívni ahhoz, hogy döntést tudjunk hozni, elég, ha mindenki belép az oldalra és megadja a saját preferenciáit az adott feladatokhoz. Ezek a feladatok és a kitölthető űrlap látható a 67. ábrán. Itt is hasonló a feladat, mint az előző példában, ahol a tárgyhoz kapcsolódó feladatokat kellett értékelni, a kutatócsoport tagjai súlyozott értékek segítségével sorba rendezik a számukra elfogadható lehetőségeket.

Rendezd sorba ezeket a feladatokat!

Rendezd sorba a számodra megfelelő feladatokat, és súlyozd őket, a súlyozott értékek összege nem lehet 100nál több. Nem kell az összes feladatot kiválasztanod, csak a számodra szimpatikusakat!

Leírás	Össz. hely	Súly
Honlap fejlesztés	[1]	
Honlap fordítás angol	[1]	
Honlap fordítás német	[1]	
Honlap frissítés - aktualitásokkal való feltöltés	[1]	
Cikk nyelvi lektorálás (angol)	[2]	
Cikk nyelvi lektorálás (magyar)	[2]	
Pályázat figyelés	[1]	
Konferencia figyelés	[1]	
TIG könyvtár feltöltése, levelezés kiadókkal	[1]	
Konferencia regisztrációk kezelése	[1]	
Adminisztráció	[1]	
Hajcsár- határidők figyelése, betartatása	[1]	
Jegyzőkönyvvezető	[1]	
Rendezvényszervező	[1]	
Pályázat író	[1]	
Fund raising	[1]	
<input type="button" value="Mentés"/>	Összesen[18]	Összesen: [0]

67 ábra – A kutatócsoport feladatainak űrlapja.

A tagok választásának eredménye a következő táblázatban látható. A táblázat első oszlopa mutatja a tagokat (S1-S15), míg a sorok a kutatócsoport tagok választását, és a hozzájuk adott értékeket mutatja.

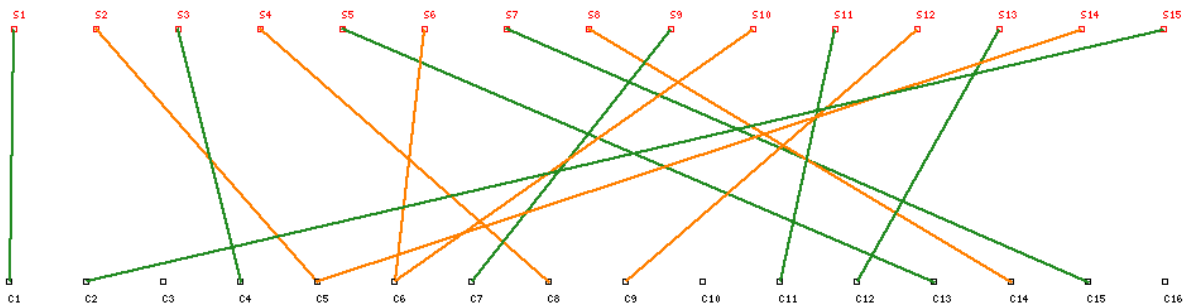
S1	C1(100)							
S2	C5(40)	C9(30)	C10(20)	C13(10)				
S3	C4(25)	C1(25)	C9(20)	C13(15)	C2(15)			
S4	C6(25)	C13(20)	C11(20)	C8(15)				
S5	C13(50)	C14(50)	C12(0)					
S6	C6(60)	C14(20)	C9(20)					
S7	C15(60)	C8(20)	C6(20)	C5(0)	C4(0)	C2(0)	C16(0)	

S8	C1(80)	C14(20)												
S9	C7(40)	C15(25)	C16(25)	C14(10)	C4(0)	C5(0)	C1(0)							
S10	C6(30)	C2(30)	C5(20)	C4(10)	C1(10)	C7(0)	C8(0)							
S11	C11(25)	C14(20)	C9(20)	C4(10)	C2(10)	C12(5)	C1(5)							
S12	C6(25)	C4(20)	C9(15)	C12(10)	C10(10)	C2(10)	C7(10)							
S13	C15(33)	C12(25)	C10(10)	C9(10)	C11(5)	C13(5)	C7(4)							
S14	C14(1)	C11(1)	C5(0)	C6(0)	C4(0)	C3(0)	C16(0)							
S15	C13(40)	C7(35)	C2(15)	C8(10)	C14(0)									

68 táblázat – A tagok választásának eredménye döntési súlyokkal

5.2.2. A helyzet értékelése

Miután beérkezett az összes tagtól a választása, utána egyből látható a rendszerben a feladat megoldása. Ez a megoldás látható a 69. ábrán.



69 ábra – A kutatócsoport feladatának szétosztása

Az ábrán világosan látszik, hogy az algoritmus végeredménye képen minden kutatócsoport tag kapott feladatot, azonban – mivel több feladat volt, mint tag – ezért volt olyan feladat (például a C3 jelzésű), amely nem talált gazdára. A párosítás ezzel együtt sikeresnek tekinthető, mert nem az volt az elsődleges célja, hogy minden feladat el legyen látva, hanem az, hogy minden tag tevékenyen részt vegyen a kutatócsoport életében.

C1	C2	C4	C5	C6	C7	C8	C9	C11	C12	C13	C14	C15
(1)	(1)	(1)	(2)	(2)	(1)	(1)	(1)	(1)	(1)	(1)	(1)	(1)
S1	S15	S3	S2	S6	S9	S4	S12	S11	S13	S5	S8	S7
(1)	(3)	(1)	(1)	(1)	(1)	(4)	(3)	(1)	(2)	(1)	(2)	(1)
			S14	S10								
			(3)	(1)								

70 táblázat – A párosítás eredménye

5.2.3. Tagok elégedettsége

A kutatócsoport tagok a hallgatókhoz hasonlóan nagy érdeklődéssel és lelkesedéssel láttak neki a feladatok értékelésének, és ők is meglepődtek, hogy nem a kutató csoportvezetője osztja ki önkényesen a feladatokat, hanem mindannyian aktívan részt vehetnek a döntéshozatalban. Miután a szoftver segítségével megalkottam a párosításokat és minden tag megkapta a feladatát, egy írásbeli felmérést csináltam az elégedettségről. Ennek eredménye a következő volt. Az írásbeli felmérés megalkotásánál figyelembe vettem az előző fejezetben bemutatott szóbeli felmérés eredményét is.

Az írásbeli kutatás (Lásd: Melléklet – Kérdőív – elégedettség vizsgálat) eredménye.

Hasonlóan a szóbeli kutatáshoz itt is megkérdeztem a résztvevőktől, hogy „Mennyire vagy elégedett a kapott feladattal?” itt 1-től 5-ig lehetett értékelni, az 5 jelentette azt, hogy teljesen elégedett a kapott feladattal, míg az egyes, azt, hogy teljesen elégedetlen. Ezeknek az értékeknek az átlaga 4,49 volt, ami azt jelzi, hogy a hallgatók kifejezetten elégedettek voltak a kapott feladattal. A relatív szórása 16%, így látható, hogy a résztvevők is nagyrészt egymáshoz közeli értékeket adtak meg. A következő kérdés arra vonatkozott, hogy „A feladat, amit kaptál, hányadik volt a preferencia listádban?”, itt az volt a lényeg, hogy emlékszik-e a saját preferencia sorrendjére, és hogy abban hányadik helyen volt végül a végső eredmény. A résztvevők mindannyian eltalálták, hogy hányadik helyen adták meg a végül győztes eredményt. Ennek a kérdésnek az volt a lényeg, hogy a résztvevők preferencia sorrendje változik-e, illetve, hogy mennyire tartják ezekben ezeket a válaszokat.

A következő kérdéscsoportban az oldal használatáról kérdeztem a résztvevőket, itt volt olyan kérdés, hogy „Mennyire volt használható az oldal”. Ennél a kérdésnél is 1-től 5-ig lehetett értékelni a kérdést, hasonlóan az első kérdéshez. A hallgatók értékelése alapján ennek az eredménye a következő lett. A válaszok átlaga 4,84, a szórás értéke 0,49, míg a relatív szóródás értéke 11% lett. Ez alapján látható, hogy a résztvevők a program használatával meg voltak elégedve. A következő kérdés arra vonatkozott, hogy mennyire volt átlátható az oldal, mindent könnyen meg lehetett találni, vagy mennyire ergonomikus a kezelőfelületek kialakítása. Az értékelés itt is 1-től 5-ig tartó skálán volt, ahol 1 volt a legrosszabb, 5 a legjobb érték. A válaszok átlaga 4,11 volt, a szórás 0,85, míg a relatív szórás értéke pedig 21%.

