



Pécsi Tudományegyetem  
Breuer Marcell Doktori Iskola

**FUNKCIONÁLIS KAPCSOLATOK MODELLEZÉSE  
BLOKK-SZERKEZETŰ EGÉSZSÉGÜGYI ELLÁTÓ  
RENDSZEREK ÉPÍTÉSZETI TERVEZÉSÉBEN**

MESKÓ DIÁNA

- PhD Doktori Értekezés -

Tudományos vezető:

Dr. habil Csébfalvi Anikó Borbála CSc PhD

**Pécs**

**2014.**

# TARTALOMJEGYZÉK

TARTALOMJEGYZÉK .....	ii
ÁBRÁK JEGYZÉKE.....	v
TÁBLÁZATOK JEGYZÉKE .....	vi
DEFINÍCIÓK GYŰJTEMÉNYE .....	vii
1. BEVEZETÉS.....	1
1.1. Motiváció .....	1
1.2. Kutatási célkitűzés .....	4
1.3. Kutatási módszertan.....	5
2. KOMPLEX ÉPÜLETRENDSZEREK FUNKCIONÁLIS TERVEZÉSE ÉS MODELLEZÉSE.....	6
2.1. Komplex épületrendszerek felépítése .....	6
2.1.1. Épületek csoportosításainak áttekintése .....	7
2.1.2. Egy- és több-blokkos épületrendszerek definiálása .....	8
2.1.3. Az épületeken belüli funkcionális egységek.....	10
2.1.4. Közlekedés és kapcsolatok az egyes egységek között .....	10
2.1.5. Megbízható komplex épületek (MKÉ).....	11
2.2. Tervezési módszerek.....	12
2.2.1. Mi is az a tervezés? .....	12
2.3. Épületek modellezése.....	14
2.3.1. Gráfelméleti alapok .....	16
2.3.2. Általános gráf formális leírása .....	16
2.3.3. Kapcsolódó fogalmak.....	17
2.3.4. Épület-komplexumok, épületek, blokkok, funkciók és a közöttük lévő kapcsolatok modellezése .....	18
2.3.5. Útkeresések gráfokban .....	19
2.4. Új tudományos eredmények: 1. Tézis.....	20
3. A MEGBÍZHATÓ ÉPÜLETEK.....	22
3.1. Védelmi stratégiák .....	22
3.1.1. Védelem vagy helyreállítás?.....	23
3.1.2. Hozzárendelt vagy megosztott védelem? .....	24

3.1.3.	Szakasz, szegmens vagy útvédelem? .....	25
3.1.4.	Statikus, dinamikus és adaptív védelem? .....	26
3.1.5.	Hibafüggő vagy hibafüggetlen? .....	26
3.2.	A kutatásom során vizsgált védelmi módszerek.....	26
3.3.	Költségszámítás .....	27
3.3.1.	Minimál-költségű többtermékes folyam probléma .....	28
3.3.2.	Egészértékű lineáris programozás.....	29
3.3.3.	Kapacitás költségfüggvénye .....	30
3.4.	Élkettőzés .....	32
3.5.	Új tudományos eredmények: 2. Tézis.....	34
4.	ÚTVONAL-KERESŐ ALGORITMUSOK.....	36
4.1.	Dijkstra-algoritmus, vagy más néven: legrövidebb-út kereső algoritmus.....	36
4.2.	Egyszerű algoritmusok ismertetése .....	37
4.3.	Meghibásodás esetére felkészült útvonal-kereső algoritmusok .....	41
4.4.	Meghibásodás esetére felkészült útvonal-kereső algoritmusok alkalmazhatósága az épülettervezés folyamatában.....	44
4.4.1.	SPP algoritmus folyamatábrája.....	45
4.4.2.	PDSP algoritmus folyamatábrája.....	48
4.4.3.	SPP-LD és PDSP-LD algoritmus folyamatábrája.....	50
4.5.	Egy egyszerű, ám annál szemléletesebb példa az algoritmusok működésére (lépésről lépésre) .....	52
4.6.	Új tudományos eredmények: 3. Tézis.....	62
5.	AZ ALGORITMUSOK KITERJESZTÉSE TÖBBTARTOMÁNYÚ ESETEKRE .....	63
5.1.	Épületek többtartományú és többtartományú hierarchikus gráf modellezése, összehasonlítása.....	63
5.2.	Új tudományos eredmények: 4. Tézis.....	66
6.	AZ ÚJ EREDMÉNYEK ALKALMAZHATÓSÁGA.....	67
6.1.	Egy példa.....	67
6.1.1.	Példa a megfogalmazott előtervezési fázis 1.-3. lépéseire - dekompozíció.....	68
6.1.2.	Példa a megfogalmazott előtervezési fázis 4. lépésére – modell analízis .....	69
6.1.3.	Példa a megfogalmazott előtervezési fázis 5. lépésére – aggregáció.....	72

6.2. EREDMÉNYEK.....	73
6.2.1. Az épületek maximális átbocsató képességének vizsgálata a három módszer segítségével .....	74
6.2.2. Épületek kihasználtsága.....	77
6.2.3. Algoritmusok futási ideje .....	77
6.3. Új tudományos eredmények: 5. Tézis.....	78
ÖSSZEGZÉS ÉS ÚJ TUDOMÁNYOS EREDMÉNYEK.....	80
KÖSZÖNETNYILVÁNÍTÁS .....	87
TÉZISEKHEZ KAPCSOLÓDÓ PUBLIKÁCIÓK .....	87
EGYÉB PUBLIKÁCIÓK.....	89
IRODALOMJEGYZÉK.....	90

## ÁBRÁK JEGYZÉKE

1. ábra Példa az egy épületből álló egészségügyi ellátó rendszerekre (Pécsi Négyszáz Ágyas Klinika).....	9
2. ábra Példa a több épületből álló, komplex egészségügyi ellátó rendszerekre I. (János kórház, Budapest).....	9
3. ábra Példa a több épületből álló, komplex egészségügyi ellátó rendszerekre II. (János kórház funkcionális egységeinek terve, Budapest).....	10
4. ábra Példa az épületek gráffal történő modellezésére .....	19
5. ábra Részben független megosztott védelmi utakkal ellátott üzemi útvonal megjelenítése .....	23
6. ábra A tartalék kapacitás megosztása: 12 vagy 7 egység .....	25
7. ábra Három kapacitás költség model védelem esetére.....	30
8. ábra Az élkettőzés illusztrációja .....	34
9. ábra SPP-LD és PDSP-LD algoritmus folyamatábrája.....	47
10. ábra PDSP algoritmus folyamatábrája .....	49
11. ábra SPP-LD és PDSP-LD algoritmus folyamatábrája .....	51
12. ábra Példa az épületek gráffal történő modellezésére .....	52
13. ábra Példa az SPP, SPP-LD és PDSP-LD algoritmusok futási eredménye.....	61
14. ábra Az épület legfelső szintű modellezése (épületek és a közöttük lévő kapcsolatok).....	64
15. ábra Az épületek kifejtése, tartományokra osztása, funkció- és kapcsolat-meghatározás – I. ....	64
16. ábra Az épületek kifejtése, tartományokra osztása, funkció- és kapcsolat-meghatározás – II.....	65
17. ábra Többtartományú gráf reprezentációja a komplex épületrendszernek.....	65
18. ábra Egytartományú modellezése egy komplex épületrendszernek.....	66
19. ábra Modell dekompozíció .....	69
20. ábra Példa az épületek gráffal történő modellezésére .....	69
21. ábra SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult gráfkép.....	71
22. ábra Modell aggregáció .....	72
23. ábra Az épület-modellekben fellépő blokkolás az összeköttetések kapacitásának függvényében (16, 22, 30 csomópontos gráf esetében) .....	76
24. ábra Az épület-modellek átlagos kihasználtsága .....	77
25. ábra Az átlagos futási idők logaritmikus skálája .....	78

## TÁBLÁZATOK JEGYZÉKE

1. táblázat Védelmi stratégiák osztályozása .....	24
2. táblázat Védelmi technikák osztályozása .....	27
3. táblázat Példa az algoritmusok működésére (lépésről lépésre) .....	53
4. táblázat SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult költségek.	62
5. táblázat 20. ábra kapcsolati mátrixsza.....	70
6. táblázat 20. ábra kapacitás mátrixsza .....	70
7. táblázat 20. ábra C költség mátrixsza.....	70
8. táblázat Igényvektorok felírása .....	71
9. táblázat SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult költségek.	72
10. táblázat Szimulációhoz használt épület-modellek tulajdonságai .....	73
11. táblázat Az épületmodellekben a referencia és a javasolt algoritmusokkal elérhető maximális átbocsátó képesség vizsgálata.....	75

## DEFINÍCIÓK GYŰJTEMÉNYE

1. definíció Egy-blokkos épületrendszer .....	8
2. definíció Több-blokkos, blokk-szerkezetű komplex épületrendszerek .....	9
3. definíció Megbízható komplex-épületrendszerek (MKÉ) .....	11
4. definíció Előtervezési fázis, előzetes funkcionális modellezés.....	14
5. definíció Épületek gráfokkal történő modellezése .....	19
6. definíció Legrövidebb útkeresés épületekben.....	19
7. definíció Páciens-forgalom, mint többtermékes folyam probléma .....	20
8. definíció Élkettőzött épület-modell .....	32
9. definíció Élkettőzött épület-modell formalizációja.....	33
10. definíció $\gamma_e'' \in \Gamma''$ arányos költség.....	33
11. definíció Védelem nélküli legrövidebb útkeresés ILP megfogalmazása .....	40
12. definíció Élkettőzött algoritmus ILP megfogalmazása .....	43
13. definíció Épületek többtartományú hierarchikus modellezése .....	65

# 1.

## BEVEZETÉS

A korszerű, nagy alapterületen elhelyezkedő, funkcionális blokkokból álló komplex egészségügyi ellátó rendszerek építészeti tervezése során a blokkok optimális elhelyezése, a blokkok közötti kapcsolatok sokasága és sokrétűsége miatt komoly feladat a tervező számára, de kihívást jelent ennek megoldása, mint matematikai probléma is. Ha hiszünk a természeti környezetbe símuló, esztétikus elhelyezés megnyugtató voltában, a természetes táji adottságok kihasználhatóságában, akkor az „optimális” elhelyezés kérdése sokkal erőteljesebben vetődik fel, mint a kis alapterületen összezsúfolt „falanszter-szerű gyógyító-gyárak” esetében.

### 1.1. *Motiváció*

A jelenlegi építészeti tervezési gyakorlat során a tervezés legelső fázisa a vázlatok és változatok elkészítése. Sok alternatíva vetődik fel ebben a szakaszban, ami igényli a teljes nyitottságot. Ekkor még nincsenek jogszabályi kötelezettségek, mégis a tervezés legnagyobb felelősséggel járó mozzanata. Jelen gyakorlat nem terjed ki arra, hogy felmérjék az épületekkel kapcsolatos igényeket, majd az épületekről, a szükséges funkciók és a közöttük lévő kapcsolatok modellezésével funkcionális tervet készítsenek, végül megvizsgálják a tervet például olyan fontos szempontok szerint, mint az épületek megbízhatósága, üzemszerűen tudnak-e akkor is működni, ha az épületen belül meghibásodás lép fel vagy felújítási munkálatok zajlanak. Tűzvédelmi és havári esetekre léteznek előírások, normák és tervezési konvenciók különösen a potenciálisan nagy kockázatú létesítmények esetén (Katona T. J. 2013), azonban nem létezik olyan módszer, mellyel automatikusan, akár a teljes épületrendszerre egyidejűleg meghatározhatók a kimenekítési útvonalak.

Korábban, a megbízható távközlő hálózatok területén folytattam kutatásokat. Kutatásaim megmutatták, hogy mennyire fontos a távközlő hálózatokban a meghibásodások esetén mielőbb védelem biztosítása. Ezt védelmi módszerek



Meskó Diána

alkalmazásával lehet biztosítani, mely védelmi módszerek arra hivatottak, hogy az üzemi útvonal mellett minél kisebb többlet-kapacitással, minél alacsonyabb költségek mellett és mielőbb biztosítsák a védelmi útvonalat a meghibásodás elkerülése céljából. A korábbi, szakirodalmakban megtalálható algoritmusokhoz képest hatékonyabb, ámbar lassabb algoritmusokat fejlesztettem ki, melyek azonban még a távközlő hálózatok esetében is elfogadható futási idővel nyújtanak eredményeket.

A doktori értekezésemben részletesen bemutatom, hogyan kapcsolódnak a távközlő hálózatokban is használatos védelmi módszerek az építészeti kutatásomhoz, hogyan adaptálhatók a védelmet is biztosító útkereső algoritmusok az építészeti tervezésbe, milyen előnyökkel jár az épületek funkcionális tervezése és a funkcióterv elemzése, vizsgálata.

A kutatás során olyan általánosan alkalmazható matematikai modellt vezettem be, melyet az optimalizálásához kapcsolódó módszertani elemek összekapcsolásával és továbbfejlesztésével értem el. A matematikai modell minden olyan területen alkalmazható, melyen az üzemszerű működéshez üzemi útvonalakat és a hozzájuk tartozó védelmi, tartalék útvonalakat szükséges definiálni, tervezni, megvalósítani. Ilyen terület lehet egy úthálózat, egy mérnöki rendszer, az adatátviteli rendszerek, illetve a témám területe is: épület-rendszerek.

Épület rendszerek esetében az úthálózat együttesen jelenti mind az épületek közötti közlekedési utakat, mind pedig az épületen belüli közlekedés minden módozatát – folyósó, közlekedő, lift, lépcsőház, de ide tartoznak egészségügyi intézmények esetén például a várótermek és a műtő helyiségek is.

Az egészségügyi ellátó rendszerek esetében kiemelt fontosságú a megfelelő úthálózat és biztonsági alternatívák felállítása, hiszen ezen létesítmények esetében emberek élete, sorsa függhet egy-egy helyesen vagy helytelenül megválasztott útvonaltól és attól, hogy a tervezéskor gondoskodtak-e arról, hogy az üzemszerű működéstől eltérő esetek: meghibásodások, fenntartási, állagmegőrzési munkálatok idején is teljes, maradéktalanul működő rendszert alkosson az intézmény.

Kutatásom során azt vizsgáltam, hogy milyen hatással van az épület megbízhatóságára egy, az építészeti tervezést megelőző funkcionális tervezési fázis, az épület komplexumok modellezése, a megbízhatóság szempontjából történő vizsgálata.

Ez a tervezési lépés az építészeti tervezést megelőző, proaktív, a problémás helyzetek fellépését előre megoldó módszerként, mintegy előtervezési lépésként fogható fel. Ennek

Meskó Diána

keretében a funkciók egymáshoz rendelését, szükségszerű és lehetséges összekapcsoltságát kell megfogalmazni. Ez azt jelenti, hogy a legkisebb helységtől a legnagyobbig, mint például a mosdóktól és a takarító felszerelés tároló helységétől a központi auláig minden funkciót listába szednek, majd meghatározzák ezen funkciók egymással való összeköttetéseit. Itt kell kitérni a szükséges és elégséges kapcsolatok meghatározására is, amit az egyes kapcsolatokhoz rendelünk.

Összefoglaltam a szakirodalomban jelenleg megtalálható útvolan-kereső algoritmusokat, melyekkel közlekedési utakat és az utak meghibásodása esetére szolgáló védelmi utakat definiálhatunk. Az általam alkotott modellen azután ilyen útvonalkereső algoritmusokkal definiálhatunk közlekedési utakat és az utak meghibásodása esetére szolgáló védelmi utakat. Ilyen meghibásodás fordulhat elő például abban az esetben, amikor egy lift nem üzemel és gondoskodni kell arról, hogy az orvosok, páciensek és hozzátartozóik probléma mentes alternatívákat tudjanak igénybe venni. De meg kell említeni az olyan rendkívüli helyzeteket is, mint például a tűz, amikor az emberek kimenekítése fontos, és jó, ha legrövidebb útvonal van a várók és a vészkijáratok között.

A beruházások folyamán legtöbbször felmerülő kérdés ebben az esetben sem elhanyagolható, miszerint: Milyen plusszköltséget fog ez generálni? A válasz korántsem egyszerű, hiszen igen sok összetevő együttese határozza meg. Függ a modell felállításától, a megválasztott üzemi-útvonalaktól, valamint természetesen a hozzájuk tartozó biztonsági utak megválasztásának módszerétől.

Az építészeti tervezési folyamat részeként megvalósítható ezen a matematikai modellen olyan útvonal-tervezés, amely az adott területen elfogadható idő alatt, elfogadható minőségű, alacsony többletköltségű megoldásokat szolgáltat.

A disszertáció megírásánál fő célom volt, hogy rávilágítsak, mennyire fontos – különösen az egészségügyi ellátó rendszerek esetében –, hogy a fizikai háttérrel adó komplex épületegyüttes robosztus legyen, tartalék útvonalak álljanak rendelkezésre a felújítási munkálatok, váratlan meghibásodások esetére és mindezt a lehető legkevesebb többletköltség befektetése mellett. Ennek megtervezéséhez, figyelembe vételéhez egy, az építészeti tervezést megelőző, 0. fázisban van lehetőség.

## **1.2. Kutatási célkitűzés**

Kutatásaim az épületek modellezését, az egy- és több-blokkos épületekben egy hiba esetére védelmet biztosító útvonalkereső algoritmusokat, valamint ezen létesítmények kihasználtságának, teljesítményének vizsgálatát érintették:

- Megfogalmaztam egy az építészeti tervezést megelőző, előtervezési fázist, melynek során az épület funkcióit gyűjtjük össze, megfogalmazzuk a funkciók közötti szükséges és elégséges kapcsolatokat.
- Az épületek egy speciális, konvex, szakaszonként lineáris költségfüggvényű modellezését definiáltam. A korábbi modellezési módszereknek volt egy közös problémája, miszerint az egyes szakaszokhoz egyetlen költségfüggvényt rendeltek. A modell megalkotásakor a fő szempont a későbbiekben bemutatásra kerülő algoritmusok megfelelő kiszolgálása volt.
- Új útvonalkereső algoritmusokat kifejlesztettem ki és formalizáltam, melyekkel rövid idő alatt keresünk elsődleges útvonalakat az épületben úgy, hogy közben a meghibásodások esetére fenntartott területek – tartalék lépcsőházak, liftek száma, a folyosók szélessége, vagyis a felhasznált területek – kapacitásának a mennyiségét a lehető legkisebbre csökkentjük, ezáltal alacsonyan tartva a többletköltségeket.
- Megmutattam, hogy milyen módon alkalmazhatók a kidolgozott algoritmusok az épületeken belüli úthálózatokra és hogyan terjeszthetők ki az épületek közötti úthálózatokra. Az algoritmusok egyszeres meghibásodást feltételeznek, vagyis hogy az épületben, egy időben, egy helyen történik az adott szakasz használatát megakadályozó – például felújítási – munkálat. Az épületekbe látogató emberek száma és a megbízhatóság iránti igények növekedése miatt úgy kell útvonalat és tartalék erőforrásokat foglalnunk az útvonalak mentén, hogy azok minél kevesebb erőforrást használjanak fel a rendelkezésükre állóból.
- Továbbá általam definiált mértékekkel jellemeztem a több-blokkos, vagyis több-tartományú és több-blokkos hierarchikus, vagyis a többtartományú-hierarchikus épület-együttesekben elérhető kihasználtságot és teljesítményt. Ezek alapján megvizsgáltam a többtartományú és többtartományú-hierarchikus referenciaépületek viselkedését.

### ***1.3. Kutatási módszertan***

Az általam megfogalmazott rendszerek működésének helyességét analitikusan vizsgáltam. Az épület modellezésére gráfelméleti módszert választottam, amely az úthálózatok modellezésére használt legelterjedtebb matematikai modell. Az elérhető javulást szimulációkkal bizonyítottam. A javasolt algoritmusok elemzését generált forgalmakkal teszhálózatokon végzettem.

A kombinatorikus optimalizálási feladaton dekompozíciót, részfeladatokra bontást alkalmaztam, majd az egyes részfeladatokon kombinatorikus gráf algoritmusokat vagy hálózati folyam algoritmusokat használtam. Ezek a legrövidebb útkeresés (Steenstrup 1995) (Dijkstra (Baase January 1983), (T. Cinkler October 1998)), illetve a minimál-költségű többtermékes folyam (MCMCF (Hu 1963) – Minimal Cost Multi-Commodity Flow), melynek lényege, hogy az épületben egyidejűleg több folyamot, igényt vezetünk el úgy, hogy ne sértsük meg az épület kapacitáskorlátaikat, valamint az összköltség minimális legyen. Ennek megoldására egy egzakt módszer az egészértékű lineáris programozási feladat (IP vagy ILP (M.S. Bazaraa 1990), (R.K. Ahuya 1993) – Integer Linear Programming). Ebben az esetben a változók csak egész értéket vehetnek fel, a használt függvények pedig csak lineárisak lehetnek. Az így megfogalmazott ILP feladatot megoldó szoftver, az ILOG CPLEX segítségével oldottam meg. Emellett heurisztikát és mohó algoritmusokat alkalmaztam. A számítógépes szimulációt C/C++ kóddal valósítottam meg.

## 2.

# KOMPLEX ÉPÜLETRENDSZEREK FUNKCIONÁLIS TERVEZÉSE ÉS MODELLEZÉSE

Ebben a fejezetben ismertetem az épületrendszerek felépítését, elemzem, hogy mitől több-blokkos és komplex egy épületrendszer. Meghatározom a megbízható komplex épületrendszerek fogalmát. Áttekintést adok az épületek tervezési folyamatának mai gyakorlatáról. Megmutatom a mai gyakorlatnak a – megbízhatóság szempontjából értelmezett – hiányosságát és javaslatot teszek egy előtervezési fázis használatára, mellyel az épületekben fellépő meghibásodások esetében is üzemképesek komplexumok tervezhetők. Bemutatom az előtervezéshez szükséges modellezési módszert, amely az épületek gráfokra történő leképezését jelenti.

### 2.1. *Komplex épületrendszerek felépítése*

Ahogy azt a bevezetőmben említettem, olyan általánosan alkalmazható matematikai modellt kívántam kifejleszteni, melyet az optimalizálásához kapcsolódó módszertani elemek összekapcsolásával és továbbfejlesztésével érek el. A matematikai modell alkalmazható minden olyan területen, melyen útvonalakat és ezen útvonalakhoz tartozó védelmi, tartalék útvonalakat szükséges tervezni, így a témám: az épület-rendszerek területén is. Az építészeti tervezési folyamat részeként a matematikai modellen egy útkereső algoritmus képes az adott területen elfogadható idő alatt, elfogadható minőségű megoldásokat szolgáltatni.

Ebben a fejezetben olyan alapfogamakat vezetek be, melyek nélkülözhetetlenek a további kutatások bemutatásához:

- az épületek felépítése alapján történő, általam választott csoportosítás:
  - egy-blokkos, nem-hierarchikus és
  - több-blokkos, hierarchikus szerkezeti felépítésű épületrendszerek;
- az épületeken belüli funkcionális egységek felosztása;

Meskó Diána

- az úrendszer fogalma, amelyet az orvosok, ügyfelek, páciensek és hozzátartozóik bejárnak.

### 2.1.1. Épületek csoportosításainak áttekintése

Az épületeket különféle szempontok szerint csoportosíthatjuk.

Egy épület szerkezetét tekintve egyetlen mondatban is különféle csoportosítások jelennek meg:

- Többszintes monolit vasbeton pillérváz, lapostetős lakóépület pilléralappal
- Egyszintes, alápincézett műhely, vasbeton alapgerendákon nyugvó a merevítő falakkal, gerendarács alappal, magastetős felszerkezettel
- Kétszintes, részben alápincézett, téglá építésű, részben lemez alappal, részben pontalappal készült, nád borítású kontyvetős lakóépület, melyhez rönkfából épített melléképület tartozik

A példákból is tisztán látszik, hogy az épületek különböző szempontok szerint csoportosíthatók, mint például:

- Felhasznált alapanyagok
- Épületszerkezet
- Alapozás
- Szintek száma
- Felépítmény
- **Épületen belüli funkciók**
- **Épület funkciója**
- **Épületek száma**
- **Épületek egymáshoz való viszonya**

Disszertációmhoz a vastaggal megjelölt csoportosítások alkalmazása szükséges:

- **Épületen belüli funkciók:** az épületen belül több funkció is megvalósítható. Ezek az osztályok, részlegek, melyeket a modellezés folyamatában figyelembe kell venni. Ezeket a funkciókat a modellezés során tartományokra osztással valósítom meg.
- **Épületek funkciója:** a komplex egészségügyi ellátó-rendszerekben az egyes épületek által ellátott közös funkciók.
- **Épületek száma**

Meskó Diána

- *egy épület* esetén egy blokkos épületről beszélünk, amelyben tartományok helyezkednek el. Ezen épületek modellje *nem hierarchikus, egytartományú modellt* hoz létre.
- Ezzel szemben léteznek a *több épületből* álló épületrendszerek. Két alcsoportba sorolhatók:
  - az épületek egyesével egy-egy funkciót valósítanak meg, ez esetben nem hierarchikus, többtartományú modellt kapunk,
  - az egyes épületek maguk is külön-külön több tartományt foglalnak magukban, ebben az esetben az épületek együttese, egy felsőbb szintű, tartományokra osztásnak felel meg, így hierarchikus, többtartományú szerkezetet hozva létre
- **Épületek egymáshoz való viszonya:** összeköttetésben állnak-e egymással az épületek vagy sem és ha igen, egyszeres avagy többszörösen összekötött kapcsolat van-e az épületek között. Ezen csoportosításhoz redukcióval, az egyes tartományok összevonásával, egyszerűsítésével nyerhető épületmodellt használunk.

### 2.1.2. Egy- és több-blokkos épületrendszerek definiálása

#### 1. definíció Egy-blokkos épületrendszer

Az épületek azon megközelítését, amikor egyetlen hatalmas épületben helyeznek el minden funkciót *egy blokkos rendszernek* nevezem és egytartományú hálózattal, gráffal modellezem. Ebben az esetben minden szervezeti egység, minden funkció egyetlen épületbe tartozik (1. ábra).

Meskó Diána



1. ábra Példa az egy épületből álló egészségügyi ellátó rendszerekre (Pécsi Négyszáz Ágyas Klinika)

## 2. definíció **Több-blokkos, blokk-szerkezetű komplex épületrendszerek**

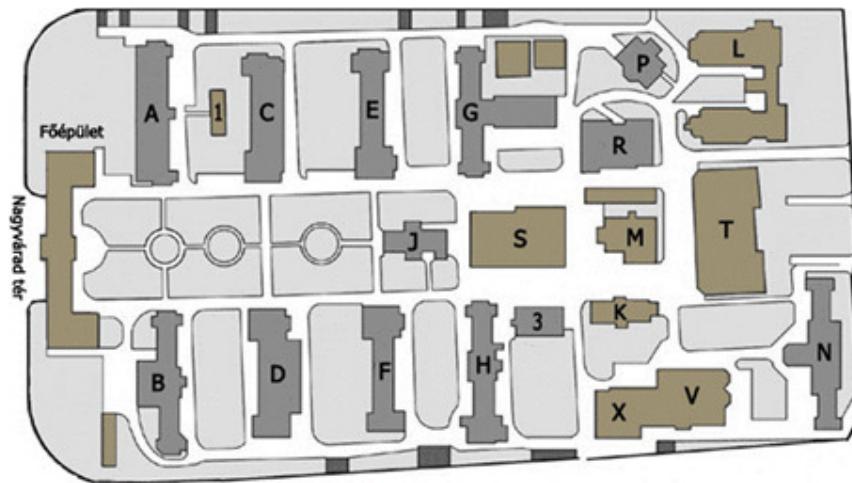
Azon épület komplexumokat, amelyek esetében szervezeti egységek és/vagy funkciók mentén bontásra kerül sor és ezek az egységek külön-külön épületekbe tagozódnak, az épületet *több blokkos rendszernek* vagy *töbttartományú hálózatnak* nevezem (2. ábra és 3. ábra).



2. ábra Példa a több épületből álló, komplex egészségügyi ellátó rendszerekre I. (János kórház, Budapest)



Meskó Diána



3. ábra Példa a több épületből álló, komplex egészségügyi ellátó rendszerekre II.  
(János kórház funkcionális egységeinek terve, Budapest)

A több blokkos rendszerek – komplex épület-együttesek – alkalmazásának megalapozottságát a napjainkban terjedő, tágas, szellős, természethez közelebb álló elrendezést biztosító szolgáltatói rendszerek igazolják. Alkalmazása még az épületrendszerekben lévő adminisztrációs, menedzsment és kontrollfolyamatok rugalmasságát is növeli.

A téma újszerűségét mutatja, hogy a szakirodalomban egyelőre nincs olyan referencia, amely figyelembe venné akár az egy blokkos, akár a több blokkos viselkedést, valamint jelenleg nincs adekvát definíció arra nézve, hogy milyen módon érdemes, hogyan kifizetődő ezt a felosztást végrehajtani.

### 2.1.3. Az épületeken belüli funkcionális egységek

Az épületeken belüli funkcionális egységek meghatározása tulajdonképpen egy lista, mely felsorolja a megvalósítandó funkciókat, például az egyes kórházi osztályokat. A lista azután részekre bontódik annak megfelelően, hogy melyik osztályon belül milyen helyiségeket (előkészítő, műtő, lábadozó, intenzív osztály, kórtermek, mosdók, közlekedő, lépcsőház és/vagy lift, stb.) szükséges megvalósítani.

### 2.1.4. Közlekedés és kapcsolatok az egyes egységek között

Mind az egy, mind a több blokkból álló épületrendszerek esetén az épületekre osztásnak, az épületen belül az emeletekre és az emeleteken belül az osztályokra, egységekre osztásnak különféle szempontjai lehetnek: lehet a szervezeti felosztás szerint,

területi elgondolások alapján, de a funkciók szerinti szétoztása is fontos szempont. Mind ezen épületek, mind pedig az épületeken belüli egységek minden esetben összeköttetésben állnak egymással. Lehet ez közvetlen vagy egy köztes tartományon keresztüli összeköttetés.

