

Miskolci Egyetem



GÉPÉSZMÉRNÖKI- ÉS INFORMATIKAI KAR

SZÁMÍTÓGÉPPEL SEGÍTETT KONCEPCIONÁLIS TERVEZÉSI
MÓDSZER

Ph.D. értekezés

KÉSZÍTETTE:

Takács Ágnes
okleveles gépészmérnök

SÁLYI ISTVÁN GÉPÉSZETI TUDOMÁNYOK DOKTORI ISKOLA
GÉPEK ÉS SZERKEZETEK TERVEZÉSE TÉMATERÜLET
TERMÉKFEJLESZTÉS ÉS TERVEZÉS TÉMACSOPORT

DOKTORI ISKOLA VEZETŐ:

Dr. Tisza Miklós
egyetemi tanár

TÉMACSOPORT VEZETŐ:

Dr. Döbröczöni Ádám
egyetemi tanár

TÉMAVEZETŐ:

Dr. Kamondi László
egyetemi docens

Miskolc, 2009.

Takács Ágnes

SZÁMÍTÓGÉPPEL SEGÍTETT KONCEPCIONÁLIS TERVEZÉSI
MÓDSZER

Doktori (Ph.D.) értekezés

Miskolc, 2009.

TARTALOMJEGYZÉK

A TÉMAVEZETŐ AJÁNLÁSA	5
ALKALMAZOTT JELÖLÉSEK	7
1. BEVEZETÉS	10
1.1. A terméktervezés szerepe	11
1.2. Célkitűzések	12
2. A TERVEZÉSMÓDSZERTAN ELMÉLETI ALAPJAI	14
2.1. Irodalmi áttekintés	14
2.2. Tudományos tervezés–tervezéstudomány	17
2.2.1. Tudományos tervezés	17
2.2.2. Tervezéstudomány	18
2.2.3. A tervezés tudománya	19
2.3. Tervező iskolák, tervezésmódszertani eljárások	19
2.3.1. Az intuíciónál az előírásig: az intuitív, a kognitív és a diszkurzív tervezői modellek	20
2.3.2. Jelentősebb tervezői modellek	24
2.3.3. A módszertani modellek szintézise	33
3. FUNKCIÓK KÖZTI KAPCSOLATOK LEÍRÁSA	36
3.1. A koncepcióképzés folyamata	36
3.2. A feladat részfeladatokra való bontása	37
3.3. A funkcióstruktúrák matematikai leírása	38
3.3.1. Fabejárési módszerek	38
3.3.2. Teljes kombinálási módszerek	40
3.3.3. Fuzzy szemlélet alkalmazása termékcsaládok leírására	41
4. FUNKCIÓSTRUKTÚRÁK GENERÁLÁSÁNAK MÓDSZEREI	43
4.1. Módszertani kritikák	43
4.2. Számítógéppel segített módszeres koncepcióépítés (CACB)	44
4.2.1. Felhasználói igények	46
4.2.2. Megoldásváltozatok generálása	47
4.2.3. Értékelési szakasz	51
4.3. Fuzzy megoldáscsalád ábrázolása háromdimenziós térben	52
4.4. A javasolt módszerek számítógépi adaptálhatósága	54
5. KONCEPCIÓGENERÁLÓ SZOFTVER	55
5.1. A VisualConcept RIV0 Felépítése	55
5.1.1. A szoftver felépítése	56
5.1.2. A szoftver felhasználói felülete	58
5.1.3. Megoldás generálás	62
5.1.4. A program dokumentációs lehetőségei	63
6. ESETTANULMÁNY–HATÉKONYSÁGVIZSGÁLAT	64

6.1. Első esettanulmány: irodabútor koncepciók képzése.....	65
6.1.1. <i>Követelményjegyzék</i>	65
6.1.2. <i>Futási eredmények</i>	65
6.1.3. <i>A futási eredmények tapasztalatai</i>	68
6.2. Második esettanulmány: Fogaskerék bolygóművek előállítása.....	69
7. ÖSSZEFOGLALÁS	70
7.1. Az új tudományos eredmények összefoglalása	71
7.2. Továbbfejlesztési irányok, lehetőségek.....	72
7.3. Köszönetnyilvánítás.....	72
8. SUMMARY	74
TÁBLÁZATOK JEGYZÉKE	76
ÁBRAJEGYZÉK.....	77
FELHASZNÁLT IRODALOM	78
PUBLIKÁCIÓK AZ ÉRTEKEZÉS TÉMÁJÁBAN	84
9. MELLÉKLETEK	87
9.1. Funkciók közti kapcsolatok szerinti generálás	87
9.1.1. <i>Elvileg helyes megoldások</i>	87
9.1.2. <i>A követelményeknek megfelelő megoldások</i>	98
9.2. Funkciók szerinti generálás	104
9.2.1. <i>Elvileg helyes megoldások</i>	104
9.2.2. <i>A követelményeknek megfelelő megoldások</i>	108
9.3. Binárisan generált bolygómű változatok	110

A témavezető ajánlása

Takács Ágnes „Számítógéppel segített koncepcionális tervezési módszer” című Ph.D. értekezéséhez

A termékfejlesztés nagy kérdése, hogy az alkotási folyamat azon fázisaiba, ahol az emberi kreativitásra, a megoldásváltozatok nagyméretű halmazának előállítására és elemző kezelésére van szükség, hogyan lehet a számítástechnika adta lehetőségeket tudományos megalapozottsággal e tevékenységsorozatba beépíteni, eszközrendszerét fejleszteni és hatékonyan alkalmazni?

Az értekezésben Takács Ágnes nagy energiával, szorgalommal és körültekintéssel végezte kutatásait, vizsgálatait. Kutatásának fő irányát a számítógéppel segített koncepcióképzés módszerének fejlesztése, megoldásváltozatok halmazának generálása és értékelése határozta meg. A termékfejlesztés folyamatában a koncepcionális változatok előállítását számos módszer segíti. A koncepcióképzés folyamatának mennyiségi és minőségi ágon történő kezelésével megteremtette annak lehetőségét, hogy matematikai modellek alkalmazása segítse a megoldásváltozatok halmazának előállítását.

A termékek funkcionális részegységeik által definiáltak. A funkcionális részegységek kapcsolataik révén struktúrákba rendezve írják le megoldásváltozatokat. A számítógépi alkalmazáshoz módszert dolgozott ki a kötött- és rugalmas funkcionális részegység halmazból véletlenszerűen generálható megoldásváltozatokra.

A termékre vonatkozó megoldásváltozatok halmazát belső paraméterrendszerével a kezelhetőségéhez le kell írni. A rugalmas funkcionális részegység halmazon megvalósított megoldáshalmaz leírására bevezette az Ω fuzzy háromdimenziós megoldásteret, mely a bonyolultsági mátrix-szal és funkcionális részegység fontossági vektorral írható le. Az α fuzzy termék-bonyolultsági-, β fuzzy funkcionális részegység fontossági- és δ fuzzy funkcionális részegység előfordulási síkok segítségével biztosította a megoldáshalmaz kezelhetőségét. Az α - β síkok metszésvonalával a megoldáscsaládba funkcionális részegységek beépülését, az α - δ síkok metszésvonalával a megoldásokban előforduló funkcionális részegységek fontosságát, a β - δ síkokkal párhuzamos síkok a funkcionális részegységek előfordulási gyakoriságát határozta meg. A megoldáshalmaz Ω fuzzy háromdimenziós megoldástérrel történő kezelése jelentős új eredmény.

Kidolgozott egy a koncepcionális fázisban alkalmazható programrendszert. A Visual Concept program előre definiált funkcionális részegység halmazon képes megoldást leíró termékstruktúrák generálására. Elkészítette a programrendszer tesztelt prototípusát, melynek működőképességét esettanulmányként kezelt példákön mutatta be.

Az elvégzett kutatás és eredményeinek bemutatása rámutatott arra, hogy tudományos megalapozottsággal és számítógépes támogatással a megoldáshalmazok kombinatorikus határai ki-
tolhatók és elkerülhetők a kombinatorikus robbanással járó megoldhatatlannak tűnő feladatok.

Takács Ágnes az elért eredményeiről rendszeresen beszámolt a különböző hazai és nemzetközi fórumokon, eleget téve a Sályi István Gépészeti Tudományok Doktori Iskola publikációs követelményeinek.

Az értekezés gondos munkát tükröz, szövegezése jól érthető, ábrái és példái a mondanivalóját jól alátámasztják, tézisei a Ph.D. cím elnyeréséhez szükséges kívánalmakat messzemenően kielégítik.

Miskolc, 2009. szeptember 16.

Dr. Kamondi László
tanszékvezető egyetemi docens

Alkalmazott jelölések

A skalár mennyiségeket normál vastagságú latin, vagy görög betűk jelölik. A vektormennyiségeket aláhúzott, normál vastagságú latin betűk jelölik. A mátrixokat félkövér latin betűk jelölik.

Alkalmazott matematikai jelölések	
$\forall$	Bármely halmazelem
$\cdot$	Skaláris szorzás
$\circ$	Diadikus szorzás
max	Maximum függvény
Latin, nagybetűvel jelölt mennyiségek	
A	Alkalmasság értékelés
B	Bonyolultsági mátrix
C_o	Konfigurációs tevékenység
C	Kapcsolati mátrix
E	Értékelés
E	Értékmátrix
F	Funkcionális részegység halmaz
F	Funkcionális részegységek mátrixa
F_t	Termékfunkciók Fuzzy halmaza
F_t'	A termékfunkciók leképezett Fuzzy halmaza
G_{B,i}	Valamely bemeneti adat
G_{K,i}	Valamely kimeneti adat
H	Hatásváltozat
K	A felhasználói igények, valamint kritériumok mátrixa
K_a	Komponensek egymásközi kapcsolata
K_i	Kiválasztás
M	Megoldásváltozat

ME	Megoldás elem
M_f	Funkcióknak való megfelelés
M_r	Követelményeknek való megfelelés
R	Elrendezés
$\tilde{R}_1$	Az F_t és az U Fuzzy halmazok közti kapcsolat
Sz	Értékelési szempont
S	Struktúra mátrix
T	Társítás
V	Megoldásváltozat
V_h	Hatáslánc változatok száma
V_{kb}	Funkcionális részegységek közti kapcsolatok bináris variálásával létrehozható megoldásváltozatok száma
V_m	Megoldásváltozatok száma
V_{opt}	Optimális megoldás
V_{sz}	Hatáslánc változatok szintenkénti száma
U	A felhasználói igények Fuzzy halmaza
U'	A felhasználói igények Fuzzy halmazának leképezése

Latin, kisbetűvel jelölt mennyiségek

a	Elemek száma
b_{ij}	A bonyolultsági mátrix egy eleme
c_{ij}	A kapcsolati mátrix egy eleme
d	Részhatás
d_t	Részhatásonkénti hatásváltozatok
é	Élek száma
$\underline{e}_f$	Az értékelési szempontok fontossági vektora
e_{if}	Valamely értékelési szempont fontossága
e_{ij}	Az értékmátrix egy eleme
f	A funkcionális részegység halmaz elemei
$\underline{f}_f$	Funkcionális részegység fontossági vektor
f_{if}	Valamely funkcionális részegység fontossága
f_{ij}	A funkcionális részegységek mátrixának egy eleme
i	Ciklusváltozó
j	Ciklusváltozó
k	Sorozat tagszáma

k_f	Kritérium fontossági vektor
k_{if}	Valamely kritérium fontossága
k_{ij}	A kritériummátrix egy eleme
m	Ciklusváltozó
n	Ciklusváltozó
$p_{\min}$	A megoldásváltozatra adható legkisebb pontszám
$p_{\max}$	A megoldásváltozatra adható legnagyobb pontszám
q	Sorrend vektor
q_i	A sorrend vektor egy adott eleme
s	Elemi súlyozási tényező
s_{ij}	A struktúra mátrix egy eleme
$\underline{s}$	Súlyozási vektor
sz	A hatáslánc szintjének száma
t	Adott részhatás sorszáma
z	A mátrix meghatározásához szükséges elemek száma

Görögbetűvel jelölt mennyiségek

α	Fuzzy termékbonyolultsági sík
β	Fuzzy funkcionális részegység fontossági sík
δ	Fuzzy funkcionális részegység előfordulási sík
$\mu(f)$	Tagsági függvény
$\mu_{F_i}(f)$	A termékfunkciók Fuzzy halmazához tartozó tagsági függvény
Ω	Fuzzy megoldástér

A dolgozatban KISKAPITÁLIS betűvel szedett személynevek a hivatkozott kutatókat jelölik. Az irodalmi hivatkozások [i]-ben szerepelnek, a Felhasznált irodalom című fejezetben szerző szerinti ABC-rendben összefoglalt munkáknak megfelelően. A saját publikációkra történő utalások [TAi] módon szerepelnek a *Publikációk az értekezés témájában* című fejezetben a publikáció típusa szerinti időrendi sorrendbe szedett cikkek alapján. A nyomtatott szövegben zárójel, idézőjel, és gondolatjel nélkül dőltbetűvel szedve publikációk címei, konferenciák elnevezései, szakmai-tudományos szervezetek nevei szerepelnek.

1. BEVEZETÉS

Az emberiség történetének kezdete óta végez tervező tevékenységet. Az ősember nem indult el úgy vadászni, hogy társaival ne beszélte volna meg előre az esetleges helyzeteket. A hadvezér addig nem indított háborút, amíg hadmérnökeivel nem vitatta meg az ellenség várható taktikáját, és az alkalmazható haditervet. Nem véletlen, hogy az ókor legkiválóbb tervezőmérnökei a hadmérnökök voltak. VITRUVIUS –*akiről a legújabb kutatások azt mutatják, hogy ő lehetett egykor Julius Caesar haditanácsadója (Kr. e. 1. század második fele)*– nem csupán a haditaktika és a korabeli hadieszközök tervezésében (*hajítófegyverek*) volt kiemelkedő, de zseniális képességei 10 kötetből álló építészeti leírásából is kitűnnek [110].

Az első műszaki dokumentáció azonban sokkal korábról maradt fenn, ugyanis Kr. e. 1100 körül 1:28 arányban papiruszon megtervezték és dokumentálták IV. Ramszesz fáraó sziklasíráját. A műszaki ábrázolás eredetének története nagyon töredékes, azonban a tervek dokumentálása iránti igény mindenképpen a Kr. előtti évszázadokra tehető. Az egyiptomiak, annak ellenére, hogy az elvont matematika területén sokkal kevesebb eredményt értek el, mint a gyakorlati számítások terén, mégis számos empirikus geometriai eredménynek voltak birtokában. Thalesz a milétoszi görög matematikus, aki kb. Kr. e. 600-ban élt, írásaiban összefoglalta az ókori egyiptomiak által ismert geometriai eredményeket és fogalmakat, beleértve Pitagorasz tételét is. Időszámításunk előtt 400 évvel megalakult az Athéni Akadémia, mely a filozófia és az irodalom aranykorának kezdetét jelentette, így a geometria tudománya háttérbe szorult. Kb. 100 évvel később, Arisztotelész tanítványa, a görög Endemosz Thalész írásai alapján foglalta össze a tudomány új eredményeit. Endemosz feljegyzéseire már csak a római történész, Prokulusz 5. században keletkezett munkái emlékeztetnek [94]. Nagyjából Endemusszal egy időben, Kr. e. 300 körül alakult meg –*Egyiptom akkori fővárosában*– Alexandriában a híres Múzeum, ami a görög tudományok egyeteme lett. Az iskola olyan matematikusoknak adott otthont, mint Euklidész, aki 'Az elemek' című munkájával az elemi geometria alapjait teremtette meg. Apollóniusz, az alexandriai iskola utolsó nagy, geometriával foglalkozó tudósa Kr. e. 250 körül a kúp metszeteinek vizsgálatával foglalkozott.