Ahogy a szóbeli beszélgetés alapján a hallgatók, úgy az írásbeli felmérés alapján a résztvevők is inkább meg voltak elégedve a szoftverrel, mint nem, nagyon örültek annak, hogy nem a véletlenül múlik, hogy ki milyen feladatot kap, hanem van lehetőségük megadni valamilyen sorrendet

5.3. További alkalmazási lehetőség

Bár a dolgozat eddigi részében egyetemi környezetben, egyetemi párosítási problémák megoldására lett a 3. fejezetben bemutatott szoftver megalkotva, kialakítása lehetővé teszi, hogy az üzleti szféra problémáira is hatékony megoldást kínáljon. Az alaphelyzet egy három műszakos folyamatos munkarendben működő vállalat munkatársainak ünnepi időszaki munkabeosztásának elkészítése. A műszakokat a 72. táblázat tartalmazza.

	December 24		December 25		December 26	
2014 Karácsonyához kapcsolódó műszakrend	06:00-14:00	S ₁	06:00-14:00	S ₄	06:00-14:00	S ₇
					08:00-16:00	S ₈
	14:00-22:00	S ₂	14:00-22:00	S ₅	14:00-22:00	S ₉
	22:00-06:00	S ₃	22:00-06:00	S ₆	22:00-06:00	S ₁₀

71 táblázat – A választható műszakok (S1-S10) táblázatos összefoglalása

A munkatársak (E1, E2, ...E10) az egyszerű Mohó algoritmus eredményeként a következő beosztást kapták volna:

	Választás	Helyezés	Érték
E1	S10	1	25
E2	S8	1	26
E3	S9	2	15
E4	S7	1	27
E5	S2	4	8
E6	S6	2	16
E7	S1	10	1
E8	S5	3	9
E9	S3	4	7
E10	S4	6	10

Összesen: **144**

72 táblázat – A Mohó algoritmus segítségével megalkotott párosítás eredménye

Ez természetesen tükrözi a munkavállalók preferenciáit, de nem teljes mértékben, hiszen az első munkavállaló kivételével minden munkavállaló csak a többiek által meghagyott műszakok közül válogathatott.

Annak érdekében, hogy a problémát, a korábbiakban már igazolt – hatékonyabb algoritmus segítségével lehessen megoldani, szükség volt a munkavállalók preferenciáira, melyeket a 73. táblázatban összegeztem. A munkatársaknak minden műszakot értékelniük kellett annak megfelelően, hogy mennyire preferálják azokat. Annak érdekében, hogy a különböző műszakok preferencia sorrendjét is megalkothassam, a munkatársaknak nem csupán sorba kellett rendezniük a műszakokat a preferenciáik szerint, de minden műszakhoz hasznossági értéket is hozzá kellett rendelniük úgy, hogy az összes műszakhoz rendelt értékek összege 100 legyen, de egyetlen műszak se kaphasson 1-nél kisebb értéket.

E1	S10 (25)	S6 (21)	S8 (19)	S7 (13)	S4 (9)	S3 (8)	S2 (6)	S9 (5)	S5 (2)	S1 (1)
E2	S8 (26)	S9 (16)	S4 (12)	S10 (11)	S3 (11)	S2 (9)	S6 (8)	S7 (6)	S5 (6)	S1 (1)
E3	S10 (42)	S9 (15)	S7 (14)	S8 (12)	S6 (8)	S5 (5)	S2 (3)	S3 (3)	S4 (1)	S1 (1)
E4	S7 (27)	S6 (14)	S2 (12)	S10 (11)	S3 (11)	S5 (10)	S8 (8)	S4 (6)	S9 (6)	S1 (1)
E5	S10 (50)	S7 (18)	S9 (8)	S2 (8)	S4 (6)	S8 (6)	S3 (5)	S6 (4)	S1 (1)	S5 (1)
E6	S10 (49)	S6 (16)	S9 (8)	S8 (7)	S4 (7)	S5 (6)	S2 (5)	S7 (4)	S3 (3)	S1 (1)
E7	S10 (58)	S6 (12)	S9 (11)	S5 (7)	S3 (5)	S7 (5)	S8 (3)	S2 (3)	S4 (1)	S1 (1)
E8	S10 (40)	S9 (20)	S5 (9)	S7 (8)	S2 (8)	S4 (6)	S8 (6)	S6 (4)	S1 (1)	S3 (1)
E9	S10 (49)	S8 (18)	S9 (9)	S3 (7)	S7 (7)	S5 (6)	S2 (3)	S6 (2)	S4 (2)	S1 (1)
E10	S10 (25)	S9 (17)	S6 (15)	S8 (13)	S3 (11)	S4 (10)	S7 (9)	S2 (3)	S5 (2)	S1 (1)

73 táblázat – A Mohó algoritmus segítségével alkotott párosítás értéke

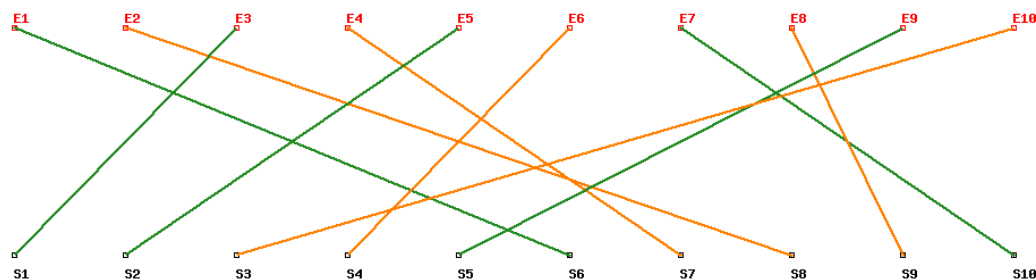
Amint a táblázatból is látszik, a Mohó algoritmus 7 munkavállaló számára járt szuboptimális eredménnyel, tehát a párosítást hatékonytalannak tekinthetjük.

A szoftver segítségével mind a Gale-Shapley, mind a Boston algoritmus futtatásra került. A párosításokat a 75. és a 77 ábrán látható diagramok mutatják.

	Választás	Helyezés	Érték
E1	S6	2	21
E2	S8	1	26
E3	S1	10	1
E4	S7	1	27
E5	S2	4	8
E6	S4	5	7
E7	S10	1	58
E8	S9	2	20
E9	S5	6	6
E10	S3	5	11

Összesen: **185**

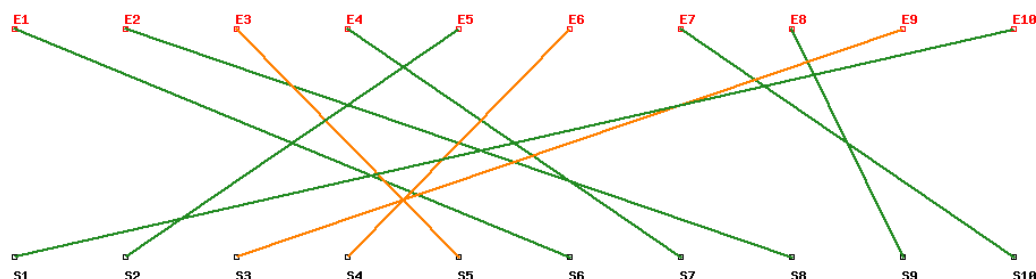
74 táblázat – A Gale-Shapley algoritmus segítségével megalkotott párosítás eredménye



75 ábra – A Gale-Shapley algoritmus segítségével megalkotott párosítás eredménye

	Választás	Helyezés	Érték
E1	S6	2	21
E2	S8	1	26
E3	S5	5	5
E4	S7	1	27
E5	S2	4	8
E6	S4	5	7
E7	S10	1	58
E8	S9	2	20
E9	S3	4	7
E10	S1	10	1
Összesen:			180

76 táblázat – A Boston algoritmus segítségével megalkotott párosítás eredménye



77 ábra – A Boston mechanizmus segítségével megalkotott párosítás eredménye

Amint az ábrákon is látszik, a két algoritmus eltérő párosítást eredményezett. Mindkettő stabilnak tekinthető, azonban – a korábbi kutatási eredményekkel összhangban – a Gale-Shapley algoritmus magasabb összesített hasznosságot (185) eredményezett, mint a Boston algoritmus (180). Az így megkapott hasznosság azonban ezzel az algoritmus-sal szignifikánsan magasabb, mint a Mohó algoritmus segítségével kialakított párosításé (144).