### **2.1.5. Megbízható komplex épületek (MKÉ)**

Az egészségügy szolgáltatók mindig is nagy hangsúlyt fektettek az épületek és az épületen belüli *funkciók elérhetőségére*. Ezzel egyidejűleg a szolgáltatásokat igénybe vevő ügyfelek: páciensek és hozzátartozóik igénye is megnőtt a *szolgáltatások megbízhatóbb elérhetőségét* illetően. Korábban nem vették figyelembe, hogy milyen plusz erőforrásokat igényelne az, hogy a legfontosabb egységek minden körülmények között, még meghibásodások esetén is hozzáférhetőek legyenek.

Az egészségügyi ellátó rendszerekben többé nem tolerálható, hogy egy meghibásodás a szolgáltatások fennakadását okozza. Meghibásodás lehet például: egy lift elromlik, egy járda nem használható vagy tüzeset van az épületben. Egy másik példa a meghibásodásra, amikor egy reflektor meghibásodik a műtőben. Az ilyen meghibásodások esetére mindig szükséges egy tartalék útvonallal rendelkeznie az épületeknek. Például egy másik lifttel, ami elegendő kapacitással bír a saját terheltségének és a meghibásodott lift használóinak kiszolgálására vagy tartalék reflektorral kell rendelkeznie a műtőnek, esetleg egy másik, rendelkezésre álló, használaton kívüli, teljes felszereltségű műtőt kell biztosítani a másik példa esetében. Ezért az egyik leginkább elvárt tulajdonsága a szolgáltatásoknak: a túlélés képessége.

### **3. definíció Megbízható komplex-épületrendszerek (MKÉ)**

Azon épületeket foglalja magában, melyek az épületben bekövetkező bármely, a normál üzemszerű használatot akadályozó meghibásodások, haváriák esetén is üzemképesek maradnak.

Az MKÉ olyan épület-rendszer, amelyben a funkcionális pontok és az útvonalak redundánsak, vagyis „párhuzamos”, egymást helyettesítő funkciók és útvonalak állnak rendelkezésre. Ezek az elemek térben szeparáltak, egyszeres meghibásodás elve érvényesül, vannak stand-by tartalékok az aktív funkciókra (pl. műtő, vagy tartalék villamosenergia-ellátás, de egy raktártűz sem béníthatja le a működést, mert van tartalék).

Ennek megvalósításához először az épületek funkcióinak és a köztük lévő kapcsolatoknak a gondos tervezésére van szükség, majd annak vizsgálatára, hogy képes-e az épület meghibásodás esetén is az üzemszerű működésre. A következő fejezetben részletezem, hogy a tervezési folyamat mely szakaszában van lehetőségünk ennek vizsgálatára, biztosítására.

## 2.2. *Tervezési módszerek*

### 2.2.1. **Mi is az a tervezés?**

Egy kicsit távolabbról, teoretikusan elemezve vizsgálom meg a tervezést. Ahogy Balázs Mihály (Balázs dátum nélk.) jegyzetében olvashatjuk, a tervezés gyakorlati menete egy kérdésfelvetéssel kezdődik, melyre választ keresünk. Ennek megkönnyítésére leíró módszert választunk. „A gyakorlati tervezésben mindvégig a folyamat modelljét keressük, mely képes a tervezés átfogó lényegét rendszerezve összegezni és megjeleníteni.”

#### 2.2.1.1. Egy terv születése

A tervezés legelső fázisa a vázlatok és változatok. Sok alternatíva vetődik fel ebben a szakaszban, ami igényli a teljes nyitottságot. Ekkor még nincsenek jogszabályi kötelezettségek, mégis a tervezés legnagyobb felelősséggel járó mozzanata.

**Struktív és evolutív** jelleg. Mit is jelent ez? Általában a dolgok fejlődésének, alakulásának két egymástól jelentősen eltérő módját különböztetjük meg: az élettelen dolgok additív módon épülnek fel: részekből épül az egész - struktív jelleg, míg az élő szervezetek születésük pillanatában egészek, magukban hordozzák lényük teljességét és fejlődésük során részleteződnek - evolutív jelleg

Az emberi alkotások is e két minta alapján jönnek létre – az anyagi jellegű alkotások struktív jellegűek, míg a szellemi alkotások evolutív jellegűek. Az építészeti terv ideális esetben éppen úgy fejlődik, mint egy élő szervezet. Fontos ez az analógia, mert megkülönbözteti az építészeti munka két lényegi elemét, **a tervet és a házat**. Míg a terv egészben születik és fejlődik, részleteződik, addig a ház részekből áll össze („kőről kőre”).

Meskó Diána

**A személyes előkép a terv magva.** Amikor a program, a helyszín, az anyagi feltételek és a tervezéshez fűződő érdekek adathalmazza személyes ismeretté válik, akkor az ismeretek összegzéseként valami kezdeti kép is formálódni kezd a tervezendő épületről. Ez a személyes előkép. A személyes előkép olyan, mint egy mag, amelyben nem csak az addigi ismeretek szummája, hanem a leendő tervről alkotott vágyaink, elképzeléseink is benne foglaltatnak. Nagyon vázlatosan már benne van az az építészeti jelleg (karakter), amelyet meg akarunk valósítani. Benne van a leendő épület hangulatából is valami. Benne van a leendő épülethez fűződő érzelmi viszonyunkból is valami. Ködös kép, mely tartalmazza már a leendő terv (épület) lényegi jellemzőit. Ez a kép a kezdetekben szinte megfoghatatlan, majd az ismeretek bővülésével egyre határozottabbá válik, egyre kontúrosabb lesz. A kicsi mag fokozatosan kicsírázik, szárba szökken. Az előkép részletei még nem láthatók, de minden lényegi sajátossága már létezik. Az előkép az információk személyes ismeretté válása során folyamatosan formálódik.

#### 2.2.1.2. Építészeti kommunikáció (Balázs dátum nélk.)

Gondolatainkat építészként is úgy kell kifejeznünk, hogy azokat mások megismerhessék és megérthessék. Bár a szavak az építészetről szólnak, a teljes mondat az életről. Bármilyen bonyolult jelentéssel bírnak is szavaink, a belőlük alkotott mondatnak egyszerűnek és érthetőnek kell maradnia.

Ebben a kommunikációban meghatározó jelentősége van a vizualitásnak, láttatásnak. A tervezés folyamatának kiteljesedésével a kommunikációba bevont területek egyre szélesebbek:

- Az építészeti rajz: általában síkszerű, kétdimenziós ábrázolást jelent. A statikus kép: klasszikus értelemben vett rajz (vázlat, szerkesztett ábra, látványkép, kollázs, fotó ...). Vizuális illúziók, amikor az észlelt kép eltér az objektív valóságtól.
- Balázs Mihály (Balázs dátum nélk.) jegyzetében kiterjeszti a rajz jelentéstartalmát. E szerint: rajz minden olyan eszköz, ami az építészeti gondolat kifejezésére alkalmas. A rajz dimenziói így lehet a vonal (1), a felület (2), a tér (3) és az idő (4). Általános használatban az időt tekintjük negyedik dimenzióknak, habár az idő, mint fogalom értelmezése ennél jóval összetettebb. A négy térdimenziójú teret elképzelni nagyon nehéz, de matematikailag és grafikusán (két dimenzióra, például monitorra vetített háromdimenziós képekkel) jól szemléltethető.
- GIF (Graphics Interchange Format) képek: Az állóképek tárolása mellett a GIF alkalmas képek animálására is. A tárolt képek megadott ritmusban váltakoznak, ezáltal folyamatos mozgóképet alkotnak. Újszerű technikák, mint amilyen a

Meskó Diána

cinemagraph: több mint egy fotó, de kevesebb, mint egy videó. Kevin Burg, web designer és Jamie Beck fotós animált gif képekből hozták létre ezt az új formát 2011-ben

- Beszéd és gesztikuláció: minden szó, mondat, mozdulat, arckifejezés vagy kézfogás: jel a külvilág számára. A beszéd és a testbeszéd legalább négydimenziós.
- Modell: klasszikus értelemben vett modell (konceptiómodell, munkamodell, reprezentációs modell, stb). Térbeli, azaz háromdimenziós.
- Mozgókép: a klasszikus értelemben vett film (rajzfilm, animáció, videó ...). Időben lejátszódó, azaz négydimenziós.

Ebbe a képbe, folyamatba illeszkedik az általam megalkotott előtervezési fázis, melynek során egy keretrendszert adunk a gondolatoknak, az elképzeléseknek, egy modellt alkotunk az épületekről, épület-rendszerről.

#### **4. definíció Előtervezési fázis, előzetes funkcionális modellezés**

Kialakítottam egy tervezési módszert: előtervezési fázis, előzetes funkcionális modellezés, amely során a funkciók listázása, a lista alábontása, a funkciók csoportokba (épületekbe) történő besorolása, a funkciók közötti kapcsolódásoknak, mint feltétel-rendszernek a definiálása történik az épület építészeti tervezése előtt.

### ***2.3. Épületek modellezése***

A modern, több-blokkos épületekben, jelen esetben az egészségügyi ellátó-rendszerekben, az épületek, blokkok és funkciók elhelyezése legalább olyan fontos, mint azok rendelkezésre állása és a szolgáltatás minősége. Nem elegendő csupán a megfelelő kapacitások biztosítása az ügyfelek – páciensek – számára, az intézmény által nyújtott szolgáltatások minősége egy egyre növekvő elvárás a részükről. A jelenlegi építészeti tervezési metódus nem követi ezt a módszert, habár hosszútávon tekintve megérné az idő és anyagi ráfordítást. Mivel a téma túl összetett, így nem igazán akad nemzetközi szakirodalom a területen. A páciensek elégedettségével tisztán mérhető egy intézmény jobb teljesítménye, gyorsabb működése és az üzemeltetési költségek csökkenése. Nayma Khan egy térbeli elrendezést mutat, amely hatással van a páciensek áramlására, ezzel növelve a működési hatékonyságot (Nayma 3-6 January 2012). Michalek, Choudhary, Papalambros (Jeremy M. 2002) egy építészeti alaprajz tervezési módszert mutat be. Ez a cikk a tervezéskor megjelenő elhelyezési problémákkal foglalkozik, de nem tárgyalja a meghibásodási eseteket, nem állítja követelménynek, hogy a módszerükkel tervezett

Meskó Diána

épületek megbízhatóak legyenek. Ahogy a 2.1.5 fejezetben megfogalmaztam, a komplex egészségügyi épületrendszerekben szükség van az épületek megbízható működésére, arra, hogy az épületben bekövetkezett meghibásodások esetében is üzemszerűen tudjon a komplexum működni.

Az integrált tervezési módszer hasznosságát taglalja a (Guide to the design and construction of high performance hospitals 2010) weboldal. Az oldal szerint az integrált tervezés elengedhetetlenül szükséges a létesítmények magas teljesítményéhez, lényegesen nagyobb energiahatékonyságú épületek érhetőek el így, mint a jelenlegi gyakorlattal. A költséghatékonyság érdekében az építészeti tervezést megelőzően, egy előzetes-lépésként szükséges a különböző funkciók és azok kapcsolatainak feltérképezése. Ez a gondolatfolyam ismét arra vezet, hogy a modern, összetett, blokk-szerkezetű épületeket elemezéséhez modellezés szükséges.

Disszertációmban egy ilyen modellezési módszert javaslom. A modellezés egyediségét az adja, hogy *többlépcsős költségfüggvény esetén is érvényben marad*, használható a fizikailag elkülönülő épület-blokkok elhelyezésének optimalizálására és a közöttük lévő kapcsolatokat is kezelni tudja. Disszertációmban nem csupán a páciensek áramlásának elemzésével foglalkozom, hanem egy matematikai modellezési módszert is megfogalmaztam, amellyel a páciensek áramlása szimulálható.

A korábbi kutatásaim során megfogalmazott adatátviteli hálózatok matematikai gráf modellezése kiterjeszthető a megbízható épületek tervezésének esetére. Az adatátviteli hálózatok esetében alkalmazott védelemmel ellátott útvonalkereső módszerek összetevői felhasználhatók a blokk-szerkezetű egészségügyi ellátó rendszerek építészeti tervezésével kapcsolatos modellezésére. Az módszer interdiszciplináris voltát, felhasználhatóságának sokrétűségét mutatja, hogy a modell és az algoritmusok épp ugyanúgy megállják a helyüket távközlési területen, mint az építészeti tervezésben (Cinkler, Meskó és Mitcsenkov 2005).

Egy fontos és kihívást jelentő különbség azonban, hogy míg az adatátviteli rendszerek esetén a modellezés legfontosabb tényezője a végrehajtási idő szemben a megoldás minőségével, addig az építészeti tervezés munkafolyamatában a modellek megoldásának minősége messze kritikusabb, mint egy 'gyors, ám csupán durva becsléssel' kapott terv, ami akár évtizedekre is meghatározza az ellátó rendszer működés folyamatát.

A vizsgálatok alapján, az elemzésre váró épületek elhelyezési problémája, és a topológia optimalizálás számos módszertani hasonlóságot mutat.

Egy olyan matematikai modellt fejlesztettem ki és részletezek disszertációmban, amely az építészeti tervezés munkafolyamatának részét képezheti, ezáltal összekapcsolva és továbbfejlesztve az átviteli rendszerek vonatkozó és használt módszereit és fejlesztve az építmények optimalizálását.

### 2.3.1. Gráfelméleti alapok

Az épületek funkcióinak és azok kapcsolatainak matematikai modellezésére az egyik leginkább alkalmas módszert, a gráf modellt használtam.

Ahogy R. L. Graham, M. Grötschel, Lovász L.-tól (R. L. Graham 1995)-ben idézik a **gráf** a matematikai gráfelmélet és számítógéptudomány egyik alapvető fogalma. A gráf dolgok (csomópontok, csúcsok) és rajtuk értelmezett összeköttetések (élek) halmaza. Egy gráfot megadhatunk csúcsainak és éleinek felsorolásával, vagy szemléletesebben egy diagram formájában, ahol a pontok felelnek meg a gráf csúcsainak, az őket összekötő ívek pedig az éleknek. A két megadási mód ekvivalens, azaz a gráf pusztán egy struktúra, semmilyen megjelenítési információt nem tartalmaz, így különböző diagramok is tartozhatnak ugyanahhoz a gráfhoz.

Alapértelmezésben a gráf irányítatlan, azaz nem teszünk különbséget „S-ből T-be”, illetve „T-ből S-be” menő élek között. Ezzel szemben az irányított gráfokban (angolosan: digráf) a két iránynak irányított élek felelnek meg.

Szintén alapértelmezésben, a gráf csúcsai címkézettek, azaz meg lehet különböztetni őket. Bizonyos problémák azonban könnyebben kezelhetők, ha nem különböztetjük meg a csúcspontokat. Persze egy-egy csúcspont így is megkülönböztethető maradhat egyéb jellemzőik alapján, mint például a vele szomszédos csúcsok száma. Hasonlóan, a gráf élei alapértelmezésben címkézettek, de előfordulhat, hogy ezt nem követeljük meg. Az olyan gráfok, amikben sem a csúcspontok, sem az élek nem címkézettek, címkézetlen gráfok. Megjegyzés: a „címkézés” szó más kontextusban is elfordul a gráfoknál, itt most az élek-csúcsok megkülönböztetésére szolgáló címkével foglalkoztunk.

### 2.3.2. Általános gráf formális leírása

Ebben a fejezetben a gráfok formális leírását mutatom be. A későbbiekben megvizsgálom a lehetséges védelmi módszerek osztályozásait és végül kitérek az általam választott és továbbfejlesztett megoldásra. Egy gráfot a következő számnégyes definiálja:  $N = (V, E, C, \Omega)$ , ahol  $N$  jelenti az épületben lévő úthálózatot (Network),  $v \in V$  a

hálózat csomópontjait (a gráf csúcsait – vertices),  $e(v_1, v_2) \in E$  a hálózat szakaszait (a gráf éleit - edges),  $C_e$  az egyes szakaszok kapacitását (a gráf éleinek kapacitását – capacity), és  $\Omega_e$  az egyes szakaszok költségét (a gráf éleinek költségét – cost).

A gráfban egy *út*, egy  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_k, v_k)$  sorozat, amennyiben  $v \in V$ ,  $e \in E$ , és a csúcsok mind különbözőek. Egy  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_k, v_k)$  utat akkor hívunk *irányított útnak*, ha  $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots, e_k = (v_{k-1}, v_k)$ . Az épületben jelenlévő igényeknek irányított üzemi, valamint védelmi utat keresünk. Egy irányított útban a forrás csúcs az *adó csomópont*,  $s$  – amely csúcs egyetlen élnek sem végpontja –, a nyelő csúcs a *vevő csomópont*,  $t$  – egyetlen élnek sem kezdőpontja. Ennek megfelelően az előző definíció a következő képpen fog alakulni:  $(v_0, e_1, v_1, e_2, v_2, \dots, v_{k-1}, e_k, v_k)$ , ahol  $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots, e_k = (v_{k-1}, v_k)$  és  $v_0 = s$ ,  $v_k = t$ . Egy *forgalmi minta* a következő képpen írható fel: Minden kielégítésre váró  $o$  forgalom, vagyis az összekötésre váró két funkció egy-egy  $T$  forgalmi mintával definiált, ahol  $b_o$  azon  $o$  forgalom sávszélessége, mely  $s_o$  forrás és  $d_o$  nyelő között fut és  $\tau'_o$  időpontban kezdődik és  $\tau''_o$  időpontig tart,  $T(o : o(s_o, d_o, b_o, \tau'_o, \tau''_o))$ .

*Üzemi út* az az elsődleges  $P'(o)$ , úgynevezett üzemi út a legrövidebb út  $b_o$  elérhető kapacitással az  $e_p$  éleken keresztül,  $P'(o) = (e'_1, e'_2 \dots e'_{|P'|})$ . *Védelmi út*: az az út, melyet védelem esetén egy másik, az előbbtől részben (vagy teljesen) független „legrövidebb” útvonalat keresünk. Teljes függetlenség esetén az üzemi út minden élétől független, míg részben független esetben minden  $e'$  él kiesése esetére, amely  $P'(o)$  üzemi útvonalában szerepel és amely meghibásodhat definiálunk védelmi útvonalat:  $P''_e(o) = (e''_1, e''_2 \dots e''_{|P''_e|})$ .

### 2.3.3. Kapcsolódó fogalmak

Az alábbiakban néhány, a későbbiekhez kapcsolódó gráfelméleti fogalmat tekintünk át. Az *épület csomópontjainak fokszáma*:  $d(v^i)$ ,  $i = 1, 2, \dots, n$ , ahol  $n$  a gráf csúcsainak, vagyis az épület funkcióinak száma. Ez megegyezik az épület gráf modelljében az egyes csomópontokhoz tartozó csúcsok fokszámával. Az *épület átlagos fokszáma*:  $D(N)$ , amelyen az épület összes csomópontja fokszámának átlagát értjük, ahogy azt a gráfok

esetében az összes csúcs fokszámának átlagával definiáljuk:  $D(N) = \frac{\sum_{i=1}^n d(v^i)}{n}$ ,  $i = 1, 2, \dots, n$ , ahol  $n$  a gráf csúcsainak, vagyis az épület funkcióinak száma.



#### **2.3.4. Épület-komplexumok, épületek, blokkok, funkciók és a közöttük lévő kapcsolatok modellezése**

A kérdés, hogy hogyan tudunk takarékoskodni a beruházás folyamán és modellezni az épület-komplexumot még bármilyen erőforrás-igényes építészeti tervezést megelőzően. Ez a probléma az előbb felvázolt matematikai gráf-elmélettel (Suurballe 1974) formalizálható. A feladat komplexitása miatt heurisztikákat alkalmaztam azt a célt szem előtt tartva, hogy közelítsük a globális optimumot. Ezek a heurisztikák dekompozíciókat, approximációkat és modellezési trükköt foglalnak magukban.

Kölcsönös megfeleltetést teszek gráf topológia és épületek között, valamint hierarchikus gráf topológia és épület-komplexumok között. A funkciók, a blokkok vagy osztályok, az épületek és funkcionális kapcsolataik, vagyis a szállítási utak, rendre megfelfeltethetők a csomópontoknak, al-tartományoknak, tartományoknak és a közöttük lévő éleknek.

Egy példán, az épület egy kis területén, részén illusztrálom, hogyan is alkalmazható az előbbieken látott leképezés az épületek modellezésére. A 4. ábra egy folyosót mutat, egyik oldaláról nyíló kórtermek, míg a másik oldalon a sebészeti osztály helységei találhatóak. Középen a közlekedőn a várakozó hozzátartozók, orvosok, paciensek.

Az egyes funkcióknak egy-egy csomópontot feleltetünk meg. Az ábra alsó felén látható két gráf reprezentáció. Tekintsük először a bal oldali példa gráfot: a bal oldali két kezelőnek két csomópont, míg a jobb oldalon elhelyezkedő három kórteremnek három csomópont felel meg a gráfban. A folyosó egy külön csomópont, melyen keresztül mindegyik helység elérhető. Míg a kórtermek között nincs szükség az átjárásra, így azon csomópontok között nincsen kapcsolat, nincsen él közöttük, addig a kezelők között célszerű az átjárhatóságot biztosítani. A jobb alsó ábra ugyanennek az épületrésznek egy alternatív leképezését mutatja be, mellyel a folyosó is egy kapcsolatként működik a különböző funkciók között.

Meskó Diána



4. ábra Példa az épületek gráffal történő modellezésére

### 5. definíció Épületek gráfokkal történő modellezése

Az épületeket, azok funkcióit, közöttük lévő kapcsolatait, átbecsítő képességüket, költségeit kölcsönösen egyértelműen megfeleltetem a matematikai gráf  $(N)$  eszközének,  $v \in V$  csomópontjainak,  $e(v_1, v_2) \in E$  éleinek,  $C_e$  kapacitásának,  $\Omega_e$  kapacitás-költségének.

$$N = (V, E, C, \Omega) \quad (1)$$

#### 2.3.5. Útkeresések gráfokban

A gráf modellben végzett legrövidebb útkeresést, Dijkstra féle algoritmussal végzik (Baase January 1983), amely megadja az s-ből t-be vezető legrövidebb utat, megadja az adó és a vevő közötti legrövidebb utat.

### 6. definíció Legrövidebb útkeresés épületekben

Amennyiben az épületeket gráffal modellezzük, a legrövidebb út keresése megfeleltethető a különböző funkciók közötti legrövidebb útkeresésnek.

A minimális költségű többtermékes folyam probléma esetén adott egy  $N(V, E, C, \Omega)$  gráf, az élein értelmezett  $b_e : E \rightarrow R^+$  kapacitás-, illetve  $c : E \rightarrow R^+$  költségfüggvénnyel. Ezen kívül ismert az igények egy  $D : \{(s_1, t_1); (s_2, t_2); \dots; (s_n, t_n)\} \subseteq V \times V$  halmaza, és az

igények  $b: (s_1, t_1) \rightarrow R^+$  nagysága. **A feladat:** egyidejűleg találni az összes  $(s_i, t_i)$  pontpárra egy olyan  $s_i$ -ből  $t_i$ -be vezető  $b_i$  kapacitású folyamat, hogy eközben minden élen teljesüljön a kapacitáskorlát: az  $e_i$  élen futó folyamatok kapacitásának összege legfeljebb akkora legyen, mint az adott él  $c_i$  kapacitása.

## 7. definíció Páciens-forgalom, mint többtermékes folyam probléma

Amennyiben az épületet gráffal, a páciens-forgalmat többtermékes folyamként modellezzük, úgy az épületekben lévő páciensek forgalmát minimális költségű többtermékes folyam problémaként írhatjuk fel.

Az épületek gráfokkal történő modellezése lehetőséget nyújt arra, hogy:

- az épület gráf modelljén a korábbiakban megfogalmazott útkereső algoritmusokat alkalmazzunk;
- a páciens-forgalmat, mint többtermékes-folyamot modellezzük;
- így többtermékes folyamproblémaként formalizáljuk az épületben közlekedő embereket;
- meghibásodásokat szimuláljunk az épületben, vagyis az épület gráf modelljén, ezzel megvizsgálva, hogy az épület megbízhatóságát;
- az egyes épületek megbízhatóság vizsgálatát kiterjeszthetjük a többblokkos épületrendszerekre, így beláthatjuk, hogy a blokk-szerkezetű komplex egészségügyi ellátórendszerek is teljesítik a megbízhatóság feltételét.

## 2.4. Új tudományos eredmények: 1. Tézis

### 1. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitcsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2012)

**Definiáltam a megbízható komplex épületek (MKÉ) fogalmát. Megfogalmaztam az MKÉ tervezéséhez szükséges előzetes, az építészeti tervezést megelőző funkcionális tervezési lépést, majd definiáltam egy egyedi gráf-leképezést, amely az előtervezés során alkalmazható és megmutattam a gráfelméleti algoritmusok, védelmi módszerek alkalmazhatóságát.**

Meskó Diána

**1.1. Definiáltam a megbízható komplex épületek (MKÉ) fogalmát, amelynek mennyiségi és minőségi attribútumai az alábbiak:**

- **robosztus, olyan épületeket foglal magában, melyek az épületben bekövetkező meghibásodások esetén is teljes kapacitással üzemképesek,**
- **redundáns elemeket és útvonalakat tartalmaznak,**
- **azonos funkciójú (redundáns) elemek térben szeparáltak,**
- **amelyet, mint rendszert funkcionálisan modelleztek és optimalizáltak az alábbi eljárás szerint.**

**1.2. Leírtam és javasoltam egy tervezési módszert: előtervezési fázis, előzetes funkcionális modellezés, amely során a funkciók listázása, a lista alábontása, a funkciók csoportokba (épületekbe) történő besorolása, a funkciók közötti kapcsolódásoknak, mint feltétel-rendszernek a definiálása történik az épület építészeti tervezése előtt.**

**1.3. Megfogalmaztam az épületek gráfokra történő leképezését: modellezésre alkalmazott módszer során épület-komplexumok, épületek, blokkok, funkciók és a közöttük lévő kapcsolatok modellezése gráfokkal. Az épületek funkcióit, közöttük lévő kapcsolatait, átbocsátó képességüket, költségeit kölcsönösen egyértelműen megfeleltetem a matematikai gráf  $(N)$  eszközének,  $v \in V$  csomópontjainak,  $e(v_1, v_2) \in E$  éleinek,  $C_e$  kapacitásának,  $\Omega_e$  kapacitásköltségének.**

**Visszavezettem a gráfokkal modellezett MKÉ-kre a gráfokon alkalmazható útkereső algoritmusokat és minimál költségű többtermékes folyam problémák alkalmazhatóságát.**

**Megmutattam, hogy a blokk-szerkezetű komplex egészségügyi ellátó rendszerek megfelelő védelmi stratégiák használatával kielégítik a megbízható komplex épületek (MÉK) fogalmát.**

### 3.

## A MEGBÍZHATÓ ÉPÜLETEK

**Az épületek rendelkezésre állása avagy „Miért is szükséges védelemről gondoskodni az egészségügyi ellátó rendszerekben?”**

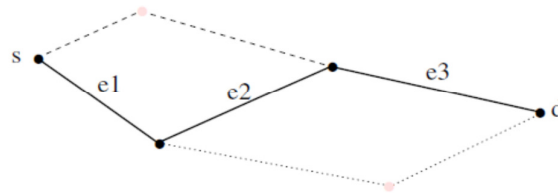
Az egészségügy szolgáltatók mindig is nagy hangsúlyt fektettek az épületek és az épületen belüli *funkciók elérhetőségére*. Ezzel egyidejűleg a szolgáltatásokat igénybe vevő ügyfelek: páciensek és hozzátartozóik igénye is megnőtt a *szolgáltatások megbízhatóbb elérhetőségét* illetően. Korábban nem vették figyelembe, hogy milyen plusz erőforrásokat igényelne, hogy a legfontosabb egységek minden körülmények között, még meghibásodások esetén is hozzáférhetők legyenek. Amennyiben az épületrendszer meghibásodás esetén is üzemszerűen működik, akkor az épületet megbízható komplex épületrendszernek nevezzük (lásd: 1.1 tézis).

Ebben a fejezetben bemutatom a szakirodalomban jelenleg fellelhető védelmi módszerek fajtáit, alkalmazásuk korlátait. Védelmi modellt javasolok, mely megoldást nyújt a nem-lineáris költségfüggvényű felírás egészértékű lineáris programmal történő megoldására. Megadom a modell matematikai leírását.

### 3.1. Védelmi stratégiák

A védelem egy olyan előre definiált módszer, amivel egy épület gördülékenyen tud üzemelni a közlekedési útvonalak, transzport hálózat egyszeres meghibásodása esetén is. Az 5. ábra egy elsődleges/üzemi útvonalat mutat  $s$  és  $d$  csomópontok között, mely az  $e_1$ ,  $e_2$  és  $e_3$  éleken fut. Jegyezzük meg, hogy ha  $e_1$  él meghibásodik, a szaggatott útvonalon és  $e_3$  élen keresztül valósítható meg a tartalék, védelmi út. Ha az  $e_3$  él hibásodik meg, a pontozott útvonalon és az  $e_3$  élen keresztül védhető az üzemi út, míg az  $e_2$  él meghibásodása esetén mind a két védelmi út megfelelő. Másik oldalról nézve azt is észrevehetjük, hogy a szaggatott útvonal mind az  $e_1$  és  $e_2$ , és a pontozott útvonal mind  $e_2$  és  $e_3$  éleket védi.