Az ókori görög geometriai eredmények a reneszánsz korszak, vagyis a modern geometria kezdetéig –*körülbelül 1000 évig*– meghatározták a rajzi ábrázolásmódot. A Római Birodalom bukása és a 13. század közt eltelt hétszáz évből ismereteink szerint csupán egyetlen jelentős, a 9. századból származó tervrajz maradt fenn, mely Sankt Gallen tervrajza néven ismert, és még ma is a svájci, Sankt Gallen-i kolostor könyvtárában őrzik. A tervrajzból azonban sosem épült kolostor, nevét a megőrzés helyéről kapta.

A történelmi korok műszaki alkotásainak létrehozói többnyire nem csupán műszaki érzékük révén váltak sikeressé, a későbbi korok által elismerté. VITRUVIUS a hadviselésen túl, az építészethez is értett, Leonardo da Vinci a műszaki megoldások nagymestere, számos festményt és freskót is festett [86]. Leonardo VITRUVIUS leírásai alapján készítette el híres Vitruvius tanulmányát, ami tulajdonképpen a manapság a terméktervezés területén nagy jelentőséggel bíró antropometria tudományának fontos alapját képezi. Leonardo tervei számos esetben nem

kerültek kidolgozásra, azonban mégis rendkívül fontosak, hiszen munkái tulajdonképpen az első műszaki dokumentációk voltak.

Watt –*akinek a gőzgép feltalálását tulajdonítjuk*–, valamint a 19. századi kortárs feltalálók és gépészmérnökök gépmodelleket készítettek, mely modelleket addig javítottak, tökéletesítettek, amíg végül úgy működött, ahogyan azt tervezőjük elgondolta [86]. Hasonló módon terveztek a mérnökök a 19-20. század fordulóján is, hiszen a kísérleti modellek még mindig a feltalálók műhelyeiben készültek, azonban a mérnöki mesterséget már tapasztalt ipari szakemberektől tanulták. HORCH a zwickau-i autógyár alapítója, például BENZ-től tanulta a gépészmérnöki szakma fortélyait. Magyarországon a mai Budapesti Műszaki és Gazdaságtudományi Egyetem jogelődje 1782-től működött. Akkoriban még csupán földmérő- és vízépítő mérnököket képeztek, azonban 1877-ben az iskola kiadta az első gépészmérnöki oklevelet.

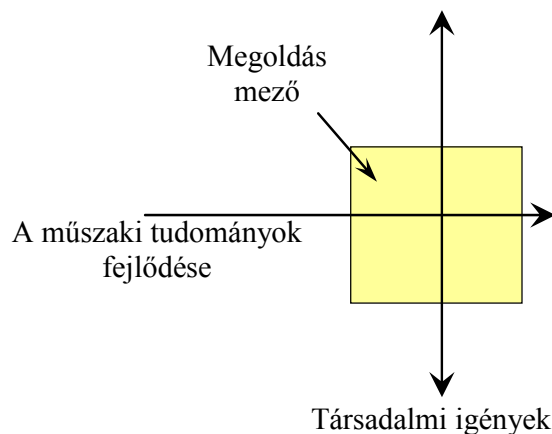
Az a szilárd nézet, hogy konstruktőr-mérnöknek születni kell, az 1920-as évek körül kezdett megdőlni. Ekkor merült fel ugyanis annak a lehetősége, hogy a tervezőmérnök a konstrukciós munka során egy lépésről lépésre haladó eljárást kövesen végig a feladatot kielégítő megoldás eléréshez. Ezt a fajta munkamódszert természetesen folyamatos ellenőrzések, mérlegelések, döntési folyamatok irányítják. Az ötlet azonban csak az 1970-es évekre forrott ki. 1977-ben a német mérnökök egyesülése, azaz a VDI (*Verein Deutscher Ingenieure*) –*mely azzal a meghatározott céllal alakult 1856-ban, hogy a „technika minden szellemi erejét egybe gyúrva” összefoglalja*– kiadta azt az irányelv gyűjteményt, mely a koncepcionális tervezésre vonatkozó összes, addig kialakított javaslatot magába foglalja [112].

Egy termék megszületése rendszerint egy meglehetősen hosszú folyamatnak, a termékfejtésnek az eredménye, amely számos lépésből áll. Ezek közül a legizgalmasabb, és legnehezebb szakasz a koncepcionális tervezés. A koncepcionális tervezés a tervezési folyamatnak az a lépcsője, amely során a mérnök a piac által támasztott követelmények és igények, valamint saját ismeretei alapján elvonatkoztatással meghatározza a termék lényeges funkcióit, a funkcióstruktúra felállításával, a megfelelő megoldási elvek keresésével és kombinálásával azt a termékötletet –*vagyis a koncepciót*–, amely a későbbi konstrukciós tervezés alapját képezi. A koncepcionális tervezés során a tervezőmérnök számos elvi megoldást vizsgálhat meg annak érdekében, hogy a végül kifejlesztésre kerülő termék tulajdonságai (*olcsó, gazdaságos, stb.*) a legjobb konstrukciót eredményezzék, és minden tekintetben megfeleljenek a piac elvárásainak. Azonban a tervező mérnök sem lehet biztos abban, hogy megtalálja-e az optimális megoldást (1. ábra), ha csupán az intuícijára, kreativitására, egyéni tapasztalataira hagyatkozik. Az intuíció általában csupán egy megoldást szolgáltat, melyet később csiszolni, tökéletesíteni lehet. Ezzel szemben a módszeres tervezés egyik legnagyobb előnye, hogy adott peremfeltételek mellett lehetőséget ad valamennyi elképzelhető megoldás változat előállítására, így téve lehetővé a kritériumokat legjobban kielégítő megoldás kiválasztását. A tervezés módszertanának, mint önálló tudományterületnek a kutatását a XX. század felgyorsult technikai fejlődése kényszerítette ki, polarizálva a jelentős tervező iskolákat.

1.1. A TERMÉKTERVEZÉS SZEREPE

A tervezőmérnök küldetése, hogy megtalálja egy-egy adott műszaki probléma optimális megoldását azon lehetőségek határain belül, melyet a műszaki tudományok fejlettségi szintje és a társadalom pillanatnyi igényei és lehetőségei határoznak meg. Az 1. ábra szerinti megoldásmező azt a megoldásteret szimbolizálja, amelyben a terméktervező mérnök a megoldást keresheti. Míg a mérnöki tudomány monoton fejlődő élő tudomány, a társadalmi igények földrajzi helytől és politikai hatásoktól is függenek, így nem tekinthetők monoton fejlődőnek, sok-

kal inkább folyamatosan változónak. A II. Világháború előtti Németország műszaki egyetemein együtt diplomázó –*emiatt hasonló műszaki tudású*– fiatal mérnökök –*az ország kettészakadása után*– a mérnöki tudás és a társadalmi igények metszéspontjában a nyugati ország-részben a Mercedest, a keleti országrészben a Trabantot hozták létre.



1. ábra. Tudomány-igény kereszt

A tervező feladata tehát, hogy a tudomány aktuális állását a mindenkori felhasználói igényekhez igazítva fejlesszen ki újabb és újabb termékeket. A műszaki tudományok fejlettségének, valamint a társadalmi igényeknek a metszéspontja a tudományos ismeretek folyamatos bővülése, alakulása révén mindig máshol van; ami tulajdonképpen nem jelent mást, mint hogy a változó igények révén a tervezőmérnöknek mindig terveznie kell, a mérnöki munka örök.

1.2. CÉLKITŰZÉSEK

Az egyre gyorsuló technikai fejlődés tette szükségessé a tervezési folyamat nagyobb egységeinek meghatározását, a koncepcióképzés különféle módszereinek kialakulását, egy új tervezői szemlélet megszületését, mely szerint egy adott problémára nem csak egy megoldás létezik, hanem meglehetősen sok. Így a tervezőmérnök legfőbb feladatának nem azt kell tekinteni, hogy a tervezési feladat egy lehetséges megoldását kidolgozza, hanem hogy az elvileg végtelen nagy megoldáshalmazban megtalálja az adott igényeknek legjobban megfelelő változatot. A disszertáció a számítógéppel segített koncepcióképzés módszertani kérdéseivel foglalkozik. Legfőbb irányvonalnak az a felismerés tekinthető, hogy a tervezésmódszertani eljárások elvi lehetőséget nyújtanak arra, hogy a tervező a tervezési feladatnak ne csak egy elvi megoldását vizsgálja meg, hanem akár valamennyit, mielőtt valamilyen szempont szerint kiválaszt egyet, amelyen a konstrukciós számításokat, a méret- és alakadásokat is elvégezve, a feladatnak egy kiváló megoldásával áll elő.

A tervezésmódszertani eljárások tanulmányozása a hagyományos eljárások irodalmára fókuszált, főleg számítógépes alkalmazhatóságuk került vizsgálat alá, hiszen a számítógépek nyújtotta lehetőségek hozták életre azt az igényt, hogy nagyszámú koncepcióváltozat előállításával a tervezői munka határfoka javítható legyen, illetve a terv várható minősége és a tervezési idő jobban becsülhető legyen. Az összes előállítható megoldásváltozat feltárása azért fontos, mert így nagyobb valószínűséggel található meg az adott feladatnak az igényeket legjobban kielégítő megoldása.

A disszertáció legfontosabb célja tehát az, hogy a klasszikus módszeres tervezési módszerek előnyeinek és hiányosságainak feltárásával, az előnyök alkalmazása, a hátrányok kiküszöbölése révén egy számítógépre adaptálható, új tervezési módszerre tegyen javaslatot a terméktervezés területén, az elméleti háttér kidolgozásával és egy erre épülő programcsomag prototípusának létrehozásával.

Ezen belül a dolgozat célja volt, hogy olyan matematikai modellekre tegyen javaslatot, melyek könnyen algoritmizálhatók, az általuk generált eredmények jól összehasonlíthatók. A bináris logika, valamint a véletlenszám generálás alkalmas volt erre.

Az irodalomkutatás során előtérbe került továbbá a fuzzy logika, mely segítségével egy adott halmaz és bizonyos elemek nem csupán úgy foghatók fel, hogy a halmaz elemei, vagy nem elemei, de úgy is értékelhetők, hogy bizonyos feltétel mellett halmazelemek, bizonyos feltétel mellett nem elemei a halmaznak. Ez további érdekes lehetőségeket vetett fel a megoldásváltozatok képzése során, mint például bonyolult megoldások, és kevésbé bonyolultak létrehozása. A számítógépes szoftver elkészítése során cél volt, hogy esettanulmányok által a program tesztelése megtörténjen. A két teljesen eltérő tanulmány lehetővé tette, hogy a szoftver termékfejlesztés és rekonstruálás közben is kipróbálásra kerüljön.

2. A TERVEZÉSMÓDSZERTAN ELMÉLETI ALAPJAI

2.1. IRODALMI ÁTTEKINTÉS

A koncepcióképzés módszeres megközelítésével kapcsolatban az elmúlt száz évben számos tanulmány foglalkozott. A szakterület kutatói rendkívül nagyszámú módszert mutattak be. Ezek mindegyike egy lépésről lépésre történő tervezési útmutató, hiszen a módszeres koncepcióképzés alap gondolata az, hogy a tervezés megtanulható tevékenységgé váljon.

A módszeres géptervezés gyökerei az 1920-as évek Németországból erednek. WÖGERBAUER [117] 1943-ban megjelent könyvében tett javaslatot arra, hogy a teljes feladatot részfeladatokra kell bontani, azokat pedig üzemi, és gyártási feladatokra. A „*Konstruktionssystematik*” kifejezést BINIEK [16] használta először 1952-ben. Az első konferencia, mely a *Conference on Design Methods (Tervezési módszerek konferencia)* elnevezést kapta, 1962 szeptemberében került megrendezésre Londonban [20].

KESSELRING [69] 1942-ben megjelent könyvében mutatta be konvergens közelítő eljárásának alapjait, amit később a VDI 2225 [115] foglalt össze. KESSELRING már 1937-től publikált az értékelő eljárásokról, mely publikációk alapjául a berlini Siemens gyárnál szerzett gyakorlati tapasztalatai szolgáltak.

Az ilmenai iskola megalapozói BISCHOFF és HANSEN voltak. HANSEN [55] az átfogó tervezési rendszerére vonatkozó elvi szempontokat 1965-ben megjelent könyvében foglalta össze, jóllehet már az '50-es évek eleje óta foglalkozott a tervezésmódszertan alapjaival.

RODENACKER [92] szerint minden olyan gép, vagy készülék, mely egy meghatározott cél, vagy funkció ellátására készül egy fizikai eseményen alapul. A tervezési folyamatot olyan információátalakításként fogja fel, mely során a megoldás kidolgozása az absztrakttól a konkrét felé halad.

A berlini iskola megalapítója– BEITZ volt. Munkássága szorosan összekapcsolódik a darms-tadti tervezőiskola megteremtőjével, PAHLlal, hiszen számos közös publikációjuk jelent meg [85].

A közép-európai gyökerek átnyúlnak Amerikába is, hiszen a cseh származású HUBKA számos alkalommal publikált közösen az osztrák származású, de Canadában élő kollégájával EDERrel [64]. HUBKA [62] a Műszaki Rendszerek Elméletének (*Teorie technických systémů*) megalkotója, a rendszerszemléletű tervezésmódszertan megalapozója, az ICED (*International Conference on Engineering Design*) elnevezésű konferencia egyik életre hívója.

A tervezési katalógusokról is ismert ROTH [93] elsők között volt, akik felismerték a grafikus számítógépekben rejlő, a módszeres tervezés területén sikerrel alkalmazható automatizálási lehetőségeket, így módszerének elvi alapjai nem csupán a táblázatosan összegyűjtött anyagok tudásbázisként való alkalmazására fókuszáltak, de az a gondolat is foglalkoztatta, hogy a teljes koncepcióképzési fázist automatizálhatóvá tegye. Ennek érdekében dolgozta ki algoritmi-kus tervezői modelljét.