Konklúzió

A vállalati életben éppúgy találkozhatunk párosítási problémákkal, melyek párosításelméleti algoritmusok segítségével hatékonyan (hatékonyabban) oldhatók meg. A 12 fejezetben bemutatott műszakkiosztás csupán egyike a lehetséges alkalmazási területeknek. A párosításelméleti algoritmusok, illetve maga a dolgozatban bemutatott szoftver erőforrás elosztási, szabadságolási, kinevezési, továbbképzési problémákat is képes kezelni. Jó alapot biztosít a megfelelő embert a megfelelő helyre munkaszervezési elv kialakításához, és hatékonyan hozzájárul a tudatosabb feladatvállalás, valamint az ennek következtében kialakuló tudatosabb, éberebb, szervezeti szempontból biztonságosabb munkavégzés létrejöttéhez is.

6. ÖSSZEFOGLALÁS, TÉZISEK

Az értekezéseimben bemutatott kutatómunka új tudományos eredményeit az alábbi tézisekben foglalom össze:

1. Tézis

A különböző megvizsgált helyzetek alapján megállapítottam, hogy a jelenleg használt random döntéseknél jobb megoldást ad bármelyik párosításelméleti algoritmus tudatos alkalmazása. [152]

2. Tézis

Bebizonyítottam, hogy amennyiben létrehozható stabil párosítás, - a Gale-Shapley algoritmus és a Bostoni Mechanizmus is megtalálják azt a stabil párosítást – az, a kognitív és pszicho-szociális rendezettség által a rendszer stabilitását is támogatja. [152], [153], [154]

3. Tézis

Bebizonyítottam, hogy amennyiben az egyik fél nem ad meg saját preferenciát akkor legenerálható számára egy racionális preferencia a másik fél preferenciája alapján, és így a helyzetre alkalmazhatóak a párosításelméleti algoritmusok. [153], [155]

4. Tézis

Bebizonyítottam, hogy kardinális preferenciák esetén a párosítási feladat megoldható szállítási feladatként is, amennyiben elfogadjuk, hogy az általa létrehozott megoldás nem lesz stabil. [155]

5. Tézis

Meghatároztam olyan mutatószámokat, amikkel lehetőség nyílik az általam vizsgált helyzetekben az algoritmusok értékelésére, ilyen mutatószám volt például a kurzusfelvétel esetén a kurzusok irányából számított összesített hasznosság. [155]

6. Tézis

Kimutattam, hogy kardinális preferenciák esetén összesített hasznosság szempontjából a Gale-Shapley algoritmus a leghatékonyabb. [155]

7. Tézis

Kérdőíves kutatás segítségével bebizonyítottam, hogy a párosítási feladat peremfeltételeinek megfelelő módon történő beállításával, növelhető a döntések eredményessége. [157]

8. Tézis

Bebizonyítottam, hogy az általam kialakított szoftver ergonomikus, és számos olyan helyzet létezik ahol a jelenleginél jobb, és a párosításban résztvevők számára elfogadhatóbb párosítás születhet a használatával. [156]

9. Tézis

Bebizonyítottam, hogy az általam készített szoftver nem csupán alap párosítási helyzetekben működik, de speciális szakértelem nélkül is könnyen adaptálható különböző párosítási helyzetekre (4.3, és 5.1. fejezet). [156]

10. Tézis

Több megvizsgált probléma esetén bebizonyítottam, hogy ha a résztvevők az eddig alkalmazott algoritmusok helyett ezt a szoftver alkalmazzák, és ennek segítségével párosításelméleti algoritmusok segítségével oldják meg a problémát, akkor sokkal elégedettebbek lesznek. (4.3. és 5.1. fejezet) [156]

11. Tézis

Kimutattam, hogy a párosítási algoritmusok, illetve a segítségükkel létrehozható stabil párosítások racionális erőforrás / feladatelosztást tesznek lehetővé. [152], [153], [154]

12. Tézis

Bebizonyítottam, hogy a megfelelően megválasztott párosításelméleti algoritmus racionális indokolhatósága és átláthatósága, a megfelelő döntéstámogató rendszer növeli a felhasználók elégedettségét és a döntés elfogadottságát. [156]

7. AZ EREDMÉNYEK HASZNOSÍTÁSA, TOVÁBBFEJLESZTÉSI LEHETŐSÉGEK

A tézisekben bemutatott kutatásaimat nagyrészt egyetemi környezetben végeztem, (Hivatkozás: [152] [153] [158] [157] [155] [154]) azonban a párosítási problémák nem csak itt jelennek meg, hanem a mindennapi életnek (szervezeti és magán) is szerves részét képezik. Párosítási probléma a magán szférában például a gyerekek különórákra való szállítása, a háztartási teendők elosztása és a nyári szabadságok gyerekekhez való igazítása. A munka világában is számos esetben találkozhatunk párosítási problémákkal. Ilyenek az erőforrás és feladatelosztási helyzetek, illetve azok a döntési szituációk, amikor kiküldetésekről, kinevezésekről, áthelyezésekről kell döntést hozni.

A párosításelméleti algoritmusok tudatos használata azonban ezekben a szférákban sem prevalensebb, mint az egyetemi életben. A random hozzárendelések pedig, - amellett, hogy sokszor még a korlátozott racionalitásnak sem tesznek eleget, - ritkán stabilak, és gyakran vezetnek szuboptimális megoldásokhoz.

A problémák stabil párosításelméleti algoritmusok segítségével történő tudatos kezelése túlzott elvárás lenne nem csupán a magán szféra problémáit illetően, de sokszor a szervezeti, intézményi helyzetekben is. Egy szoftveres megoldás azonban számos helyzetben megelőzheti, megoldhatja a problémát. A dolgozatban bemutatott szoftver segítségével nagyon egyszerűen meghatározható lenne például, hogy ki mikor mehet szabadságra, kihez milyen feladatokat érdemes hozzárendelni [156].

A visszajelzések alapján a szoftver kezelői felülete egyértelmű, és könnyen használható, ergonomikus. A felhasználóbarát jellegén túl azonban előnye, hogy külső beavatkozás nélkül is képes párosításokat létrehozni, valamint az, hogy könnyen adaptálható bármilyen helyzetre, legyen szó egy az egyhez, több az egyhez, vagy több a többhöz típusú párosítási problémáról.

IRODALOMJEGYZÉK

- [1] R. E. Freeman, *Strategic Management: A stakeholder approach*, Boston: Pitman, 1984.
- [2] R. Ackoff, *Creating the Corporate Future: Plan or be Planned*, New York: John Wiley, 1981.
- [3] A. H. Booz, *Convergence of Enterprise Security Organizations*, ASIS, ISSA. , 2005.
- [4] Risk and Insurance Management Society (RIMS), „Enterprise Risk Management Survey,” 03 2013. [Online]. Available: https://www.rims.org/resources/RIMStore/Documents/2013_RIMS_Compensation_Survey.pdf. [Hozzáférés dátuma: 05 12 2015].
- [5] A. Maslow, *A lét pszichológiája felé*, Budapest: Ursus Libris, 2003.
- [6] K. Lazányi, „A SZERVEZETI BIZTONSÁG ÉS A MUNKAHELYI STRESSZ KAPCSOLATA.” Taylor: *Gazdálkodás- és Szervezéstudományi Folyóirat: A Virtuális Intézet Közép-Európa Kutatására Közleményei*, 2016.
- [7] M. Enyedi, *Döntéelmélet*, Budapest: BMF KGK, 2005.
- [8] V. H. VROOM, *Work and Motivation*, New York: John Wiley & Sons, 1964.
- [9] J. S. ADAMS, „Toward an Understanding of Inequity,” *Journal of Abnormal Psychology*, %1. kötet November, p. 422–436, 1963.
- [10] K. Lazányi, „A biztonsági kultúra,” Taylor: *Gazdálkodás- és Szervezéstudományi Folyóirat - A Virtuális Intézet Közép-Európa Kutatására Közleményei*, %1. kötet7 (1.2), pp. 398-405, 2015.
- [11] F. H.-T. C. Trompenaars, *Riding The Waves of Culture: Understanding Diversity in Global Business*, New York.: McGraw –Hill, 1997.
- [12] H. E. Schein, *Organizational Culture and Leadership*, San Francisco: Jossey-Bass, 1985.
- [13] P. Buckowski, *The Social Horizon of Knowledge*, Amsterdam: Rodopi, 1984.