Meskó Diána



5. ábra Részben független megosztott védelmi utakkal ellátott üzemi útvonal megjelenítése

Ha szeretnénk, hogy bármely él meghibásodása esetére a funkciók és a blokkok között rendelkezünk elegendő tartalék erőforrással, akkor megfelelő védelmi módszerről is gondoskodnunk szükséges. A meghibásodás lehet egyszeres vagy többszörös. Mivel azonban a meghibásodás problémája magasabb szintre aggregálható, így itt az egyszeres meghibásodásra összpontosítok. Különböző kritériumok alapján számos csoportosítás hozható létre az eltérő védelmi és helyreállító technikákból.

### 3.1.1. Védelem vagy helyreállítás?

A bekövetkező meghibásodásokra történő reagálás és az azt megelőző felkészülés alapján beszélhetünk védelemről vagy helyreállításról. Védelem esetén előzetesen, még a meghibásodás előtt keresünk és foglalunk az igények számára egy (vagy több) védelmi utat, és a hiba bekövetkezésekor azonnal átkapcsolunk a védelmi útra. Helyreállítás használatakor előzetesen nem keresünk alternatív megoldásokat valamely hálózatrész kiesésének esetére, hanem a hiba megtörténtekezékor kezdünk a hálózatban más utat keresni az érintett igények számára – ezáltal kevesebb erőforrást használunk, és gyorsabb lesz az igények elvezetésének számítása ideje, de előfordulhat, hogy nem találunk a hibát elkerülő alternatív utat a forgalmaink számára.

Meskó Diána

I. táblázat Védelmi stratégiák osztályozása

Osztályozás	Stratégia, pro és kontra
Helyreállítás	Az útvonalak helyreállítása valós időben történik a pillanatnyilag rendelkezésre álló erőforrások felhasználásával. + Kevés többletköltség – Nagyon lassan reagál – Bizonyos esetekben nincs helyreállításhoz szükséges kapacitás
Védelem	Előre meghatározott védelmi útát feltételezzük + Mindig van háttér útvonal az üzemi útvonalhoz + Gyorsan reagál – Több addicionális költséggel jár az előző módszerhez képest

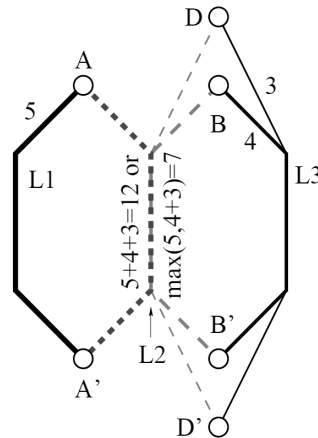
### 3.1.2. Hozzárendelt vagy megosztott védelem?

Nyilvánvaló, hogy védelem alkalmazása többlet erőforrásokat igényel a hálózattól. Egyszerű, de kevésbé takarékos megoldás a **hozzárendelt védelem** alkalmazása. Ekkor minden elvezetett igény üzemi útja mellé keresünk egy attól független, diszjunkt (R. 1999.), (Suurballe 1974), vele azonos kapacitású védelmi utat, mindkettőt lefoglaljuk a hálózatban, és azt más forgalmak nem használhatják. Ez számítási idő szempontjából üdvös megoldás, tekintettel arra, hogy ekkor mindössze annyi a feladatunk, hogy egy helyett két utat keressünk és foglaljunk, viszont a két út miatt kétszeres erőforrásigényt jelent.

Bonyolultabb, de takarékosabb megoldás a **megosztott védelem** alkalmazása. Ennek alapelve a következő: mint már a bevezetőben is említettem, a felhasznált eszközök megbízhatósága miatt feltehető, hogy csak egyszeres hibákra kell felkészülnünk. Emiatt két különböző igény (valós forgalmat szállító) üzemi útja, amely nem használ közös erőforrásokat, sosem fog egyszerre meghibásodni, azaz sosem fog egyszerre mindkét igény forgalma átkerülni a védelmi útjára. Így pedig a két védelem használhat közös erőforrásokat. A probléma lényege az, hogy hogyan válasszunk egy üzemi útvonalat és egy vagy több védelmi útvonalat egy igény számára. Egy példán keresztül megvilágítom, hogy miért létfontosságú a modellezési módszer. Ebben a példában három-három funkcionális egység áll összeköttetésben három-három éllel: L1, L2 és L3. Adott az 5, 4 és 3 emberből álló három orvosi csapat. Hogy biztosak lehessünk abban, hogy mindhárom orvosi csapat elérje a cél-területet, védelmi-útvonalakat kell meghatározni minden egyes hiba esetére a közlekedési útvonaluk mentén. Az 6. ábra mutatja a gráf-megfeleltetését a három-három funkciónak: A, B és D, valamint A', B' és D'

Meskó Diána

csomópontok és a három orvosi csapat: a hálózati forgalom, melynek elvezetéséről gondoskodni kell A, B és D-ből A', B' és D'. A forgalmi igényük 5, 4 és 3 egység egyesével. Minden igény rendelkezik egy üzemi úttal (folyamatos vonal) és védelmi úttal (szaggatott vonal). 3 élen lehetséges meghibásodás.



6. ábra A tartálék kapacitás megosztása: 12 vagy 7 egység

Feltételezzük azt az esetet, amikor minden csomópont-pár a folytonos vonallal jelölt útvonalakat használja, mint üzemi útvonal. Ha L1 él meghibásodik, akkor az AA' védelmi útvonalnak (L2 élen keresztül) 5 egységnyi forgalom kapacitást kell átvennie. Abban az esetben azonban, ha L3 hibásodik meg, akkor BB' és DD' egyidőben kellene használják a tartalék útvonalukat, ami azt jelenti, hogy ekkor L2-nek elegendő kapacitást kellene biztosítani mindkét meghibásodott útvonal számára egyidőben (azaz 7 egységnyi kapacitás szükséges a szolgáltatás fenntartásához). Abban az esetben azonban, ha egyidőben több él meghibásodását feltételeznénk, például L1 és L3 élek meghibásodását, akkor a két igény kapacitásának összegét kellene a közös védelmi szakaszon lefoglalni, ami a példában 12 egységnyi kapacitásnak felel meg. Ez azt jelenti, hogy megosztható védelemmel dolgozva az L2 élen 12 egységnyi kapacitás helyett 7 egységnyi kapacitás elegendő. A gyakorlati példánál maradván ez például azt jelenti, hogy az épület egy liftejének meghibásodása esetére, egy L2 liftre van szükség, ami 7 embert tud kiszolgálni.

### 3.1.3. Szakasz, szegmens vagy útvédelem?

Egy épület szomszédos funkciókból épül fel, melyet pont-pont összeköttetésekkel ábrázolhatunk. Kézenfekvőnek tűnik az egyes **élek védelme**: minden szomszédos csomópontpár közti összeköttetéshez rendelhetünk egy alternatív útvonalat, amelyre a szakasz kiesése esetén az addig ez eredeti élen átfolyó forgalmat átirányítjuk.



Az is megfelelő megoldás, ha a két végpont közötti teljes útvonalra keresünk a forrás és nyelő pontpár között egy, az üzemtől teljesen független, alternatív útvonalat, ekkor **útvédelemről** beszélünk.

Egy élnél hosszabb, de egy egész útnál rövidebb hálózatrészek védelmét **szegmensvédelemnek** nevezzük.

#### 3.1.4. Statikus, dinamikus és adaptív védelem?

**Statikus védelem** esetén minden pontpárra előre meghatározunk egy üzemi és egy védelmi útvonalat, a beérkező igények esetén pedig ezekhez tartjuk magunkat.

**Dinamikus a védelem**, amennyiben az egyes pontpárok között meghatározott útvonalak időről időre változnak.

**Adaptív védelemről** akkor beszélünk, ha a védelmi utak folyamatosan változnak a hálózat állapotának és forgalmi viszonyainak megfelelően.

Nyilvánvaló, hogy ezek egyre hatékonyabb, ugyanakkor egyre komplexebb megoldások, melyek egyre kevesebb, előzetesen rendelkezésre álló információt igényelnek a hálózatról, illetve az azt érő terhelésről.

#### 3.1.5. Hibafüggő vagy hibafüggetlen?

A **hiba-függetlenség** azt jelenti, hogy egy algoritmus az adott igény védelmét a hiba helyétől függetlenül határozza meg. Tehát, ha az üzemi útjának bármely alkotóeleme sérül is meg, a forgalom mindig ugyanarra a védelmi útra helyeződik át.

**Hibafüggő** ellenkező esetben: amikor különböző helyen bekövetkezett hibák esetén egyazon üzemi úthoz más és más védelmet rendelünk.

### 3.2. *A kutatásom során vizsgált védelmi módszerek*

A védelmek különböző fajtáinak összefoglalóját a 2. táblázat tartalmazza. Az általam megalkotott modellen használt védelmi módszerek:

- megosztottak;
- rész-hálózatokon alapulnak;

- részben független utakat használnak;
- hibafüggő, illetve hibafüggetlen megoldásokat is tartalmaznak;
- bármilyen egyszeres meghibásodás esetére túlélést garantál, de néhány többszörös meghibásodás esetére is működik.

2. táblázat Védelmi technikák osztályozása

Osztályozás	Módszer, pro és kontra
Hozzárendelt védelem	Minden egyes üzemi út rendelkezik egy kizárólag számára foglalt készenléti úttal + Egyszerű és gyors – Rendkívül magas erőforrás-igényű
Megosztható védelem	A védelem számára allokált erőforrásokat próbálja megosztani + Csak korlátozott erőforrásokat használ – Hosszabb számítási igényű
Útvédelem (végponttól-végpontig tartó)	A teljes út egyetlen független úttal védett (Suurballe 1974), (R.K. Ahuya 1993)
Él-védelem	A forgalmat az él két végpontja között vezeti újra.
AI-hálózat védelem	A védelmi tartományok végpontjai közötti védelem (hálózati csoportok)
Szegmens védelem (Ho P. H. 2002)	A teljes út bizonyos részeinek védelme (szélsőséges esetei: útvédelem és szakaszvédelem)
Statikus védelem	Minden egyes csomópont-párra egy üzemi és egy vagy több védelmi utat foglalunk és ezen nem változtatunk az idő múlásával sem
Dinamikus védelem	Amikor az utak újra-konfigurálhatók
Adaptív védelem	A védelmi útvonalak minden új igény érkezésekor megváltozhatnak, hogy azonnal alkalmazkodjanak a megváltozott forgalomi és a hálózati feltételeknek megfelelően.

### 3.3. *Költségszámítás*

A cél, hogy meg tudjuk határozni, hogy mennyibe fog kerülni egy hiba fellépése esetén egy lehetséges védelmi útvonal, például egy lift meghibásodása. Lássuk a

megoldási lehetőségeket a példa esetére: egy másik lift lehet használatban, amely ki tudja szolgálni az eredeti kapacitását, valamint a meghibásodott lift igényeit is ezen felül; esetleg egy lépcsőház használható helyette. Általánosságban elmondható, hogy egy tartalék lift költsége sokkal magasabb, mint a lépcsőház használata. Azonban, ha a meghibásodott üzemi lift a rehabilitációs osztály és a mozgássérültek osztálya között helyezkedik el, akkor ebben az esetben mindenképpen gondoskodni kell egy tartalék liftről, ami legalább a mozgáskorlátozottak szállítását megoldja. Ez többlet-költséggel jár.

Amennyiben a meghibásodások esetére fenntartott lift vagy lépcsőház más meghibásodások esetére is tartalék útvonalat szolgáltatnak, úgy ennek a meghibásodásnak a kapacitás igényét is figyelembe kell venni. Abban az esetben azonban, ha egyszerre egy meghibásodást feltételezünk, a tartalék útvonalak oszthatóak a védelmi kapacitásokon, a két külön meghibásodás esetére elegendő egy fenntartott liftet biztosítani, így annak költségét elégséges egyszeresen figyelembe venni.

A számítások modellezéséhez matematikai formalizációkat alkalmaztam. A következő fejezetekben áttekintem a szükséges matematikai eszközöket, mint amilyen a minimál-költségű többtermékes folyam probléma és az egészértékű lineáris programozás kérdésköre. Ezen eszköztárat felhasználva azután bemutatom az általam definiált gráf modellt, mellyel konvex, lineáris szakaszokkal közelíthető költségfüggvényű feladatok optimális megoldását keresem egészértékű lineáris programozással.

### 3.3.1. Minimál-költségű többtermékes folyam probléma

A **Minimál Költségű Többtermékes Folyam** (Minimum-Cost Multi-Commodity Flow – MCMCF) probléma jelentése, hogy egy hálózatban egyidejűleg több különböző igényt próbálunk meg elvezetni úgy, hogy az összes folyam megfeleljen a kapacitás korlátoknak és közben minimális költségűek legyenek. Az MCMCF számításigénye meghatározorra az algoritmus futási idejét mind elméletben, mind pedig a gyakorlatban. Emlékeztetőül a minimális költségű többtermékes folyam probléma.

Ahogy azt a 2.3.2 fejezetben láthattuk, ez formálisan a következő képpen néz ki:

Adott egy  $G(V, E)$  gráf, az élein értelmezett  $b_e : E \rightarrow R^+$  kapacitás-, illetve  $c : E \rightarrow R^+$  költségfüggvénnyel. Ezen kívül ismert az igények egy  $D : \{(s_1, t_1); (s_2, t_2); \dots; (s_n, t_n)\} \subseteq V \times V$  halmaza, és az igények  $b : (s_1, t_1) \rightarrow R^+$  nagysága.

**A feladat:** egyidejűleg találni az összes  $(s_i, t_i)$  pontpárra egy olyan  $s_i$ -ből  $t_i$ -be vezető  $b_i$  kapacitású folyamatot, hogy eközben minden élen teljesüljön a kapacitáskorlát: az  $e_i$  élen futó folyamatok kapacitásának összege legfeljebb akkora legyen, mint az adott él  $c_i$  kapacitása.

### 3.3.2. Egészértékű lineáris programozás

A többtermékes folyamatokat *egészértékű lineáris programmal* (M.S. Bazaraa 1990), (R.K. Ahuya 1993) (Integer Linear Programming – ILP) tudjuk megfogalmazni. Mit is takar ez a kifejezés?

A **lineáris program** (LP) jelentése, hogy a változók és segédváltozók tetszőleges értéket vehetnek fel, és a problémát lineáris egyenletek segítségével adjuk meg. (Az ilyen módon megfogalmazott problémák polinom időben megoldhatók, melyet először N. K. Karmarkar (Karmarkar 1984)-ben fogalmazott meg. Bár a bizonyítottan polinomiális megoldóalgoritmus a gyakorlatban kevéssé használható.)

Amennyiben megkötést teszünk arra, hogy minden változó csak egész értéket vehet fel, akkor **egészértékű lineáris programhoz** (ILP) jutunk. Az ILP, egy a közgazdaságtanban, mérnöki, matematikai munkákban alkalmazott módszer, mely egész értékű változók egy lineáris kombinációjának értékéhez (célfüggvény) keres a változók lineáris kombinációira vonatkozó egyenlőséget vagy egyenlőtlenségeket kielégítő minimális vagy maximális értéket.

Amennyiben csak a változók egy részére teszünk olyan megkötést, hogy csak egész értéket vehetnek fel, másik részük lehet folytonos, akkor **kevert egészértékű lineáris programnak** (Mixed Integer Linear Programming – MILP) nevezzük. A következőkben leírt és általunk megfogalmazott probléma definíciók mindegyike MILP. Az MILP helyett lehetett volna heurisztikus közelítő algoritmusokat (G. Ausiello 1999) is alkalmazni, melyek gyorsabb futási idejűek és elég jól megközelítik az optimális megoldást. Munkám során azonban nem ezt tárgyalom.

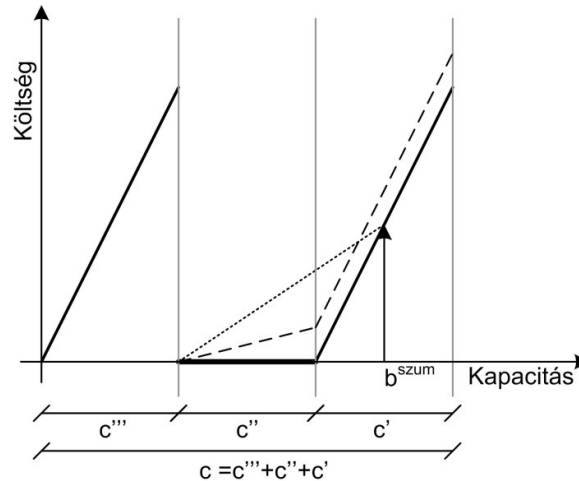
Emlékeztetőül a kevert egészértékű lineáris program formális megfogalmazása a következő: adott a változóknak a korlátot jelentő lineáris kombinációit tartalmazó  $\mathbf{A} \in R^{m \times n}$  mátrix (Rózsa 1991), a korlátok értékeit tartalmazó  $\mathbf{b} \in R^m$  vektor, és a  $\mathbf{c} \in R^n$  célfüggvény, melyekre:  $\mathbf{Ax} \leq \mathbf{b}$ ,  $\mathbf{x} \geq 0$ . Keressük  $\min_{\mathbf{x}} \{\mathbf{c} \cdot \mathbf{x}\}$  vagy  $\max_{\mathbf{x}} \{\mathbf{c} \cdot \mathbf{x}\}$  értékét.

A lineáris programozás alapfeladatának részalmazát képező ILP probléma NP-teljes (Michael R. Garey 1979), de léteznek a gyakorlatban alkalmazható futási idejű közelítő módszerek a megoldására (Wolsey 1998), (Vazirani 2003), (Michalewicz 1995). A probléma megoldásakor a CPLEX programcsomagot használtam (ILOG CLPEX, User's Manual dátum nélk.).

### 3.3.3. Kapacitás költségfüggvénye

Az algoritmusnak, mellyel meghatározható a takarékosan foglalandó háttér útvonalak kapacitás mennyisége a (Cinkler T. 7-11 June 1999) az alapötlete. Minden  $e$  élre a teljes kapacitás  $c_e$ , melyek költsége a következő képpen alakul (7. ábra alsó része):

- $c_e''$  az üzemi útvonalak számára foglalt kapacitás költsége (a védelem számára nem használható, hiszen ott üzemi útvonalak futnak);
- $c_e'$  a (megosztható) védelem számára foglalt, tartalék kapacitás (ezen kapacitás az üzemi útvonalak diszjunkt, független voltának függvényében alacsony költséggel használható);
- $c_e - c_e'' - c_e'$  szabad, nem lefoglalt és nem használt kapacitás, mely kapacitások az eredeti él-költségen foglalhatók.



7. ábra Három kapacitás költség model védelem esetére

Ennek fényében a „legrövidebb” út azt jelenti, hogy az az útvonal, ami a legkevesebb erőforrás foglalást igényli a kapacitás-költség függvényében. Tehát a legalacsonyabb erőforrás foglalást a  $c - c'' - c'''$  kapacitásokból, vagyis hogy minél kevesebb  $c'$  kapacitást foglaljunk.

Meskó Diána

Hogy egy  $b_o$  kapacitású  $o$  igény védelmi útvonalát elvezessük, csak annyi kapacitást szükséges foglalnunk minden egyes élen, amennyivel az igény kapacitása meghaladja az adott élen megosztható védelmi kapacitás számára foglaltat. A célunk, hogy az útvonal összes élére nézve ezen költségek összege minimális legyen. Ez az egyetlen mérőszám, melyet az igények számára történő útvonal tervezése során használunk szükséges. Ezen  $P_e(o)$  utakat a  $P'_e(o)$  út részben független, megosztott védelmi útjainak nevezzük (lásd az 3.1.3 fejezetet).

A 7. ábra felső része az előbbieken bemutatott kapacitás-eloszláshoz tartozó három kapacitás modellt mutat:

- **Folyamatos vonal:** a  $C''$ , védelem számára megosztható kapacitás költsége nulla, míg a  $C - C'' - C'''$ , szabad kapacitás költsége az eredeti élköltséggel egyenlő. Ebben az esetben a megosztható kapacitás ingyen használható, míg a nem lefoglalt szabad kapacitás az eredeti kapacitásköltségért foglalható. A megoldás veszélye, hogy a nullaköltségű élek mentén hurkok alakulnak ki az utakban.
- **Pontozott vonal:** egy alternatív megoldás lehet erre a problémára. Egyetlen lineáris költségfüggvénnyel közelítjük a két, egyenként lineáris szegmenset. Azonban az eredmény így csupán szub-optimális lesz.
- Az általam javasolt megoldás a **szaggatott vonallal** bemutatott költségmodell: a megosztható kapacitásnak egy igen alacsony, de nem nulla kapacitás-költséget definiáltam, hogy elkerüljem a hurkok kialakulását az útvonalakon, azonban a költségfüggvény értékétől függően még így is kialakulhatnak szükségtelenül hosszú utak.

Erre vonatkozó kutatásaim során definiáltam a  $\gamma_e''$  **arányos költség** fogalmát, melyet a 2. tézisben részletesen tárgyalok. Vizsgálataim azt mutatták, hogy  $\gamma_e'' = 0,1$  esetén, vagyis az eredeti költség 0,1-szerese esetén lehet elérni a legmegfelelőbb úthosszúságot a védelmi útvonalak tekintetében és ezzel együtt ebben az esetben lesz optimális az épületkihasználtság. A szabad kapacitások költségének ebben az esetben is az eredeti élköltséget alkalmazzuk.

Azonban az általam javasolt, optimális eredményt adó költség-függvény esetén is (7. ábra szaggatott vonal) adott marad a probléma: a keresett célfüggvény nem-lineáris, két, egymástól független, lineáris szakasszal meghatározott költségfüggvényt kaptunk.

Ennek feloldására alkottam meg az **él kettőzést**.

### 3.4. *Élkettőzés*

Ebben a fejezetben az *élkettőzést* (Link Doubling – LD) mutatom be.

Az LD önmagában nem egy algoritmus, hanem egy modellezési trükk. Több, a megosztható kapacitás költségét a szabad kapacitás költségétől megkülönböztető, javasolt algoritmusnak is ez az alapja. Abban az esetben is használható, amikor egyszerre több megosztható védelmi utat is újra elvezetünk. Mivel a két-szakaszos kapacitásköltségfüggvény egy nem-lineáris költségfüggvény, így nem alkalmazható sem Egészértékű Lineáris Programozási megfogalmazásban (Integer Linear Programming – ILP), sem a Dijkstra (Baase January 1983) algoritmussal. Ezért szükség van egy linearizálási megoldásra.

Lássunk erre egy példát. Adott egy hálózat, egy épület, amely a megosztott védelmet támogatja. Ebben az épületben szeretnénk tartalék útvonalat keresni egy meghibásodás esetére. Ez azt jelenti, hogy szeretnénk elvezetni egy  $b_o$  kapacitású  $o$  igényt. Eredeti kapacitásköltséggel csupán arra a kapacitás-mennyiségre lesz szükség, ami meghaladja az adott éleken a diszjunkt üzemi útvonalakkal rendelkező megosztható védelem számára már korábban lefoglalt kapacitásokat. Hiszen a  $b_o$  kapacitás számára foglalásra kerülő kapacitás a következő két összetevőből áll:

- egyrészt áll abból az alacsony költségű megosztható védelemből, amelyen a  $b_o$  kapacitású  $o$  igény osztozni tud a már korábban elvezetett igényekkel;
- másrészt a megosztható kapacitást meghaladó kapacitás mennyiségéből.

Ez azt jelenti, hogy egy élen két, lineáris szegmensből álló költségfüggvényű kapacitás lesz jelen. Az élettőzéssel linearizálhatjuk a két-szakaszos költségfüggvényt és lehetőségünk van bármely egyszeres élhibára megtalálni az optimális megosztható védelmi útvonalakat.

### 8. definíció *Élkettőzött épület-modell*

Az épület gráffal történő megfogalmazásakor egy modellezési trükköt alkalmaztam. Azokat éleken, melyeken egyidejűleg megosztható védelem számára foglalt kapacitás és szabad kapacitás is jelen van, megkettőztem. Ezzel megvalósítható, hogy egyazon valós szakaszon, a gráf megfelelőjében egy élen a megosztható védelem számára alacsonyabb, míg a szabad, ez idáig nem használt kapacitás esetén az eredeti élköltséggel számolunk úgy, hogy a költségfüggvény élenként lineáris marad (7. ábra

Meskó Diána

szaggatott vonal), ezzel egyidejűleg elkerüljük a szükségtelenül hosszú és hurokba kerülő utakat.

Az új gráf modell jelentősége, hogy konvex, lineáris szakaszokkal közelíthető költségfüggvényű osztatlan többtermékes folyamatok esetén is megtalálhatjuk egészértékű lineáris programozás segítségével a minimális költségű megoldást.

Lássuk, hogy hogyan is működik ez.

### 9. definíció Élkettőzött épület-modell formalizációja

$$N' = (V', E', C', \Omega', E'', C'', \Gamma'') \quad (2)$$

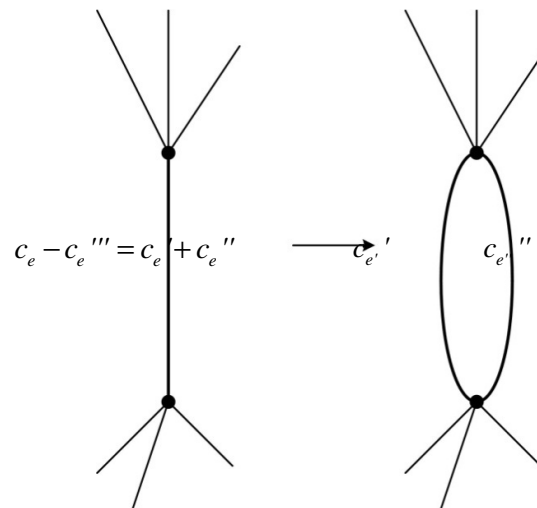
ahol  $N'$  az épület új modellje, az új modell paraméterei pedig a következők (emlékeztetőül az  $N = (V, E, C, \Omega)$  definícióját lásd az 1. definíciónál):  $v' \in V'$ ,  $V \equiv V'$  a gráf csomópontjai,  $e'(v'_1, v'_2) \in E'$ ,  $E \equiv E'$  a gráf szakaszai,  $C'$  az  $e' \in E'$  szakaszok kapacitása, amely kapacitás az eredeti modell  $e \in E$  élének szabad kapacitása:  $C'_e \equiv C_e$ ,  $\omega'_e \in \Omega'$  az  $e' \in E'$  szakaszok költsége, továbbá az  $e''(v'_1, v'_2) \in E''$  a gráf újonnan bevezetett szakaszai, amelyek kapacitása  $c''_e \in C''$  minden  $e'' \in E''$  szakaszra, amely kapacitás az eredeti gráf  $e \in E$  élének megosztható kapacitása:  $C''_e \equiv C_e$ , végül a  $\gamma''_e \in \Gamma''$  az  $e'' \in E''$  szakasz arányos költségét jelenti.

### 10. definíció $\gamma''_e \in \Gamma''$ arányos költség

Bevezettem a  $\gamma''_e \in \Gamma''$  arányos költség fogalmát, melyet az  $e''(v'_1, v'_2) \in E''$  élhez tartozó  $\gamma''_e \in \Gamma''$  arányos költséget az  $e'(v'_1, v'_2) \in E'$  szakasz  $\omega'_e \in \Omega'$  költségéből származtatjuk oly módon, hogy a  $\gamma''_e$  és  $\omega'_e$  egymással egyenes arányban vannak. A gráfba oda húzunk be  $e''(v'_1, v'_2) \in E''$  élet  $c''_e \in C''$  kapacitással,  $\gamma''_e \in \Gamma''$  költséggel, ahol az eredeti gráf modell  $e(v_1, v_2) \in E$  élén  $c_e$  megosztható kapacitás van és  $v_1 \equiv v'_1, v_2 \equiv v'_2$ .



Meskó Diána



8. ábra Az élkettőzés illusztrációja

Az ötlet lényege, hogy a 8. ábra szerint két párhuzamos éllel helyettesítsük a korábbi, két lineáris szegmensű költségfüggvénnyel bíró élt. Ezen párhuzamos éleknek ilyen módon már egyesével lineáris költségfüggvénye lesz (7. ábra). Az él, amely a megosztható tartalék kapacitást reprezentálja  $c_e''$  kapacitású, míg a másik él  $c_e'$  kapacitással a szabad kapacitást mutatja.

A hátránya, hogy legrosszabb esetben az élek száma megduplázódik és így a futási idő hosszabb lesz, míg az előnye az optimális eredmény.

A megosztható védelmmel ellátott utak elvezetéséhez kevert egészértékű lineáris programozás (Mixed Integer Linear Programming – MILP) formalizáció szükséges. Élketűzés felhasználásával a probléma lineárisra és megvalósíthatóvá válik.

### 3.5. Új tudományos eredmények: 2. Tézis

#### 2. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2012)

**Definiáltam a megbízható komplex egészségügyi ellátó rendszerek, MKÉ épület funkcióinak és funkció-kapcsolatainak modellezésére egy új, gráf alapú modellt (az 1.3. tézisben megfogalmazott gráf modell továbbfejlesztése):**

- 2.1. Az épületek modellezésére megalkottam egy új épület-modellt, melyen a nem-lineáris költségfüggvényű, védelmet is biztosító útvonalkereső algoritmusok lineáris programozási feladatként megfogalmazva megoldhatók. A modellben a nem-lineáris költségfüggvényű éleket megkettőzöm. Az új éleken a költségfüggvény konvex, lineáris szakaszokkal írható fel. A modellezési módszert él kettőzésnek neveztem.**
- 2.2. Bevezettem a  $\gamma_e'' \in \Gamma''$  arányos költség fogalmát, mely az élkettőzött él eredeti költségéből lineáris aránnyal számítható.**
- 2.3. Egészértékű lineáris programozási feladatként megfogalmaztam az élkettőzött épületmodellt.**

## 4.

### ÚTVONAL-KERESŐ ALGORITMUSOK

A disszertációban kidolgozott algoritmusok bármely virtuális vagy valós útvonalak, úthálózatok esetén alkalmasak útvonalkeresésre, ha a hálózatot fel szeretnénk készíteni egyszeres hibák elleni védelemre. Mindenki számára könnyen elképzelhető és leképezhető egy épületen belüli útrendszer egy úthálózat képére.