1. táblázat. A tervezésmódszertan mérföldkövei

MEGJELENÉS ÉVE	SZERZŐ	MÉRFÖLDKÖVEK	ORSZÁG
1852	Redtenbacher [91]	Prinzipien der Mechanik und des Maschinenbau	Németország
1854	Reuleaux [90]	Konstruktionslehre für den Maschinenbau	Németország
1881	Bach [7]	Die Maschinenelemente	Németország
1928	Erkens [43]	Beiträge zu Konstruktionserziehung	Németország
1942	Kesselring [69]	Die starke Konstruktion	Németország
1943	Wögerbauer [117]	Die Technik des Konstruierens	Németország
1948	Zwicky [122]	The morphological method of analysis and construction	USA
1950	Nieman [83]	Maschinenelemente	Németország
1952	Biniek [16]	Konstruktionssystematik. Feingerätetechnik	Németország
1954	Kesselring [71]	Technische Kompositionslehre	Németország
1956	Altschuller [3]	Theory of inventive problem solving	Oroszország
1957	Matousek [79]	Konstruktionslehre des allgemeinen Maschinenbaus	Németország
~1960	Cross [21]	Design Ability	Anglia
1965	Hansen [55]	Konstruktionssystematik	Németország
1963-1971	Leyer [76]	Maschinenkonstruktionslehre	Németország
1970	Rodenacker [92]	Methodisches Konstruieren	Németország
1973	Hubka [62]	Design Science, rendszerszemléletű tervezés	Európa
1973	Koller [72]	Eine algorithmische-physikalische orientierte Konstruktionsmethodik	Németország
1973	VDI [111]	VDI 2221	Németország
1973	Tajnaírói [103]	Hajtóműelmélet (hajtómű változatok módszeres feltárása)	Magyarország
1974	Roth [93]	Aufbau und Handhabung von Konstruktionskatalogen	Németország
1976	Tajnaírói [104]	Szerszámgép morfológia	Magyarország
1977	Pahl és Beitz [85]	Konstruktionslehre	Németország
1978	Suh [100]	Axiomatic design	USA
1980	Yoshikawa [118]	General design theory	Japán
1985	Pugh [89]	Total design	Anglia
~1985 óta	Eder [40]	Transformation system	Canada
1989	Cross [21]	Engineering design methods	Anglia
1989	Mittal és Frayman [81]	Configurative design	USA
1992	Bercsey és Vajna [12]	Autogenetische Konstruktionstheorie	Magyarország, Németország
1993	Linde és Hill [77]	WOIS	Németország
1999	Furka [46]	Kombinatorikus kémia	Magyarország
2001	Otto [84]	Reverse engineering techniques in product design	USA

A közép-európai módszerekkel –*logikai felépítésének tekintetében*– szemben áll ALTSHULLER tervezői szemlélete, a Feltalálói Problémamegoldás Módszere, a Теория Решения Изобретательских Задач (*Theorija Reshenija Izobretatel'skih Zadach*), vagyis a TRIZ módszer [3]. A szabadalmi hivatalnok a benyújtott szabadalmakat tanulmányozva jutott arra a gondolatra, hogy vajon nem tartalmazznak-e a találmányi leírások valamiféle rejtett törvényszerűségeket. Számos szabadalmat vizsgált meg, végül az 1970-es évekre összeállította javasolt módszerét.

A TRIZ módszerhez nagyon hasonló szemléletű eljárást dolgozott ki LINDE és HILL [77], az Ellentmondásorientált Innovációsstratégiát (*Widerspruchsorientierte Innovationsstrategie*), vagyis a WOIS módszert, melynek alapjait 1988-ban, LINDE fogalmazta meg disszertációjában. A WOIS a TRIZ alapkoncepcióit használja, de megpróbál olyan módszereket, mint a QFD, brainstorming, szinektika, rendszerelmélet, értékelemzés konzisztens módon integrálni. LINDE a WOIS-ban, a TRIZ-zel ellentétben jövőbeni fejlesztési módszerek felfedezésére helyezi a hangsúlyt. Számára a hajtóerőt műszaki innovációk kifejlesztése jelenti. Sokkal inkább értékeli az innovatív feladatállítások generálását, mint a megoldásukhoz szükséges dolgokat. Magyarországi vonatkozásban említést kell tenni a budapesti tervező iskoláról, mely a terméktervezés módszertanának és korszerű eszközrendszerének fejlesztésével, kutatásával foglalkozik. Ennek a kutatási területnek a hazai megalapozója BERCSEY [10], [11], [12], [13], hiszen számos idegen nyelvű irodalom magyarra fordításában működött közre, emellett számos jegyzet [14], [15] összeállítását irányította, melyek a tervezésmódszertan tudományának hazai oktatását segítik. Mindemellett új módszerek kidolgozásával is foglalkozott, mint például az Autogenetikus Algoritmus, amelyet VAJNÁVAL közösen végzett [12], [109]. HORVÁTH [9] a terméktervezéshez szükséges számítógépes rendszerek kutatásával-fejlesztésével foglalkozik. A miskolci tervezőiskolát TERPLÁN és TAJNAFŐI alapozta meg. A géptervezés általános elveinek terméktervezésre történő alkalmazását DÖBRÖCZÖNI [14], [15] és KAMONDI [14], [15], [68] vezette be az oktatásba. TAJNAFŐI [102], [103], [104] a szerszámgépekkel kapcsolatosan dolgozott ki több jelentős módszert. Ezen módszerek felhasználásával például LIPÓTH [78] és TAKÁCS [105] számítógépes struktúrageneráló módszereket dolgozott ki.

Az irodalomkutatás során feldolgozott publikációk közül számos tanulmány született a világ más területein (*Észak-Európa, Japán, Amerika*). Ezek azonban túlnyomó többségben más tudományterületeken (*pl.: számítástechnika,*) kutadják a tervezésmódszertan eszközeit.

A konfiguratív feladat fogalma a számítástechnika területéről származik, MITTAL és FRAYMAN 1981-ben tett javaslatot a fogalom bevezetésére [81]. TIIHONEN és társai [106], [107] a szoftver termékcsaládok konfiguratív tervezésére dolgoztak ki módszert. BROWN 1998-ban úgy vélte [18], hogy a konfiguratív tervezés a terméktervezésre közvetlenül nem alkalmazható, mert maga a folyamat számos lépését tekintve pontatlan. DECIU és társai [35] azonban a ZADEH [119] által kidolgozott fuzzy halmazelmélet segítségével alkalmazták a konfiguratív tervezést az ipari terméktervezés területén, így a BROWN által említett, a funkciók kapcsolódására vonatkozó pontatlanságot kiküszöbölték.

A kombinatorikus kémia fogalma FURKÁTÓL [46], [47], [48] származik. Tapasztalatai során megállapította, hogy egy új gyógyszer kifejlesztéséhez rendkívül nagyszámú vegyületet kell előállítani. Kutatásai során különböző peptid- és fehérje láncok kapcsolatait vizsgálta. Ezek alapján vetette fel azt a lehetőséget, hogy a kombinatorika segítségével viszonylag gyorsan szinte végtelen számú vegyület generálható.

Az 1. táblázat a tervezésmódszertan legjelentősebb eredményeit foglalja össze. A táblázatból kitűnik, hogy nagyon sok német mérnök, módszertan kutató foglalkozott a módszeres tervezés fejlesztésével, így az az igény, hogy a tervezés tanulható, tanítható képességgé váljon, Európából indult és terjedt el az egész világon.

2.2. TUDOMÁNYOS TERVEZÉS–TERVEZÉSTUDOMÁNY

A tervezésmódszertan kutatói mindig arra törekedtek, hogy megkülönböztessék a tervezést és a tudományt. ALEXANDER 1964-ben úgy fogalmazott, hogy a tudósok feladata az, hogy meghatározzák a létező szerkezetek elemeit, míg a tervezők feladata az, hogy megformálják az új szerkezetek elemeit [1].

Ezzel szemben GLYNN [51] 1985-ben rámutatott arra, hogy a tudósok, a tervezőkhöz hasonlóan megalkotják azon feltevéseiket és elméleteiket, amelyek a tények utáni kutatásaikat irányítják. Szintén ebben az évben LEVY azt a kijelentést tette, hogy a tervezést és a technológiát nem volna célszerű a tudománytól és a társadalomtól elszigetelni [75].

Habár még mindig tehető olyan kritikai megkülönböztetés, miszerint a módszer létfontosságú lehet a tudomány számára –*ahol a módszer igazolja az eredményeket*–, de nem a tervezés számára –*ahol az eredményeket nem kell megismételni* [20]. Az is nyilvánvaló, hogy a gyakorlati szakemberek –*akár a tudomány, akár a tervezés terén*– nem kell, hogy a módszertan területén is jártasak legyenek, csupán a módszerek kompetencia szintű alkalmazási készségével kell rendelkezniük. Ahogy BARTLETT 1958-ban fogalmazott, „*A kísérletezőknek speciális módszereket precízen kell tudni alkalmazni, de legkevésbé sem szükséges a módszertant összességében ismerni. Hiszen a kiváló módszertani szakemberek sem sikeres kísérletezők*” [20]. Ha a kísérletezőt a tervezővel helyettesítjük, a fenti megállapítás a tervezésre is igaz.

A *Design Research Society* 1980-ban megrendezett *Design: Science: Method* elnevezésű konferenciája lehetőséget adott arra, hogy ezek a megállapítások megvitatásra kerüljenek. ARCHER [6] úgy fogalmazott cikkében, hogy „*a tervezés a tudományhoz hasonlóan egy világszemléletű módszer és annak rendszerbe való foglalása*”. Mind a tudomány, mind a tervezés lényegében a megismerésen alapul [25], [33].

Manapság sokkal megalapozottabb nézetek léteznek mind a tudomány, mind a tervezés kapcsán, mint a 1960-as években. Ahogy LEVY (1985) írta, „*a tudomány többé már nem egyetlen speciális világnézet alapján kialakított, egyedüli fix módszertan értelmében fogható fel. Sokkal inkább a problémaazonosító, problémaszervező és problémamegoldó tevékenységek kiterjesztett ésszerűsítéseként*” [75].

Azonban néhány területen a régi felfogás rendkívül lassan évül el. A *Research in Engineering Design* 1. évfolyamának 1. számában megjelent vezércikk kihangsúlyozta, hogy a folyóirat célja bizonyítani, hogy a tervezés nem művészet, hanem tudomány.

Az elmúlt évszázadban a tervezésmódszertan kutatóinak különféle nézőpontjai különféle felfogásokat, fogalmakat eredményeztek. Az így kialakult polémiában célszerű tisztázni a tudományos tervezés, a tervezéstudomány, illetve a tervezés tudománya fogalmakat.

2.2.1. Tudományos tervezés

A múltszázad első felében a tervezés számos területén –*mint például az anyagtudomány, a gépészeti tudomány, az építészeti tudomány területén*– figyelhető meg a tudományos alátámasztás rohamos fejlődése. A tervezés és a tudomány kapcsolatának definiálására egy viszonylag egyszerű mód WILLEM [116] szerint az, hogy a tervezés tudományos ismeretekbe vetett bizalma által, valamint a tudományos ismeret gyakorlati feladatok során történő alkalmazása révén a tervezés a tudományt láthatóvá, kézzel foghatóvá teszi. Tehát a tudományos tervezés a tudományos ismereteken alapuló, az intuitív és nem intuitív módszereket is egyesítő tervezéssel azonosítható.

2.2.2. Tervezéstudomány

A tervezéstudomány kifejezést elsőként valószínűleg GREGORY [53] alkalmazta a *Conference on The Design Method* elnevezésű, 1965-ben megtartott konferencián [29]. A tervezéstudomány fejlesztését mások is célul tűzték ki; például az ICED konferenciák, valamint a *Workshop Design Konstruktion (WDK)*, valamint a később létrejött *International Society for Design Science*. Az a törekvés, hogy a tervezéstudomány kifejlődjön a tervezési folyamat formalizálásának kísérletéhez vezetett –*egy egyszerű, logikus folyamathoz*–, amely formális nyelveken, és elméleteken alapul.

A tervezés tudományosítása iránti vágyakozás a modern tervezési mozgalomban visszavezethető az ötletekig. A konstruktőr DOESBURG, az 1920-as években a következőket írta [20]: „*Egy új objektum tervezéséhez új módszerre van szükségünk, mely egy úgynevezett tárgyalagos rendszer*”. Nem sokkal később, 1929-ben a neves építész, LE CORBUSIER a következőket mondta [20]: „*Egy ház felépítése meghatározott funkciók szabályos sorrendjéből áll. Ezen funkciók szabályos sorrendje irányított jelenség. Ezen irányítás pontossá, gazdaságossá és gyorsá tétele a modern építészeti tudomány legfőbb feladata*”. Érdemes megfontolni tehát azt a tényt, hogy míg a tervezés irányított jelenséggé válása iránti igény az építészet területén már 1929-ben, az ipari tervezés területén csak az 1960-as évek elején fogalmazódott meg. Elképzelhető tehát, hogy az építészeti irányított tervezés területén megjelenő új irányzatok ipari tervezés területén történő alkalmazásának elemzése, és folyamatos bevezetése gyorsabbá tenné más műszaki alkotások fejlődésének folyamatát is.

HANSEN [55] úgy fogalmazta meg a tervezéstudomány célját, hogy segítségével „*a tervezés és az ahhoz kapcsolódó tevékenységek törvényszerűségei felismerhetők, valamint szabályok fejlesztethetők ki*”. Ez azt jelentené, mintha a tervezéstudomány nem volna több mint „*szisztematikus tervezés*”, vagyis a tervezési tevékenységek szisztematikus lépések szerinti rendezése. HUBKA és EDER [64] Hansen megállapítását a tervezéstudomány egy sokkal szűkebb értelmezésének tekintették, mint saját felfogásukat, mely szerint „*A tervezéstudomány a tervezés területén fellelhető, egymással logikai kapcsolatban álló tudáshalmaz gyűjteményéből (rendszeréből) áll és tartalmazza a műszaki információ és tervezésmódszertan fogalmait... A tervezéstudomány a tervezendő rendszerek szabályos jelenségeinek meghatározását és rendszerezését, valamint a tervezési folyamat problémáját jelenti. A tervezéstudomány a természettudományok alkalmazott ismeretének olyan információvá történő alakításával is foglalkozik, mely a tervezőmérnök feladatának kidolgozásához megfelelő*”.

Ez a meghatározás átnyúlik a tudományos tervezés témakörébe is olyan értelemben, hogy magába foglalja nem csak a termékek tervezésének tudományos alátámasztását, de a tervezési folyamat és módszertan rendszerező ismereteit is. HUBKA és EDER szerint a tervezéstudomány fontos alkotóelemei a következők:

- a természettudományok, valamint a humán tudományok alkalmazott ismeretanyaga,
- a műszaki rendszerek elmélete,
- a tervezési folyamatok elmélete,
- a tervezésmódszertan.

ANDREASEN [5] a tervezéstudomány két fontos területére is rámutatott, melyeket HUBKA (gépészmérnökök számára) vázolt fel: a tervezési folyamat elmélete (*általános eljárások, módszerek, eszközök*), valamint a géprendszerek elmélete (*a műszaki rendszerek osztályozása, modellezése, stb.*). Ez segít meghatározni a tervezés tudományát, beleértve mind a folyamatot és terméket, mind pedig a tudást és elméletet.

Mindez úgy összegezhető, hogy a tervezéstudomány a tervezésnek egy szervezett, ésszerű és módszeres megközelítése; nem csupán a termékek tudományos alátámasztására szolgál. A tervezés bizonyos értelemben maga is egy tudományos tevékenység.

2.2.3. A tervezés tudománya

A tervezéstudomány és a tervezés tudománya közt a határvonal nem húzható meg élesen, minthogy a tervezés tudománya, magába foglalja a tervezéstudomány fejlesztését (*legalábbis néhány kutató ezt szeretné elérni*). Például a tervezés tudománya a gyakorlati tapasztalat alapján a hatékony tevékenység tudományának is nevezhető, és SIMON [99] a *The Sciences of the Artificial*-ben úgy fogalmazott, hogy a tervezés tudománya a tervezési folyamatra vonatkozó, elemző, részben formalizálható, részben empirikus, tanítható elméletek halmaza.