- [14] J. Gribbin, Schrödinger macskája - Kvantumfizika és valóság, Budapest: Akkord Kiadó, 2012.
- [15] C. McNamara, „Organizational culture,” 2006. [Online]. Available: www.managementhelp.org. [Hozzáférés dátuma: 27 03 2016].
- [16] A. M. K. a. S. K. Tsohou, „Analyzing the role of cognitive and cultural biases in the internalization of information security policies: Recommendations for information security awareness programs,” *Computers & Security*, %1. kötet52, 2015.
- [17] B. F. a. S. L. Paul SLOVIC, „ Behavioral Decision Theory Perspectives on Risk and Safety.” *Acta Psychologica* , %1. kötet56, pp. 183-203, 1984.
- [18] S. a. A. A. Hügelschäfer, „On confident men and rational women: It’s all on your mind (set).” *Journal of Economic Psychology*, %1. kötet41, pp. 31-44, 2014.
- [19] P.-C. e. a. Liao, „Influence of person-organizational fit on construction safety climate.” *Journal of Management in Engineering* , %1. kötet31(4), pp. 14-49, 2013.
- [20] J. e. a. De Fine Licht, „When does transparency generate legitimacy? Experimenting on a context-bound relationship.” *Governance* , %1. kötet27.1, pp. 111-134, 2014.
- [21] M. Persson, P. Esaiasson és M. Gilljam, „The effects of direct voting and deliberation on legitimacy beliefs: an experimental study of small group decisionmaking.” *European Political Science Review*, %1. kötet5.03, pp. 381-399, 2013.
- [22] M. EREZ, P. C. EARLY és C. L. HULIN, „The Impact of Participation on Goal Acceptance and Performance: A Two-Step Model,” *Academy of Management Journal*, %1. kötetMarch, p. 50–66, 1985.
- [23] P. C. EARLY, G. B. NORTHCRAFT, C. LEE és T. R. LITUCHY, „Impact of Process and Outcome Feed-back on the Relation of Goal Setting to Task Performance,” *Academy of Management Journal*, %1. kötetMarch, p. 87–105, 1990.
- [24] L. Bartha, Pszichológiai alapfogalmak kis enciklopédiája, Budapest: Tankönyvkiadó, 1987.

- [25] P. Z. Zoltainé, *Döntéelmélet*, Budapest: Alinea Kiadó, 2005.
- [26] M. Enyedi, *Bevezetés a döntéelméletbe*, Budapest: Ligatura kiadó, 1997.
- [27] J. Kindler, *Fejezetek a döntéelméletből*, Budapest: BKE-Aula Kiadó, 1991.
- [28] J. Kornai, *Anti-equilibrium*, Budapest: Közgazdasági és Jogi Könyvkiadó, 1971.
- [29] H. Simon, *Administrative Behavior* (3rd ed.), New York: The Free Press, 1976.
- [30] H. Simon, *The New Science of Management Decisions*, New York: Harpere and Brothers, 1960.
- [31] S. H. Haeckel, *Presentation to the information planning*, Cambridge: Marketing science Institute, 1987.
- [32] Renn, „Concept of Risk: A Classification,” in *Social Theories of Risk*, G. Krimsky, Szerk., Westport, Praeger, 1992, pp. 53-82.
- [33] I. Krómer, „Természeti katasztrófák: Kockázatok és Bizonytalanságok,” *ELEKTROTECHNIKA*, %1. kötet104: (11), pp. 19-24, 2011 .
- [34] P. Slovic, „Perception of risk,” *Science*, %1. kötet236, pp. 280-285, 1987.
- [35] M. Power, *Organized Uncertainty: Designing a World of Risk Management*, Oxford: Oxford University Press, 2008.
- [36] E. H. Bowman, „A Risk/Return Paradox of Strategic Management,” *Sloan Management Review*, %1. kötet21, pp. 17-33, 1980.
- [37] E. H. Bowman, „Risk Seeking by Troubled Firms,” *Sloan Management Review*, %1. kötet23, pp. 33-42, 1982.
- [38] P. Medvegyev, „Vélekedések kockázatról és bizonytalanságról,” *Bankszövetség*, %1. kötet10, 2011.
- [39] A. Kaufmann, *A döntés tudománya. Bevezetés a praxeológiába*, Budapest: Közgazdasági és Jogi Könyvkiadó, 1982.
- [40] A. Chikán, *Operációkutatás és döntéelmélet 2. Bevezetés a döntéelméletbe (Főiskolai jegyzet)*, Budapest: Műszaki Könyvkiadó, 1978.

- [41] G. A. Miller, The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, *The Psychological Review*, 1956, pp. 81-97.
- [42] G. J. March, *Bevezetés a döntéshozatalba*, Budapest: Panem kiadó, 2000.
- [43] D. Kahnemann és A. Tversky, „Prospect theory: An Analysis of decision under Risk,” *Econometria*, %1. kötet47, %1. szám2, 1979.
- [44] M. H. Bazerman, *Judgment in managerial decision making*, New York: John Wiley & sons, 1990.
- [45] T. Sterbenz, *Korlátozott racionalitás a sportmenedzseri döntésekben (doktori (PhD) értekezés)*, Sopron: Nyugat-Magyarországi Egyetem, 2007.
- [46] H. Simon, *A vezetői döntés új tudománya*, Budapest: Statisztikai Kiadó Vállalat, 1982.
- [47] J. W. Forrester, *Industrial Dynamics*, MIT Press, 1961.
- [48] E. McDermid, „A $3/2$ approximation algorithm for general stable marriage,” in *Automata, Languages and Programming*, Glasgow, University of Glasgow, 2009, pp. 689-700.
- [49] D. Manlove, R. Irving, K. Iwama, S. Miyazaki és Y. Morit, „Hard variants of stable marriage,” *Theoretical Computer Science*, p. 261–27, 2002.
- [50] R. Gibbons, *Bevezetés a játékelméletbe*, Budapest: Nemzeti Tankönyvkiadó Rt., 2005.
- [51] A. Simonovits, *Bevezetés a Játékelméletbe: vázlat*, Budapest: MTA Közgazdaságtudományi Kutatóközpont, 2007.
- [52] J. Neumann, „Zur Theorie der Gesellschaftspiele.” *Mathematische Annalen*,, %1. kötet100, pp. 295-320, 1928.
- [53] L.-G. Svensson, „Strategy-Proof Allocation of Indivisible Goods,” *Social Choice and Welfare*, pp. 557-567, 1999.
- [54] J. –. M. O. Neumann, *Theory of Games and Economic Behavior*, 3 szerk., Princeton: Princeton University Press, 1953.

- [55] Á. L. Kóczy, „A Neumann-féle játékelmélet,” *Közgazdasági Szemle*, %1. kötet1, pp. 31-45, 2006.
- [56] F. -. P. M. -. S. A. -. S. T. Forgó, „Játékelmélet (elektronikus jegyzet),” Budapest, 2005.
- [57] D. B. Gillies, „Solutions to general non-zero-sum games,” *Tucker–Luce*, p. 47–85, 1959.
- [58] L. S. Shapley, „On balanced sets and cores,” *Naval Research Logistics Quarterly*, %1. kötet14, p. 453–460, 1967.
- [59] W. F. Lucas, „A game with no solution.,” *Bulletin of the American Mathematical Society*, %1. kötet74, p. 237–239. , 1967.
- [60] O. Bondareva, „Some Applications of Linear Programming Methods to the Theory of Cooperative Games,” *Problemy Kybernetiki*, %1. kötet10, pp. 119-139., 1963.
- [61] L. Zhou, „A New Bargaining Set of an N-Person Game and Endogenous Coalition Formation,” *Games and Economic Behavior*, %1. kötet6, pp. 512-526, 1994.
- [62] P. Shenoy, „On Coalition Formation: A Game-Theoretical Approach,” *International Journal of Game Theory*, %1. kötet8, pp. 133-164., 1979.
- [63] E. W. Packel, „A stochastic solution concept for n-person games.,” *Mathematics of Operations Research*, %1. kötet6, p. 349–362, 1981.
- [64] A. Sengupta és K. Sengupta, „Viable proposals,” *International Economic Review*, %1. kötet35, p. 347–359, 1994.
- [65] L. Á. Kóczy és L. Lauwers, „The Minimal Dominant Set is a Non-Empty Core-Extensio,” *Games and Economic Behavior*, %1. kötet61(2), p. 277–298., 2007.
- [66] M. Agastya, „Perturbed adaptive dynamics in coalition form games,” *Journal of Economic Theory*, %1. kötet89, p. 207–233, 1999.
- [67] Y. Yang, „On the accessibility of the core,” *Games and Economic Behavior*, %1. kötet69, pp. 194-199, 2010.