Az épületben fellépő meghibásodások vagy felújítási munkálatok idejére is biztosítani kell a közlekedési útvonalakat. Különösen igaz ez az egészségügyi ellátó intézmények esetében. Tűz és havári esetekben különösen a potenciálisan nagy kockázatú létesítmények esetén szükséges olyan módszer, mellyel automatikusan, akár a teljes épületrendszerre meghatározhatók a kimenekítési útvonalak.

Ilyen leképezést azonban nem csak egy épületben lévő közlekedési rendszerrel lehet párba állítani, hanem akár a távközlésben meglévő hálózatokkal is. Példaként hozhatjuk a többprotokollós címkekapcsolást (Multi Protocol Label Switching – MPLS (J. Guichard 2001)), amely egyre inkább átveszi az IP (Internet Protocol) forgalom szállításának feladatát (Black 2000), (S. Halabi 2000). Ebben az esetben az elvezetendő adatmennyiség és a megbízhatóság iránti igény növekedése miatt úgy kell útvonalat és tartalék erőforrásokat foglalnunk a címkekapcsolt utak számára, hogy azok minél kevesebb erőforrást használjanak fel a rendelkezésükre állóból.

#### ***4.1. Dijkstra-algoritmus, vagy más néven: legrövidebb-út kereső algoritmus***

A felhasznált gráf-modellnek köszönhetően a gráfelméletben ismeretes algoritmusok (R. 1999.) egyszerűen, közvetlenül felhasználhatóak.

Az üzemi útvonalak keresése megfelel a gráf-modellben a következő feladatnak: Keressünk több pontpár között egyidejűleg úgy útvonalakat, hogy minden élen és csúcson

Meskó Diána

legfeljebb annyi halad át, amennyi annak kapacitása, s legyen az útvonalak összhossza (összköltsége) a legkisebb. Ez a feladat NP-teljes, így nincs rá általános hatékony algoritmus. Egyetlen pontpárra meg tudjuk határozni az optimális utakat, és általában ezen algoritmusok segítségével keresünk közelítő megoldásokat az általános feladatra. Tipikusan, egymás után sorba vesszük a pontpárokat, és a még szabad kapacitásokat használva keresünk köztük egy üzemi és egy védelmi útvonalat, lefoglaljuk és lépünk tovább a következő pontpárra. Az utak keresésére két módszer terjedt el:

- Első esetben futtatunk egy Dijkstra algoritmust. Ez megadja az adott pontpár közötti legrövidebb útvonalat. Ennek éleit kitöröljük az épületet reprezentáló gráfból, majd a kapott részgráfban futtatunk egy második Dijkstra algoritmust. Ezzel biztosítjuk, hogy a két útvonal egymástól él-független lesz.
- A másik eset, amikor a Suurballe-féle (Suurballe 1974) módszert alkalmazzuk, vagyis két csomópont között keresünk két olyan él-független útvonalat, amely kettő összege a legrövidebb.

A kétszer futtatott Dijkstra algoritmus ugyan legrövidebb üzemi útvonalat talál, de lehet, hogy a védelmi útvonal már igen hosszú lesz. Olyan eset is előfordulhat, hogy ugyan létezik két éldiszjunk út a pontpár között, az első Dijkstra futtatása és az élek kitörlése után azonban már nincs lehetőség védelmi útvonal kijelölésére.

A Suurballe módszernek az az előnye, hogy ha létezik útvonalpár, akkor azt mindig megtalálja. Mégpedig olyat talál, aminek az összhossza legrövidebb lesz. Azonban a Suurballe módszernek is van hátránya: lehet, hogy az üzemi útvonal jóval hosszabb, mint amit a Dijkstra algoritmus találna, pedig ennek rövidege nagyon fontos szempont, míg a védelmi útvonal lehet hosszabb is.

#### ***4.2. Egyszerű algoritmusok ismertetése***

Hozzárendelt védelem esetén ezek után a következő módon futhat egy mohó algoritmus:

1. Kiválasztunk egy olyan pontpárt, amihez még nem rendeltünk üzemi és védelmi útvonalakat.
2. Ideiglenesen kitöröljük a gráfból azokat az éleket, amelyeken kisebb a kapacitás, mint amekkora kapacitású útvonalakra van szükség a két kiválasztott pont között.

Meskó Diána

3. Suurballe módszerével keresünk köztük két diszjunkt utat. A rövidebbik lesz az üzemi útvonal, a hosszabbik a védelmi.
4. Az ideiglenesen kitörölt éleket visszatesszük, a két útvonal által használt éleken levonjuk a kapacitásból az utak kapacitását.
5. Ha van még feldolgozatlan pontpár, akkor az 1. ponttól folytatjuk.

Egy másik szokásos módszer az, hogy először végigmegyünk az összes pontpáron, de az előzőtől eltérően csak az üzemi útvonalakat keressük meg, majd ha már megvan az összes üzemi útvonal, azután jelöljük ki a védelmi útvonalakat. Ebben az esetben természetesen a Dijkstra algoritmust használjuk az utak keresésére.

Megosztott védelem esetén előnyösebb a Dijkstra algoritmus használata. A következő módon nézhet ki ennek segítségével egy egyszerű algoritmus az üzemi és védelmi útvonalak kijelölésére, szakaszvédelem használatával:

#### 1. Üzemi útvonalak kijelölése

- a. Kiválasztunk egy olyan pontpárt, amihez még nem rendeltünk üzemi útvonalat.
- b. Ideiglenesen kitöröljük a gráfból azokat az éleket, amelyeken kisebb a kapacitás, mint amekkora kapacitású útvonalra van szükség a két kiválasztott pont között.
- c. Dijkstra módszerével megkeressük közöttük a legolcsóbb utat.
- d. Az ideiglenesen kitörölt éleket visszatesszük, az új útvonal által használt éleken levonjuk a kapacitásból az út kapacitását.
- e. Ha van még feldolgozatlan pontpár, akkor az 1a ponttól folytatjuk.

#### 2. Védelmi útvonalak kijelölése minden egyes $e$ élre

- a. Kitöröljük az  $e$  élt a gráfból.
- b. Kiválasztjuk azokat a pontpárokat, amelyekhez tartozó üzemi útvonalak használják az  $e$  élt.
- c. „Kitöröljük” ezeket az útvonalakat, azaz minden olyan élen, amelyen átmegy egy ilyen út, hozzáadjuk a szabad kapacitáshoz az út kapacitását.
- d. Minden egyes a 2b pontban kiválasztott pontpárra:
  - i. Ideiglenesen kitöröljük a gráfból azokat az éleket, amelyeken kisebb kapacitás maradt az üzemi útvonalak kiválasztása és a 2c pontbeli módosítás után, mint amekkora kapacitású útvonalra van szükség a két kiválasztott pont között.
  - ii. A gráf éleihez rendeljük új költségfüggvényt a következő módon: Minden olyan élen, ahol még nincs kijelölt védelmi út, legyen annyi a

Meskó Diána

- költség, amennyibe kerül ott lefoglalni az adott kapacitást. Egy olyan élen, amin van védelemnek lefoglalt kapacitás, és ez nagyobb, mint az aktuálisan szükséges, legyen  $0$  a költség. Amennyiben nem nagyobb, legyen akkora a költség, mint amennyibe kerül az adott kapacitásnyira növelni ott a védelemnek lefoglalt helyet. (Azaz a szükséges kapacitás és a meglévő védelmi kapacitás különbségének lefoglalásához szükséges költség. Ezen  $0$  kapacitás veszélyét láthattuk a 3.3.3 fejezetben, azonban ettől most tekintsünk el.)
- iii. Dijkstra módszerével keressünk legolcsóbb utat a két pont között. Ez legyen a kijelölt védelmi útvonal az adott pontpárhoz az  $e$  él meghibásodása esetére.
  - iv. A most megtalált út éleit vizsgáljuk meg: Ha azon az utunk kapacitásánál nagyobb kapacitású védelem volt kijelölve korábban, akkor vonjuk ki belőle az utunk kapacitását. (Hogy ne ütközzön két olyan védelmi útvonal, ami ugyanazon  $e$  él kerül el.) Ha volt ugyan védelmi kapacitás lefoglalva, de kevesebb, mint utunk kapacitása, akkor töröljük ki ezt a védelmi kapacitást. Ha nem volt rajta védelmi útvonal számára foglalt kapacitás, akkor nem változtatunk semmi.
  - v. A 2(d)i pontban törölt éleket állítsuk vissza.
  - vi. Ha van még olyan a 2b pontban kiválasztott pontpár, amihez még nem kerestünk védelmi útvonalat, akkor folytassuk a 2(d)i ponttól.
- e. A 2a pontban törölt élt tegyük vissza a gráfba, a 2c pontban módosított kapacitásokat állítsuk vissza. Az összes 2b-beli pontpáron menjünk végig, és az  $e$  él elkerülésére talált védelmi útvonalának éleihez adjuk hozzá védelmi-kapacitásként annak kapacitását. Ha van még olyan él, aminek a meghibásodását nem vizsgáltuk, akkor folytassuk a 2a ponttól.

A módszer nem sokkal bonyolultabb a hozzárendelt védelemnél leírtaknál, csak más (kisebb) költségekkel kell számolni a védelmi útvonalak keresésénél és vigyázni kell arra is, hogy egy rögzített  $e$  él meghibásodásakor elvezetendő igényeket kielégítő utak számára diszjunkt védelmi-kapacitásokat foglaljunk le.

Az előzőekben ismertettem néhány egyszerű közelítő módszert az üzemi és védelmi útvonalak meghatározására. Mint már említettük, a feladat valójában egy NP-teljes probléma, így nem várható, hogy hatékony algoritmussal optimális megoldást kapjunk. A korábban ismertetett Dijkstra vagy Suurballe algoritmusain alapuló megoldásokban sok

múlik azon, hogy a pontpárokat milyen sorrendben vesszük, és az ilyen módszerek gyakran igen költséges útvonal-rendszert határoznak meg.

A jobb eredmény érdekében – az NP-teljes feladatoknál általánosan alkalmazott – közelítőmódszereket és heurisztikákat (G. Ausiello 1999) szokás bevetni: Ezek közül a legismertebbek a szimulált lehűtésen alapuló, valamint a genetikus algoritmusok. A heurisztikák „jószágától” függően ezek igen gyorsak lehetnek, és jól közelíthetik az optimumot.

Bizonyítottan optimális megoldást kaphatunk, ha megoldjuk a feladatunkat leíró 3.3.1 fejezetben ismertetett *minimál költségű többtermékest folyam* problémát a 3.3.2 fejezetben részletezett *egészértékű lineáris programozási feladatként*, ami természetesen egy NP-teljes probléma, így bele kell törődnünk, hogy nagy méretű gráfokra nem feltétlen tudjuk (véges idő alatt) megoldani.

Az általam javasolt algoritmusok esetében az üzemi utak megkeresésekor a legfontosabb szempont a költségek minimalizálása, ennek megfelelően Dijkstra-algoritmust alkalmaztam. A következőkben ILP (egészértékű lineáris programként) felírom az előzőekben ismertetett egyszerű algoritmusokat.

Elsőként a védelem nélküli útválasztást ismertetem. Ebben az esetben csupán arról gondoskodunk, hogy az igényeknek merre biztosítsunk üzemi útvonalat. Ezen algoritmust alkalmazva az útvonalválasztás gyorsabb, azonban a hibák fellépésétől a helyreállításig sokkal hosszabb időre van szükség, mint bármely más, védelmet is biztosító algoritmus esetében: egy szakasz kiesésekor ugyanis újra kell számolni az útvonalakat és ki kell jelölni egy újat, ami igen sok időbe telhet.

## **11. definíció Védelem nélküli legrövidebb útkeresés ILP megfogalmazása**

A probléma ILP (Integer Linear Programming) feladatként formalizálható.

$E$  : az élek halmaza, amiben az egyes szakaszokat  $e$ -vel jelöljük

$F$  : a felmerülő igények halmaza, melyben egy igényt  $f$ -fal jelölünk

$x_e^f$  : egy bináris folyammutató változó az  $e$  szakaszon  $f$  termék esetén

$d^f$  : a kapacitás igény  $f$  termék esetén

$C_e$  : az  $e$  szakasz kapacitása

Meskó Diána

$x_{i,j}^f$  ( $x_{ki}^f$ ):  $f$  igény esetén az  $i$  ( $k$ ) csomópontból  $j$ -be ( $i$ -be) vezető élek.

**Célfüggvény:**

$$\text{minimalizáljuk } \sum_{e \in E} \sum_{f \in F} x_e^f d^f \quad (3)$$

**Korlátok:**

$$\sum_{f \in F} x_e^f d^f \leq C_e, \text{ minden } e \in E \text{ szakaszra} \quad (4)$$

$$\sum_{j=1}^N x_{ij}^f - \sum_{k=1}^N x_{ki}^f = \begin{cases} 0, & \text{ha } i \neq s \text{ és } i \neq t \\ 1, & \text{ha } i = s_o \\ -1, & \text{ha } i = t_o \end{cases}, \quad (5)$$

minden  $i$  csomópontra és  $f$  termékre.

$$x_e^f \in \{0,1\}, \text{ minden } e \in E \text{ szakaszra} \quad (6)$$

és  $f \in F$  termékre

(8) Ez az egyenlet a célfüggvény. Az  $x_e^f$  igényt súlyozzuk a  $d^f$  kapacitásigénnyel, és ennek az összes él esetére vett szummáját próbáljuk igényenként csökkenteni.

(9) Semelyik  $f$  igény esetén sem léphetjük át egyetlen  $e$  szakaszon sem az adott  $e$  szakasz kapacitását.

(10) Folyam-megmaradási törvény

(11)  $x_e^f$  egy bináris folyammutató változó, amely 0 értéket vesz fel, ha az adott  $e$  szakaszt nem használja  $f$  igény, ill. 1 értéket vesz fel, amennyiben használja  $e$  szakaszt.

### 4.3. Meghibásodás esetére felkészült útvonal-kereső algoritmusok

Rövid leírást adok néhány védelemről is gondoskodó algoritmusról. Majd bemutatom a referencia algoritmus és az általam javasolt algoritmusok közötti legfontosabb különbségeket:



Meskó Diána

Az *end-to-end SPP* (end-to-end Shared Path Protection, végponttól végpontig Megosztott Útvédelem) algoritmus az igények védelmi útvonalainak lefoglalt kapacitásokat nem rendezi át az újabb védelmi útvonalak keresésekor, az üzemi útvonalakat végponttól végpontig a kapacitásokat megosztva védi.

A *link-by-link SPP* (Link-by-Link Shared Path Protection, Szakaszonkénti Megosztott Útvédelem) algoritmus az igények védelmi útvonalainak lefoglalt kapacitásokat nem rendezi át az újabb védelmi útvonalak keresésekor, az üzemi útvonalak minden egyes szakaszához külön-külön keres kapacitás megosztással védelmi útvonalakat, melyek a teljes üzemi útvonaltól függetlenek lesznek.

A *PDSP* (Partially Disjoint Shared Protection, Részben Független Megosztott Védelem) algoritmus az igények védelmi útvonalainak lefoglalt kapacitásokat nem rendezi át az újabb védelmi útvonalak keresésekor, az üzemi útvonal egyes szakaszaihoz külön-külön keres védelmi útvonalat. Ezen védelmi útvonalakról bizonyosan csak annyit állíthatunk, hogy az adott szakasztól függetlenek lesznek, azonban nem tudjuk előre megmondani, hogy csak a vizsgált szakasztól, vagy az üzemi útvonalnak akár egy nagyobb szegmensétől, esetleg a teljes üzemi útvonaltól független a védelmi útvonal.

Az *SPP-LD* (Shared Path Protection – with Link Doubling, Megosztott Útvédelem Élkettőzéssel) algoritmus a védelmi útvonalak kapacitásait átrendezheti, és végponttól végpontig védi az egyes üzemi útvonalakat.

A *PDSP-LD* (Partially Disjoint Shared Protection – with Link Doubling, Részben Független Megosztott Védelem – Élkettőzéssel) algoritmus a védelmi útvonalaknak lefoglalt kapacitásokat átrendezheti, és a védelmi útvonal keresésénél csak azt a szempontot veszi figyelembe, hogy a meghibásodott élt ne használja a védelmi útvonal (úgy, mint a PDSP algoritmus esetén).

Terveztem és egészértékű lineáris programozási feladatként formalizáltam két új, megosztott védelmi módszert alkalmazó, az utakat adaptívan újrendező útvonalkereső algoritmust: *SPP-LD* és *PDSP-LD*. Formalizáltam a hagyományos hálózati- és költségmodellen alapuló megosztott védelmet alkalmazó referencia algoritmust, melyet a szakirodalom megosztható útvédelemként definiál (*SPP*).

Az általam javasolt algoritmusok az igények számára Dijkstra algoritmusával keresnek üzemi útvonalat, majd egyszeres meghibásodás esetére az általam megalkotott élettözött épület-modellben megosztott védelmi utakat biztosítanak, az egyik esetben az új igény teljes üzemi útjának, a másikban az üzemi út egy szegmensének, egy a védelem és a

helyreállítás közötti speciális módszerét alkalmazva. A védelem számára lefoglalt erőforrásokat többtermékes folyamatként, adaptívan újrendezik, amennyiben az, jobb épület-kihasználtságot eredményez.

## 12. definíció **Élkettozött algoritmus ILP megfogalmazása**

**Célfüggvény:**

$$\min \sum_{o \in T_e} \left\{ \sum_{e' \in E'} x_{e'}^o \omega_{e'} + \sum_{e'' \in E''} x_{e''}^o \gamma_{e''} \right\} \quad (7)$$

Ahol  $E'$  a szabad kapacitású  $\omega_{e'} \in \Omega'$  költségű éleket,  $E''$  a megosztott kapacitású  $\gamma_{e''} \in \Gamma''$  költségű éleket,  $e$  az eredeti épület-modellben a vizsgált él,  $T_e$  az elvezetendő igények halmaza, az  $x_{e'}^o$  és  $x_{e''}^o$  változók értéke jelzi az  $o$  igény védelmi útjának  $e'$  és  $e''$  élre eső részét (amely az eredeti gráf  $e$  élén fog megjelenni).

**Korlátok:**

$$\sum_{\forall j \in V', j \neq i} x_{ij}^f - \sum_{k=1}^N x_{ki}^f = \begin{cases} 0, & \text{ha } i \neq s \text{ és } i \neq t \\ b_o, & \text{ha } i = s_o \\ -b_o, & \text{ha } i = t_o \end{cases} \quad (8)$$

minden  $o \in T$  és  $v' \in V'$  esetén

$$\sum_{o \in T_e} x_{e'}^o \leq c_{e'} \quad \text{minden } e' \in E' \text{ esetén} \quad (9)$$

$$\sum_{o \in T_e} x_{e''}^o \leq c_{e''} \quad \text{minden } e'' \in E'' \text{ esetén} \quad (10)$$

$$0 \leq x_e^o \leq b_o \quad \text{minden } e \in E \text{ és } o \in T_e \text{ esetén} \quad (11)$$

$$z_{ik}^o \in \{0,1\} \quad (12)$$

$$\sum x_{ik}^o = b^o \cdot z_{ik}^o \quad (13)$$

(12) egyenlet a célfüggvény, (13) a folyam-megmaradási törvény, (14) – (15) kapacitáskorlátok az  $e' \in E'$  és  $e'' \in E''$  élekre, (16) a folytonosságot biztosító korlát, (17) egy bináris folyammutató segédváltozó, (18) az osztatlanságot biztosító korlát – a

Meskó Diána

korábbiakban bevezetett épület-modell esetén egy igény védelmi útja egy csomópontból csak az újonnan bevezetett él páron osztható.

Az SPP-LD és PDSP-LD algoritmusok közötti különbség az üzemi útvonaltól való függetlenség, amely az előbbi esetében a következő képpen adható meg:

$$x_e^o = 0 \quad (14)$$

bármely  $o \in T_e$  és  $e \in WP_o$  esetén

Ahol  $o \in T_e$  igény üzemi útvonala  $e \in WP_o$  élekből áll, míg utóbbi esetében csak a vizsgált  $e$  éltől való függetlenséget követeljük meg:

$$x_e^o = 0 \quad (15)$$

bármely  $o \in T_e$

#### ***4.4. Meghibásodás esetére felkészült útvonal-kereső algoritmusok alkalmazhatósága az épülettervezés folyamatában***

A legfontosabb kérdés, hogy miért is van szükség ezekre a modellekre és a védelmet is biztosító algoritmusokra az egészségügyi ellátó rendszerekben. Ahogyan arra már korábban is kitértem, az egészségügyi ellátórendszerekben nem megengedhető, hogy bizonyos funkciók, helyiségek ne legyenek elérhetők azért, mert például felújítási munkálatokat végeznek az épületben. Ezen funkciók zavartalan elérhetőséget igénylenek. Egy épületrendszer esetében ennek biztosítása csak körültekintő és gondos előzetes tervezéssel valósítható meg. Figyelembe kell vennünk már a tervezés kezdetén a meghibásodások, javítások lehetőségét. Ezt különféle képpen tehetjük. Optimalizálást végzünk a tervezés kezdeti szakaszában, mint egy elő-tervezési fázist. Ekkor modelleznünk szükséges az épületet, melyben meg kell határoznunk az épületen belüli helyiségeket és az azok között elérhető kapacitásokat, mint például egy folyosó átbocsátó képessége, az ajtók szélessége vagy egy várakozó helyiség kapacitása.

Ebben a modellben szimuláljuk az igények vagyis a páciensek közlekedését. Erre kiválóan alkalmas módszer a folyamatokkal való szimuláció, mellyel – ahogy azt szintén a

korábbiakban már láthattuk –  $s$ -ből  $t$ -be, forrásból nyelőbe vezető utat keresünk megfelelő kapacitással és minimális költséggel. Közben természetesen figyelemmel kísérjük az aktuálisan elérhető kapacitások mennyiségét. Amennyiben az épületben valahol szűk keresztmetszet alakul ki, a páciensek egy része nem tud közlekedni, blokkolódnak, akkor ott kapacitás növelésre, az épület megfelelő átstruktúrálására van szükség. Ez a modell esetében azt jelenti, hogy ha egyes élek mentén az igényeket nem tudjuk elvezetni, az igények blokkolásra kerülnek. Az épület átstruktúrálásával nyert épületet ismét modellezzük, majd újra szimulálunk.

Az igények elvezetését a modellben különféle módon végezhetjük. Ezeket a módokat algoritmusoknak hívjuk. A következőkben részletesen bemutatom a referencia algoritmusnak használt módszert, illetve az általam fejlesztett optimalizáló algoritmusokat. Az optimalizáláskor az algoritmus részeként a korábban megfogalmazott Dijkstra-algoritmust (legrövidebb út keresése adott hálózatban) alkalmazom. Az általam javasolt optimalizálási algoritmushoz felhasználok a 2.1 tézisben megfogalmazott modellt. Ezt hasonlítom a szakirodalomban is megtalálható, korábban felírt algoritmusokhoz.

Az algoritmusok egyes lépéseinek megjelenítésére legelterjedtebben használt eszköz a folyamatábra. Lássuk azokat az algoritmusokat, melyek releváns eredménnyel szolgáltak.

#### 4.4.1. SPP algoritmus folyamatábrája

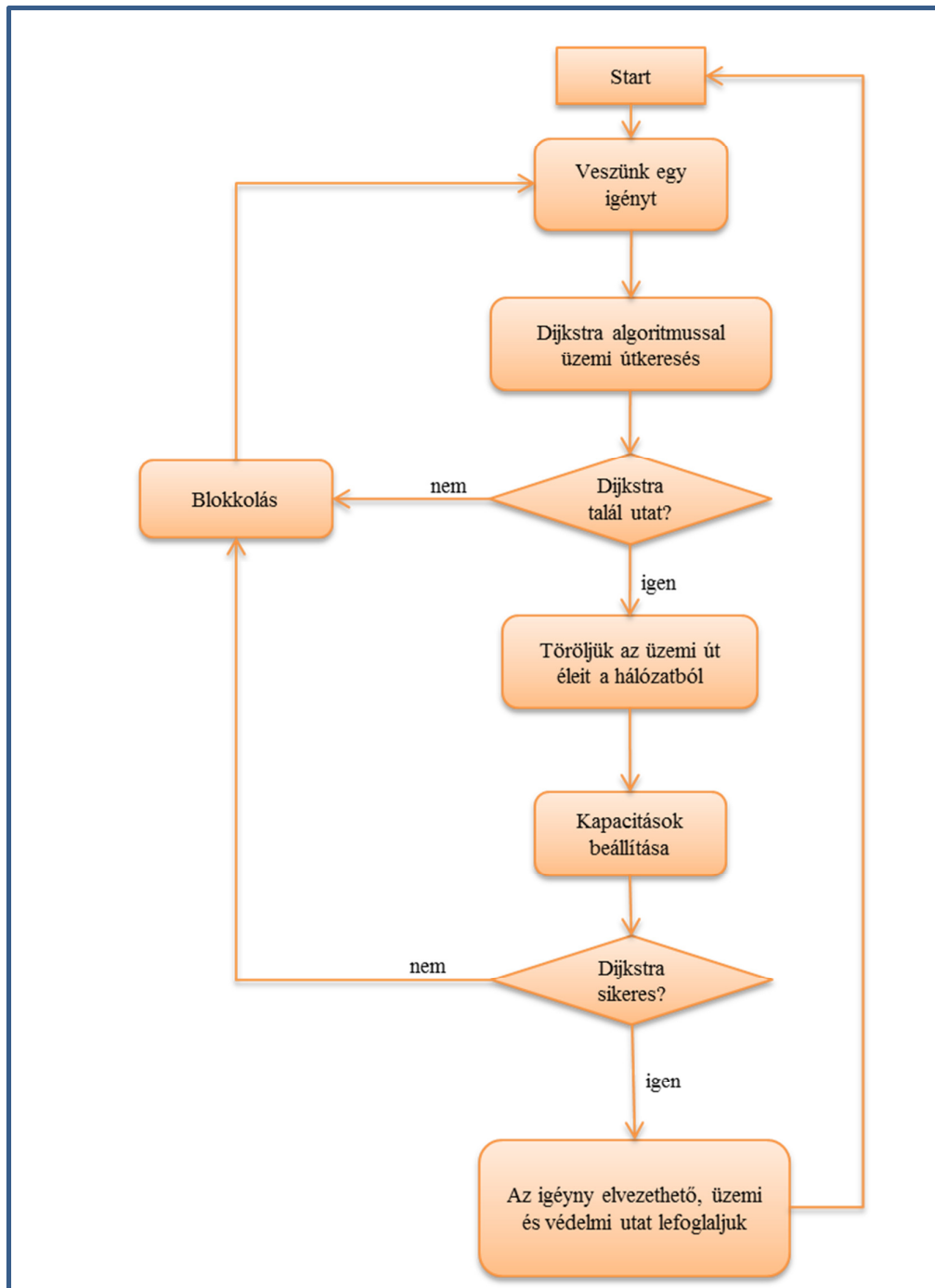
A legegyszerűbb módszer a hiba-független, dinamikus út-védelem. Működése a következő ábrán (9. ábra) követhető nyomon: az igényeknek először Dijkstra algoritmusával próbálunk meg üzemi utat foglalni. Ha nem sikerült, blokkolás következik be. Ez egészségügyi ellátó rendszerek esetében nem megengedhető, hiszen ez azt jelentené, hogy bizonyos funkciók nem lesznek elérhetők. Ilyen esetben újra kell gondolni az épület-modellt. Ellenkező esetben megpróbálunk számára védelmet is keresni szintén Dijkstra algoritmusával, de előtte szükség van az egyes élek költségeinek beállítására.

Ennek menete a következő: az üzemi út minden egyes  $l'$  élére, valamint a védelmi út minden szóba jöhető (az üzemi úttól független)  $l''$  élére kiszámítjuk a  $C_{l',l''}$  értéket, melynek jelentése: az adott  $l'$  él meghibásodása esetén az  $l''$  élen az igény összesen  $C_{l',l''}$  megosztható kapacitást talál. Végül vesszük a  $\min\{C_{l',l''}\}$  értéket, ez lesz az  $l'$  élre adódó költség kiszámításakor a  $C''$  érték. Ezek után képezhetjük a  $C_{sum} - C_{igény}$  értéket, amely

Meskó Diána

megmondja, hogy mennyi szabad kapacitás lefoglalására lesz szükség az adott élen, és e kettő megfelelően súlyozott összege adja meg az  $l''$  él költségét.

Végül a függetlenség biztosítása érdekében az üzemi út éleit átmenetileg töröljük a gráfból, ezt követően lefuttatjuk a Dijkstra-algoritmust, és amennyiben ez talál védelmi utat, az igényt elfogadjuk, üzemi és védelmi útjait lefoglaljuk a gráfban – ellenkező esetben blokkolás történik.



9. ábra SPP-LD és PDSP-LD algoritmus folyamatábrája

#### 4.4.2. PDSP algoritmus folyamatábrája

Partially Disjoint Shared Path Protection (PDSP), vagyis megosztott védelem részben független utak alkalmazásával. Ezen algoritmus dinamikus, hiba-függő szegmensvédelemet valósít meg.

Ahogy az algoritmus nevéből is látszik, nem az üzemi utaktól teljesen független védelmi utakkal dolgozik. Az algoritmus kezeli azt a korábban is említett feltevést, miszerint egyidőben csak egy hiba fordul elő a vizsgált épületrészben. Így ez a módszer egy üzemi út egy éléhez rendelt védelme, az üzemi út azon életől független, amelynek kiesésére készülünk, miközben bármely más üzemi élet használhatunk. Ez okból tartozik az algoritmus a hiba-függő módszerek körébe: adott esetben az üzemi út minden éléhez más és más védelmet rendelünk a hiba helyétől függően.