Ez a nézet nem éppen helytálló. Ahogyan azt GRANT [52] írta: *„A tervezésmódszertani szakemberek, valamint a tervezők véleménye szerint a tervezési tevékenység maga nem tudományos tevékenység, és nem is lesz az, vagyis a tervezés maga egy nem tudományos, vagy más szóval 'a-tudományos' (nem a hagyományos értelemben vett tudományos) tevékenység”*. Habár GRANT azt is leszögezte, hogy *„a tervezés tudománya tudományos tevékenység is lehet, vagyis a tervezés, mint tevékenység a tudományos kutatás tárgyává válhat.”*

A tervezés tudományának egy hasonló nézetét vallotta GASPARDKI [50] is: *„A tervezés tudományát, a metatudományhoz (a tudományt magyarázó tudományhoz) hasonlóan olyan altudományok összességéként kellene értelmeznünk, mely altudományok kognitív érdekei célpontjában a tervezés áll”*.

Így ezen utóbbi nézet szerint a tervezés tudománya a tervezés elveinek, gyakorlatainak és eljárásainak tana. CROSS szerint a tervezésmódszertan *„magába foglalja azokat az ismereteket, melyek leírják, hogy egy mérnök hogyan tervez és gondolkodik, a tervezési folyamathoz megfelelő struktúrák előállítását, új tervezői módszerek, technikák és eljárások fejlesztését és alkalmazását, valamint a természetről való elmélkedést és a tervezői tudásnak, illetve annak a tervezési problémákra történő alkalmazásainak mértékét”* [22].

A tervezés tudománya tehát a feladatoknak arra a halmazára utal, amely a kutatás „tudományos” –vagyis rendszeres, megbízható– módszerei révén megkísérli fejleszteni a tervezésről alkotott szemléletünket, a tervezés hatásfokának javítása (vagyis a tervezés minőségének javítása, és a tervezési idő rövidítése) végett.

2.3. TERVEZŐ ISKOLÁK, TERVEZÉSMÓDSZERTANI ELJÁRÁSOK

Az intuitív gondolkodás a rendszerszemléletű tervezés szerves része, a tervezési folyamat minden lépésében megfigyelhető. SHIGLEY [98] szerint a kreativitás lényege a döntéshozó képesség, amely végigkíséri a tervezőmérnök munkáját a teljes tervezési folyamat során.

Egy tervezési feladat optimális megoldásának keresése közben a megoldással kapcsolatos szemlélet mindig változik. A feladat megismerésének folyamata teszi szükségessé, hogy a tervezőmérnök elképzelése a feladatmegoldásról folyamatosan változzon, hiszen amikor elkezd dolgozni, még nagyon keveset tud az adott termékről, azonban minél alaposabb piac- és szabadalomkutatást végez, annál több tényezőt tud figyelembe venni. Ahogyan a mérnök egyre több részletét ismeri meg a feladatnak, úgy változik nézete a termék-megoldással kapcsolatban is. Ennek megfelelően PÓLYA [87] a feladatmegoldást négy szakaszra osztja fel, melyek:

- a feladat megértése,

- a részfeladatok egymáshoz kapcsolódásának vizsgálata a megoldás alapötletének megtalálásához, valamint a megoldási terv elkészítéshez,
- a terv végrehajtása,
- a megoldás elemzése.

Előfordulhat, hogy a probléma megoldásának keresése közben olyan komplex ötlet merül fel, hogy már a feladat megértésének fázisában „*látszik*” a megoldás, azonban ez rendkívül ritka. PÓLYA úgy véli, hogy ha nem gyullad ki a megoldás szikrája az első pillanatban, akkor be kell tartani az általa javasolt lépéseket. Szerinte a legrosszabb eset az, amikor annak ellenére, hogy nincs meg a frappáns, intuitív ötlet, a megoldás keresője mégis elhagyja a négy szakasz valamelyikét, és anélkül kezdi el kidolgozni a megoldást, hogy megértette volna, mik a pontos követelmények. PÓLYA rendkívül nagy figyelmet fordít a feladatrészek folyamatos ellenőrzésére.

Ebben a fejezetrészen az egyes tervezőiskolák alapvető jellemzői, illetve a különféle tervezésmódszertani eljárások kerülnek bemutatásra. A fejezet végén a módszerek szintézise kerül tárgyalásra.

2.3.1. Az intuíciónál az előírásig: az intuitív, a kognitív és a diszkurzív tervezői modellek

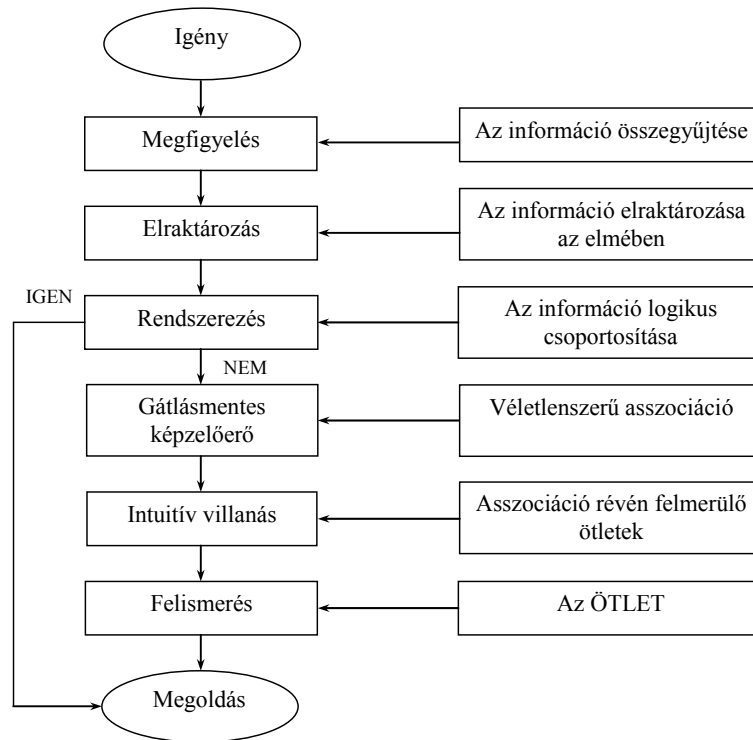
SELYE [97] az intuíciónál tudattalan intelligenciaként említi, mely révén elmélkedés, vagy következtetés nélkül jutunk ismeretekhez. Felfogása szerint az intuiciónál racionális megértés nélküli, közvetlen felfogást, vagy megismerést jelent. *„Az intuiciónál az eredetiség, a találékonyság, a leleményesség minden változatának gyújtószikrája. Az a villanás, amely a tudatos gondolkodás és a képzelőerő között érintkezést teremt”*.

Az 2. ábra szerint az alkotó elme a megfigyelés során a tényeket összegyűjti, elraktározza, majd pedig a csoportosítás során valamilyen logika szerint rendszerezi azokat. Sok esetben már ekkor megjelenik az az ötlet, ami a feladat megoldását elősegíti, azonban számos esetben előfordul, hogy az ember tudattalan képessége, a képzelőerő segítségével, véletlenszerű asszociációk hosszú iterációs láncolata révén felmerülő ötletek, intuitív villanások közül felismeri az *Ötletet*, így jutva el a feladat megoldásához. Az intuiciónál tehát az a képesség, mely az absztrahálás során felvillanó alkalmazható ötleteket a tudatba hozza. Az intuitív villanás, a „megsejtés”, mely annak ellenére, hogy az 2. ábra szerinti lépések váltották ki, nem vezethető le azokból a formális logika alkalmazásával.

HANSEN [55] ezzel szemben úgy értelmezi az intuitív munkát, hogy az ötlet többnyire egyszerű, és világos gondolatmeneten alapszik, hiszen minden új idea az agyban raktározott tapasztalatokból, megfigyelésekből, ezek kombinációjából születik, csupán maga a folyamat –*amely révén az újító termék, vagy gép megszületik*–, jelentősen gyorsabb. Ennek megfelelően a valóságban soha nem beszélhetünk véletlenről, vagyis zseniális ötletéről. HANSEN szerint az intuitív folyamat mesterséges lelassításával feltárulnak azok az elemek, melyekből a megoldás felépült, létrejönnek azok a megoldások is, melyeket az agy automatikusan elvetett volna. Ilyen formán maga az intuitív gondolkodás rekonstruálható és az intuiciónál tapasztalatszerzéssel fejleszthető. A kevesebb tervezői tapasztalattal rendelkező mérnöknek azonban lehetősége van arra, hogy *„mesterséges intuiciónál”* ugyanúgy megtalálja a feladat elvégzését optimálisan biztosító megoldást, mint a tapasztaltabb konstruktőrök.

Összességében megállapítható, hogy az intuitív problémamegoldás minden forrás szerint a megszerzett tapasztalatokból eredő ötleteken alapul. A fogalom meghatározása szempontjából lényegtelen, hogy az intuiciónál segítségével megszülető megoldások milyen lépéseken keresztül jöttek létre. HANSEN elmélete arra világít rá, hogy az intuitív tervezés pótolható, jól közelíthetően rekonstruálható egy olyan módszer segítségével, amely lépésről lépésre támogatja a ter-

vezőt a logikus gondolkodásban. A tervezési módszerek kialakításával már számos elméleti kutató foglalkozott, ennek megfelelően különböző szemléletek alakultak ki. Leggyakrabban az intuitív, a kognitív, és a diszkurzív modellek szerinti felosztást említi a szakirodalom. Ennek a fejezetnek a legfőbb célja, hogy ezeket a modelleket bemutassa.



2. ábra. Az intuíción Selye alapján

Az intuitív tervezési folyamatot a felismerés hirtelen tudatossá válása indítja el. Ez a feladatmegoldási módszer nem csupán a tervezőmérnök rátermettségétől, de kreativitásától, munkahelyi környezetétől is függ. A kreativitás nem tanulható meg, azonban fejleszthető, az egyéni tapasztalatoktól, tanulással elsajátított –*műszaki, gazdasági, jogi, orvosi, stb.*– ismeretektől, kultúrától, intelligenciától függő individuális képesség.

Egy-egy műszaki probléma intuitív megoldása révén már számos szabadalom született az évszázadok során, és minden bizonnyal még fog is születni. Azonban a módszer számos szempontot tekintve hátrányos, ezek közül a két legfontosabb a következő:

- megfelelő ötlet ritkán születik a megfelelő időpontban, mivel az intuíción emberi akarat nem kényszeríthető ki; tehát az intuíción alapuló mérnöki munka, időben nem tervezhető.
- nem rendelkezik mindenki ugyanakkora kreativitással, valamint szakmai tapasztalattal; tehát az intuitív tervezés minősége nem homogén, nagyon függ a tervező személyétől.

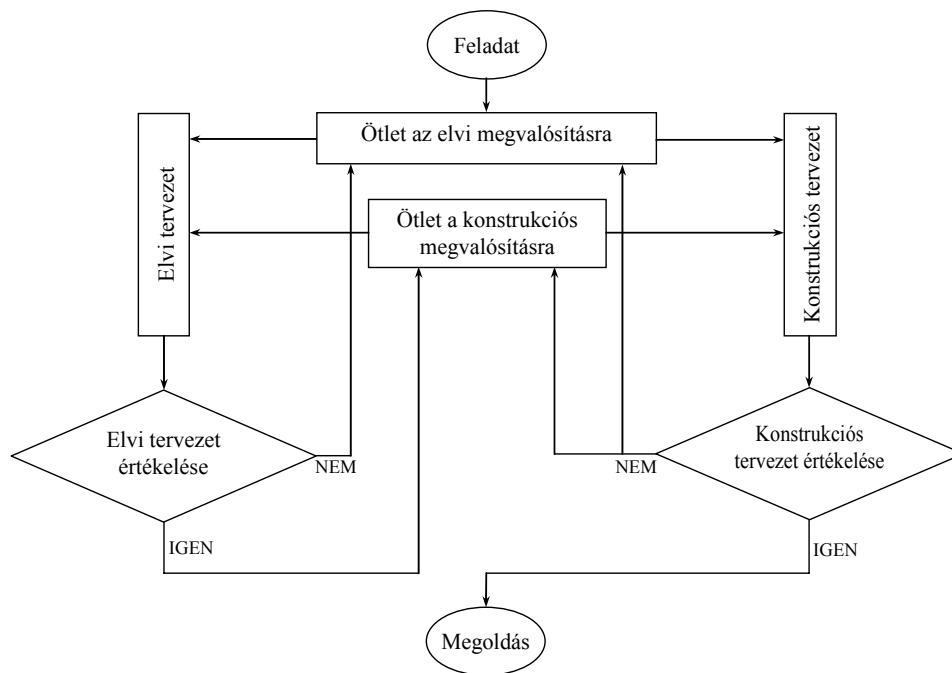
Emiatt arra kell törekedni, hogy valamilyen útmutatásnak megfelelően a tervező, egy előre meghatározott folyamatot járjon végig, mely egy elvárt minőségű műszaki tervet produkál elvárt idő alatt.

2.3.1.1. Az intuitív tervezési modell

Az 3. ábra az irodalomkutatás során feltárt intuitív tervezést leíró modellek alapján készült [102], [103]. A modell jellemzője, hogy a feladatmegoldás két szinten –*az elvi megoldás- és a*

tényleges konstrukció szintjén– valósul meg. Tehát egy-egy tervezési feladat elvégzése során, több szinten több fajta ötlet szükséges.

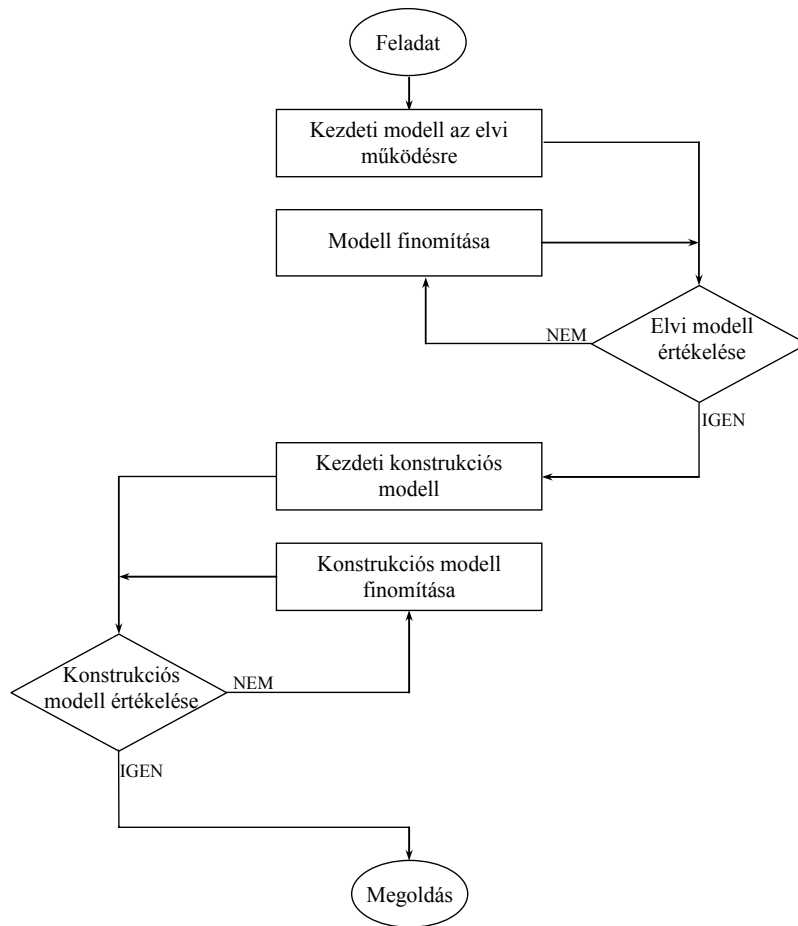
A modell fontos jellemzője, hogy a tervező mérnök annyiszor jut vissza az ötlet szintjére, ahányszor a funkcionális értékelés során nem bizonyul megfelelőnek az ötletből kifejlesztett tervezet; az intuitív tervezés tehát nem monoton előrehaladó folyamat. A munka során a tervező bármelyik szintről bármelyik szintre kerülhet csapongó ötletei miatt. Az intuitív tervezési modell alkalmazása esetén a terv várható minősége, valamint a várható tervezési idő előre nem definiálható.