- [68] A. Abdulkadiroglu és T. Sönmez, „Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems,” *Econometrica*, %1. kötet66, %1. szám, pp. 689-701, 1998.
- [69] L. Ehlers, „Coalitional Strategy-Proof House Allocation,” *Journal of Economic Theory*, %1. kötet105(2, pp. 298-317., 2002.
- [70] H. Ergin, „Consistency in House Allocation Problem,” *Journal of Mathematical Economics*, %1. kötet34, pp. 77-97, 2000.
- [71] D. Gale és L. S. Shapley, „College admissions and stability of marriage,” *American Mathematical Monthly*, %1. kötet69, pp. 9-15, 1962.
- [72] A. E. Roth, „Stability and Polarization of Interests in Job Matching,” *Econometrica*, %1. kötet52, pp. 47-57, 1984/b.
- [73] A. E. Roth, „The Economics of Matching: Stability and Incentives,” *Mathematics of Operations Research*, %1. kötet7, pp. 617-628, 1982.
- [74] A. E. Roth és A. Postlewaite, „Weak versus Strong Domination in a Market with Indivisible Goods,” *Journal of Mathematical Economics*, %1. kötet4, pp. 131-137, 1977.
- [75] R. Irving és L. P., „The complexity of counting stable marriages,” *SIAM Journal on Computing*, %1. kötet15(3), p. 655–667, 1986.
- [76] R. Irving, „Stable marriage and indifference,” *Discrete Applied Mathematics*, %1. kötet48, p. 261– 272, 1994.
- [77] Z. Király., „Better and simpler approximation algorithms for the stable marriage problem.,” in *Proceedings of ESA '08: the 16th Annual European Symposium on Algorithms*, 2008.
- [78] A. Hylland és R. Zeckhauser, „Internships,” *J. Assoc. American Medical Colleges*, %1. kötet22, pp. 45-46, 1947.
- [79] A. Hylland és R. Zeckhauser, „The Internship Matching Plan,” *J. Medical Educ.*, %1. kötet27, p. 46, 1952.

- [80] M. Balinski és T. Sönmez, „A tale of two mechanisms: Student placement,” *Journal of Economic Theory*, %1. kötet84, p. 73–94, 1999.
- [81] W. Darley, „The Seventh Annual Report of the National Intern Matching Program,” *J. Medical Educ.*, %1. kötet34, pp. 38-46, 1959.
- [82] J. S. Graettinger és E. Peranson, „The Matching Program,” *New England J. Medicine* , %1. kötet304, pp. 1163-65., 1981.
- [83] A. E. Roth, „The evolution of the labor market for medical interns and residents: a case study in game theor,” *Journal of Political Economy*, %1. kötet6, p. 991 – 1016, 1984.
- [84] J. M. Stalnaker, „The Matching Program for Intern Placement: The Second Year of Operation,” *J. Medical Educ.* , %1. kötet28, pp. 13-19, 1953.
- [85] J. Turner, „Intern Selection: Wanted, an Orderly Plan.,” *J. Assoc. American Medical Colleges*, %1. kötet20, pp. 26-32, 1945.
- [86] A. E. Roth és E. Peranson, „The redesign of the matching market for American physicians: some engineering aspects of economic design.,” *The American Economic Review*, %1. kötet89, p. 748 – 752, 1999.
- [87] I. M. Rutkow és A. H. Glasgow, „How Medical Students View the Application and Reviewing Procedure for Surgical Residency,” *J. Medical Educ.*, %1. kötet53, %1. szám , pp. 5-7, 1978.
- [88] L. E. Dubins és D. A. Freedman, „Machiavelli and the Gale-Shapley Algorithm,” *American Mathematical Monthly*, %1. kötet88(7), pp. 485-494, 1981.
- [89] L. Zhou, „On a conjecture by Gale about one-sided matching problems,” *Journal of Economic Theory*, %1. kötet52(1), pp. 123-135, 1990.
- [90] J. Schummer, „Eliciting Preferences to Assign Positions and Compensation,” *Games and Economic Behavior*, %1. kötet30(2), pp. 293-318, 2000.
- [91] A. Abdulkadiroglu és T. Sönmez, „House Allocation with Existing Tenants,” *Journal of Economic Theory*, %1. kötet88, pp. 233-260, 1999.

- [92] L. Ehlers, B. Klaus és S. Pápai, „Strategy-proofness and population-monotonicity in house allocation problems,” *Journal of Mathematical Economics*, %1. kötet38(3), p. 329–339, 2002.
- [93] E. Miyagawa, „House Allocation with Transfers,” *Journal of Economic Theory*, %1. kötet100(2), pp. 329-355, 2001.
- [94] E. Miyagawa, „Strategy-Proofness and the Core in House Allocation Problems,” *Games and Economic Behavior*, %1. kötet38(2), pp. 347-361, 2002.
- [95] G. Demange és D. Gale, „A Strategy-proof Allocation Mechanism for Two-sided Matching Markets,” in *Mimeographed. Paper presented at the IMSSS-Economics workshop, Stanford, Calif., Stanford Univ., 1983.*
- [96] D. Gale és M. Sotomayor., „Some remarks on the stable matching problem.,” *Discrete Applied Mathematics*, %1. kötet11, p. 223–232, 1985.
- [97] L. S. Shapley és H. Scarf, „On Cores and Indivisibility,” *J. Math. Econ*, %1. kötet1, pp. 23-28., 1974.
- [98] Romero és Medina, „Implementation of stable solutions in a restricted matching market,” *Review of Economic Design*, %1. kötet3(2), pp. 137-147, 1998.
- [99] M. K. I. Halldórsson, S. Miyazaki és H. Yanagisawa, „Randomized approximation of the stable marriage problem,” *Theoretical Computer Science*, %1. kötet325(3), pp. 439-465, 2004.
- [100] Elias, A. David és E. Roy., „Matching of Couples in the NRMP,” *New England J. Medicine*, %1. kötet302, pp. 1425-26., 1980.
- [101] A. Roth, „The College Admissions Problem is Not Equivalent to the Marriage Problem,” *Journal of Economic Theory*, %1. kötet36, pp. 277-288, 1985.
- [102] M. Halldórsson, K. Iwama, S. Miyazaki és H. Yanagisawa., „Improved Approximation of the stable marriage problem.,” *ACM Transactions on Algorithms*, %1. kötet3(3), 2007.

- [103] A. Abdulkadiroğlu, P. Pathak és E. Roth, „The New York City High School Match,” *American Economic Review, Papers and Proceedings*, %1. kötet95, pp. 364-367, 2005/b.
- [104] M. Baïou és M. Balinski., „Student admissions and faculty recruitment,” *Theoretical Computer Science*, %1. kötet322(2):, p. 245–265, 2004.
- [105] M. Schneider, P. Teske és M. Marschall, *Choosing Schools: Consumer Choice and the Quality of American Schools*, Princeton: Princeton University Pres, 2000.
- [106] A. E. Roth, „New physicians: a natural experiment in market organization,” *Science*, %1. kötet250, pp. 1524 - 1528, 1990.
- [107] R. W. Irving és D. F. Manlove, „ Finding large stable matchings,” *Journal of Experimental Algorithmics*, %1. kötet14, 2009.
- [108] J. R. Henig, *Rethinking School Choice, Limits of the Market Metaphor*, Princeton : Princeton University Press, 1994.
- [109] Office of Educational Research and improvement, *Getting Started: How Choice Can Renew Your Public Schools*, Washington, D.C.: U.S. Government Printing Office, 1992.
- [110] A. Abdulkadiroğlu és T. Sönmez, „School choice: A mechanism design approach,” *American Economic Review*, %1. kötet93, p. 729–747, 2003.
- [111] H. Ergin és T. Sönmez, „Games of school choice under the boston mechanism,” *Journal of Public Economics*, %1. kötet90, p. 215–237, 2006.
- [112] A. Abdulkadiroğlu, P. A. Pathak, A. E. Roth és T. Sönmez, „The Boston public school match,” *American Economic Review*, %1. kötet95, p. 368–371, 2005.
- [113] S. Glazerman és R. H. Meyer, „Public school choice in Minneapolis,” in *Midwest approaches to school reform*, T. A. Downes és W. A. Testa, szerk., Chicago, Federal Reserve Bank of Chicago, 1994, p. 110–126.
- [114] P. Biró, „Stabil párosítási modellek és ezeken alapuló központi párosító programok,” *Sigma*, %1. kötet37, p. 153–175, 2006.