Így még nagyobb mozgásteret adunk az algoritmusnak, mint az előzőleg említett élenkénti megosztott útvédelem módszerénél.

Lássuk tehát a pontos működés leírását (folyamatábra: 10. ábra), mely az előző (SPP) módszerhez nagymértékben hasonlít: először üzemi utat keresünk Dijkstra algoritmusával, majd beállítjuk az egyes élek  $C_{l',l'}$  értékeit és költségét. Ezután a részleges függetlenség biztosítása érdekében töröljük a gráfból azt az egy élt, amelyet vizsgálunk (és amely kiesésére készülünk), majd ismét futtatunk egy Dijkstra útkeresést a megmaradt részgráfban. Ez esetben természetesen előfordulhat, hogy az üzemi és a védelmi út egy része egybeesik, csak kis szakaszon térnek el egymástól, és persze az is megeshet, hogy két, egymástól teljesen független utat kapunk. Egyedül azt tudjuk teljes bizonyossággal kijelenteni, hogy az  $l$  él kieséséhez rendelt védelmi út nem tartalmazza az  $l$  élt. Ezt megismételjük minden üzemi élre.

Ha a Dijkstra-algoritmus végig sikerrel jár, az igényt elfogadjuk, üzemi és védelmi útjaihoz szükséges kapacitásokat lefoglaljuk, ellenkező esetben pedig blokkoljuk az igényt.





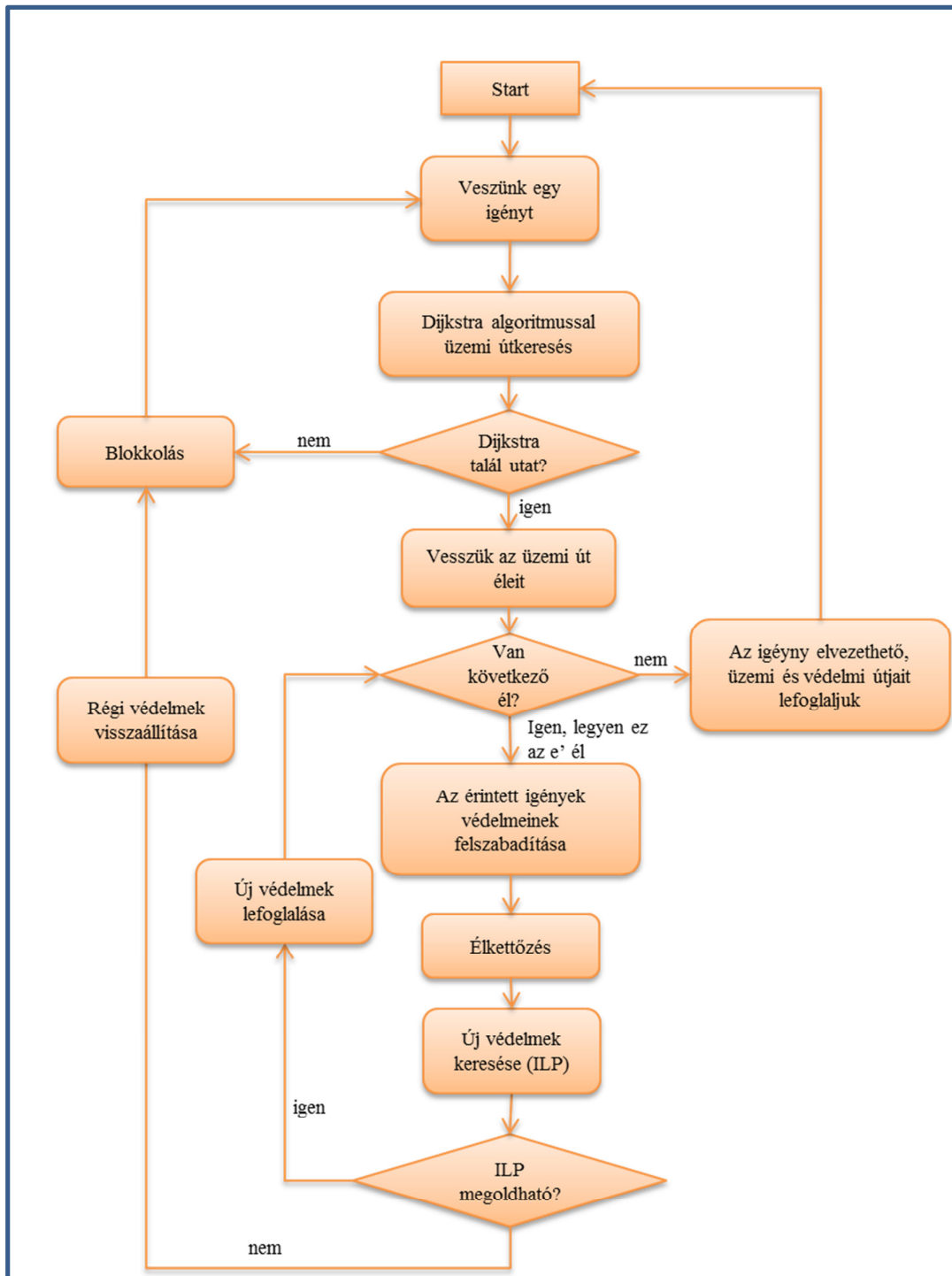
#### 4.4.3. SPP-LD és PDSP-LD algoritmus folyamatábrája

Az eddigi megoldásokhoz hasonlóan itt is Dijkstra-algoritmussal keresünk üzemi utat a gráfban. A különbség a védelmi út keresésében van: az SPP-LD a szakaszonkénti útvédelem (SPP) továbbfejlesztésének tekinthető, azonban ebben az esetben a védelmi út keresésekor nem csak az új igény védelmi útjának próbálunk helyet keresni, hanem a már lefoglalt igények egy csoportját is újra elvezetjük.

Ez azt jelenti, hogy egy  $e$  él kiesésének szimulálásakor (és a megfelelő védelmek keresésekor) az összes olyan igény védelmi útját töröljük, amely üzemi élként használta az  $e$  élet. Ezen igényeknek egyidejűleg próbálunk védelmet találni abban a részgráfban, amelyet a kiesett  $e$  él átmeneti törlése után kapunk. A következő pontban leírt ILP probléma megoldása adja az új utakat – és amennyiben talált érvényes megoldást, lefoglaljuk az összes védelmet, az új üzemi utat, és elfogadjuk az igényt.

Felmerülhet a kérdés, hogy miért csak az  $e$  él kiesése által érintett védelmi utakat rendezzük át, és miért nem az összeset? Ennek magyarázata az adódó matematikai probléma komplexitása. A gráfban összesen jelenlévő igények száma  $|V^2|$ -tel arányos, míg az adott élen futó védelmi utak száma  $|V^2|/|E|$ -vel. A lineáris programozási feladat változóinak száma tehát így kezelhető szintre csökken.

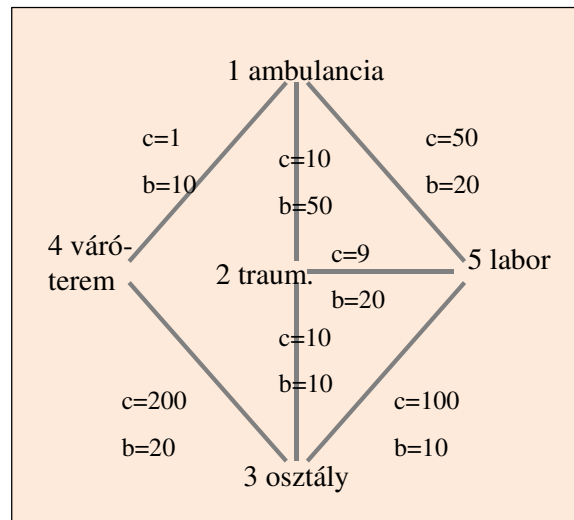
Ellenkező esetben visszaállítjuk az eredeti üzemi utak rendszerét, és az új igényt blokkoljuk.



11. ábra SPP-LD és PDSP-LD algoritmus folyamatábrája

#### 4.5. Egy egyszerű, ám annál szemléletesebb példa az algoritmusok működésére (lépésről lépésre)

Az algoritmusok által nyújtott teljesítmények közötti lényeges különbségeket egy példán keresztül mutatom be, melyben egy épületrészt modelleztem. Ezen a példán szemléltethető, hogy a gyakorlatban hogyan hozhatók létre egy-egy épületegység megfogalmazására a 2. tézisben bemutatott modellek, hogy alkalmazhatók a modellen az 3.3. tézisben felírt algoritmusok és hogy miért is szükséges az egészségügyi ellátórendszerekben ezen előzetes tervezési fázis alkalmazása. Ebben a példában a következő helyiségeket modelleztem: váróterem, melyből az ambulanciára és az osztályra közvetlenül lehet eljutni, míg a traumatológiára azonban csak az ambulanciáról és az osztályról. A labor legyen elérhető három helyről is: ambulancia, traumatológia és az osztály.

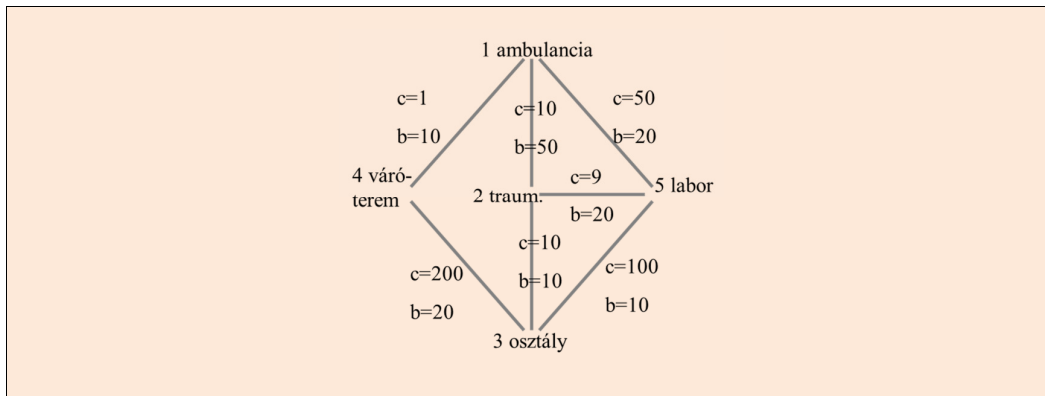


12. ábra Példa az épületek gráffal történő modellezésére

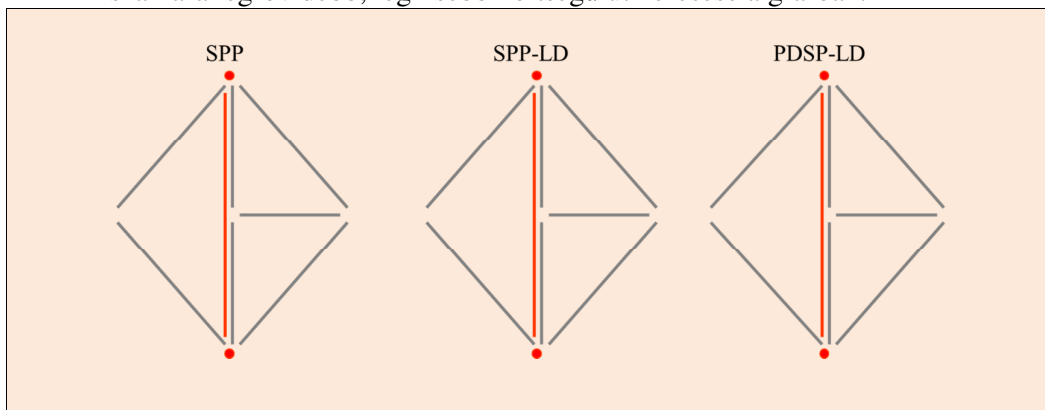
A következőkben a leírt épületrészen, vagyis részgráfon mutatom be lépésről lépésre az algoritmusok működését és a közöttük lévő legfontosabb különbségeket.

Meskó Diána

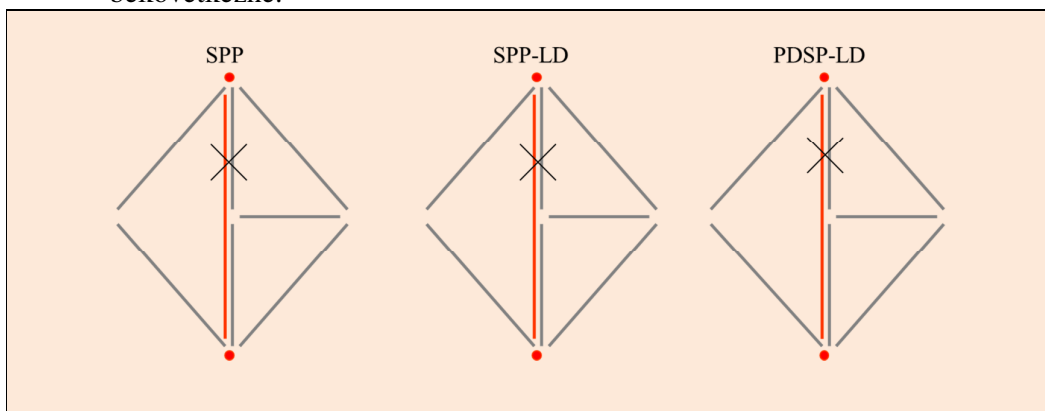
3. táblázat Példa az algoritmusok működésére (lépésről lépésre)



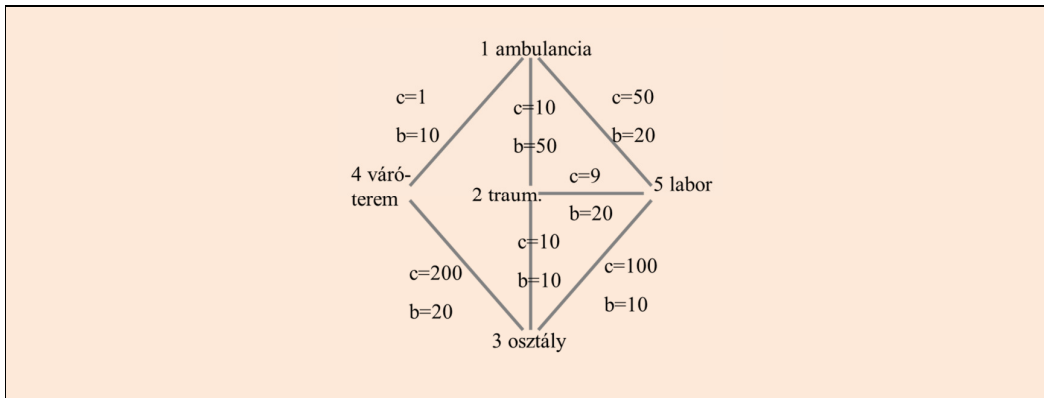
1. Lépés: érkezik a gráfba egy igény (1-3 csomópontok között). Üzemi útvonal számára legrövidebb, legkisebb költségű út keresése a gráfban.



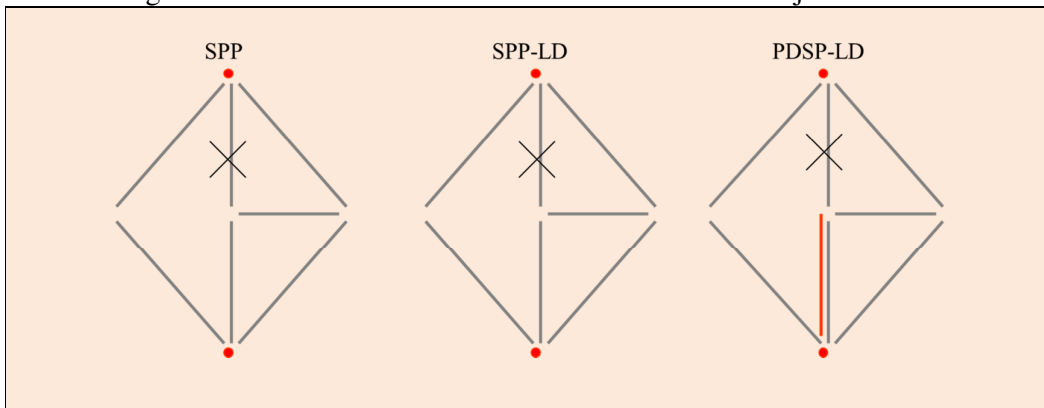
2. Lépés: (Él)meghibásodás esetére, a meghibásodott éltől független legrövidebb, legkisebb költségű út keresése arra az esetre, ha a meghibásodás bekövetkezne.



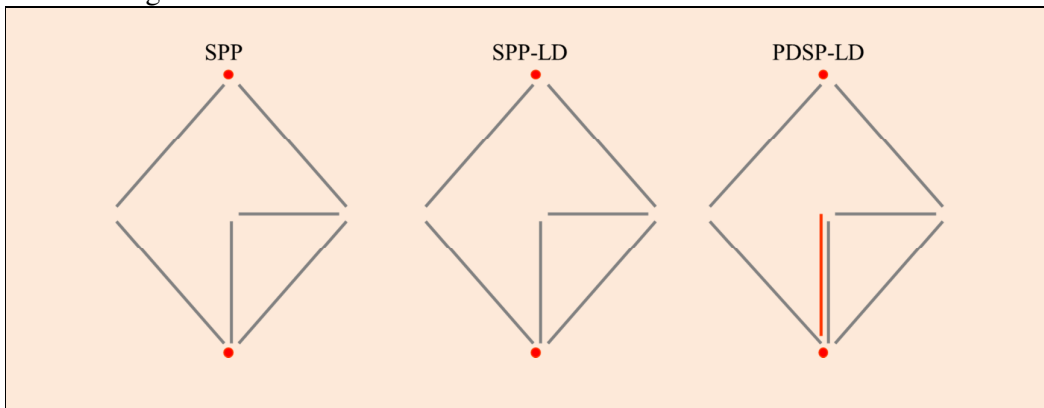
Meskó Diána



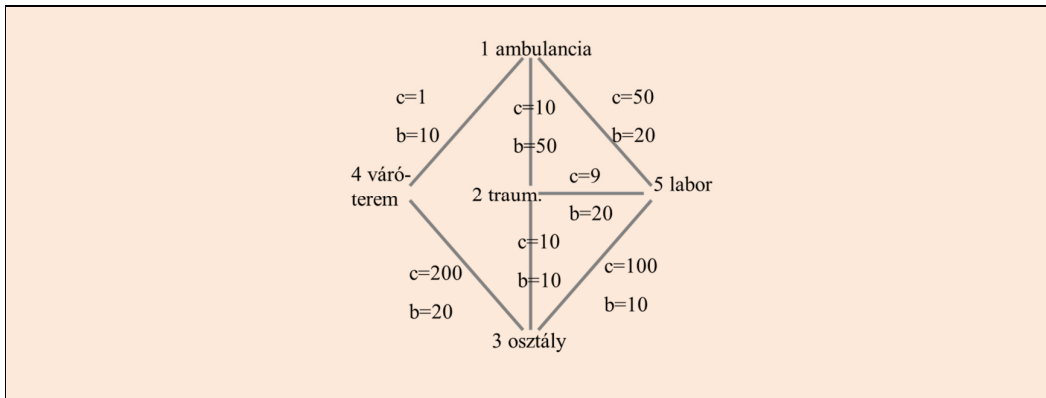
3. Lépés: töröljük a gráfból a meghibásodott él által érintett üzemi útvonalakat. SPP – Shared Path Protection – Megosztható Útvédelem – algoritmusok esetén a teljes üzemi útvonalat, míg PDSP – Partially Disjoint Shared Protection – Részben Független Megosztható Védelem – esetén csupán a meghibásodás által érintett üzemi útvonal szakaszát töröljük.



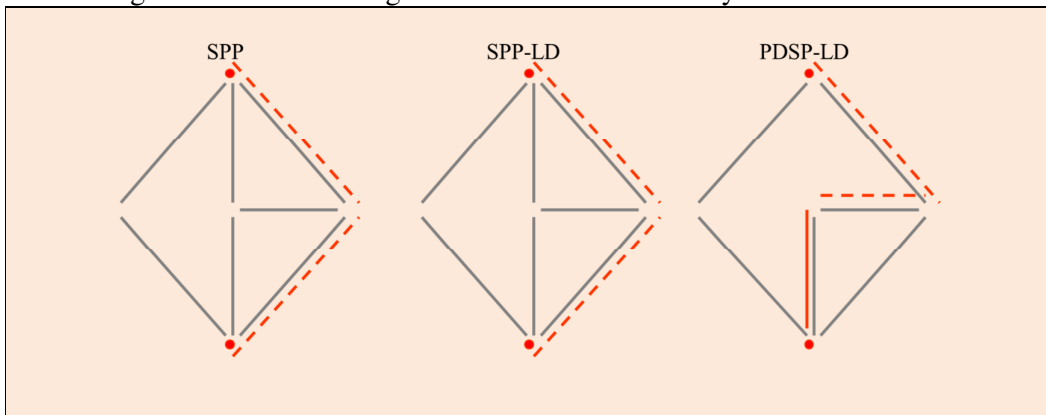
4. Lépés: Töröljük a gráfból a meghibásodott élt, ezzel szimuláljuk annak meghibásodását.



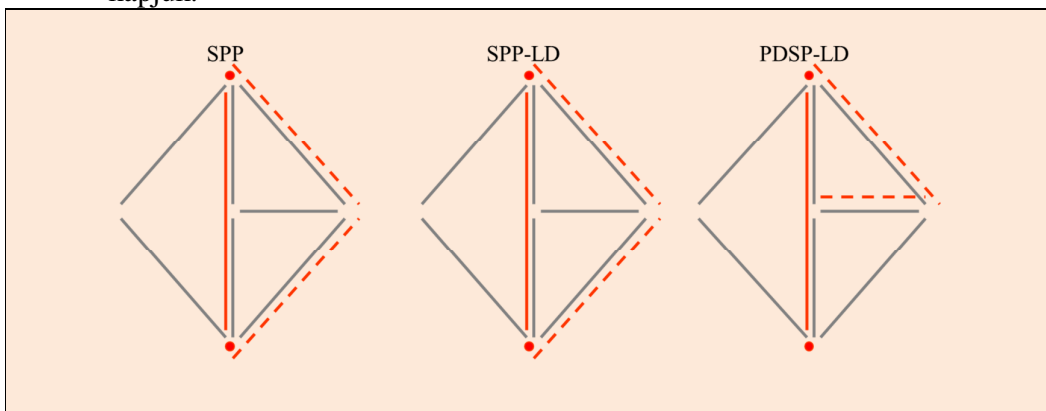
Meskó Diána



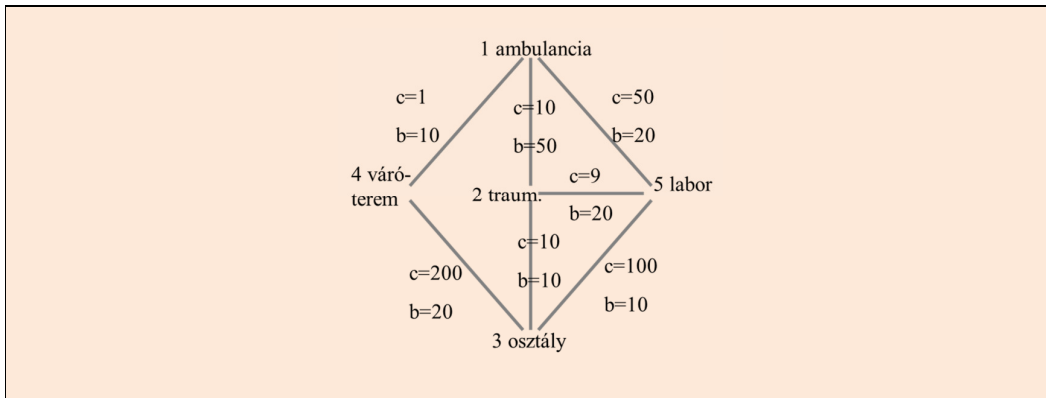
5. Lépés: Védelmi útvonalak keresése a megmaradt részgráfban, PDSP algoritmus esetén a megmaradt üzemi útvonal hiányzó szakaszára.



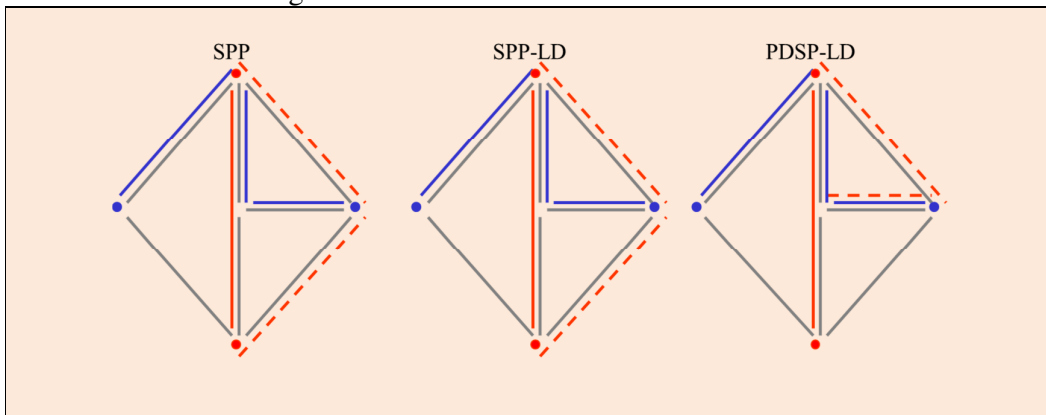
6. Lépés: a szimuláció végeztével az üzemi útvonalak számára lefoglalt kapacitást és a védelmi kapacitásokat is rendezve a következő gráfkepet kapjuk.



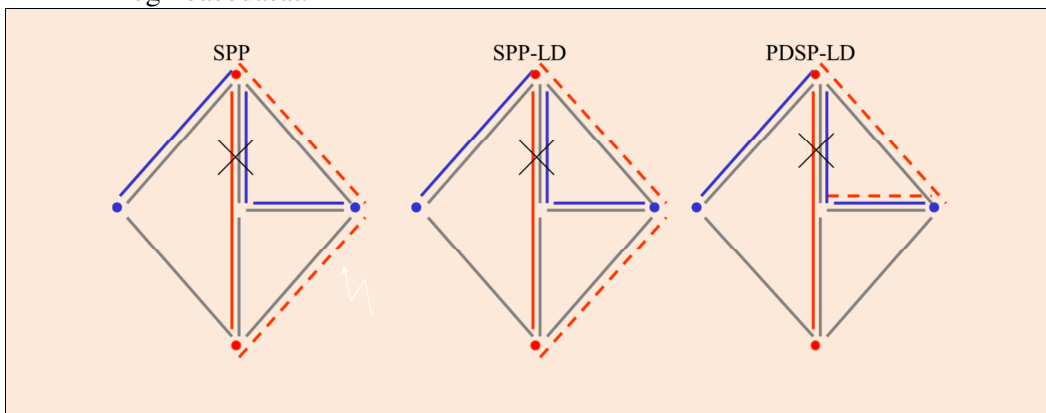
Meskó Diána



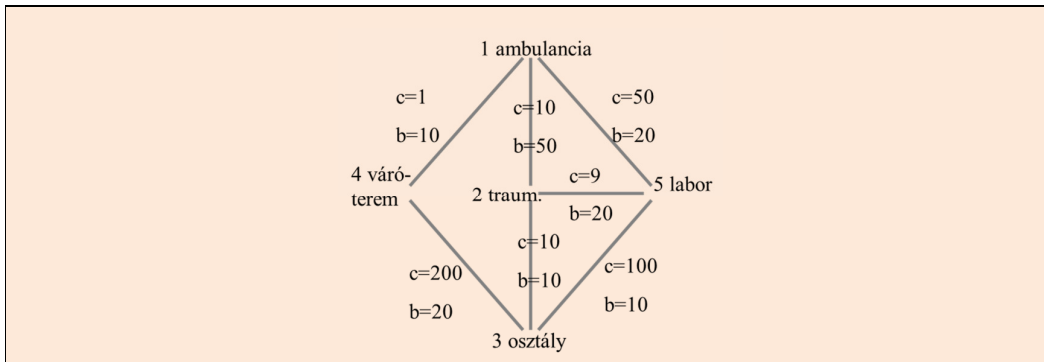
7. Lépés: a gráfban jelenlévő újabb igény számára is foglalunk legrövidebb, lekisebb költségű üzemi útvonalat.



8. Lépés: a példa kedvéért szimuláljuk ebben a helyzetben a közös szakasz meghibásodását.

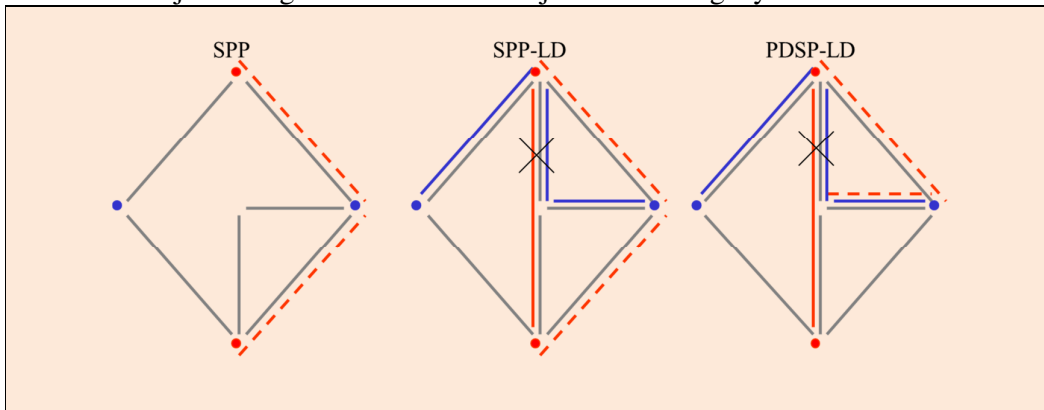


Meskó Diána



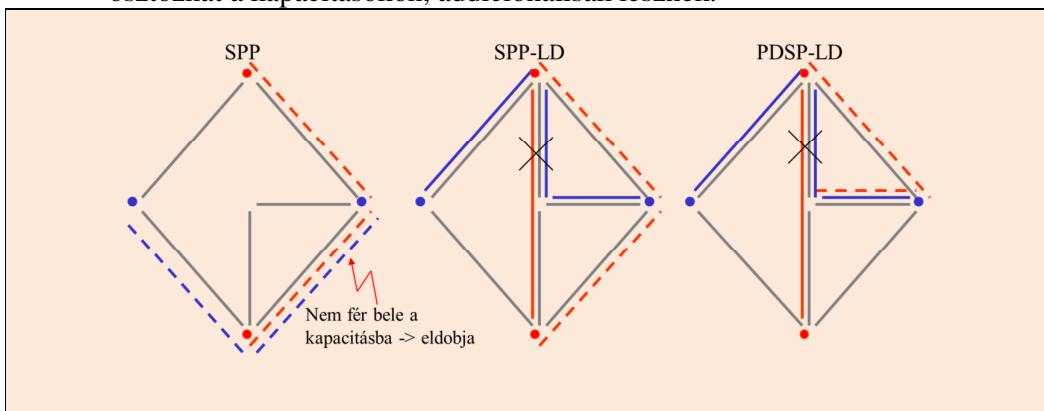
9. Lépés: **SPP** esetén

Töröljük a meghibásodott élt és a rajta elvezetett igényeket.