3. ábra. Az intuitív tervezés modellje

2.3.1.2. Kognitív tervezési modell

A kognitív modell folyamatstruktúrája az intuitív tervező rendszerek folyamatához hasonlítható leginkább, ám ebben az esetben egy nagyrészt nyílt, előrehaladó rendszerről beszélünk, amelyben a feladat megoldásainak különféle szintjein lokális ciklusok ágyazódnak. Az intuíciónak a folyamat végéig nagy jelentőséggel bír –*hiszen a tervezőnek folyamatosan új ötletekre van szüksége*–, azonban megjelennek az előíró szemlélet fontos jellemzői is, vagyis maga a tervezés előírt lépések szerint halad. Ezen kognitív folyamatok legfontosabb jellemzője, hogy megpróbálják leírni, szimulálni a tervezőmérnök munka közbeni gondolkodási tevékenységét. A kognitív szemlélet szerint folytatott tervezés esetén a terv minősége szintén nem határozható meg előre, de a várható tervezési idő a folyamat nyitott, nagyrészt előre haladó jellege miatt jobban becsülhető, mint az intuitív modellnél.



4. ábra. A kognitív tervezési modell OSHUGA alapján [61]

2.3.1.3. A diszkurzív tervezési modell

Az előíró, vagy diszkurzív modellek nem jelentik az intuíciónak teljes háttérbe szorítását, azonban segítik a konstruktort az adott feladat optimális megoldásának megtalálásában intuitív ötlet hiányában is azáltal, hogy előírják a tervezőnek, hogy milyen lépéseket kell elvégeznie az adott tervezési probléma megoldásához. Mindemellett lehetőséget biztosítanak arra, hogy akár egy teljesen új koncepció jelenjen meg a tervezési folyamat végén, mint legjobb megoldás. A diszkurzív módszerek az intuitív módszerekhez hasonlóan tehát szintén a legjobb megoldást keresik, azonban ezt a legjobb megoldást több lehetőség közül választják ki.

A diszkurzív tervezés folyamata két nagy szakaszra, egy bővítő- és egy szűkítő folyamatra bontható. A két folyamat határán a megoldásváltozatok halmaza áll, mely halmaz elemeinek száma attól függ, hogy hány megoldáselemet lehet összekombinálni. Minél nagyobb a megoldáselemek száma, annál nagyobb megoldáshalmaz keletkezik. Túl sok megoldáselem esetén, megnő annak az esélye is, hogy előáll az ún. 'Kombinatorikus robbanás' jelensége. Ez azt jelenti, hogy az egyedi megoldáselemek variálásával elvileg kialakítható megoldásváltozatok száma meghaladja az emberi áttekinthetőséget –*mely képesség akár egyénenként is más lehet*. A kombinatorikus robbanás kezelésének határai számítógépes támogatással jelentősen kitolhatók.

Az intuíciónak –*így az egyén műszaki ismeretei és tapasztalata*– azonban ebben az esetben sem hanyagolható el, mivel a megoldáselemek feltárása során a tervező mérnöknek kreativitására is szüksége van amellet, hogy a gyakorlat során megismert lehetőségeket számba veszi.

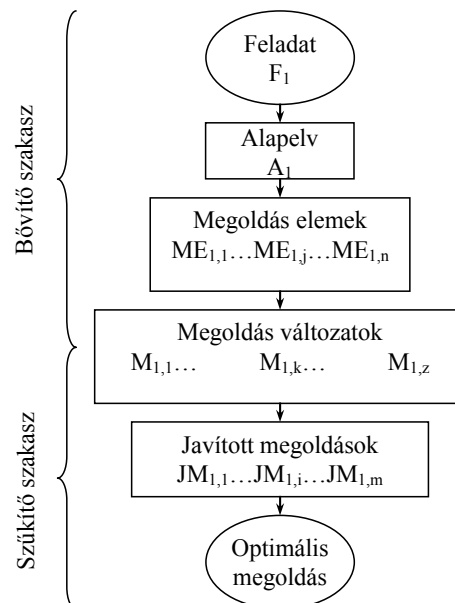
A diszkurzív tervezési folyamat függ az intuíciónál a legkevésbé, emiatt a terv minősége nem függ jelentősen a tervező intuitív képességeitől, csak a tervezés alá vont megoldáselemek számától. Mivel ez a tervezési folyamat monoton előre haladó, a várható tervezési idő jól kalkulálható.

2.3.2. Jelentősebb tervezői modellek

A fejezet azoknak a módszereknek a jellemző tulajdonságait foglalja össze, melyek olyan logikai felépítésűek, hogy kisebb változtatásokkal számítógépre is adaptálhatók, illetve azokat a módszereket, melyek már eleve algoritmizált felépítésűek.

2.3.2.1. Alaprendszer felállítása: Hansen tervezői modellje

HANSEN [55] munkássága során a tervezési folyamatot összrendszerként vizsgálta, amely összrendszer alaprendszerek logikailag egymáshoz fűzött láncszemeiből áll. Az alaprendszer tulajdonképpen egy-egy részfeladat megoldását jelenti, mely a teljes tervezés során többször megismétlődhet, természetesen mindig a folyamat egy magasabb logikai szintjén.



5. ábra. Az alaprendszer HANSEN alapján

HANSEN szerint az alaprendszer felépítésének első lépcsője az alapelv kidolgozása a feladat absztrahálása révén. Ezután következik az alapelvben feltárt probléma elemeihez a megoldáselemek ($ME_1, \dots, ME_i, \dots, ME_n$) feltárására és megfogalmazására, mely megoldáselemek számát minél nagyobbra kell növelni azért, hogy a tervezőmérnök minél több lehetőség közül választhasson. Harmadik lépésként a megoldáselemeket kombinálni kell egymással, így állíthatók elő a megoldásváltozatok ($M_1, \dots, M_j, \dots, M_m$). A szelektálás, amely során a megoldásváltozatok nagyszámú halmazából az optimális megoldást kell megtalálni, általában egy kétlépcsős elemzőeljárás, a hibaanalízisből –a használhatatlan megoldások elvetése, a kedvezőbb megoldások hibáinak kijavítása– és az értékelésből áll. A műszaki értékelés során a javított megoldásoknak konkrét kritériumoknak kell eleget tenniük. Az értékelés végeredményeként a

tervező azt a legjobb megoldást kapja meg, mely a célkitűzésben megfogalmazott feltételeket a legjobban képes kielégíteni a kombinációs úton előállított megoldásváltozatok közül.

2.3.2.2. A tervezés algoritmizálása: Koller tervezői modellje

RODENACKER hipotézise szerint a műszaki rendszerekben csak az energia, anyag és jel állapota, tulajdonságai valamint áramlásának iránya és nagysága változhat. KOLLER [72], [73] ezt az elméletet dolgozta tovább. Munkássága során célul tűzte ki a tervezési folyamat algoritmizálhatóvá tételét, annak érdekében, hogy később a tervezői munka számítógéppel támogatható legyen. Módszertanának alapja, a tervezés folyamatának a lehető legtöbb munkalépésre történő felbontása.

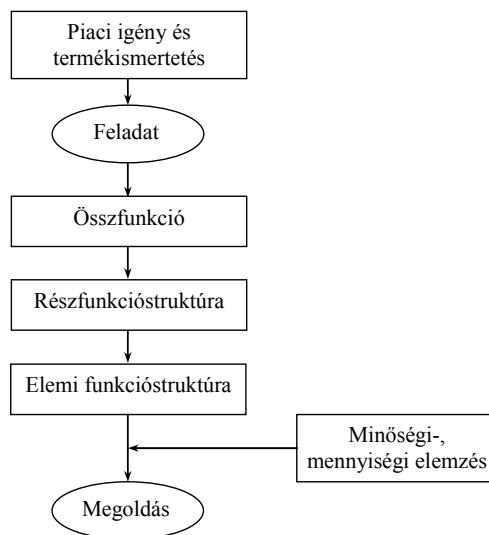
KOLLER az egyes munkalépéseket olyan feketedobozként fogja fel, amelyben a bemenő adatból kimenő adat jön létre:

$$G_{B,i} \Rightarrow G_{K,i} \quad (1)$$

A bemenő és a kimenő adat lehet anyag, energia, vagy információ. KOLLER szerint azt a jelenséget, amely a teljes tervezési folyamat egy-egy feketedobozában lejátszódik, 12 fizikai alpművelet valamelyikével le lehet írni. A teljes megoldás ezeknek az alpműveleteknek a láncolatával teljes mértékben meghatározható, vagyis a tervezési feladat megoldásának összfunckiója ezek összességéből adódik:

$$\sum G_{B,i} \Rightarrow \sum G_{K,i} \quad (2)$$

Számítógépes alkalmazás esetén szükséges a folyamat egyes lépéseinek elvégzésére vonatkozó logikai szabályok felállítása. Ennek következtében a funkcióelemzés megoldása egy olyan funkcióstruktúra, mely logikailag és fizikailag összekapcsolt alpműveletekből áll.



6. ábra. KOLLER tervezői módszere

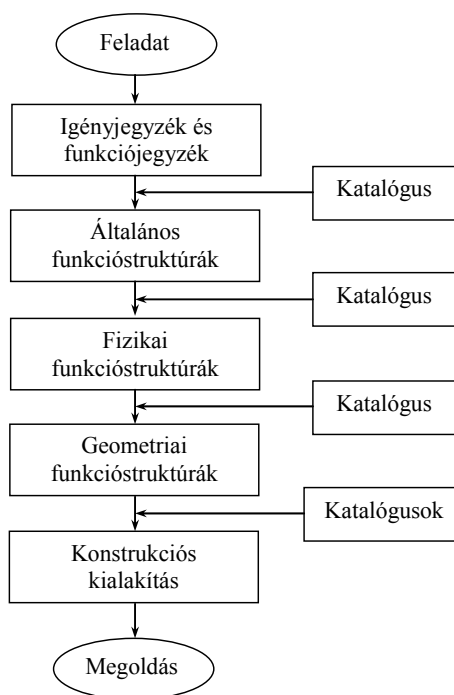
2.3.2.3. Tervezés katalógussal: Roth tervezői modellje

A tervezői katalógusok a tervezési problémák ismert és bizonyított megoldásainak gyűjteményei. A tervezési katalógusokon alapuló módszeres tervezés alapgondolata az, hogy a konstruktőr a legjobb megoldás keresése során ne csak saját ötletét vegye figyelembe, hanem használja fel mások dokumentált tudását.

2. táblázat. Tervezői katalógusok

FELHASZNÁLÁSI TERÜLET	SZERZŐ
Alapelvek tervezői katalógusokhoz	Roth
Kötések	Diekhöner, Ersoy, Ewald, Fuhrmann, Gießner, Grandt, Hinterwalder, Kastner, Kollmann, Kopowski, Lohkamp, Roth, Wölse,
Vezetékek, csapágys	Diekhöner, Ewald, Roth
Hajtástechnika, energiatermelés, erőátviteli vezetékek	Ewald, Jung, Kopowski, Raab, Roth, Schneider
Kinematika, mechanizmuselmélet	Schneider, Raab, Roth, VDI 2727 Blatt 2, VDI 2222 Blatt 2
Hajtás	Diekhöner, Ewald, Lohkamp, VDI 2222 Blatt 2
Biztonságtechnika	Neudorfer
Ergonómia	Neudorfer
Kikészítő eljárások	Roth

Az ismertebb tervezői katalógusok a felhasználási terület alapján a 2. táblázat szerinti csoportokra oszthatók. A táblázat azt is bemutatja, hogy az egyes területeken kik állítottak össze tervezési katalógust.



7. ábra. A teljes tervezési folyamat ROTH alapján

Ahogy a 7. ábra is szemlélteti, ROTH módszerének alapja a feladat pontos megfogalmazása, mely magába foglalja a funkciójegyzék és a követelményjegyzék összeállítását. A funkciójegyzék egy olyan lista, mely a fizikai hatások, hatáselvek meghatározásából megfogalma-

zott funkciókat sorolja fel. ROTH ezeket a funkciókat általánosan fogalmazta meg. Teóriája háromféle funkcióstruktúrát ismertet, melyek katalógusok segítségével egymásba átvihetők. ROTH szerint egy feladatot (*összfeladatot*), mely folyamatok egész sorát tételezi fel, könnyebb akkor megoldani, ha azt részfeladatokra lehet bontani. Ennek a módszernek az az elv az alapja, mely szerint az összfeladatot olyan egyszerű feladatokra kell visszavezetni, melynek már léteznek jól működő, ismert megoldásai.

ROTH az általános funkciókból (*energia-/anyag-/információ tárolása- /vezetése-/átalakítása-/megváltoztatása-/összekapcsolása*) általános funkcióstruktúrákat épít fel. Ezek az általános funkcióstruktúrák egy-egy adott szerkezet elvi működését írják le. Ezekhez katalógus segítségével fizikai elveket, hatásokat, axiómákat rendel, így fizikai funkcióstruktúrákat épít fel, melyek adott esetben logikai,- vagy vektoriális funkcióstruktúrával is helyettesíthetők. Ezen funkcióstruktúrákból katalógus segítségével geometriai funkcióstruktúrákat hoz létre, melyek egy-egy részegység mechanizmusok formájában történő egymáshoz való kapcsolódására mutatnak lehetőséget.

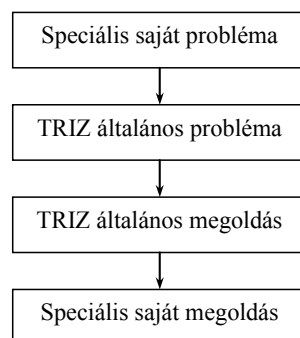
ROTH különös figyelmet szentel a követelményjegyzék összeállításának. Abból indult ki, hogy az igények listájának nem egy meghatározott tervezési szakaszhoz kell kötődnie, hanem az egész tervezési folyamatra érvényesnek kell lennie, ezért a követelményjegyzéket a feladat megfogalmazásától kezdve folyamatosan bővíteni kell.

ROTH módszertani elmélete teljes egészében az általa összeállított tervezési katalógusokra épül. Ezeknek a táblázatoknak a legfőbb előnye amellet, hogy a már meglévő elveket, megoldásokat rendszerezik, hogy számos kiegészítő tulajdonságot tartalmaznak a műszaki megoldás fizikai, mechanikai tulajdonságait illetően. A 2. táblázat szerint ROTH nem csupán a módszer alapjait fektette le, de a technika számos területén aktívan részt vett a különböző katalógusok összeállításában.