- [115] A. Abdulkadiroglu, P. Pathak, A. E. Roth és T. Sonmez, „Changing the Boston School Choice Mechanism,” National Bureau of Economic Research, %1. kötet Working Paper 11965, pp. 1-59, 2006.
- [116] D. Epple és R. E. Romano, „Competition Between Private and Public Schools: Vouchers and Peer Group Effects,” American Economic Review, March, %1. kötet 88(1), %1. szám 6. . ", pp. 33-62, 1998.
- [117] H. B. Leonard, „Elicitation of Honest Preferences for the Assignment of Individuals to Positions.,” The Journal of Political Economy, %1. kötet 91, pp. 461-79, 1983.
- [118] A. E. Roth, „Conflict and Coincidence of Interest in Job Matching: Some New Results and Open Questions,” Mathematics of Operations Research, %1. kötet 10(3), pp. 379-389, 1985.
- [119] A. Hylland és R. Zeckhauser, „The Efficient Allocation of Individuals to Positions,” Journal of Political Economy, %1. kötet 87, pp. 293-314, 1979.
- [120] A. S. J. Kelso és V. P. Crawford, „Job Matching, Coalition Formation, and Gross Substitutes,” Econometrica, %1. kötet 50(6), pp. 1483-1504, 1982.
- [121] A. E. Roth, „A Natural Experiment in the Organization of Entry-Level Labor Markets: Regional Markets for New Physicians and Surgeons in the United Kingdom,” American Economic Review, %1. kötet 81(3), pp. 414-440, 1991.
- [122] E. H. Cole, P. Nickerson, P. Campbell, K. Yetzer, N. Lahaie, J. Zaltzman és J. S. Gill, „The Canadian Kidney Paired Donation Program: A National Program to Increase Living Donor Transplantation,” Transplantation, %1. kötet 99, pp. 985-990, 2015.
- [123] W. C. H. R. F. P. Cantwell L, „Four years of experience with the Australian kidney paired donation programme,” Nephrology, %1. kötet 20, pp. 124-131, 2015.
- [124] S. Flechner, D. Leiser, R. Pelletier, M. Morgieovich, K. Miller, L. Thompson, S. McGuire, J. Sinacore és G. Hil, „The Incorporation of an Advanced Donation Program Into Kidney Paired Exchange: Initial Experience of the National Kidney Registry,” American Journal of Transplantation, %1. kötet 15(10), pp. 2712-7, 2015.

- [125] S. Malik és E. Cole, „State of the Art Practices and Policies in Kidney Paired Donation,” *Current Transplantation Reports*, %1. kötet1(1), pp. 10-17, 2014.
- [126] J. S. Graettinger, „Graduate Medical Education Viewed from the National Intern and Resident Matching Program,” *Journal of Medical Education*, %1. kötet5(1), pp. 703-15, 1976.
- [127] A. Sudarshan és S. Zisook, „National Resident Matching Program,” *New England Journal of Medicine*, %1. kötet305, pp. 525-26, 1981.
- [128] K. J. Williams, V. P. Werth és J. A. Wolff, „An Analysis of the Resident Match,” *New England Journal of Medicine*, %1. kötet304, pp. 1165-66, 1981.
- [129] J. R. Green és J.-J. Laffont, *Incentives in Public Decision Making*, Amsterdam: North-Holland, 1979.
- [130] M. Jackson és P. Sonnenschein, „Overcoming Incentive Constraints by Linking Decisions,” *Econometrica*, %1. kötet75, pp. 241-257, 2007.
- [131] paulgb, „<https://github.com/paulgb/>,” 2010. [Online]. Available: <https://github.com/paulgb/Python-Gale-Shapley>. [Hozzáférés dátuma: 30 szeptember 2014].
- [132] M. Bhojasia, „<http://www.sanfoundry.com/>,” 2014. [Online]. Available: <http://www.sanfoundry.com/java-program-gale-shapley-algorithm/>. [Hozzáférés dátuma: 30 szeptember 2014].
- [133] sephlietz, „[sephlietz.com](http://www.sephlietz.com/),” 2014. [Online]. Available: <http://www.sephlietz.com/gale-shapley/>. [Hozzáférés dátuma: 30 szeptember 2014].
- [134] E. Szücs, *Hasonlóság és model*, Budapest: Műszaki, 1972.
- [135] E. Szücs, *Similitude and Modelling*, Amsterdam: Elsevier, 1980.
- [136] E. Szücs, *Technika és rendszer*, Budapest: Tankönyvkiadó, 1981.
- [137] B. Halassy, *Az adatbázisstervezés alapjai és titka*, Budapest: IDG Magyarországi Lapkiadó Kft, 1995.

- [138] M. d. Raffai, *Információrendszerek fejlesztése és menedzselése*, Győr: Novadat Bt, 2003.
- [139] B. Halassy, *Adatmodellezés*, Budapest: Tankönyvkiadó, 2000.
- [140] A. Keszthelyi, „Remarks on the Efficiency of Information System,” *Acta Polytechnica Hungarica*, %1. kötet7, pp. 153-161, 2009/b.
- [141] P. Szikora, „Better data model makes less work?,” in *Proceedings of FIKUSZ 2009: Symposium for young researchers*, Á. L. Kóczy, Szerk., Budapest, Budapest Tech, 2009/a, pp. 195-203.
- [142] P. Szikora, „Measured Performance of an Information System,” in *MEB 2009 – 7th International Conference on Management, Enterprise and Benchmarking: Proceedings*, G. Kadocsa, Szerk., Budapest, BMF, 2009/b, pp. 267-272.
- [143] A. Keszthelyi, „How to Measure an Information System's Efficiency?,” in *MEB 2009 – 7th International Conference on Management, Enterprise and Benchmarking: Proceedings*, K. György, Szerk., Budapest, Budapesti Műszaki Főiskola, 2009/a, pp. 213-219.
- [144] A. Keszthelyi, „The Role of Data Modelling in Information System Efficiency.,” Budapest, 2009/c.
- [145] C. H. Mullin és D. H. Reiley, „Recombinant estimation for normal-form games, with applications to auctions and bargaining.,” *Games and Economic Behavior*, %1. kötet54, p. 159–182, 2006.
- [146] P. Biró, „Student Admissions in Hungary as Gale and Shapley Envisaged,” *University of Glasgow*, Glasgow, 2008.
- [147] P. Biró, T. Fleiner, R. Irving és D. Manlove, „The College Admissions problem with lower and common quotas,” *Theoretical Computer Science*, %1. kötet411, pp. 3136-3153, 2010.
- [148] R. Irving és D. Manlove, „Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems,” *Journal of Combinatorial Optimization*, %1. kötet16(3), p. 279–292, 2008.