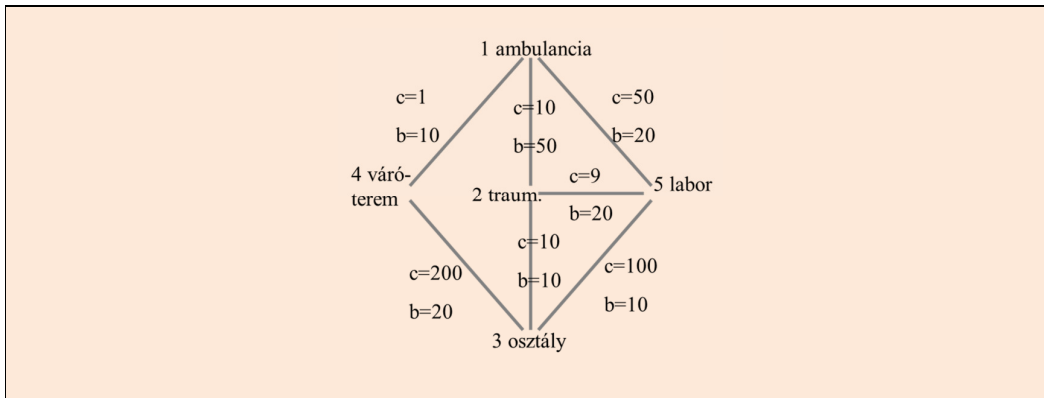
10. Lépés: **SPP** esetén

Megpróbáljuk elvezetni azokat a kapacitásokat, melyek érintettek. A piros védelmi útvonalat nem érinti a meghibásodás, így azt nem módosítjuk. Így azonban a 3-5 csomópontok között nincsen megfelelő mennyiségű kapacitás ( $b=10$ ) a már lefoglalt piros igény mellé a kék igény számára is védelmet foglalni, hiszen ennek az élnek a meghibásodása esetében a két igény nem osztható a kapacitásokon, addicionálisak lesznek.



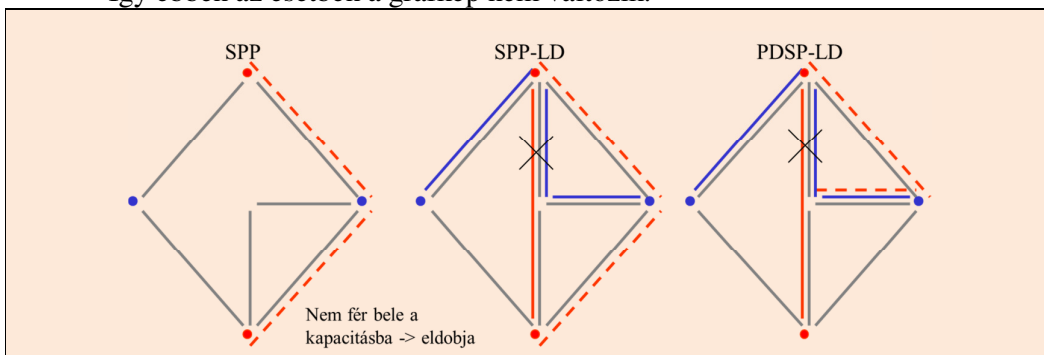


Meskó Diána



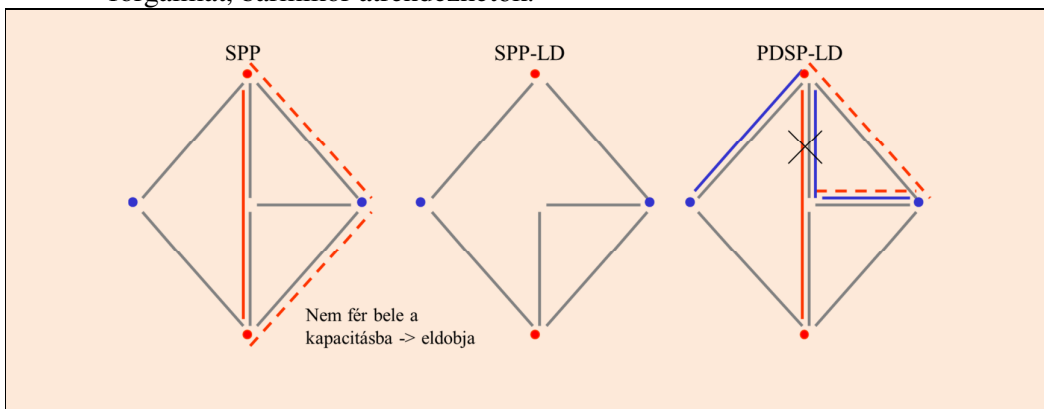
11. Lépés: **SPP esetén**

SPP algoritmus esetében ez a két igény nem szolgálható ki egyidejűleg, hiszen a közös szakasz miatt készülni kell a közös védelmi út használatra. Így ebben az esetben a gráfkép nem változik.

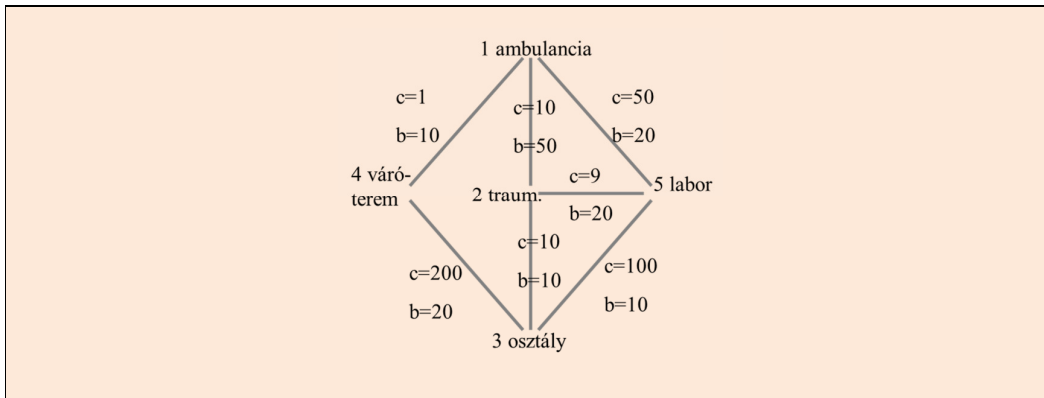


12. Lépés: **SPP-LD esetén**

Kitörljük a meghibásodott élt, a rajta futó üzemi útvonalakkal és a hozzájuk tartozó védelmi útvonalakat is, hiszen ezen kapacitások nem hordoznak forgalmat, bármikor átrendezhetők.

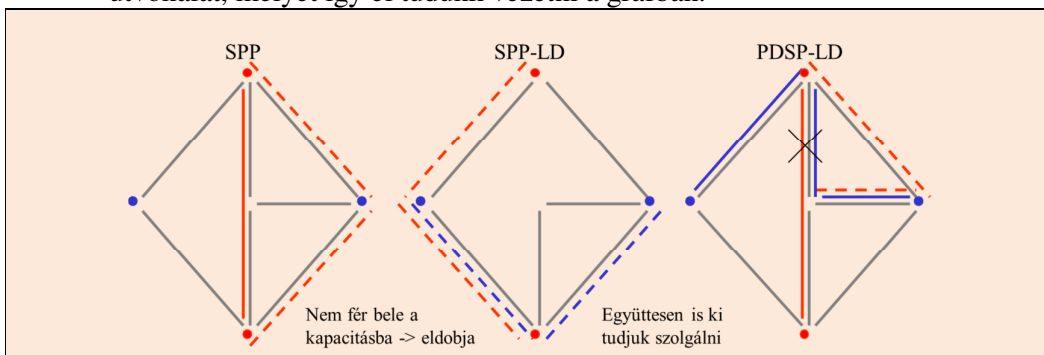


Meskó Diána



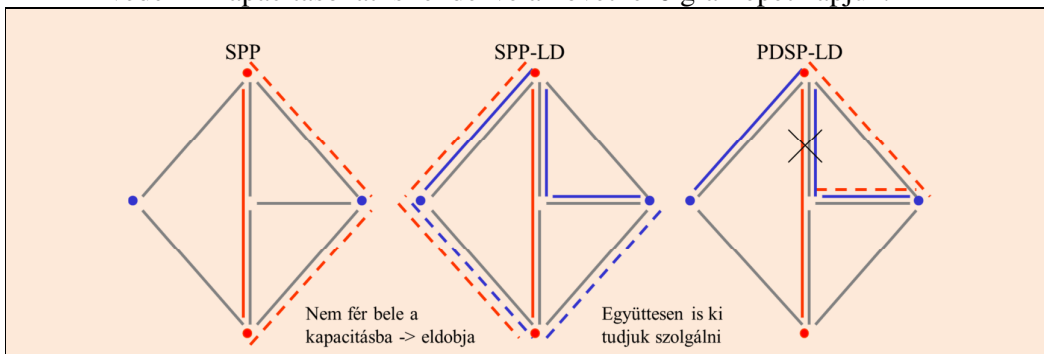
13. Lépés: **SPP-LD esetén**

MCMCF formalizációval az LD – Link Doubling – Élkettőzések épületmodellben egyidőben keresünk a jelen esetben két igénynek védelmi útvonalat, melyet így el tudunk vezetni a gráfban.

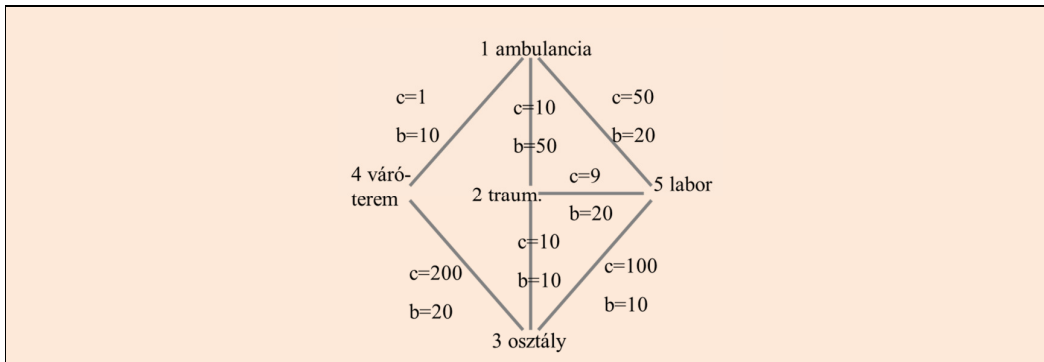


14. Lépés: **SPP-LD esetén**

a szimuláció végeztével az üzemi útvonalak számára lefoglalt kapacitást és a védelmi kapacitásokat is rendezve a következő gráfkepet kapjuk.

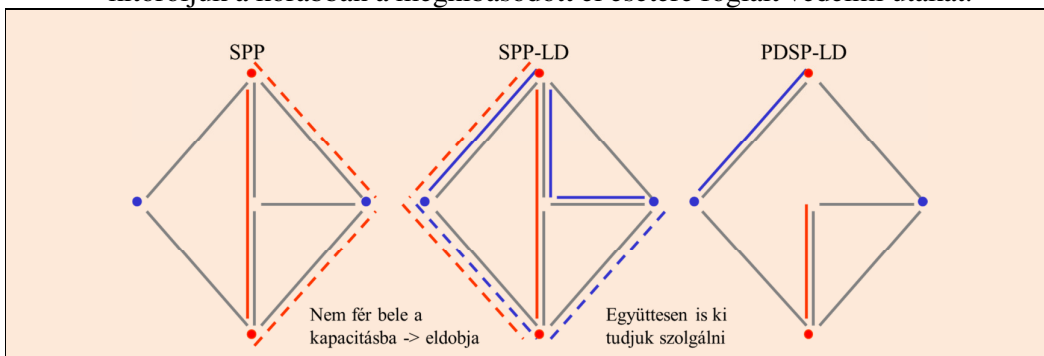


Meskó Diána



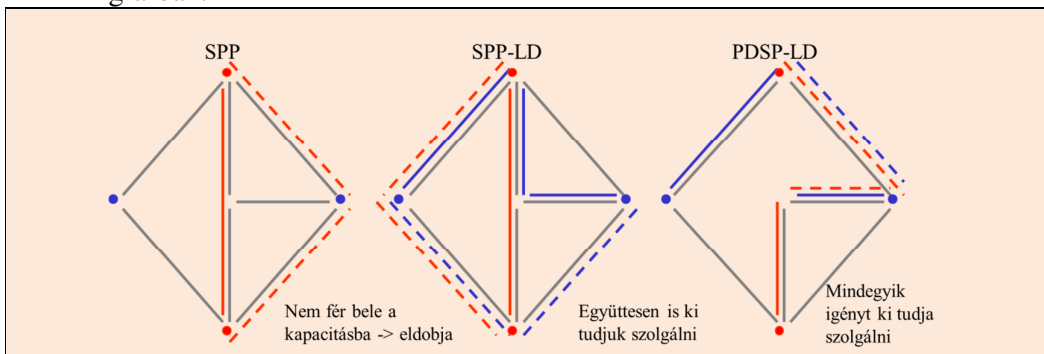
15. Lépés: **PDSP-LD** esetén

Hasonlóan az SPP-LD algoritmushoz itt is élünk azzal a tudással, miszerint a védelmi utak nem szállítanak forgalmat, így azok újrarendezhetőek. Tehát egyrészt kitöröljük a meghibásodás által érintett üzemi szakaszokat, valamint kitöröljük a korábban a meghibásodott él esetére foglalt védelmi utakat.

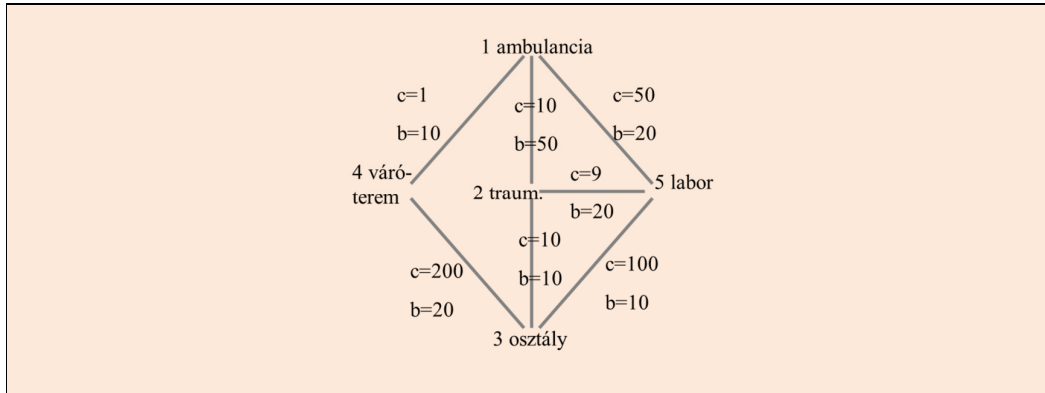


16. Lépés: **PDSP-LD** esetén

Az így kapott hiányos épületrészben, részgráfban keresünk védelmi utakat a hiányzó szakaszok számára. MCMCF formalizációval az LD – Link Doubling – Élkettőzéses épület-modellben egyidőben keresünk a jelen esetben két igénynek védelmi útvonalat, melyet így el tudunk vezetni a gráfban.

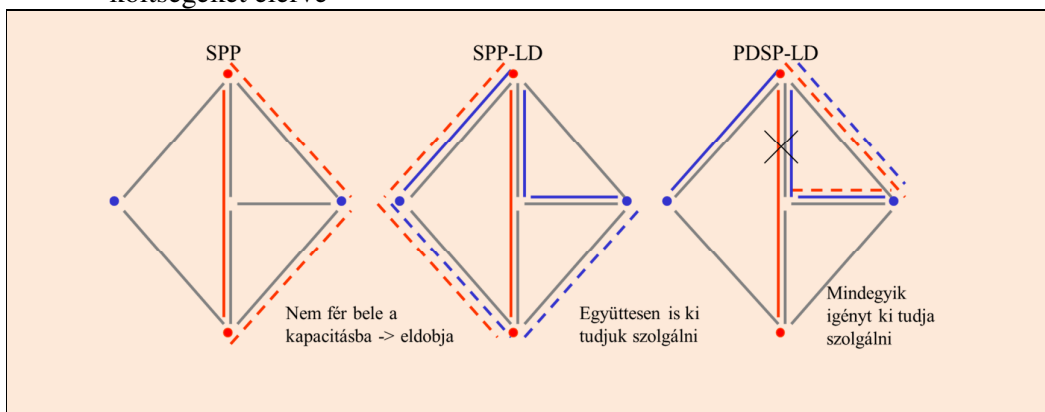


Meskó Diána



### 17. Lépés: PDSP-LD esetén

a szimuláció végeztével az üzemi útvonalak számára lefoglalt kapacitást és a védelmi kapacitásokat is rendezve a következő gráfkepet kapjuk. Ebben az esetben is mindkét igényt el tudtuk vezetni a következő összesített költségeket elérve



13. ábra Példa az SPP, SPP-LD és PDSP-LD algoritmusok futási eredménye

Az algoritmusok futtatásának végeztével kialakuló állapotot mutatja a 13. ábra. Jól látható, hogy ugyanazon a modellen az SPP algoritmus nem volt képes az összes igényt elvezetni, míg az SPP-LD és a PDSP-LD algoritmusok sikerrel jártak.

Azonban a védelmi kapacitások tekintetében is jelentős különbséget mutat a költségek tekintetében a két általam definiált algoritmus. Míg SPP-LD esetében a védelem számára tartalékban lefoglalt kapacitások mennyisége 501 egység, addig PDSP-LD esetében ez 120 egység. Úgy tűnhet, hogy a modell szándékosan túlzó értékeket mutat az egyes éleken, azonban ezek a költségkülönbségek fakadhatnak például abból, hogy az egyik él egy nagyobb kapacitású liftet jelent, míg a másik él csupán egy folyosó kiszélesítését takarja. Azonban a különbség akkor is szembeszökő, ha egy teljesen egységes mértékkel

számolunk, mégpedig a felhasznált élek darabszámával. Ebben az esetben a következő képpen alakulna a két javasolt algoritmus: SPP-LD esetében 4-szeresen, míg PDSP-LD esetében 3-szorosan használunk éleket (lásd a 4. táblázatban).

4. táblázat SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult költségek

	SPP	SPP-LD	PDSP-LD
Üzemi utak költsége	$(10+10)$ =20	$(10+10)+(1+10+9)$ =40	$(10+10)+(1+10+9)$ =40
Védelmi utak költsége	$(50+100)$ =150	$(1+200)+(100+200)$ =501	$(50+9+10)+(1+50)$ =120
Védelmi élek száma	-	4	3
Eldobott igények	1	0	0

#### 4.6. Új tudományos eredmények: 3. Tézis

##### 3. Tézis

(Meskó 2014 , Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Mitscenkov, Meskó és Cinkler 2007 - [Teljes dokumentum](#), [BME PA közlemény](#), Diána Meskó, Gábor Viola, Tibor Cinkler 2006 - [Google scholar hash](#), [WoS](#), [Google scholar](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)

Definiáltam két új útkereső algoritmust, melyek a megbízható épületek előtervezési fázisában, az épület modelljén legrövidebb útkereséssel adnak üzemi útvonalat és a meghibásodás által érintett páciens-forgalmat, mint minimál költségű többtermékes folyam problémát ábrázolva keresnek költséghatékony, takarékos, megosztott védelmi útvonalakat. A definiált algoritmusok az általam megalkotott, 2.2. tézisben formalizált modellen optimális megoldást közelítő eredményt adnak.

- 3.1. Terveztem két új, megosztott védelmet alkalmazó, a védelmi utakat adaptívan újraparendező útvonalkereső algoritmust: SPP-LD és PDSP-LD.
- 3.2. Egészértékű lineáris feladatként formalizáltam a hagyományos hálózati- és költség-modellen alapuló megosztott védelmet alkalmazó referencia algoritmust, melyet a szakirodalom megosztható útvédelemként definiál (SPP).
- 3.3. Egészértékű lineáris programozási feladatként formalizáltam az általam megalkotott két új, megosztott védelmet alkalmazó, a védelmi utakat adaptívan újraparendező útvonalkereső algoritmust: SPP-LD és PDSP-LD (Shared Path Protection with Link Doubling és Partially Disjoint Shared Path Protection with Link Doubling).

## 5.

### AZ ALGORITMUSOK KITERJESZTÉSE TÖBBTARTOMÁNYÚ ESETEKRE

Ebben a fejezetben azt tárgyalom, hogy hogyan terjeszthető ki a modellezés és az algoritmusok több-tartományú vagyis multi-blokk rendszerű épületek esetére. A korábbiakban láthattuk, hogy az egyes épület részeket hogyan modellezzük gráfokkal. Ha azonban a teljes épület-rendszert együttesen kezelni, átláthatatlanul nagy és nehezen kezelhető modellhez jutnánk. Ezért a feladatot a következő módon bontjuk könnyen átlátható és kezelhető egységekre:

- A feladaton dekompozíciót hatunk végre, részfeladatokra bontjuk, majd így modellezzük.
- Az így kapott modelleken alkalmazzuk az optimalizálási algoritmusokat.
- Majd ezek megoldásait azután aggregáljuk, összevonjuk egy nagyobb, komplex egységbe olvasszjuk, ezzel egy hierarchikus feladat megfogalmazást kapunk.

Így jutunk el a **több-tartományú hierarchikus modell**hez.

A multi-blokk rendszerű épületekről 2.1.2 alfejezetében (Egy- és több-blokkos épületrendszerek definiálása), a modell-alkotás folyamatáról a disszertáció 2.3 fejezetében (Épületek modellezése) írtam részletesen, míg a jelen fejezet hivatott megmutatni a több-tartományú és több-tartományú hierarchikus modellezés lényegét.

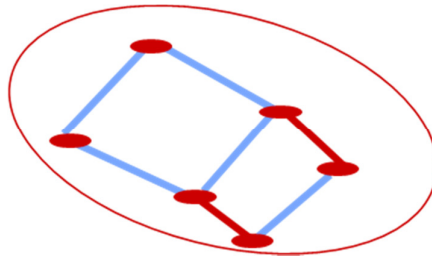
#### ***5.1. Épületek többtartományú és többtartományú hierarchikus gráf modellezése, összehasonlítása***

Ahogy azt a 2.3.4 alfejezetben megfogalmaztam, kölcsönös megfeleltetést teszek gráf és épület között. Egy épület-komplexumot azonban egy egyszerű gráffal nehézkes lenne megfogalmazni, modellezni, így azokat **hierarchikus többtartományú gráf topológiára** képezem le. A funkciók, a blokkok vagy osztályok, az épületek és

Meskó Diána

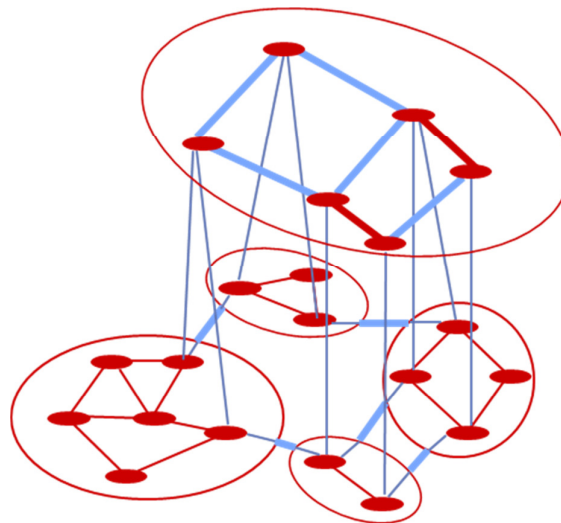
funkcionális kapcsolataik, vagyis a szállítási utak, rendre megfektethetők a csomópontoknak, al-tartományoknak, tartományoknak és a közöttük lévő éleknek.

Egy példát mutat be az alprobléma felrajzolására a 14. ábra.



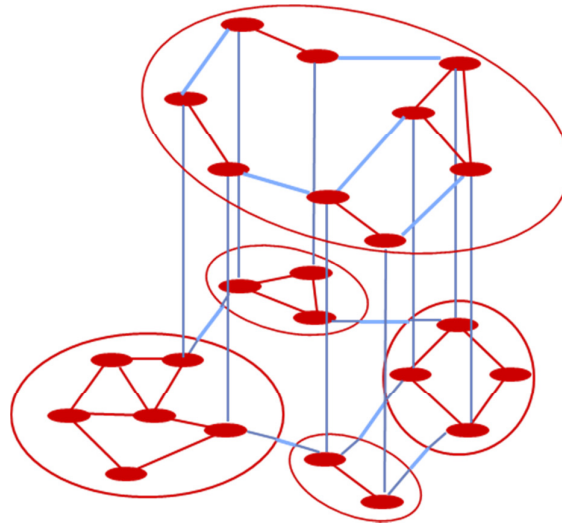
14. ábra Az épület legfelső szintű modellezése (épületek és a közöttük lévő kapcsolatok)

Az alprobléma megfogalmazásának dekompozícióját mutatja a 15. ábra és a 16. ábra. A dekomponált problémát úgy képzelhetjük el, hogy a korábban egy (vagy az összeköttetések számától függően néhány) csomóponttal reprezentált épületet felosztjuk megfelelő számú csomópontra és összeköttetésre. Így a korábban egy-egy épületet reprezentáló csomópont, mely több különböző osztályt és azok minden funkcióját magába foglalta, most egy-egy új tartományt eredményeznek, egy-egy épületet mutatnak, melyekben a csomópontok ettől fogva az osztályok azok minden funkciójával. Ez egy újabb modellt eredményez.



15. ábra Az épületek kifejtése, tartományokra osztása, funkció- és kapcsolat-meghatározás – I.

Meskó Diána



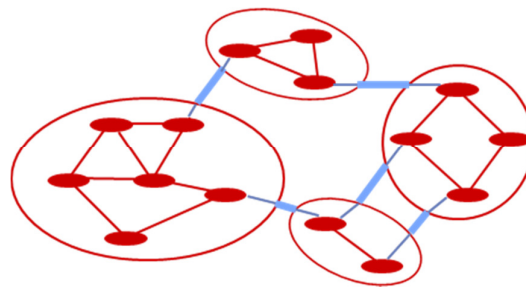
16. ábra Az épületek kifejtése, tartományokra osztása, funkció- és kapcsolat-meghatározás – II.

Amennyiben ezt a műveletet iteratíván újból elvégezzük és a tartományokat tovább bontjuk osztályokra, akkor a korábbi csomópontokból altartományok, míg az új gráf csomópontokjaiból már az osztályokon belüli funkciók lesznek.

### 13. definíció **Épületek többtartományú hierarchikus modellezése**

Ezzel a megközelítéssel lehetséges akár többszintű hierarchikus, többtartományos modell készítése az épület-komplexumokról (R. 1999.). A 15. ábra szemlélteti leginkább, hogy a modellezés többtartományú hierarchikus modellt eredményez. Minden egyes dekompozíciós szinten szükség van annak vizsgálatára, hogy egyszeres meghibásodás esetére a teljes szolgáltatás, az épület minden része változatlanul elérhető-e.

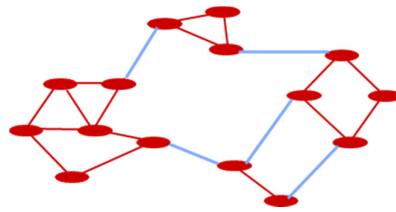
A 17. ábra és a 18. ábra mutatja, ahogy a komplex feladatról lebontjuk a felső szintet, így az alsóbb szinten lévő, kisebb rész-feladatokat lehet részletesen, aprólékosan megtervezni, az optimalizálást elvégezni.



17. ábra Többtartományú gráf reprezentációja a komplex épületrendszernek



Meskó Diána



18. ábra Egytartományú modellezése egy komplex épületrendszernek

Az optimalizálás végeztével a részfeladatok megoldásai aggregálhatók. Ehhez a megfelelő funkciókat blokkokba, altartományokba (helyiségeket osztályokba), a blokkokat tartományokba (osztályokat épületekbe) rendezzük.

Az előbbieken alapján, hogy ha a védelemről gondoskodó algoritmusokat a dekompozícióval nyert egyes hierarchia szinteken és az egyes tartományokon külön-külön alkalmazzuk, akkor a részeredmények összevonása, aggregálása után a teljes épületrendszerre elmondható, hogy az felkészült az egyszeres meghibásodásokra.