2.3.2.4. A Feltalálói Problémamegoldás Módszere

A TRIZ elméletet 1946-ban ALTSHULLER [3], kezdte kifejleszteni az akkori Szovjetunióban. A módszer mozaikszavas neve az orosz, Теория Решения Изобретательских Задач (*Theorija Reschenija Izobretatel'skij Zadach*) elnevezésből származik, mely nem más, mint a „Feltalálói Problémamegoldás Módszere”.

A TRIZ elmélet megalkotásának első lépése az általános alapelvek, vagyis az ún. *40 alapvető irányelv* megfogalmazása volt, amely alapelvek listáját a felmerült problémák vizsgálata során tapasztalt szabályszerűségek alapján állították össze, melyhez számos szabadalmat tanulmányoztak.



8. ábra. A TRIZ feladatmegoldási módszere

A TRIZ elmélet szerint a fejlesztés öt különböző kategóriában valósulhat meg [3]:

- Első kategória: nyilvánvaló, már létező rutin megoldások, a vizsgált szabadalmak 32%-a erre a szintre sorolható.
- Második kategória: létező rendszer kisebb fejlesztése, a vizsgált szabadalmak 45%-a sorolható erre a szintre.
- Harmadik kategória: egy létező rendszer alapvető fejlesztése, ismert megoldások segítségével. Ezek a találmányok a vizsgált szabadalmak 18%-át teszik ki.
- Negyedik kategória: olyan találmányok, melyek új elvek felhasználásával elégítik ki a találmány elsődleges funkcióit. Jellemzően nem a már bevált megoldásokon alapulnak, inkább a tudományból veszik az újdonság alapját szolgáló ötletet. Az ALTSCHULLER csoportja által vizsgált szabadalmaknak 4%-a sorolható a fejlesztés negyedik szintjére.
- Ötödik kategória: nagyon ritka tudományos felfedezés, mely alapvetően új rendszer kialakítását jelenti. ALTSCHULLER kutatásai szerint a megvizsgált szabadalmaknak csupán 1%-a sorolható a fejlesztésnek erre a szintjére.

A Feltalálói Problémamegoldás Módszere –*csakúgy, mint a többi módszeres tervező eljárás*– egy speciális eszközrendszer segítségével keresi a legjobb megoldást. Az eljárás lényege: a felmerülő problémát általánosítva kell megoldani, ezáltal adódik a saját probléma megoldása.

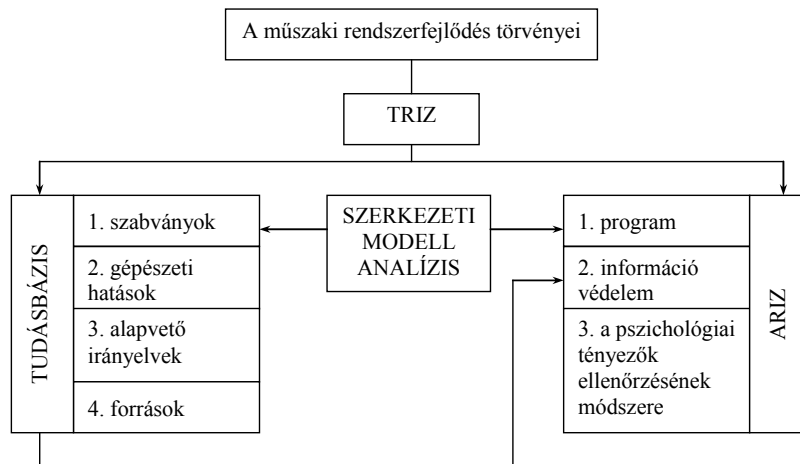
A TRIZ módszer alap gondolata, hogy az adott feladat ellentmondásokra épül; a módszer ezen konfliktusok feloldásával keresi a megoldást, a megoldás megtalálását pedig az irányelvekkel segíti. Az ellentmondásokat két csoportra lehet osztani:

- technikai ellentmondások,
- fizikai ellentmondások.

Gyengülő paraméterek	1. Mozgó objektum tömege	2. Álló objektum tömege	3. Mozgó objektum hossza		4. Álló objektum hossza	5. Mozgó objektum területe		...
Javuló paraméterek	1. Mozgó objektum tömege	2. Álló objektum tömege	15	8	-	29	17	
		-			-	38	34	
		-	8: ellensúly 15: dinamika 29: pneumatika és hidraulika 34: újratöltés és helyreállítás		1	-		
		-			35			
	8	15	-		-	15	17	
	29	34			-		4	
	-	35	28	-		-		
	-	40	29					
	2	17	-		14	15	-	
	29	4			18	4		
...							...	

9. ábra. Az ellentmondási mátrix részlete

A technikai ellentmondások esetében a cél az ellentmondás teljes feloldása. Tehát annak a törvényszerűségnek a kiküszöbölése, hogy egy adott mérnöki paraméter javítása egy másik mérnöki paraméter gyengülését eredményezi. A fizikai ellentmondás azt fejezi ki, hogy egyszerre kell teljesülnie két egymásnak ellentmondó állapotnak. Ebben az esetben valamilyen szeparációs eljárást kell alkalmazni.



10. ábra. A TRIZ szerkezeti felépítése

A technikai ellentmondások feloldásához nyújt segítséget az ún. *ellentmondási mátrix*. Ez egy 39x39-es táblázat, sorai az adott feladatra jellemző javuló tulajdonságokat tartalmazzák, oszlopai pedig az ezzel párhuzamosan gyengülő jellemzőket. Az i -edik sor és a j -edik oszlop metszéspontjában található e_{ij} mező azokat a kódokat tartalmazza, melyek a 40 alapelv valamelyikére utalnak. Az ellentmondási mátrix tehát egy olyan adatbázis, mely azokat az ismert megoldásokat foglalja magába, melyek képesek az ellentmondások kiküszöbölésére.

ALTSHULLER 1959 és 1985 között fejlesztette ki az ARIZ-t (*Algorithm of Inventive Problems Solving*), vagyis a feltalálói problémamegoldás algoritmusát, mely magába foglal egy olyan programot, mely a felmerülő ellentmondások kiküszöbölése révén a probléma megoldását adja. A TRIZ szerkezeti felépítését a 10. ábra szemlélteti. A TRIZ tudásbázisát az ARIZ egy szerkezeti modell analízis révén hasznosítja, mely az adott tervezői problémának egy olyan formális nyelvre lefordított modellje, amit az ARIZ kezelni tud.

2.3.2.5. A konfiguratív termékek és a termék-konfigurációs folyamat

Számos iparág versenyképessége követeli meg a hatékony tervezést, aminek alapja a nagyszámú termékváltozat előállítása és elemzése. Ennek a problémának a megoldására TIHONEN [106], [107] és társai egy olyan módszert dolgoztak ki a számítástechnika területén, mely az előzőekben ismertetettéktől jelentős mértékben eltér. A módszer a termékanalízisen alapul, azonban a tervezési folyamat nem a klasszikus szekvenciák mentén tagolódik. Ahhoz, hogy mindez áttekinthető legyen, nem csupán a szakterület irodalmi vonatkozásait kell megvizsgálni, de érdemes kitekinteni más műszaki területekre is. MANNISTÖ [106] és társai a felépítési folyamatot két alapvető részre bontották, a felépítési feladatra, valamint a felépítési termékre.

A konfigurációs feladat egyik első definíciója 1981-ből származik. MITTAL és FRAYMANN [16] úgy fogalmazták meg a konfigurációs folyamatot, hogy adott:

- egy fix, előre megfogalmazott komponenshalmaz,

- a kívánt felépítés valamilyen megfogalmazása,
- néhány kritérium az optimális kiválasztás érdekében.

Ezek segítségével fel kell építeni egy, vagy több konfigurációt, melyek kielégítik a követelményeket, vagy rávilágítanak az azokban rejlő ellentmondásokra. A felépítés leírása két lényeges dologból tevődik össze:

- egy komponens leírható tulajdonsághalmazzal, más komponensekhez való csatlakozási pontokkal, valamint a csatlakozási pontokban működő kényszerekkel, melyek jellemzik az egyes komponenseket,
- a konfiguráció egy komponens halmaz, valamint a halmazban lévő komponensek közötti kapcsolatok leírása.

Természetesen ezzel az elmélettel kapcsolatosan is fellép a kombinatorikus robbanás problémája, hiszen sok komponens esetén a termékváltozatok száma rendkívüli módon megnő, így az eljárás egy bonyolult, nehezen kezelhető, időigényes folyamattá válik.

BROWN [18] azt mondja, hogy MITTAL és FRAYMAN nézetei a konfiguratív szemlélet informatikai területeken való alkalmazhatóságáról nem elég pontosak, ily módon a terméktervezésben közvetlenül nem vehetők át. BROWN leginkább a termékösszetevők, vagyis a komponensek közti kapcsolatok pontatlan leírását véli kritikusnak.

BROWN továbbá azt a megállapítást tette, hogy a konfiguratív tevékenység logikai kapcsolat esetén a komponensek kiválasztása, a komponensek közti kapcsolatok felállítása, valamint a komponensek értékeléséből tevődik össze, melyet a következő szimbolikus egyenlettel fogalmazott meg [18]:

$$C_o = K_i + T + E \quad (3)$$

Az értékelés további két résztevékenységből tevődik össze, úgymint a termék alkalmasságának értékelése, valamint a követelményeknek való megfelelés vizsgálat. Ezt a következő szimbolikus egyenlet fejezi ki BROWN szerint:

$$E = A + M_r \quad (4)$$

Logikai kapcsolat esetén tehát, ha a komponens kiválasztása nem megfelelően történik, akkor a társítás sem lesz sikeres, így az értékelés kedvezőtlen megoldást eredményezhet.

BROWN a specifikus kapcsolat modellezésére is tett megfogalmazást. Ekkor a konfiguratív tevékenység a komponensek kiválasztásából, a komponensek közti elvonatkoztatott kapcsolatok felállításából, a komponensek közti jellegzetes kapcsolat feltárásából *–különös tekintettel a térbeli elhelyezkedésre–*, valamint az értékelésből áll:

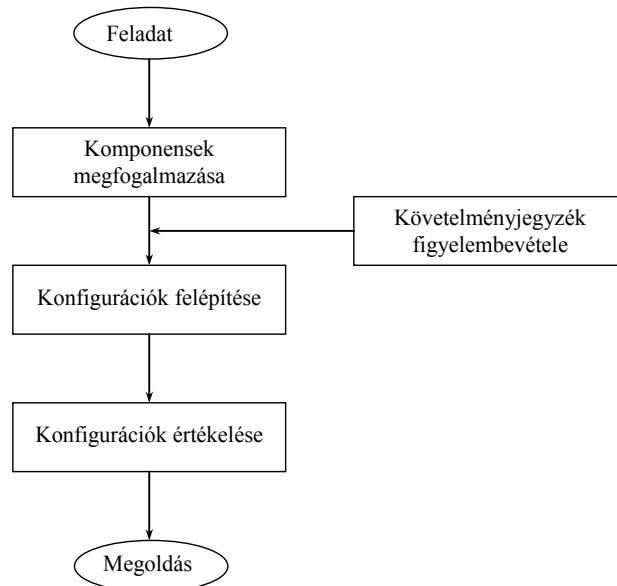
$$C_o = K_i + K_a + R + E \quad (5)$$

Ez a folyamat részben logikainak tekinthető, mivel a viszonyba állítás során olyan logikai kapcsolatok kerülnek megfogalmazásra, mint például az egyik komponens a másik mellett van, vagy érinti azt. Ezzel szemben az elrendezési fázisban pontos meghatározásokra kerül sor az egyes komponensek térbeli, valamint egymáshoz viszonyított helyzetét tekintve.

BROWN elmélete szerint a konfiguratív szemlélet legnagyobb hiánya, hogy míg a tervezési folyamat valós értékeket eredményez, a konfiguratív modellben nincsen méret, vagy kiterjedés. Minthogy a konfiguratív tervezést leginkább az informatika, azon belül is a szoftverfejlesztés területén alkalmazták sikerrel, így a legtöbb példa is erről a területről származik. A módszer legfontosabb alappillére, hogy az egyes termékcsaládok minden egyes terméke a vásárlók igényeinek megfelelően készül, az előre definiált konfiguratív modell alapján, ami leírja a lehetséges termékváltozatok egy halmazát.

TIIHONEN [106], [107] és társai a konfiguratív terméket a következő alaptulajdonságokkal jellemezték:

- minden átadott egyedi terméknek ki kell elégítenie az egyedi követelményeket,
- a terméket elő kell tervezni, hogy a különböző felhasználói igényeknek megfeleljen,
- minden egyedi terméket előre definiált komponensek vagy modulok kombinációjaként határoznak meg,
- a termékeknek előre megtervezett általános szerkezetük van,
- az eladás-kézésítési folyamat részeként nincs szükség kreatív, vagy innovatív tervezésre, tehát a folyamat csak szisztematikus változatképzést igényel.



11. ábra. A termékfelépítési folyamat

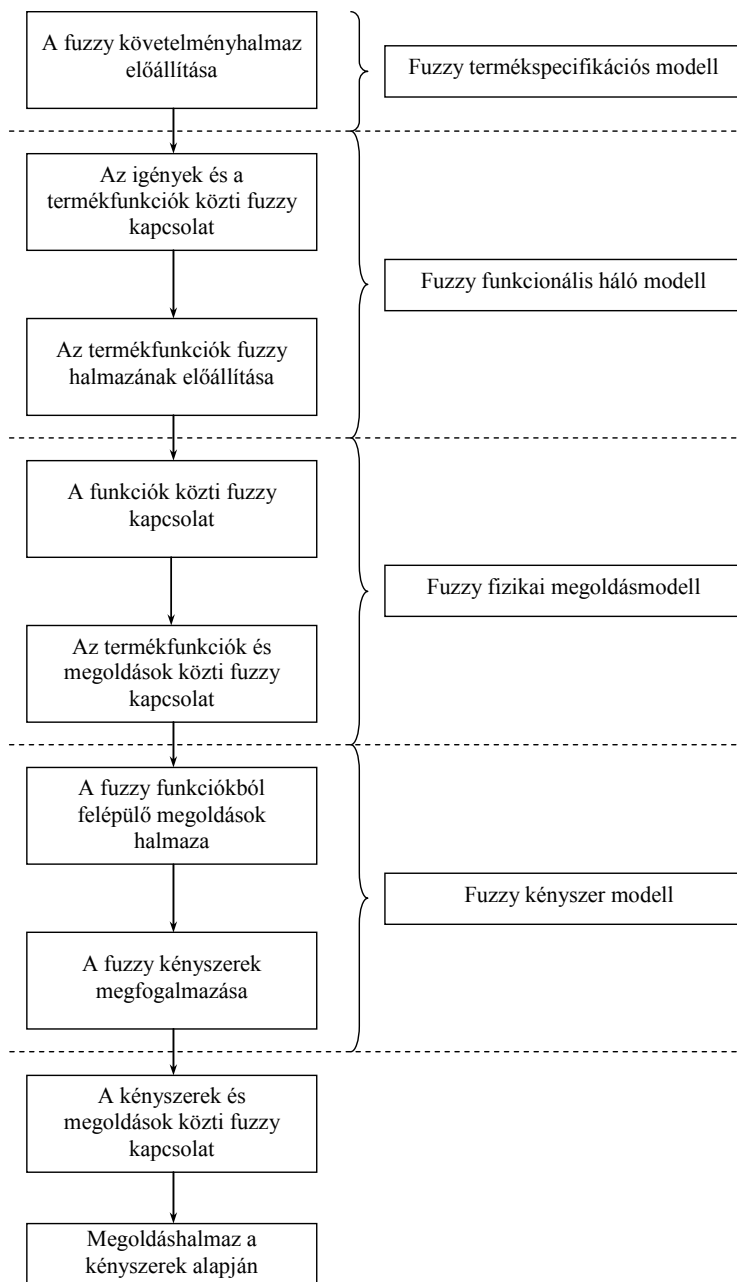
Ezek az alapjellemzők is tükrözik, hogy TIHONEN módszere nem fektet hangsúlyt a termékötlet kidolgozására, hiszen a számítógépes szoftverfejlesztés során nem is azok a tervezési elvek használatosak, mint egy műszaki alkotás esetében. Ily módon azt mondhatjuk, hogy a BROWN által megfogalmazott hiány, miszerint a TIHONEN-féle konfiguratív tervezés nem foglalkozik a parametrizálhatósággal, helyén való. Hiszen a műszaki tervezés során szükség van méretmegadásra, míg egy számítógépes szoftver tekintetében mindez elhanyagolható. Azonban megállapítható, hogy a tervezési folyamat összetettségéből adódóan vannak olyan tervezési lépések, amelyek során hasznosan alkalmazható egy, a TIHONENÉHEZ HASONLÓ KONFIGURÁCIÓS MODELL. Az erre legalkalmasabb mozgástér a koncepcionális tervezés fázisa. Ekkor ugyanis még nincsenek konkrét dimenziói a gépészeti termékeknek, így módon a gépészeti tervezés során is haszonnal alkalmazható a TIHONEN és társai által a számítógépes termékcsaládokra kidolgozott elmélet.