- [149] A. E. Roth és X. Xing, „Turnaround Time and Bottlenecks in Market Clearing: Decentralized Matching in the Market for Clinical Psychologists,” *Journal of Political Economy*, %1. kötet 105(2), pp. 284-329, 1997.
- [150] Á. L. Kóczy, „Központi felvételi rendszerek. Taktikázás és stabilitás,” *Közgazdasági Szemle*, %1. kötet LVI. évf., p. 422–442., 2009.
- [151] Á. L. Kóczy, „A magyarországi felvételi rendszerek sajátosságai Magyarországon,” *Közgazdasági Szemle*, %1. kötetLVII. évf, p. 142–164, 2010.
- [152] P. Szikora, „Hatékonyság-vesztés egy egyszerű centralizált párosítási mechanizmusban,” Komárno, 2013.
- [153] P. Szikora, „Matching Theory Applied - The Case of Distribution of Tasks Among Agents With Preferences,” *Managerial Challenges of the contemporary Society*, %1. kötet7, pp. 146-151, 2014.
- [154] P. Szikora, „How matching algorithms can bring forth more effective decisions in situations with information deficiency,” *Science journal of Busienss and Managament*, %1. kötet3, 2015/c.
- [155] P. Szikora, „Allocating time-bound tasks – an application of matching theory,” *SEFBIS Journal*, %1. kötet2015/1, 2015/b.
- [156] P. Szikora, „Practical application of matching algorithms in case of a task allocation problem,” *SERBIAN JOURNAL OF MANAGEMENT*, %1. kötet10, 2015/e.
- [157] P. Szikora, „Hallgatói döntések racionalitásának vizsgálata párosításelméleti eszközökkel,” *TAYLOR Gazdálkodás és Szervezéstudományi folyóirat*, %1. kötet2015/1, 2015/a.
- [158] P. Szikora, „Párosítás elméleti problémák megoldásának lehetőségei, és a döntések racionalitásának vizsgálata,” *TAYLOR Gazdálkodás és Szervezéstudományi folyóirat*, %1. kötet2015/1, 2015/d.
- [159] G. Szabó és I. Borsos, „Evolution and Extinction of Families in a Cellular Automaton.,” *Physical Review E*, %1. kötet49, p. 5900–5902, 1994.

[160] G. Szabó és A. Szolnoki, „Selfishness, Fraternity, and Other-Regarding Reference in Spatial Evolutionary Games,” *Journal of Theoretical Biology*, %1. kötet299, p. 81–87, 2012.

AZ ÉRTEKEZÉS TÉMÁJÁHOZ KAPCSOLÓDÓ TUDOMÁNYOS KÖZLEMÉNYEK

- [141] P. Szikora, „Better data model makes less work?,” in Proceedings of FIKUSZ 2009: Symposium for young researchers, Á. L. Kóczy, Szerk., Budapest, Budapest Tech, 2009/a, pp. 195-203.
- [142] P. Szikora, „Measured Performance of an Information System,” in MEB 2009 – 7th International Conference on Management, Enterprise and Benchmarking: Proceedings, G. Kadocsa, Szerk., Budapest, BMF, 2009/b, pp. 267-272.
- [152] P. Szikora, „Hatékonyság-vesztés egy egyszerű centralizált párosítási mechanizmusban,” Komárno, 2013.
- [153] P. Szikora, „Matching Theory Applied - The Case of Distribution of Tasks Among Agents With Preferences,” Managerial Challenges of the contemporary Society, %1. kötet7, pp. 146-151, 2014.
- [154] P. Szikora, „How matching algorithms can bring forth more effective decisions in situations with information deficiency,” Science journal of Busienss and Managament, %1. kötet3, 2015/c.
- [155] P. Szikora, „Allocating time-bound tasks – an application of matching theory,” SEFBIS Journal, %1. kötet2015/1, 2015/b.
- [156] P. Szikora, „Practical application of matching algorithms in case of a task allocation problem,” SERBIAN JOURNAL OF MANAGEMENT, %1. kötet10, 2015/e.
- [157] P. Szikora, „Hallgatói döntések racionalitásának vizsgálata párosításelméleti eszközökkel,” TAYLOR Gazdálkodás és Szervezéstudományi folyóirat, %1. kötet2015/1, 2015/a.

[158] P. Szikora, „Párosítás elméleti problémák megoldásának lehetőségei, és a döntések racionalitásának vizsgálata,” TAYLOR Gazdálkodás és Szervezéstudományi folyóirat, %1. kötet2015/1, 2015/d.

ÁBRAJEGYZÉK

1	ábra – A probléma meghatározása [11] alapján.....	9
2	ábra – A probléma meghatározása [11] alapján.....	10
3	ábra – probléma megoldásának módjai [11] alapján	11
4	ábra – Döntési folyamat [15] alapján.....	12
5	ábra - Az információ hierarchiamodellje Haeckel (1987) alapján.....	15
9	ábra – a különböző cselekvési lehetőségek halmaza, [11] alapján	27
10	ábra - Az operációkutatás és annak válogatott problémái	32
11	ábra – Az egyértelmű hozzárendelés	34
12	ábra – A kölcsönösen egyértelmű hozzárendelés	35
13	ábra – A többértelmű hozzárendelés.....	35
14	ábra – A párosítás- és a döntéelmélet kapcsolata	41
17	ábra – A fogalmi szintű modell ábrája.....	56
18	ábra – A logikai tervezés ábrája.....	57
19	ábra – az adminisztrációs oldal kezdő képernyője.....	58
20	ábra – Az események szerkesztésének oldala.....	59
21	ábra - Új esemény hozzáadásának az űrlapja.....	59
22	ábra - Már létező esemény módosításának az űrlapja.....	59
23	ábra – Egy már létező esemény lehetőségei	60
24	ábra – Az események felhasználókhöz és a feladatokhoz tartozó preferencia sorrendje.....	61
25	ábra - Egy adott eseményhez tartozó párosítás grafikus ábrája	62
26	ábra – Egy adott eseményhez tartozó párosítás táblázatos ábrázolása	62
27	ábra – Egy adott eseményhez tartozó lehetőségek.....	63
40	ábra – Párosítás Gale-Shapley algoritmus segítségével (R=934, U=31,11).....	83
41	ábra – Párosítás Boston mechanizmussal (R=804, U=23,53).....	83

47	ábra – 2 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	89
48	ábra – 3 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	90
49	ábra – 4 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	90
50	ábra – 2 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	91
51	ábra – 3 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	91
52	ábra – 4 hely van a kurzuson (az x tengely a kiválasztható helyek száma).....	92
54	ábra - A különböző párosítási algoritmusok által generált párosítások száma a választható helyek függvényében	93
56	ábra - A különböző párosítási algoritmusok által generált hasznosság értékek a választható helyek függvényében	95
59	ábra – A kitöltők születési évük szerinti megoszlása	98
60	ábra – A kitöltők felvételük évében való életkoruk szerinti megoszlása.....	99
61	ábra – A hallgatók megoszlása az alapján, hogy hány helyre adták be a jelentkezésüket.....	99
62	ábra – A hallgatók eredménye a felvételin	101
63	ábra – A szervezeti magatartás tárgyhoz tartozó űrlap	104
65	ábra – A szervezeti magatartás tárgy feladat kiosztásának eredménye	105
67	ábra – A kutatócsoport feladatainak űrlapja.	108
69	ábra – A kutatócsoport feladatának szétosztása	109
75	ábra – A Gale-Shapley algoritmus segítségével megalkotott párosítás eredménye 113	
77	ábra – A Boston mechanizmus segítségével megalkotott párosítás eredménye ...	113

TÁBLÁZATJEGYZÉK

6	táblázat – példa a determinisztikus esetre.....	18
7	táblázat – példa az ismert valószínűségek esetére	19
8	táblázat – példa az ismeretlen valószínűségek esetére.....	22
15	táblázat – A különböző párosítás elmélet algoritmusok összehasonlítása [100]	47
16	táblázat – A párosítási algoritmusok használata Európában, [100]	48
28	táblázat - Hallgatói és fogadó intézményi preferencia sorrendek.....	67
29	táblázat- A jelenleg alkalmazott párosítás eredménye.....	69
30	táblázat – A hallgatóbarát Gale-Shapley algoritmus eredménye.....	70
31	táblázat – Az egyetembarát Gale-Shapley algoritmus eredménye	71
32	táblázat – Az eredeti jelentkezés eredménye	74
33	táblázat – A párosítás eredménye csak a sorrendet megadó hallgatók alapján	75
34	táblázat – A párosítás eredménye minden hallgató alapján.....	76
35	táblázat – Párosítás csak a sorrendet megadó hallgatók alapján.....	78
36	táblázat – A párosítás eredménye minden hallgatóra Boston mechanizmus segítségével.....	79
37	táblázat – A hallgatók preferencia sorrendje, ha minden feladatot sorba rendezhetnek	82
38	táblázat – Hallgatók által a táblánál történő feladatválasztás (R=345, U=60).....	82
39	táblázat – Párosítás Gale-Shapley algoritmussal (R=934, U=31,11).....	82
42	táblázat – Párosítás Boston mechanizmussal (R=804, U=23,53).....	83
43	táblázat – A hallgatói cserék alapján elért „R” pontszámok a különböző algoritmusokkal	84
44	táblázat - Az adatok normális eloszlásának vizsgálata	84
45	táblázat – A statisztikai elemzés eredménye.....	85
46	táblázat – A jelentkezések átlaga	87