## 5.2. Új tudományos eredmények: 4. Tézis

### 4. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Mitcsenkov, Meskó és Cinkler 2007 - [Teljes dokumentum](#), [BME PA közlemény](#), Diána Meskó, Gábor Viola, Tibor Cinkler 2006 - [Google scholar hash](#), [WoS](#), [Google scholar](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)

**Megfogalmaztam, hogy dekompozíció alkalmazásával hogyan írható fel a teljes épületrendszer, az épületek többtartományú és többtartományú hierarchikus modellje. Megvizsgáltam, hogy a kapott modelleken alkalmazható-e az 1. tézisben felírt előtervezési fázis, valamint hogy a 3. tézisben felírt SPP és az általam kidolgozott SPP-LD és PDSP-LD módszerek.**

**4.1. Definiáltam az épületek többtartományú és többtartományú hierarchikus modelljét, mely az előtervezési fázis során a funkcionális modell többszöri dekompozíciójával jön létre.**

**4.2. Megmutattam, hogy az általam alkotott védelmi módszerek kiterjeszthetők többtartományú és többtartományú hierarchikus esetekre. Megmutattam, hogy a multi-blokkos épületekben történő meghibásodások, javítások esetére is alkalmazhatók a javasolt, takarékos védelmi módszerek.**

## 6.

### AZ ÚJ EREDMÉNYEK ALKALMAZHATÓSÁGA

Ebben a fejezetben megmutatom az általam megadott és megfogalmazott definíciók, tézisek, modellek és algoritmusok gyakorlati alkalmazhatóságát. Megmutatom, hogy miért szükséges és hogyan alkalmazható az egészségügyi ellátórendszerekben az előzetes tervezési fázis, a modellalkotás és iteratív, algoritmikus analízis alkalmazása. Szimulációs eredményekkel támasztom alá az általam javasolt módszer és algoritmusok hatékonyságát.

#### 6.1. *Egy példa*

Az előzetes tervezési-fázisban (pre-planning phase) modellezzük az épületrendszert, dekomponáljuk a modellt, alkalmazzuk a meghibásodás esetére védelmet biztosító algoritmusokat, iteratív lépések alkalmazásával eljutunk az épületrészek egy optimális funkciótervéhez, majd aggregáljuk a kapott eredményeket a teljes épületegyüttesre. Az általam javasolt folyamat a következőképpen foglalható össze:

1. első lépésként definiáljuk az épület komplexum épületeit és az azokba besorolt funkciókat: **felső hierarchia szintű modell**,
2. a feladatot részekre bontjuk, dekomponáljuk, megadjuk az egyes épületek funkcionális tervét: **tartományok modellje**,
3. amennyiben szükséges az átláthatósághoz, könnyebb követhetőséghez, akkor tovább bontható a feladat: **a tartományok altartományokra osztása**
4. minden hierarchia szinten és minden tartományra elvégezzük a következő lépéseket:
  - 4.1. az így nyert tartományok (altartományok) funkcióit és azok összekötöttségét határozzuk meg: **szomszédossági mátrix megadása**
  - 4.2. funkciók és összeköttetések kapacitása: **kapacitás mátrix definiálása**

Meskó Diána

- 4.3. megadjuk, hogy az egyes egységeket, a közlekedési utakat milyen költséggel lehet megvalósítani, használni (a költségdefiníció a korábbiakban látott módon kiterjeszhető a funkciók költségére): **költség-mátrix**
- 4.4. megadjuk a várható ügyfél-, páciens-, látogató- és/vagy beteg-áramlást: **várható folyamatok definíciója**.
- 4.5. megvizsgáljuk, hogy az előzetesen tervezett összeköttetések, költségek és kapacitások mellett milyen várható többletköltség szükséges, hogy az épület, a tervezett felújítási munkálatok, váratlan meghibásodások javítása idején is teljes szolgáltatást nyújtson. Ez magában foglalja a szűk keresztmetszetek felfedését és az iteratív újratervezési lépést, melyben a kapacitásokat növeljük, új funkciókkal és összeköttetésekkel bővítjük a modellt: **egyszeres meghibásodások ellen védelmet nyújtó algoritmusok alkalmazása**
5. a részfeladatok megoldásainak összevonásával elérjük a teljes épületrendszer funkcionális modelljét: **az épületrendszer aggregált modellje**.

### 6.1.1. Példa a megfogalmazott előtervezési fázis 1.-3. lépéseire - dekompozíció

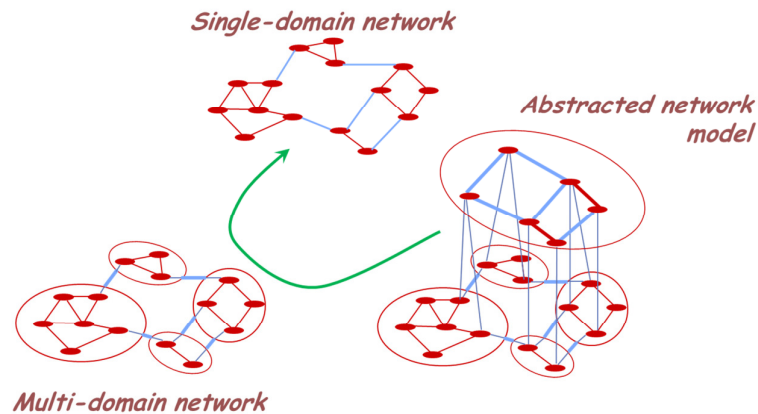
Dekomponált többtartományú hierarchikus modell készítése.

Az előbbieken láthattuk, hogy mit jelent a teljes feladat dekomponálásával nyert részfeladatok megfogalmazása, modellezése és iteratív optimalizálása. A részfeladatok megoldása után a teljes épületegyüttesre nézve is szükségünk van egy hasonló optimalizációra. Ilyenkor következik a dekompozícióval nyerhető feladatrészek eredményeinek aggregációja. De mit is jelent ez?

A teljes feladatot részekre osztottuk, optimalizáltuk, majd egybe szeretnénk gyúrni a megoldásokat, miközben egy újabb modellhez jutunk, ami egy újabb feladatot eredményez. Ezen újabb feladatot is meg kell oldanunk, hogy a teljes, egész feladatra nézve mondhassuk, hogy optimális megoldást javasolunk.

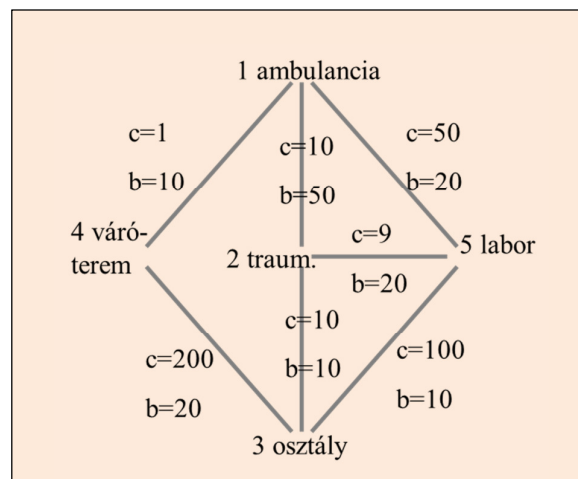
Emlékeztetőül idézzük fel a 14. ábra - 18. ábra sorozatban megvizsgált dekompozíció egyszerűsített példájára. A 19. ábra egy összesítést mutat. Dekompozíció során a zöld nyíl mentén haladunk.

Meskó Diána



19. ábra Modell dekompozíció

### 6.1.2. Példa a megfogalmazott előtervezési fázis 4. lépésére – modell analízis



20. ábra Példa az épületek gráffal történő modellezésére

#### 6.1.2.1. Szomszédossági mátrix leírása, a modell megfogalmazása

Ezt a gráfot legegyszerűbben egy szomszédossági mátrixszal definiálhatjuk. A szomszédossági mátrix metszéspontjaiban  $1$  szerepel, ha a két csomópont (funkció) között van kapcsolat, míg  $0$ , ha nincs közöttük kapcsolat definiálva. Az előbbi példa a következő kapcsolati mátrixszal definiálható:

Meskó Diána

5. táblázat 20. ábra kapcsolati mátrixsza

	váróterem	ambulancia	traumatológia	osztály	labor
váróterem	0	1	0	1	0
ambulancia	1	0	1	0	1
traumatológia	0	1	0	1	1
osztály	1	0	1	0	1
labor	0	1	1	1	0

#### 6.1.2.2. Kapacitás mátrix készítése

Hasonlóképpen, ha ezt a mátrixot a kapcsolatok közötti kapacitásokkal töltjük fel, azzal megadható az egyes kapcsolatok, élek kapacitása. Az előbbi példa B mátrixsza:

6. táblázat 20. ábra kapacitás mátrixsza

	váróterem	ambulancia	traumatológia	osztály	labor
váróterem	0	10	0	20	0
ambulancia	10	0	50	0	20
traumatológia	0	50	0	10	20
osztály	20	0	10	0	10
labor	0	20	20	10	0

#### 6.1.2.3. Költség mátrix készítése

Ha ezt a mátrixot továbbfejlesztjük, megadható az egyes kapcsolatok, élek költsége. Az előbbi példa C mátrixsza:

7. táblázat 20. ábra C költség mátrixsza

	váróterem	ambulancia	traumatológia	osztály	labor
váróterem	0	1	0	200	0
ambulancia	1	0	10	0	50
traumatológia	0	10	0	10	9
osztály	200	0	10	0	100
labor	0	50	9	100	0

#### 6.1.2.4. Várható folyamatok definíciója

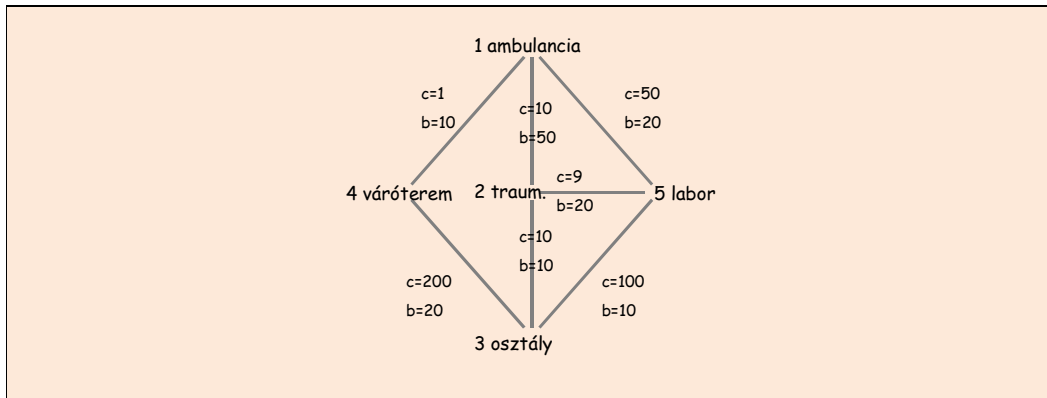
Ezen egyszerűsített modellben legyenek az érkező igények egységesen 10-10 egység kapacitás-igényűek. Ez jeleníti meg az épületben közlekedő emberek, páciensek, látogatók, orvosok, ellátó személyzet számát, de ide sorolhatók a gépek, ágyak számára szükséges kapacitás-igény is. A folyamatok a következő számhármassok sorozatával definiálható:

Meskó Diána

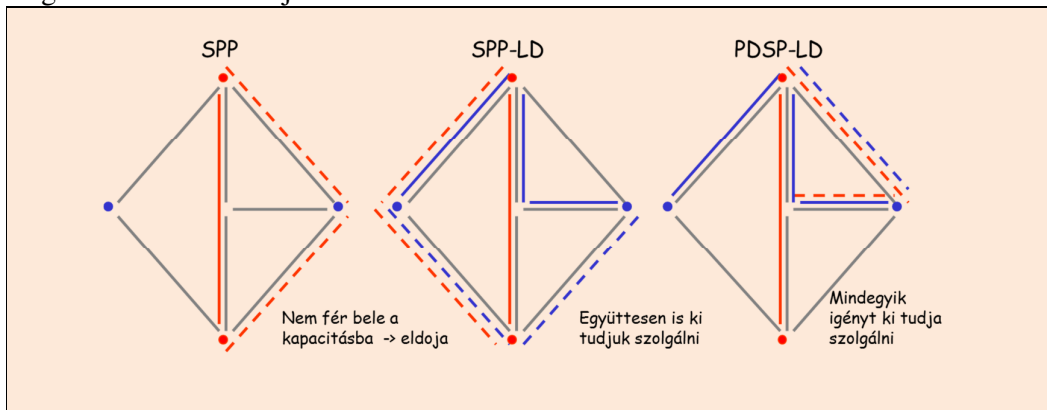
8. táblázat Igényvektorok felírása

Forráscsomópont	Nyelőcsomópont	Kapacitás igény
1	3	10
4	5	10
...	...	...

6.1.2.5. Az épületrészben a referencia és a javasolt algoritmusokkal végzett optimalizálás eredményei (részletezve a 1.1 fejezetben)



Az algoritmusok futtatásának végeztével kialakuló állapotot mutatja a 21. ábra. Ahogy a korábbi analízisből látható: ugyanazon a modellen, ugyanolyan igények esetén az SPP algoritmus nem volt képes az összes igényt elvezetni, míg az SPP-LD és PDSP-LD algoritmusok sikerrel jártak.



21. ábra SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult gráfkép

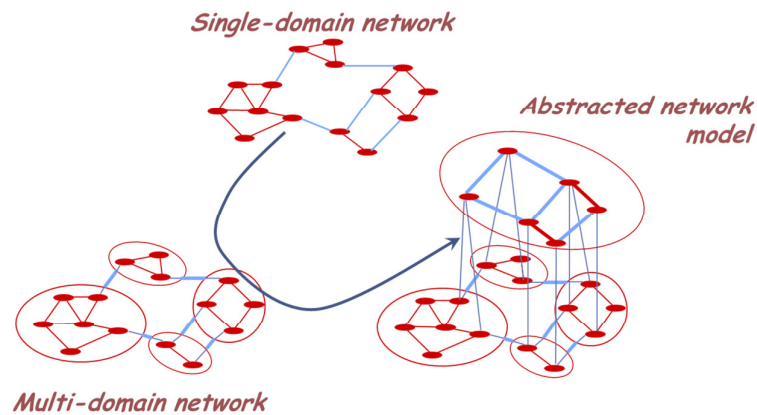
Hasonlóan itt is a korábbi elemzést idézem: a védelmi kapacitások tekintetében is jelentős költség-különbséget mutat a két általam definiált algoritmus. Az SPP-LD esetében a védelem számára tartalékban több kapacitást szükséges lefoglalni.

9. táblázat SPP, SPP-LD, PDSP-LD algoritmusok futtatása végén kialakult költségek

	SPP	SPP-LD	PDSP-LD
Üzemi utak költsége	$(10+10)$ =20	$(10+10)+(1+10+9)$ =40	$(10+10)+(1+10+9)$ =40
Védelmi utak költsége	$(50+100)$ =150	$(1+200)+(100+200)$ =501	$(50+9+10)+(1+50)$ =120
Védelmi élek száma	-	4	3
Eldobott igények	1	0	0

### 6.1.3. Példa a megfogalmazott előtervezési fázis 5. lépésére – aggregáció

Az előző lépésekben az egyes épületrészek funkcionális tervezését optimalizáltuk. Majd optimalizáltuk az épületrészek közötti kapcsolatokat is. Ebben a lépésben annyit teszünk, hogy a korábbi dekompozíciós lépéseket megfordítjuk, és az optimalizált épületrészek terveit **aggregáljuk**, ahogy azt a 22. ábra mutatja.



22. ábra Modell aggregáció

## 6.2. EREDMÉNYEK

Három módszer teljesítményét hasonlítjuk össze három különböző épület-modellen. A módszerek az SPP, SPP-LD és PDSP-LD, míg az épület-modellek: 16, 22 és 30 funkciót magukban foglaló épületrészek modelljei. Az épületben közlekedő pácienseket, hozzátartozókat, vagyis az embereket, valamint a gépeket, ágyakat együttesen igényekkel definiáljuk, melyeknek üzemi és védelmi utakat biztosítunk az épületben történő meghibásodások esetére. A modellezett épületek és az alkalmazott igények összefoglaló adatait 10. táblázat tartalmazza.

10. táblázat Szimulációhoz használt épület-modellek tulajdonságai

Épület-modellek	16 funkció	22 funkció	30 funkció
Modellezett funkciók száma	16	22	30
Összekötöttség	0,59	0,49	0,47
Szimuláció során elvezetett igények száma	646	709	1320
Egyidőben érkező igények száma	3,23	3,545	5,28
Szimulált időegységek	200	200	250
Átlagos jelenlétük az modellben	15,55	25,06	19,89

Vizsgálataim során különböző beállítások mellett figyeltem meg az algoritmusok teljesítményét. Alacsony kapacitás beállítárról indultak a szimulációk és lépésről lépésre a gráf összes élén egységesen megemelt kapacitásokkal haladtunk, míg mindhárom módszer el tudta vezetni az összes igényt. A valóságban ezen emeléseket az élek egyenként történő vizsgálatával kell végezni, azonban a teljesítmény-elemzéshez ez volt az adekvát módszer.

Ugyanakkor meg kell említeni, hogy ez a skálázás megfeleltethető az igények skálázásának, hiszen az, hogy egy épület kapacitásait csökkentjük, megfeleltethető az épületben közlekedő emberek számát növelésével. Tehát ha arra keressük a választ, hogy mennyi ember kiszolgálására alkalmas egy-egy épületrész, akkor akár az épületben lévő valós kapacitások lefelé történő skálázásával, akár az épületben közlekedők számának felskálázásával megadhatjuk a kérdésre a választ. Szimulációs szempontból az előbbi volt alkalmasabban megvalósítható.



A szimulációk során a 90%-os blokkoltságtól a nagyjából 0% blokkoltságig végeztem a vizsgálatokat. A szimuláció során érkező igények számossága elegendő volt ahhoz, hogy az épület kezdeti, tranzienis viselkedésétől eltekintsünk.

A szimulációkhoz a kód C++ nyelven íródott, az MILP megoldásához ILOG CPLEX 9.030-at használtunk.

### **6.2.1. Az épületek maximális átbocsátó képességének vizsgálata a három módszer segítségével**

A 23. ábra azt mutatja, hogy az adott épület-modellek esetében az épület kapacitásának függvényében hogyan változik az épület kiszolgáló képessége, vagyis hogy az igények hány százalékát nem tudja kiszolgálni az adott épület. Általánosságban az mondható, hogy az SPP-LD algoritmus csupán kicsivel hozott jobb eredményeket, mint az SPP algoritmus (néhány esetben az egyszerűbb SPP algoritmus teljesített jobban), azonban a PDSP-LD algoritmus jelentős javulást eredményezett a másik kettőhöz képest.

Emlékezzünk vissza, hogy mi is volt az eredeti kérdés: az épületek maximális átbocsátó képességének vizsgálata.

A 16 funkciót megjelenítő modell esetében az SPP algoritmussal az épületben 200 egységnyi kapacitással érjük el, hogy minden közlekedőt kiszolgáljunk, míg az SPP-LD algoritmus esetében ez már 170 egységnyi kapacitásnál, a PDSP-LD esetében pedig már 140 egységnyi kapacitásnál bekövetkezik. Ez azt jelenti, hogy az SPP algoritmushoz képest az SPP-LD algoritmust alkalmazva 15%-kal kevesebb, míg a PDSP-LD alkalmazva 30%-kal kevesebb kapacitásra van szükség az adott igények kiszolgálásához. Ahogy azt az előbbiekből láttuk, kapacitások felskálázása megfeleltethető az igények lefelé skálázásának, vagyis SPP-LD alkalmazásával 15%-kal több, míg PDSP-LD alkalmazásával 30%-kal több páciens esetén is azt mondhatjuk, hogy az épület az egyszeres meghibásodások esetében is fennakadások nélkül, teljesen üzemképes marad.

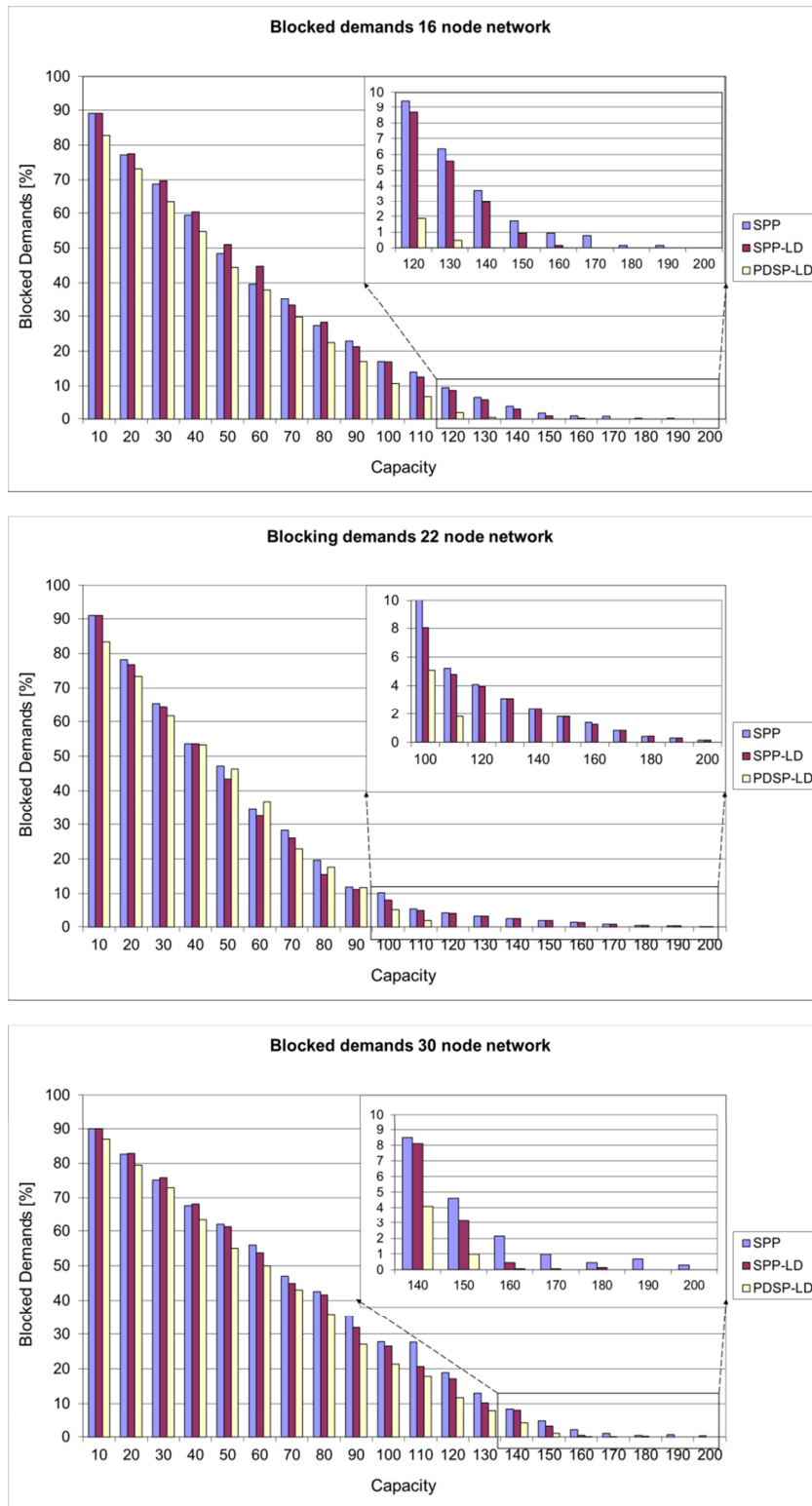
A 22 funkciót megjelenítő modell esetében ez a következő képpen néz ki: az SPP referencia algoritmushoz képest az SPP-LD algoritmus nem hoz szignifikáns javulást, azonban a PDSP-LD algoritmus 40%-kal, míg a 30 funkciójú épületet modellező gráf esetében az SPP referencia algoritmushoz képest az SPP-LD 10%-kal, a PDSP-LD 20%-kal több páciens kiszolgálása esetén is fennakadások nélkül, teljesen üzemképes marad az egyszeres meghibásodások mellett.

Meskó Diána

11. táblázat Az épületmodellekben a referencia és a javasolt algoritmusokkal elérhető maximális átbecsátó  
képesség vizsgálata

	SPP	SPP-LD	PDSP-LD
16 funkciós	100%	85%	70%
22 funkciós	100%	100%	60%
30 funkciós	100%	90%	80%

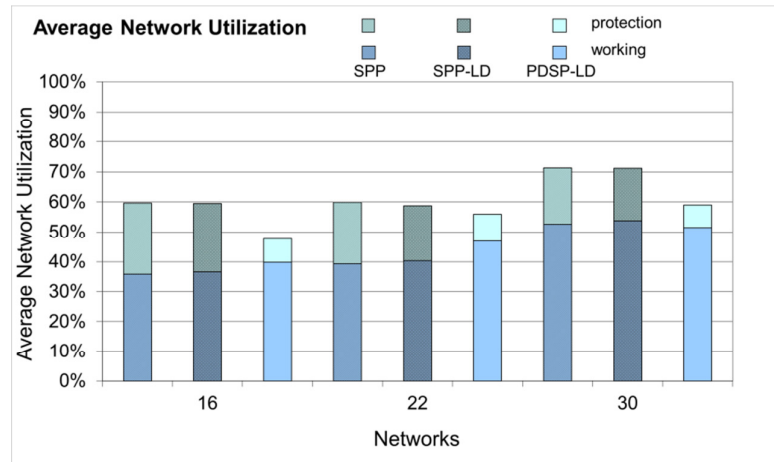
Meskó Diána



23. ábra Az épület-modellekben fellépő blokkolás az összeköttetések kapacitásának függvényében (16, 22, 30 csomópontos gráf esetében)

### 6.2.2. Épületek kihasználtsága

A 24. ábra segítségével azt szemléltetem, hogy a szimuláció során alkalmazott épületmodellek egyes élein milyen kapacitás eloszlást eredményeznek a különböző algoritmusok. Az itt látható, összes élre nézett átlagos kapacitás eloszlások azon szimulációk esetében álltak elő, amikor a blokkolás 1% alatti értéken volt.



24. ábra Az épület-modellek átlagos kihasználtsága

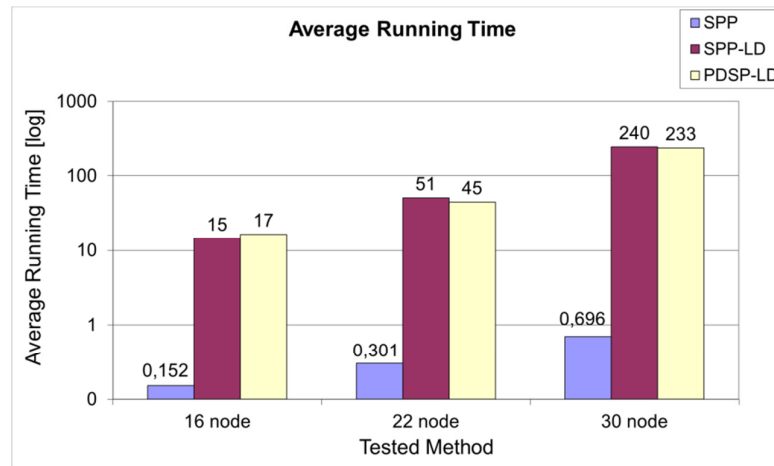
Szimulációkkal igazoltam, hogy a PDSP-LD algoritmus alacsonyabb összkapacitásmennyiséget foglal az épületmodellekben, mint a referencia SPP és a javasolt SPP-LD algoritmusok. Mindazonáltal a kapacitás eloszlás tekintetében, az üzemi útvonalak számára foglalt kapacitás mennyisége a PDSP-LD algoritmus esetében magasabb, ami azt jelenti, hogy a védelmi kapacitások számára jelentősen kevesebb többlet erőforrásra van szükség. Míg ebben a tekintetben az SPP és az SPP-LD hasonló kapacitásmennyiséget foglalt az üzemi útvonalak számára és csak alig észrevehetően kevesebb kapacitást a védelmi útvonalak számára.

### 6.2.3. Algoritmusok futási ideje

A 25. ábra logaritmikus skálán mutatja az összes futtatás átlagos futási idejét. Ez azt jelenti, hogy a szimulációk során volt olyan épület-modellünk is, amelyben csaknem minden igényt már az üzemi útvonal keresésekor el kellett dobni, hiszen nem volt elegendő a kapacitás számára és akadt olyan is, amikor minden igényt el tudtunk vezetni. Ez utóbbi eset jelentősen megnöveli a számítási időt, hiszen a Minimál Költségű Többtermékes Folyamok (MCMCF) Kevert Egészértékű Lineáris Programozási feladatként (MILP) történő megoldása polinomiális idejű, vagyis az igények számával

Meskó Diána

hatványozottan nő. Habár ez lenne az üzemi útvonalak keresésének optimális módja, mégis emiatt nem alkalmazható módszer, hogy egy egész épület teljes modelljére egyidejűleg, az összes felmerülő igény számára üzemi útvonalat keressünk. Ahogy az az ábrán látható, az MILP (Kevert Egészértékű Lineáris Programozás) megoldása szignifikánsan hosszabb időt igényel.



25. ábra Az átlagos futási idők logaritmikuskálájára

### 6.3. Új tudományos eredmények: 5. Tézis

#### 5. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitcsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)

Több minta-rendszeren számítógépes szimulációs eljárással numerikusan, számításokkal bizonyítottam, hogy az általam javasolt tervezési módszertan alkalmazásával megbízható komplex épületeket lehet tervezni, valamint megmutattam az általam alkotott algoritmusok költség- és kapacitás-hatékonyságát.

5.1. Megmutattam a minta-rendszereken, hogy az általam definiált előtervezési-fázist alkalmazva, mellyel az építészeti tervezést megelőzően, az épület funkcionális modelljét megalkotva, egy hiba esetére is minimális többletköltséggel üzemképes komplex egészségügyi ellátó rendszerek tervezhetőek.