2.3.2.6. Fuzzy-logikán alapuló terméktervezési módszer

DECIU, OSTROSI és FERNEY [35] tettek javaslatot a konfiguratív termékek tervezésének gépészeti alkalmazására. Módszerük nem csupán a konfigurálhatóságon alapul, de a fuzzy logika elemeit is felhasználták az általuk javasolt közelítés kidolgozása során.

A fuzzy logika tulajdonképpen nem más, mint a hagyományos, kétértékű logika többértékűre való kiterjesztése, ahol a következtetési eljárások különböznek a hagyományos logikában megismert módozatoktól. A személyre szabhatóság és a feladat sajátosságainak figyelembevétele igényli az alternatív következtetési eljárásokat, a fuzzy logikában ennek megfelelően

többféle lehetőség adódik. A fuzzy rendszerek a fuzzy logikán alapulnak; olyan leképezések, melyek m dimenziós fuzzy halmazokat képeznek le n dimenziós fuzzy halmazokra. Az f leképezést a rendszer becsüli, anélkül, hogy pontos ismeretekkel rendelkezne róla.



12. ábra. Többszörös Fuzzy modell [35] alapján

„SELYE [97] azt írja, hogy „a természetben semmi sem jellemezhető hiánytalanul, mert minden egységében, bárhogyan határozzuk is meg, újabb részletek fedezhetők fel a további kutatás segítségével”. Úgy véli, csupán az elvont gondolkodás (pl.: a matematika) területén érvényes az az állítás, hogy egy fogalmat pontosan meg lehet határozni. A természetben azonban számos alkalommal előfordulnak esetek, amikor egy ilyen jellegű értékadás –bár pontosnak tűnik– mégsem határozza meg eléggé a leírni kívánt folyamatokat.

A konfigurálható tervezés egyik alapgondolata az, hogy minden körülmények között szem előtt kell tartani a felhasználói igényeket. Minthogy a konfiguráció képzés előre definiált komponensek segítségével jöhet létre, azonban a felhasználóknak az egyes termékekkel szemben támasztott követelményei nem konkrét, fixen meghatározható igények, a fuzzy logika alkalmazása kézenfekvőnek tűnik. DECIU, OSTROSI és FERNEY kidolgoztak egy olyan eljárást, mely kihasználja a konfigurálható tervezés előre definiált eszközeit, valamint a fuzzy logikának azt a lehetőségét, hogy bizonyos döntési helyzetek esetén többértékű megoldást biztosít. Ez azt jelenti, hogy ugyan már a tervezés első pillanatában meg kell hozni bizonyos döntéseket –*előre definiálni kell a komponenseket*–, azonban ez teszi lehetővé, hogy a tervező mérnöknek a további lépések során ne kelljen egyes megoldásokat határozottan elvetnie.

A bizonytalanság a tervezési folyamat szerves része. Ez nem logikai pontatlanságot, sokkal inkább az előzetes, még nem teljes, vagy fuzzy tervezői leírás valódi pontatlanságát jelenti. Annak érdekében, hogy azokat a bizonytalanságot keltő jellemzőket, melyek a konfigurációs tervezői folyamatot meghatározzák, a módszer kidolgozói a fuzzy halmazokkal való megközelítést javasolják. A fuzzy halmazokkal való megközelítés különösen alkalmas a bizonytalan információk kezelésére, minthogy a megoldások halmazát a fontosság különböző szintjeivel látja el.

A módszer alkotói egy többszörös fuzzy modellel történő megközelítésre tettek javaslatot (12. ábra). A fuzzy modellek a tervezési folyamatot egészítik ki, szem előtt tartva a felhasználói követelményeket. Megközelítésük a következő modellekből épül fel:

- fuzzy termékspecifikációs modell,
- fuzzy funkcionális háló-modell,
- fuzzy fizikai megoldásmodell,
- fuzzy kényszer-modell.

A felhasználói igényeket a felhasználói szükségletek határozzák meg. A felhasználó igények azonban sok esetben nem határozhatók meg pontosan. Ha például a vásárló részéről felmerül egy olyan igény, hogy a termék (*pl. egy autó visszapillantó tükrre*) stabil legyen, akkor ezt, a klasszikus tervezői modell szempontjából nem lehet pontosan definiálni.

A többszörös fuzzy modellel irányított konfigurálható módszer leginkább termékcsaládok tervezése esetén alkalmazható eredménnyel. Akkor hasznos ez az eljárás, amikor az a cél, hogy minél többféle terméket, termékváltozatot hozzunk létre. Azonban lényeges megemlíteni, hogy a disszertációban javasolt módszer alapötletét az ebben a fejezetben ismertetett eljárás adta.

2.3.3. A módszertani modellek szintézise

Az intuitív tervezés nagymértékben támaszkodik a benyomásokra, az egyéni kreativitásra, a tervező mérnök korábbi tapasztalataira. Azonban ezekkel a képességekkel és előzményekkel nem minden konstruktőr rendelkezik. Ez tette szükségessé az irányelvek, a különböző módszerek szerint történő előíró –*vagy más néven diszkurzív*– tervezést. Ezeknek az eljárásoknak köszönhetően a koncepcionális tervezés, az elvek és hatások megfogalmazása, az elvonatkoztatás egyszerűbbé válik, a mérnök társadalom ismeretei gyorsabban bővülnek.

Az intuitív és a kognitív tervezés nagy hátránya a diszkurzív tervezéshez képest, hogy csupán egy megoldást vizsgál, tökéletesít. Ezzel szemben a diszkurzív tervezés lehetővé teszi, hogy az irányelvek segítségével, módszeresen létrehozott megoldásváltozatok közül, értékeléssel válassza ki a tervező mérnök a legjobb megoldást. Megállapítható tehát, hogy az irányelvek, valamint a tervezői katalógusok, vagyis összefoglaló néven az előíró módszerek nélkülözhe-

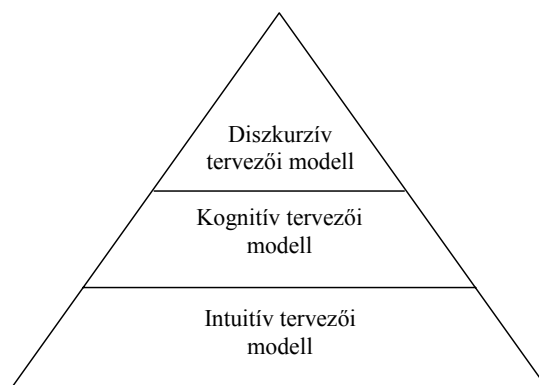
tetlenség a modern tervezés során, hiszen napjainkban számos lehetőséget kell megvizsgálnia a tervező mérnöknek, hogy új, vagy újszerű megoldást tudjon előállítani.

Az előíró tervezés eszközei az irányelvek, a katalógusok és az algoritmusok. A tervezés módszertani eljárások irányelvek segítségével adnak tanácsot, hogy bizonyos variációk közül melyiket érdemes választani, katalógusok szisztematikusan felépített tudástárából javasolnak jól működő elvet, hatást, funkciót, konstrukciós megoldást, az adott probléma konkrét műszaki realizációját.

3. táblázat. A tervezési modellek jellemzői

	INTUITÍV MODELL	KOGNITÍV MODELL	DISZKURZÍV MODELL
Tervezési szintek közti átjárhatóság	Bármelyik szintről bármelyik szintre lehet jutni attól függetlenül, hogy hol tart a tervezési folyamat.	Csak a szinteken belül van visszacsatolás (elvi tervezésen belül/ konstrukciós tervezésen belül). A szintek között csak előrehaladás van.	Van visszacsatolás a teljes tervezés során, de alapvetően folyamatos előrehaladás jellemző.
A várható tervezési idő	Előre nem kalkulálható	Becsülhető	Előre kalkulálható
A terv várható minősége	Előre nem kalkulálható	Előre nem kalkulálható	Előre kalkulálható
Megoldásváltozatok száma	Kevés	Kevés	Számos
Az intuíció fontossága	Nagyon fontos	Fontos	A legkevesebb intuíció szükséges

A módszeres tervezés eszközszerének hatékonyabb alkalmazása érdekében célszerű ezen eszközök (*például: a ROTH-féle katalógusok, és a KOLLER-féle logikai operátorok*) automatizálása, számítógépi alkalmazása. Ez nem annyira a távoli jövő feladata, hiszen napjainkban is léteznek már számítógépes tudás-táblázatok, azonban ezek még korántsem teljes körűen alkalmazzák a már meglévő tervezői katalógusokat.



13. ábra. A pszichológiai szemléletű tervezési modellek hierarchiája

A katalógusok számítógépes adaptálása mellett figyelembe kell venni a genetikus algoritmusok hatékonyságát a termékoptimalásban. A genetikus algoritmusok alkalmazása lehetővé teszi egy adott populációból a peremfeltételeknek leginkább megfelelő megoldás kiválasztását. Ez az eljárás azonban az ALTSCHULLER [3] által említett fejlesztési szintek közül legfeljebb a harmadik szint elérését teszi lehetővé; mivel a termékpuláció minden új tagja egy korábbi

megoldásból származtatható, kizárható az elvileg is új megoldások genetikai kialakulása. Ily módon a legfiatalabb egyed is visszavezethető a kiindulási populáció valamely egyedére.

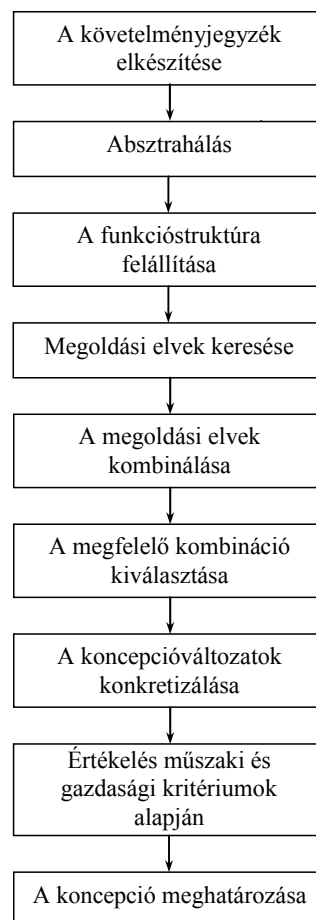
Tehát a módszeres tervezés jövőbeni célja, a már meglévő számítógépes eszközök fejlesztése, bővítése, új lehetőségek feltárása, az altschulleri fejlesztési szintek minél magasabb szintű ki-elégítése CAD-rendszerbe integrálható lehetőséggel.

Az 3. táblázat az intuitív-, a kognitív- és a diszkurzív tervezői modellek jellemző tulajdonságait foglalja össze. Az intuíciónak a fontosságát a különféle tervezői modellekben a 13. ábra szemlélteti. A piramis alakja (*a vízszintes metszék*) mutatja az intuíciónak a szükségességét és mennyiségét az egyes tervezési modellekben. Végül megállapítható, hogy az intuíció minden tervezési folyamat során jelen van, és néhol fontos szereppel bír. Arra nincs garancia, hogy a diszkrét megoldáselemek variálásán alapuló módszeres tervezési technika biztosan jobb eredményt hoz, mint az intuitív. Az viszont bizonyos, hogy a megoldáselemek számának növelése növeli a módszeres tervezési technika eredményének biztonságát.

3. FUNKCIÓK KÖZTI KAPCSOLATOK LEÍRÁSA

A legtöbb tervezési módszer szerint a termékek a koncepcionális tervezés szakaszában funkcióstruktúrákkal írhatók le. A hagyományos elméletek a terméket leíró funkcióstruktúrát általában a termék működési elvét képező fizikai elvek alapján építik fel. A hagyományos módszeres tervezés anyag-, energia-, és információ áramot vizsgál, valamint ennek a három tényezőnek egy terméken belül megfigyelhető kapcsolatait írja. A termékleíró funkcióstruktúrák matematikai megfogalmazására, generálására az irodalom kevés javaslatot tesz. A jelen fejezet a hagyományos koncepcióképzés részfeladat megoldásainak matematikai leírásának lehetőségeit foglalja össze, különös tekintettel a funkcióstruktúrákra.

3.1. A KONCEPCIÓKÉPZÉS FOLYAMATA



14. ábra. A koncepcióképzés folyamata [85]

A 14. ábra a koncepcióképzés fázisának lépéseit szemlélteti. A tervező a kész követelmény-jegyzék alapján elvonatkoztat azért, hogy a probléma lényegét felismerje. Az absztrahálás révén az összfunctió és a részfunctiók is megfogalmazásra kerülnek, így a functióstruktúra is elkészíthető. A tervezőmérnök ezután olyan megoldási elveket keres, melyek kielégítik az egyes részfunctiókat. Ezeket a megoldási elveket úgy kell kombinálni, hogy az egyes változatok az összfunctiót kielégítsék. Ezek közül a változatok közül a műszaki és gazdasági kritériumok alapján kerül kiválasztásra a megfelelő koncepció.

3.2. A FELADAT RÉSZFELADATOKRA VALÓ BONTÁSA

Az előző fejezetben bemutatott három fő irányzat –*intuitív, kognitív, diszkurzív eljárások*– közül a további tárgyalás szempontjából a tervezési folyamat koncepcionális szakaszának diszkurzív megközelítése a leglényegesebb. Ezeket a módszereket általánosan szemlélve egyértelműen megállapítható, hogy a koncepcióképzés egyes szegmensei, szekvenciái meghatározott sorrendben követik egymást, az egyes elméletek csupán abban különböznek, hogy az egyes szegmensekben előírt feladatot milyen segédeszközökkel hajtja végre a tervező mérnök. Ezek a segédeszközök elméletenként átfedésekkel változnak, a módszerek kidolgozói ezek alkalmazását nem előírják, csupán javaslatot tesznek használatukra.