53	táblázat - A különböző párosítási algoritmusok által generált párosítások száma a választható helyek függvényében	93
55	táblázat – A különböző párosítási algoritmusok által generált hasznosság értékek a választható helyek függvényében	94
57	táblázat – A vizsgált algoritmusok által generált értékek viszonya	96
58	táblázat – A Levene-próba eredménye	97
64	táblázat – A hallgatók választása döntési súlyokkal	104
66	táblázat – A párosítás eredménye	105
68	táblázat – A tagok választásának eredménye döntési súlyokkal	109
70	táblázat – A párosítás eredménye	109
71	táblázat – A választható műszakok (S1-S10) táblázatos összefoglalása	111
72	táblázat – A Mohó algoritmus segítségével megalkotott párosítás eredménye	111
73	táblázat – A Mohó algoritmus segítségével alkotott párosítás értéke	112
74	táblázat – A Gale-Shapley algoritmus segítségével megalkotott párosítás eredménye	112
76	táblázat – A Boston algoritmus segítségével megalkotott párosítás eredménye ..	113

MELLÉKLETEK

Melléklet –SQL adatbázis létrehozó script

A MySQL kezelő számára az alábbi sql-script definiálja az adatbázist, feltéve, hogy az üres adatbázist a MySQL-adminisztrátor már létrehozta, és a megfelelő adatbázis-jogosultságokat számomra beállította. Ez az alábbi parancsokkal történhet:

```
create database matchingprogram;
grant all on matchinguser.* to 'matchingprogram'@'localhost';
use matchingprogram;
DROP TABLE `ESEMENY_VALASZTAS`;
DROP TABLE `SZEMELY_VALASZTAS`;
DROP TABLE `ESEMENYELEM`;
DROP TABLE `ESEMENY`;
DROP TABLE `SZEMELY`;
```

```
CREATE TABLE `SZEMELY` (
  `SzemelyAzon` bigint(20) unsigned NOT NULL auto_increment,
  `Nev` varchar(32) default "",
  `Neptunkod` varchar(6) default "",
  `JelszoSHA1` varchar(20) default "",
  `Aktiv` tinyint(1) unsigned default 0,
  PRIMARY KEY (`SzemelyAzon`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ESEMENY` (
  `EsemenyAzon` bigint(20) unsigned NOT NULL auto_increment,
  `Megnevezes` varchar(32) default NULL,
  `MaxPont` int(4) unsigned default 100,
  `Aktiv` tinyint(1) unsigned default 0,
  PRIMARY KEY (`EsemenyAzon`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ESEMENYELEM` (
  `EsemenyAzon` bigint(20) unsigned NOT NULL,
  `Sorszam` tinyint(1) unsigned NOT NULL,
  `Megnevezes` varchar(32) default "",
  `MaxValaszt` int(4) unsigned default 0,
  `Aktiv` tinyint(1) unsigned default 0,
  PRIMARY KEY (`EsemenyAzon`,`Sorszam`),
  FOREIGN KEY (`EsemenyAzon`) REFERENCES `ESEMENY` (`EsemenyAzon`)
```



```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `SZEMELY_VALASZTAS` (  
  `EsemenyAzon` bigint(20) unsigned NOT NULL,  
  `Sorszam` tinyint(1) unsigned NOT NULL,  
  `SzemelyAzon` bigint(20) unsigned NOT NULL,  
  `Pontszam` int(4) unsigned default 0,  
  PRIMARY KEY (`EsemenyAzon`,`Sorszam`,`SzemelyAzon`),  
  FOREIGN KEY (`EsemenyAzon`,`Sorszam`) REFERENCES `ESEMENYELEM`  
  (`EsemenyAzon`,`Sorszam`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ESEMENY_VALASZTAS` (  
  `EsemenyAzon` bigint(20) unsigned NOT NULL,  
  `Sorszam` tinyint(1) unsigned NOT NULL,  
  `SzemelyAzon` bigint(20) unsigned NOT NULL,  
  `Pontszam` int(4) unsigned default 0,  
  PRIMARY KEY (`EsemenyAzon`,`Sorszam`,`SzemelyAzon`),  
  FOREIGN KEY (`EsemenyAzon`,`Sorszam`) REFERENCES `ESEMENYELEM`  
  (`EsemenyAzon`,`Sorszam`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Melléklet – Kérdőív – elégedettség vizsgálat

TIG elégedettség vizsgálat

*Kötelező

Nemed *

- Férfi
 Nő

Korod *

- 20-29
 30-39
 40-49
 50-59

Feladat értékelése

Mennyire vagy elégedett a kapott feladattal? *

1 2 3 4 5

Teljesen elégedett Teljesen elégedetlen

A feladat, amit kaptál benne volt-e a preferencia listádban? *

- Igen
 Nem
 Nem emlékszem

A feladat, amit kaptál hányadik volt a preferencia listádban? *

Szerinted lehetett volna jobban is szétosztani a feladatokat? *

Válaszodat indokold!

Mennyire volt könnyen használható az oldal? *

1 2 3 4 5

Teljesen egyértelmű volt Segítséget kellett kérnem hozzá

Ergonómiaailag mennyire volt megfelelő az oldal? *

1 2 3 4 5

Átlátható Átláthatatlan

Mi tetszett legjobban az oldalon?

Mit változtatnál az oldalon?

Melléklet – Kérdőív – Felsőoktatási jelentkezések

Felsőoktatási felvételi

Kérlek töltsd ki ezt a rövid tesztet, a teszt kitöltésének a becsült ideje max. 5 perc.

*Kötelező

Melyik felsőoktatási intézményben tanulsz? *

- Óbudai Egyetem
- Egyszerre több intézményben is tanulok
- Egyéb:

Ha több helyen is tanulsz, akkor kérlek sorold fel itt őket

Milyen tagozaton tanulsz/tál? *

- Nappali
- Levelező
- Távoktatás
- Egyéb:

Nemed *

- Férfi
- Nő

Születési évéd *

számmal pl.1978

Felvételeid éve *

Mikor felvételiztél oda, ahol most tanulsz?

Hány különböző helyre jelentkezél egyszerre?*

Elégnek tartottad a választható helyek számát?*

- Igen
- Nem

Mi határozta meg a választásaidat?*

- Biztosan legyen olyan hely ahova felvesznek
- Legyen olyan is a listán, ahova nagyon szeretnék bekerülni (még ha kevés is az esély, hogy felvesznek)
- Egyéb:

Milyen helyeket adtál meg?*

- Csak állami finanszírozottat
- Csak költségtérítéssel
- Vegyesen
- Egyéb:

Ha lett volna lehetőségem több helyet megadni, akkor a lista*

- végére tettem volna még olyan szakot, ahova biztosan bekerülök
- elejére tettem volna még olyan szakot, ahova kicsi volt az esélyem, hogy bekerülök, de nagyon örültem volna, ha sikerül
- Egyéb:

Az általad hanyadik helyen megjelölt szakra nyertél végül felvételt?*

- Első
- Második
- Harmadik
- Negyedik
- Ötödik
- Egyéb:

Volt sikertelen felvételig, amikor sehova sem vettek fel?*

- Igen
- Nem

Sikertelen felvételi esetén

Miért volt sikertelen a felvételi?*

- Túl kevés helyet adtam meg (Nem használtam ki a lehetőségeket)
- Túl kevés helyet adtam meg (A lehetőségek maximumát választottam)
- Kevesebb pontom lett, mint előre terveztem
- Magasabb pontszám kellett, mint amit terveztem
- Egyéb:

Mikor volt a sikertelen felvételig?*

Évet adj meg (pl. 2010)

Ha több lehetőséget is megadhattam volna, akkor*

- biztosan felvettek volna valahova
- nem adtam volna meg több helyet akkor sem

A sikertelen felvételi után más szempontok szerint válogattad a felvételiben megjelölhető helyeket?

- Igen
- Nem

Ha megváltoztattad a stratégiádat, akkor mit tettél másképp?