- 5.2. Szimulációs eljárást alkalmazva, numerikus eredményekkel alátámasztva megmutattam, hogy az SPP algoritmushoz képest az SPP-LD és a PDSP-LD javasolt algoritmusokkal alacsonyabb többletköltség és többletkapacitás mellett érhető el ugyanaz az épület-kihasználtság.**
- 5.3. Minta-rendszereken szimulációval igazoltam, hogy a PDSP-LD algoritmus kevesebb kapacitást foglal úgy, hogy az üzemi kapacitások mennyisége nagyobb arányú az egymáshoz képest hasonlóan teljesítő SPP és SPP-LD-hez képest.**

## ÖSSZEGZÉS ÉS ÚJ TUDOMÁNYOS EREDMÉNYEK

Értekezésem elején célul tűztem ki, hogy korszerű, nagy alapterületen elhelyezkedő, de ugyanakkor tájba simuló blokk-szerkezetű komplex egészségügyi ellátó rendszerek építészeti tervezése során a blokkok optimális elhelyezéséhez, a blokkok közötti kapcsolatok tervezéséhez nyújtok segítséget figyelembe véve azt, hogy az épületek megbízható épületek legyenek.

Az építészeti tervezés folyamatát vizsgáltam. Azt találtam, hogy az épületek meghibásodás esetére történő vizsgálata, ellenőrzése teljesen hiányzik a tervezési folyamatból. Bár jogszabályi előírások, normák rendelkezésre állnak a tervezés folyamán, azonban nincs a kezünkben olyan eszköz, mellyel ez automatizálható, ellenőrizhető lenne.

A disszertációmban megfogalmazott eszközökkel meghibásodások, tűz vagy havária esetén is üzemképes, megbízható épületek tervezhetők. Az általam definiált eszközök a következők: előtervezési fázis; speciális, több tartományú hierarchikus, élkettőzött gráf; védelmet is biztosító, többtermékes folyam problémák megoldására alkalmas, matematikai útkereső algoritmusok; épületek üzemszerű működésének modellezése és az épületek meghibásodásának szimulációja.

Az optimalizálási eszköz a fizikailag elkülönülő blokkok helyének meghatározásában, a blokkok közötti funkcionális kapcsolatok kezelésében játszanak szerepet.

A korábbi, több éves, adatátviteli rendszerek matematikai modellezése területén folytatott kutatásom során megismert módszertani elemeket felhasználtam a blokk-rendszerű egészségügyi ellátó rendszerek építészeti tervezéséhez kapcsolódó modellek kifejlesztésében. Erre a "természetes" analógia adott lehetőséget.

Egy fontos különbség a kutatás teljes ideje alatt jelen volt: míg az adatátviteli rendszerek esetében – érthető okokból – a modellek futtatási sebessége fontosabb tényező, mint a modellek futtatási eredményének minősége, addig az építészeti tervezés folyamatában a modellek eredményének minősége elsődleges szempontként kezelendő, mivel egy "gyorsan kapott, de csupán hozzávetőleges elhelyezés" évtizedekre

Meskó Diána

meghatározza egy ellátó rendszer működését. Fontos az alapfeladatot szem előtt tartanunk: egy épületet több tíz évre tervezünk. Az algoritmusok futtatási ideje elenyésző ahhoz képest, mint amennyi tervezési problémát, szűk keresztmetszeteket előzetesen feltárhatunk az általam megfogalmazott modell és algoritmusok alkalmazásával.

A 2. fejezetben bevezettem a *több-blokkos* és a *több-blokkos hierarchikus épületrendszerek* fogalmát, valamint vizsgáltam az épületeken belüli funkcionális egységeket és azok kapcsolatát, illetve az egyes épületek közötti kapcsolatokat. Definiáltam a *megbízható komplex épületrendszerek* fogalmát.

Kitértem a tervezés módszertanára és arra, hogy a tervezési folyamat 0. tervezési fázisában van lehetőség biztosítani, hogy az épület komplexumok megbízható épületek legyenek. Ehhez az épületek modellezése szükséges. A modellezésre a gráf modellezést alkalmaztam és beláttam, hogy ezen modellen a gráfokon alkalmazható eljárások végrehajthatók, így az épületeken belüli útkeresés megfeleltethető a gráfokban történő útkeresésnek.

Az elért eredményeket az 1. tézis ismerteti.

## 1. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2012)

**Definiáltam a megbízható komplex épületek (MKÉ) fogalmát. Megfogalmaztam az MKÉ tervezéséhez szükséges előzetes, az építészeti tervezést megelőző funkcionális tervezési lépést, majd definiáltam egy egyedi gráf-leképezést, amely az előtervezés során alkalmazható és megmutattam a gráfelméleti algoritmusok, védelmi módszerek alkalmazhatóságát.**

**1.1. Definiáltam a megbízható komplex épületek (MKÉ) fogalmát, amelynek mennyiségi és minőségi attribútumai az alábbiak:**

- **roboosztus, olyan épületeket foglal magában, melyek az épületben bekövetkező meghibásodások esetén is teljes kapacitással üzemképesek,**
- **redundáns elemeket és útvonalakat tartalmaznak,**
- **azonos funkciójú (redundáns) elemek térben szeparáltak,**



- amelyet, mint rendszert funkcionálisan modellezték és optimalizáltak az alábbi eljárás szerint.
- 1.2. Leírtam és javasoltam egy tervezési módszert: előtervezési fázis, előzetes funkcionális modellezés, amely során a funkciók listázása, a lista alábontása, a funkciók csoportokba (épületekbe) történő besorolása, a funkciók közötti kapcsolódásoknak, mint feltétel-rendszernek a definiálása történik az épület építészeti tervezése előtt.
  - 1.3. Megfogalmaztam az épületek gráfokra történő leképezését: modellezésre alkalmazott módszer során épület-komplexumok, épületek, blokkok, funkciók és a közöttük lévő kapcsolatok modellezése gráfokkal. Az épületek funkcióit, közöttük lévő kapcsolatait, átbocsátó képességüket, költségeit kölcsönösen egyértelműen megfeleltetem a matematikai gráf  $(N)$  eszközének,  $v \in V$  csomópontjainak,  $e(v_1, v_2) \in E$  éleinek,  $C_e$  kapacitásának,  $\Omega_e$  kapacitásköltségének.

Visszavezettem a gráfokkal modellezett MKÉ-kre a gráfokon alkalmazható útkereső algoritmusokat és minimál költségű többtermékes folyam problémák alkalmazhatóságát.

Megmutattam, hogy a blokk-szerkezetű komplex egészségügyi ellátó rendszerek megfelelő védelmi stratégiák használatával kielégítik a megbízható komplex épületek (MÉK) fogalmát.

A 3. fejezetben összefoglaltam a szakirodalomban fellelhető védelmi stratégiákat, melyek csoportosítását ismertettem. Meghatároztam a kutatásom irányát: megosztott, rész-hálózatokon alapuló, részben független utakat használó, hibafüggő, illetve hibafüggetlen megoldások, melyek bármilyen egyszeres meghibásodás esetére túlélést garantálnak, de néhány többszörös meghibásodás esetére is működnek.

Vizsgálataimat a továbbiakban a védelmet is biztosító útkereső algoritmusok területén folytattam. Azt tapasztaltam, hogy amennyiben megosztott védelmet szeretnénk biztosítani, abban az esetben az algoritmusok vagy nem-lineáris költségfüggvényt eredményeznek, melyre azután nem alkalmazhatók az egzakt megoldást nyújtó egészértékű vagy kevert egészértékű lineáris megfogalmazások, vagy durva becsléssel lehet csupán a költségfüggvényt megadni, ami az elérhető eredmények romlásához vezet. Felismertem, hogy a szakirodalomban ismertett védelemről is gondoskodó útkereső algoritmusok problémája, hogy nem-lineáris költségfüggvényű problémához vezetnek.

Meskó Diána

Ennek feloldására megalkottam egy speciális, élkettőzött épület-modellt, melynek megadtam a matematikai formalizációját is. Bevezettem a  $\gamma_e'' \in \Gamma''$  arányos költség fogalmát, mellyel megadható, hogy az eredeti élköltséghez képest milyen arányban van a megosztható kapacitás költsége.

Az elért eredményeket az 2. tézis ismerteti.

## 2. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2012)

**Definiáltam a megbízható komplex egészségügyi ellátó rendszerek, MKÉ épület funkcióinak és funkció-kapcsolatainak modellezésére egy új, gráf alapú modellt (az 1.3. tézisben megfogalmazott gráf modell továbbfejlesztése):**

- 2.1. Az épületek modellezésére megalkottam egy új épület-modellt, melyen a nem-lineáris költségfüggvényű, védelmet is biztosító útvonalkereső algoritmusok lineáris programozási feladatként megfogalmazva megoldhatók. A modellben a nem-lineáris költségfüggvényű éleket megkettőzöm. Az új éleken a költségfüggvény konvex, lineáris szakaszokkal írható fel. A modellezési módszert él kettőzésnek neveztem.**
- 2.2. Bevezettem a  $\gamma_e'' \in \Gamma''$  arányos költség fogalmát, mely az élkettőzött él eredeti költségéből lineáris aránnyal számítható.**
- 2.3. Egészértékű lineáris programozási feladatként megfogalmaztam az élkettőzött épületmodellt.**

Az 4. fejezetben definiáltam két új útkereső algoritmust, melyek a megbízható épületek előtervezési fázisában, az épület modelljén legrövidebb útkereséssel adnak üzemi útvonalat és a meghibásodás által érintett páciens-forgalmat, mint minimál költségű többtermékes folyam problémát ábrázolva keresnek költséghatékony, takarékos, megosztott védelmi útvonalakat. A definiált algoritmusok az általam megalkotott, 2.2. tézisben formalizált modellen optimális megoldást közelítő eredményt adnak.

Az elért eredményeket az 3. tézis ismerteti.

### 3. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Mitcsenkov, Meskó és Cinkler 2007 - [Teljes dokumentum](#), [BME PA közlemény](#), Diána Meskó, Gábor Viola, Tibor Cinkler 2006 - [Google scholar hash](#), [WoS](#), [Google scholar](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)

Definiáltam két új útkereső algoritmust, melyek a megbízható épületek előtervezési fázisában, az épület modelljén legrövidebb útkereséssel adnak üzemi útvonalat és a meghibásodás által érintett páciens-forgalmat, mint minimál költségű többtermékes folyam problémát ábrázolva keresnek költséghatékony, takarékos, megosztott védelmi útvonalakat. A definiált algoritmusok az általam megalkotott, 2.2. tézisben formalizált modellen optimális megoldást közelítő eredményt adnak.

- 3.1. Terveztem két új, megosztott védelmet alkalmazó, a védelmi utakat adaptívan újrendező útvonalkereső algoritmust: SPP-LD és PDSP-LD.
- 3.2. Egészértékű lineáris feladatként formalizáltam a hagyományos hálózati- és költség-modellen alapuló megosztott védelmet alkalmazó referencia algoritmust, melyet a szakirodalom megosztható útvédelemként definiál (SPP).
- 3.3. Egészértékű lineáris programozási feladatként formalizáltam az általam megalkotott két új, megosztott védelmet alkalmazó, a védelmi utakat adaptívan újrendező útvonalkereső algoritmust: SPP-LD és PDSP-LD (Shared Path Protection with Link Doubling és Partially Disjoint Shared Path Protection with Link Doubling).

A 5. fejezetben megfogalmaztam, hogy dekompozíció alkalmazásával hogyan írható fel a teljes épületrendszer, az épületek többtartományú és többtartományú hierarchikus modellje. Megvizsgáltam, hogy a kapott modelleken alkalmazható-e az 1. tézisben felírt előtervezési fázis, valamint a 3. tézisben felírt SPP és az általam kidolgozott SPP-LD és PDSP-LD módszerek.

Az elért eredményeket az 4. tézis ismerteti.

### 4. Tézis

(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Mitcsenkov, Meskó és Cinkler 2007 - [Teljes dokumentum](#), [BME PA közlemény](#), Diána Meskó, Gábor Viola, Tibor Cinkler 2006 - [Google scholar hash](#), [WoS](#), [Google scholar](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)

Megfogalmaztam, hogy dekompozíció alkalmazásával hogyan írható fel a teljes épületrendszer, az épületek többtartományú és többtartományú hierarchikus modellje. Megvizsgáltam, hogy a kapott modelleken alkalmazható-e az 1. tézisben felírt előtervezési fázis, valamint hogy a 3. tézisben felírt SPP és az általam kidolgozott SPP-LD és PDSP-LD módszerek.

**4.1. Definiáltam az épületek többtartományú és többtartományú hierarchikus modelljét, mely az előtervezési fázis során a funkcionális modell többszöri dekompozíciójával jön létre.**

**4.2. Megmutattam, hogy az általam alkotott védelmi módszerek kiterjeszthetők többtartományú és többtartományú hierarchikus esetekre. Megmutattam, hogy a multi-blokkos épületekben történő meghibásodások, javítások esetére is alkalmazhatók a javasolt, takarékos védelmi módszerek.**

A 6. fejezetben bemutattam a dekompozíció működési elvét, majd példaként egy épületrészen végigvezettem az általam javasolt teljes előtervezési fázist. Végül szimulációs eljárást alkalmazva megvizsgáltam, hogy a korábbi téziseimben megfogalmazott modellen az általam kidolgozott algoritmusok milyen hatékonysággal teljesítenek.

Az elért eredményeket az 5. tézis ismerteti.

## **5. Tézis**

*(Meskó 2014, Meskó, Csébfalvi 2013 - [DOI](#), [Scopus](#), Tibor Cinkler, Diána Meskó, Attila Mitsenkov, Gábor Viola 2005 - [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Cinkler Tibor, Meskó Diána, Viola Gábor, Tapolcai János 2005 - [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Meskó, Csébfalvi 2014)*

**Több minta-rendszeren számítógépes szimulációs eljárással numerikusan, számításokkal bizonyítottam, hogy az általam javasolt tervezési módszertan alkalmazásával megbízható komplex épületeket lehet tervezni, valamint megmutattam az általam alkotott algoritmusok költség- és kapacitás-hatékonyságát.**

**5.1. Megmutattam a minta-rendszereken, hogy az általam definiált előtervezési-fázist alkalmazva, mellyel az építészeti tervezést megelőzően, az épület funkcionális modelljét megalkotva, egy hiba esetére is minimális**

**többletköltséggel üzemképes komplex egészségügyi ellátó rendszerek tervezhetőek.**

- 5.2. Szimulációs eljárást alkalmazva, numerikus eredményekkel alátámasztva megmutattam, hogy az SPP algoritmushoz képest az SPP-LD és a PDSP-LD javasolt algoritmusokkal alacsonyabb többletköltség és többletkapacitás mellett érhető el ugyanaz az épület-kihasználtság.**
- 5.3. Minta-rendszereken szimulációval igazoltam, hogy a PDSP-LD algoritmus kevesebb kapacitást foglal úgy, hogy az üzemi kapacitások mennyisége nagyobb arányú az egymáshoz képest hasonlóan teljesítő SPP és SPP-LD-hez képest.**

## KÖSZÖNETNYILVÁNÍTÁS

Mindenek előtt köszönetemet fejezem ki jelenlegi témavezetőmnek, *Csébfalvi Anikó Borbálának*, amiért rávilágított témám inter-diszciplináris voltára és a végsőig támogatott ezen disszertáció elkészülésében.

Szeretném megköszönni korábbi témavezetőmnek, *Cinkler Tibornak*, hogy megismertette velem a modellezési módszerek, útvonalkeresési algoritmusok matematikai megfogalmazásának szépségét, megmutatta a mérnöki kutatás mibenlétét és témavezetői útmutatást adott a kutatásom során.

Hálás vagyok mindazoknak, akik segítettek, buzdítottak és méltatlanul nélkülöztek engem a doktori fokozat megszerzésének időszakában – különösen *Gyermekeimnek*, *Családomnak* és közeli *Barátaimnak*.

A kutatás az Európai Unió és Magyarország támogatásával, az Európai Szociális Alap társfinanszírozásával a TÁMOP 4.2.4.A/2-11-1-2012-0001 azonosító számú „Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése konvergencia program” című kiemelt projekt keretei között valósult meg.

## TÉZISEKHEZ KAPCSOLÓDÓ PUBLIKÁCIÓK

Meskó Diána: Optimization Phase Using Graph Modelling for Reliable Building Complexes, *International Journal of Optimization in Civil Engineering* (ISSN: 2228-7558) (megjelenés alatt), 2014.

Meskó Diána: Planning pre-phase of architectural planning of reliable complex health-care systems: Előzetes, funkcionális tervezési fázis egy megbízható, komplex egészségügyi ellátórendszer építészeti tervezéséhez, *III. Interdiszciplináris Doktorandusz Konferencia 2014*, Pécs, Magyarország, 2014.04.15-2014.04.17. p. 204-207. Paper O21.13. 3 (2014) (ISBN: [978-963-642-597-5](https://doi.org/10.1007/978-963-642-597-5)) Befoglaló mű

Meskó Diána

link(ek): [Teljes dokumentum](#)

Könyvrészlet/Absztrakt/Tudományos

Diána Meskó, Anikó Csébfalvi: Modeling method in the architectural planning of reliable complex health-care systems, *Pollack Periodica: An International Journal for Engineering and Information Sciences* 8:(3) pp. 35-46. (2013)  
Link(ek): [DOI](#), [Scopus](#), [Teljes dokumentum](#)  
Folyóiratcikk/Szaccikk/Tudományos

Diána Meskó, Anikó Csébfalvi: Shared Protection Methods in the Architectural Planning of complex health-care, *Architectural, Engineering and Information Sciences - 9th International PhD & DLA Symposium: Abstracts Book*. Konferencia helye, ideje: Pécs, Magyarország, 2013.10.21-2013.10.22. Pécs: University of Pécs Pollack Mihály Faculty of Engineering, 2013. p. 106. ([ISBN ISBN 978-963-7298-54-7](#))  
Befoglaló mű link(ek): [OSZK](#)  
Könyvrészlet/Absztrakt/Tudományos

D. Meskó, A. B. Csébfalvi: Modelling the functional connections between the block-structures in the architectural planning of complex health-care systems, *Architectural, engineering and information sciences: 8th International PhD & DLA Symposium*, University of Pécs Pollack Mihály Faculty of Engineering and Information Technology, Pécs, Magyarország, 2012.10.29-2012.10.30. Pécs: PTE PMMK, 2012. p. 107. (ISBN: [978 963 7298 48 6](#))  
Könyvrészlet/Konferenciaközlemény/Tudományos

Attila Mitcsenkov, Diána Meskó, Tibor Cinkler: „Adaptive protection methods”, *Híradástechnika folyóirat Vol. LXII*, Budapest, 2007/7 szám, p. 2-9  
Link(ek): [Teljes dokumentum](#), [BME PA közlemény](#)  
Folyóiratcikk/Szaccikk/Tudományos

Diána Meskó, Gábor Viola, Tibor Cinkler: „A Hierarchical and a Non-Hierarchical European Multi-Domain Reference Network: Routing and Protection”, *IEEE (szerk.), Telecommunications Network Strategy and Planning Symposium, 2006.: NETWORKS 2006. 12th International*, New Delhi, India, Novembre 6 -9, 2006., IEEE, 2006. pp. 101-105. (ISBN: [3-8007-2999-7](#)), Link(ek): [Google scholar hash](#), [WoS](#), [Google scholar](#), [BME PA közlemény](#)  
Könyvrészlet/Konferenciaközlemény/Tudományos Független idéző: 3 Független idéző: 4 Összesen: 7

Tibor Cinkler, Diána Meskó, Attila Mitcsenkov, Gábor Viola: “Adaptive Shared Protection Rearrangement”, *Design of Reliable Communication Networks, DRCN 2005*, Nápoly, Olaszország, 2005.10.16-2005.10.19. Ischia: pp. 429-435., Link(ek): [DOI](#), [Google scholar hash](#), [WoS](#), [Scopus](#), [Teljes dokumentum](#), [Google scholar](#), [BME PA közlemény](#), Befoglaló mű link(ek): [Teljes dokumentum](#)  
Egyéb konferenciaközlemény/Konferenciaközlemény/Tudományos, Független idéző: 4 Független idéző: 3 Összesen: 7

Cinkler T, Mesko D, Viola G, Tapolcai J: Routing with partially disjoint shared path (PDSP) protection, *IEEE (szerk.), 1st Conference on Next Generation Internet Networks Traffic Engineering*, Róma, Olaszország, 2005.04.18-2005.04.20. New York: IEEE Press, 2005. pp. 47-52. (ISBN:[0-7803-8900-x](#))

Meskó Diána

Link(ek): [DOI](#), [WoS](#), [Scopus](#), [BME PA közlemény](#), Befoglaló mű  
link(ek): [BME PA közlemény](#),  
Könyvrészlet/Konferenciaközlemény/Tudományos

## EGYÉB PUBLIKÁCIÓK

- Mitscenkov Attila, Meskó Diána, Cinkler Tibor: „Forgalomhoz alkalmazkodó védelmi módszerek”, HIRADÁSTECHNIKA LXII:(7) pp. 2-9. (2007), Link(ek): [Teljes dokumentum](#), [BME PA közlemény](#)  
Folyóiratcikk/Szaccikk/Tudományos
- Meskó Diána: Adobe Acrobat 7, Budapest: PANEM, 2007. 416 p. Sorozat: (Egyszerűen), (ISBN:[9789635454662](#))  
Könyv/Kézikönyv/Tudományos
- A. Pataki, D. Meskó, G. Viola, T. Cinkler: Architectural Consideration for Multi-Domain Networks, Transcom 2007. Zilina, Szlovákia, 2007.06.25-2007.06.27.pp. 1-4.  
Link(ek): [BME PA közlemény](#),  
Egyéb konferenciaközlemény/Konferenciaközlemény/Tudományos
- Diána Meskó, Tibor Cinkler, Attila Mitscenkov, Gábor Viola, János Nagy: “Adaptive Shared Protection Methods”, WTC 2006: World Telecommunications Congress 2006, Budapest, Magyarország, 2006.04.30-2006.05.03.pp. 1-8., Link(ek): [Teljes dokumentum](#), [BME PA közlemény](#),  
Egyéb konferenciaközlemény/Konferenciaközlemény/Tudományos
- Meskó Diána, Mitscenkov Attila, Viola Gábor, Cinkler Tibor: Adaptive Shared Protection Methods, High Speed Networks Laboratory Workshop 2006 Spring. Budapest, Magyarország, 2006.05.,  
Egyéb konferenciaközlemény/Nem besorolt/Tudományos
- Meskó Diána, Mitscenkov Attila, Viola Gábor, Cinkler Tibor: Routing with partially disjoint shared protection, High Speed Networks Laboratory Workshop 2005 Spring. Budapest, Magyarország, 2005.05.,  
Egyéb konferenciaközlemény/Nem besorolt/Tudományos
- Meskó Diána, Viola Gábor, Cinkler Tibor: Special shared protection for MPLS, High Speed Networks Laboratory Workshop 2004 Spring, Budapest, Magyarország, 2004.04.,  
Egyéb konferenciaközlemény/Nem besorolt/Tudományos
- Meskó Diána: „A legtakarékosabb védelmi stratégiák nyomában”, Magyar Távközlés, Budapest, 2004. november, p. 26-30.,  
Folyóiratcikk/Szaccikk/Tudományos
- Meskó Diána, Viola Gábor: „Útvonalválasztás MPLS hálózatokban takarékos tartalékolással”, HIRADÁSTECHNIKA LVIII.: pp. 10-18. (2003),



Meskó Diána

Link(ek): [Teljes dokumentum](#)

Folyóiratcikk/Szaccikk/Tudományos

Diána Meskó, Tibor Cinkler, Gábor Viola, János Tapolcai: „Shared Protection with Partially Disjoint Paths”, EUNICE 2003: 9th Open European Summer School and IFIP Workshop on Next Generation Networks. 258 p., Balatonfüred, Magyarország, 2003.09.08-2003.09.10. Budapest: Budapest University of Technology and Economics, 2003. pp. 1-8., (ISBN:[963-421-576-9](#)),

Link(ek): [Egyéb URL](#), [BME PA közlemény](#)

Befoglaló mű link(ek): [Egyéb URL](#), [BME PA közlemény](#)

Könyvrészlet/Konferenciaközlemény/Tudományos

Független idéző: 1 Összesen: 1

Meskó Diána, Becker Johanna Cecília, Viola Gábor, Cinkler Tibor: Protection of the VoIP VPN, High Speed Networks Laboratory Workshop 2003 Autumn, Budapest, Magyarország, 2003.11

Egyéb konferenciaközlemény/Nem besorolt/Tudományos

Diána Meskó, Gábor Viola, János Tapolcai, Tibor Cinkler: „Routing with restoration in MPLS networks”, Materials, Technologies, Design, Maintenance and their Application in the Field of Transportation. Proceedings of the 20th International Colloquium Žilina, Slovakia, 2003. Zilina, Szlovákia, 2003, p. 138., Zilina, Szlovákia, 2003.06.23-2003.06.25.

Egyéb konferenciaközlemény/Konferenciaközlemény/Tudományos

Meskó Diána, Viola Gábor, Tapolcai János, Cinkler Tibor: Routing with restoration in MPLS networks, High Speed Networks Laboratory Workshop 2002 Autumn. Budapest, Magyarország, 2002.11

Egyéb konferenciaközlemény/Nem besorolt/Tudományos

Tudományos Diákköri Konferencia 2002 I. Helyezés, ugyanezen dolgozattal Országos Tudományos Diákköri Konferencia részvétel 2003 március: Meskó Diána, Viola Gábor: „Útvonalválasztás helyreállítással MPLS hálózatokban”, 2002. November

## IRODALOMJEGYZÉK

Baase, S. *Computer Algorithms: Introduction to Design and Analysis*. Addison-Wesley Publishing Company, ISBN: 0-201-00327-9, January 1983.

Balázs, Mihály. „Tervezőmódszertan tantárgy 2014 – előadások szövegvázlata.” dátum nélk. [http://www.kozep.bme.hu/wp-content/uploads/2014/01/01\\_TMT\\_jegyzet\\_2014.pdf](http://www.kozep.bme.hu/wp-content/uploads/2014/01/01_TMT_jegyzet_2014.pdf).

Black, U. *IP Routing Protocols*. Prentice Hall, 2000.

- Cinkler T., Laborczi P., Horváth Á. „Protection through thrifty configuration.” *Proceedings of the 16th International Teletraffic Congress*. Edinburgh, UK, 7-11 June 1999.
- G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi. „Complexity and Approximation.” *Springer-Verlang*, ISBN 3-540-65431-3. szám (1999).
- Guide to the design and construction of high performance hospitals*. Homepage of BetterBricks, powerful energy ideas by NEEA. 2010. [http://www.betterbricks.com/graphics/assets/documents/DC\\_Healthcare\\_Broch02\\_08\\_final.pdf](http://www.betterbricks.com/graphics/assets/documents/DC_Healthcare_Broch02_08_final.pdf).
- Ho P. H., Mouftah H. T. „A framework for service-guaranteed shared protection in WDM mesh networks.” *IEEE Communication Magazine*, Vol. 40, No. 2, 2002: pp. 97–103.
- Hu, T.C. „Multi-Commodity Network Flows.” In *Operations Research 11*, 344-360. 1963.
- ILOG CLPEX, User's Manual*. dátum nélk. <http://www.ilog.com/products/cplex/>.
- J. Guichard, I. Pepelnjak. *MPLS and VPN architectures*. Cisco Press, 2001.
- Jeremy M., Ruchi C., Papalambros P. „Architectural layout design optimization.” *Engineering Optimization* Vol. 34, No. 5. szám (2002): 461–484.
- Karmarkar, N. K. „A new polynomial-time algorithm for linear programming.” *Proceedings of the 16th Annual ACM Symposium on Theory of Computing*. 1984.
- M.S. Bazaraa, J.J. Jarvis, H.D. Sherali. *Linear Programming and Network Flows*. John Wiley & Sons, ISBN: 0-471-63681-9, 1990.
- Michael R. Garey, David S. Johnson. *Computers and Intractability*. W. H. Freeman and Co., 1979.
- Michalewicz, Zbigniew. *Genetic Algorithms + Data Structures = Evolution Programs* . 3rd edition, Springer-Verlag, 1995.
- Nayma, Khan. „Analyzing patient flow: Reviewing literature to understand the contribution of space syntax to improve operational efficiency in healthcare settings.” *Short paper presented at the 8th International Space Syntax Symposium*. Santiago de Chile, Chile, 3-6 January 2012.
- R. L. Graham, M. Grötschel, Lovász L. „Gráf - Handbook of combinatorics. MIT Press, 1995. nyomán.” 1995. <http://hu.wikipedia.org/wiki/Gráf>.
- R., Bhandari. „Survivable networks.” In *Algorithms for diverse routing*. Kluwer Academic Publishers, 1999.
- R.K. Ahuja, T.L. Magnanti, J.B. Orlin. *Network Flows: "Theory, Algorithms and Applications"*. Prentice Hall, ISBN 0-13-617549-x, 1993.
- Rózsa, Pál. *Lineáris algebra és alkalmazásai*. Tankönyvkiadó, 1991.
- S. Halabi, D. McPherson. *Internet Routing Architectures*. Second Edition, Cisco Press, 2000, ISBN: 1-57870-233-X.

Meskó Diána

Steenstrup, M. *Routing in Communications Networks*. Prentice Hall, ISBN: 0-13-010752-2, 1995.

Suurballe, J. W. „Disjoint paths in a network.” *Networks*. 1974.

T. Cinkler, I. Andersson, M. Zethzon. „Greedy Algorithms for Topological Design.” *NETWORKS'98, 8th International Telecommunication Network Planning Symposium*. Sorrento, Italy, October 1998.

Vazirani, Vijay V. *Approximation Algorithms*. Springer-Verlag, 2003.

Wolsey, Laurence A. *Integer Programming*. Wiley, 1998.