SELYE [97] szerint bármit kutassunk is, az első kérdés az, hogy milyen kisebb egységekből áll, és milyen nagyobb rendszerben foglal helyet egy bizonyos egység. A módszeres tervezés gyakorlatában is ez a nézet tűnik a legcélravezetőbbnek. A tervezési feladat a legkönnyebben úgy oldható meg, ha a teljes feladat minél mélyebben egymásba ágyazott, kisebb részekre, feladat egységekre van felosztva. A feladatmegoldás kulcsa tulajdonképpen abban rejlik, hogy a tervező a rá bízott feladatot milyen alapegységekre bontja. A tématerület kutatói ezeket az alapegységeket különbözőképpen nevezték el, ami gyakran nehézkessé teszi az egyes elméletek közti kapcsolatok megállapítását, az azonos lépések felismerését.

KOLLER [72], [73] azt javasolja, hogy a koncepcionális tervezés során az „*összfunktio*”-t „*részfunktiostruktúrák*”-ra kell felosztani. HANSEN [55] az „*eredő funktio*”-t egymással egy logikai szinten lévő „*funktio*”-ra osztja, míg ROTH [93] az „*összfeladat*”-ot bontja „*részfeladat*”-okra.

TAJNAFŐI [104] úgy foglalja össze a módszertani eljárások során alkalmazott functiók jellemzőit, hogy a functióelemeket a feladatelemekből alakítják ki úgy, hogy azok célokat határozzanak meg és ne konkrét eszközöket. A konkrét eszközök pontosan meghatározzák a megvalósítás módját, ezáltal már a problémamegoldás kezdetén csökkentenék a lehetséges megoldások számát. Egy cél több megoldáselemmel is megvalósítható.

Többnyire minden módszer azt ajánlja, hogy a teljes feladatot részenként kell vizsgálni, a részfeladatokat pedig még kisebb elemekre, functiókra kell bontani. A functiók meghatározzák az egyes részfeladatokat alkotó elemek feladatát, de nem korlátozzák le, hogy az egyes elemek milyen fizikai/mechanikai/stb... elv szerint működjenek.

Az egyes elméletek alapján a functiókról a következő fontos megállapítások fogalmazhatók meg:

- a functiót úgy kell megfogalmazni, hogy azzal a tervező ne korlátozza a kialakítás elvi lehetőségeit a termékmegvalósítás során.
- a functiók közt a legtöbb módszer esetében hierarchia figyelhető meg, tehát a functió fogalom jelentése a feladatmegoldás különféle szintjein, különféle módon jelenik meg.

A koncepcionális tervezés során a terméktervező csupán a műszaki functióra fókuszál, vagyis hogy a későbbi termék, vagy annak egyes részei képesek-e ellátni a tőlük elvárt feladatot.

3.3. A FUNKCIÓSTRUKTÚRÁK MATEMATIKAI LEÍRÁSA

A módszeres tervezés területén, ahogyan már említésre került, a termék-funkcióstruktúrák inkább fizikai kapcsolatokat leíró struktúrák, ily módon a gráfelméletben használatos fogalmak alig jelennek meg a módszeres tervezési elméletekben. A disszertáció azonban különös hangsúlyt fektet a termékek gráfokkal történő leírására, illetve a leírás törvényszerűségeire, ezért meg kell vizsgálni, hogy az egyes struktúrák milyen matematikai módszerekkel írhatók le.

3.3.1. Fabejárési módszerek

A következőkben a hagyományos tervezésmódszertani eljárások során használatos fabejárési módszerek kerülnek bemutatásra.

A morfológiai mátrix az egyes részfunkciókhoz tartozó megoldáselveket (*pl.: egy adott feladat megoldható pneumatikusan, de hidraulikusan is*) foglalja magába, a 15. ábra szerinti módon. A mátrix alapján az egyes részfeladatokra egy megoldáselvet kell kiválasztani. Az egyes részfunkciókat megvalósító megoldáselvekhez tartozó számokból álló számsor az egyes megoldásváltozatokat jelenti.

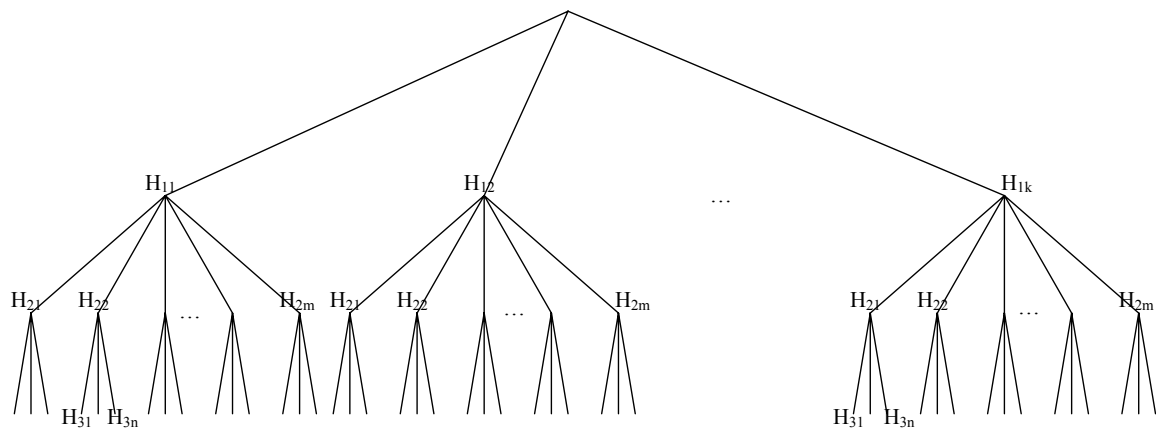
Részhatások	A részhatások hatásváltozatai				
$d_1 = \sum_{i=1}^k H_i$	H ₁₁	H ₁₂	H ₁₃	...	H _{1k}
$d_2 = \sum_{i=1}^m H_i$	H ₂₁	H ₂₂	H ₂₃		H _{2m}
$d_3 = \sum_{i=1}^n H_i$	H ₃₁	H ₃₂	H ₃₃		H _{3n}
...					
$d_t = \sum_{i=1}^z H_i$	H _{t1}	H _{t2}	H _{t3}		H _{tz}

15. ábra. Morfológiai mátrix

A morfológiai mátrix segítségével összekapcsolt megoldásváltozatok kiválasztására BIRKHOFER [17] tett javaslatot, a mátrix alapján kidolgozható megoldások, úgynevezett megoldásfával történő leírásával. A megoldásfa minden szintjén más-más részfunkció megoldásváltozatai kapcsolódnak az előző szint részfunkcióihoz (16. ábra). A megoldásfa megmutatja egy adott feladat összes, ily módon előállítható megoldását. A kiinduló szögpontról, azaz a fa gyökeréből elindulva, a fa bármely következő élét választva valamely végélhez lehet jutni. Az így bejárt útvonalak más-más megoldást adnak.

A fa olyan összefüggő gráf, melyben nincs kör, bármely két szögpontját egyetlen út köti össze. A körnélküli gráf minden komponense fa, ezért a teljes gráfot erdőnek is szokták nevezni. A gyökérből, vagy kezdő élből induló utak szétválnak. Minden ág egy végélben és annak végpontjában végződik, melyből már nem indul több él. A legegyszerűbb fa egyetlen élből, és 2 szögpontról áll.

Egy k számú összefüggő komponensből álló erdő esetén, a komponensek mindegyike fa. Minden fa-komponensben eggyel kevesebb él van, mint amennyi a szögponatok száma. A k számú komponensből álló n szögponatú erdőben $p-k$ számú él van.



16. ábra. Megoldásfa

A mátrix úgy működik, hogy az egyes sorokban található megoldási lehetőségek közül mindig ki kell választani egyet, és ahhoz kell kapcsolni a következő sorban lévő kiválasztott elemet. Így különböző hatásláncok adódnak. Ha a mátrix egy részfeladatot foglal össze, akkor a kiválasztott hatáslánc egy másik részfeladat morfológiai mátrixából adódó hatáslánccal összekapcsolható.

A mátrix elemeinek kombinálása számos hatásláncot eredményezhet. A 15. ábra szerinti példát alapul véve az összes részhatások száma a

$$d = \sum_{i=1}^n H_i \quad (6)$$

szerint határozható meg. Az összefüggés megmutatja, hogy a morfológiai mátrix egyes sorai-ban hány hatásváltozat található. A hatásváltozatok számának meghatározása után össze kell szorozni azokat, melyből a hatáslánc változatok száma adódik:

$$V_h = \prod_{t=1}^m d_t = \prod_{t=1}^m \left[\sum_{i=1}^n H_i \right]_t \quad (7)$$

A fenti összefüggésből látható, hogy egy néhány hatásból álló részfeladat is igen nagyszámú hatáslánc változattal adható meg. Hiszen maga a morfológiai mátrix eleve felteszi, hogy a hatáslánc a hatások mindegyikét tartalmazza. Abban az esetben, amikor a hatáslánc egyes hatásaihoz tartozó hatásváltozatok száma megegyezik ($d_1 = d_2 = \dots = d_s = d$), a hatáslánc változatok számát adó egyenlet a következő képpen módosul:

$$V_h = d^i \quad (8)$$

A megoldásfa szintjein az egyes szintekhez tartozó hatáslánc változatok száma ekkor a következő összefüggéssel határozható meg:

$$V_{sz} = d^{sz} \quad (9)$$

A változatok számának csökkentésére, vagyis a megfelelő megoldás kiválasztására szolgál a BIRKHOFFER [17] által javasolt megoldásfa. BIRKHOFFER elmélete szerint minden egyes részfel-

adatra megfogalmazható egy morfológiai mátrix, mely alapján egy-egy megoldásfa értelmezhető. Részfeladatonként tehető kikötés arra vonatkozóan, hogy mely hatást érdemes végül majd a teljes koncepcióban figyelembe venni. A hatáslánc változatok száma így módon csökkenthető.

3.3.2. Teljes kombinálási módszerek

Egy új gyógyszer felfedezéséhez több tízezer új vegyületet kell előállítani és megvizsgálni. Azonban egy közel egy évtizedes múlttal rendelkező, magyar tudományos eredmény lehetővé tette több millió új vegyület előállítását néhány nap alatt, mindemellett egy új tudományág, a kombinatorikus kémia kialakulását is eredményezte [46], [47]. Az új eljárással egy hét alatt több mint hatmillió vegyületet lehet előállítani, melyek közül elemzéssel választják ki a szükséges variánsokat.

Ez a tudományos eredmény rávilágít a hagyományos módszeres tervezésnek arra a nagy hiányosságára, hogy a különböző módszerekkel számos megoldás ugyan előállítható, azonban ezekkel a módszerekkel nem állítható elő minden képezhető változat az adott termék-elemek/funkciók alapján. Minden funkció minden funkcióval történő variálása természetesen számos használhatatlan, működésképtelen megoldást nyújtana, de ezek különféle értékelő eljárások segítségével elvethetők.

Néhány korábbi saját tanulmány [TA 7], [TA 4] már foglalkozott a teljes variálás lehetőségével. Az eljárás az ismétléses variáció, valamint a bináris logika elvén alapul, vagyis az adott funkciók mindegyikét mindegyikkel variálva az összes lehetséges funkcióstruktúra legenerálható.

A bináris logika, a 0 és az 1 érték, vagyis két érték közül választ. A Bináris Struktúragenerálás éppen ezt használja ki, hiszen azt vizsgálja, hogy az egyes megoldásokat jelentő funkcióhalmazokban benne van-e az adott funkció (*benne van=1*), vagy nincs benne (*nincs benne=0*). Az ismétléses variáció, mint matematikai fogalom azt fejezi ki, hogy n elemből, ha az elemek ismétlődhetnek, hányféle módon képezhető egy k tagú sorozat. Ha ezt a korábbi bináris alapú logikával összevetjük, akkor a struktúragenerálás esetében az n számú elem a bináris logikai igennek és nemnek felel meg. A sorozat tagszámát a funkciók száma adja meg. Tehát az ismétléses variációra felírható

$$V_n^{k,i} = a^k \quad (10)$$

egyenlet szerinti általános összefüggés a

$$V_m = 2^f \quad (11)$$

egyenlet szerint alakítható.

4. táblázat. Struktúra mátrix (S)

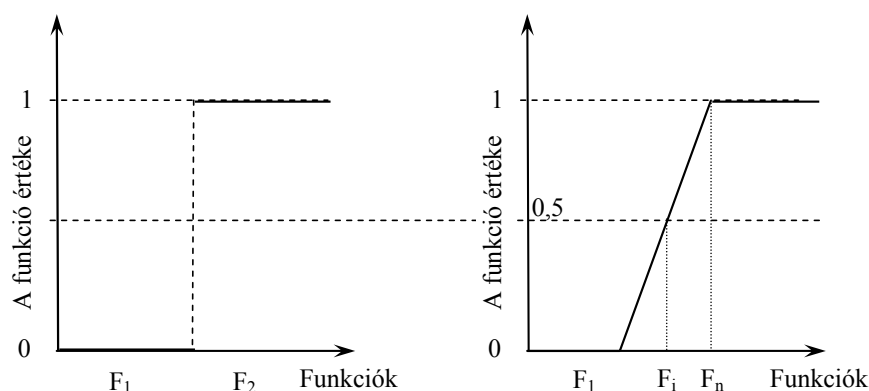
f_1	f_2	f_3	...	f_{n-1}	f_n
0	0	0		0	0
0	0	0		0	1
0	0	0		1	0
0	0	0		1	1
...					
1	1	1		1	1

A (11) egyenlet azt mutatja meg, hogy f funktiószám esetén hány funktióstruktúra generálható a teljes variáció szerint. Az 4. táblázat szerinti struktúra mátrix n számú funktió esetén az egyes funktióknak a funktióstruktúrát alkotó elemhalmazba való tartozását szemlélteti. Belátható, hogy már nagyon kevés funktiószám esetén is olyan nagyszámú megoldás állítható elő, hogy a kombinatorikus robbanás kiküszöbölése csak a számítógép segítségével lehetséges. A korábbi tanulmány során egy számítógépi program is készült, mely az itt leírt elméletre épült [TA 8]. A szoftver a rendelkezésre álló funktiókból egy felhasználó által formális nyelven előírt szabálykészletnek megfelelő összes lehetséges megoldást generálta. A szabálykészlet azt írta le, hogy egy adott funktió melyik funktióval nem kapcsolódhat. A tanulmány bebizonyította, hogy a kombinatorikus robbanás határai számítógépes segítséggel jelentős mértékben kitolhatók, vagy akár teljesen ki is küszöbölhetők. Bebizonyosodott, hogy a számítógéppel segített funktióstruktúra generálás során a figyelembe vett lehetőségek közül a legjobb megoldás választható ki az adott feladat megoldására.

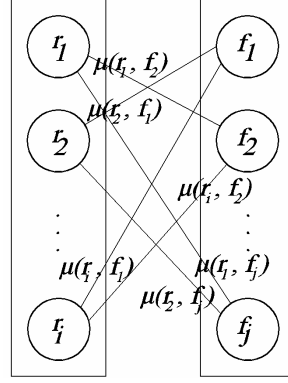
3.3.3. Fuzzy szemlélet alkalmazása termékcsaládok leírására

A bináris logika alapjait megfogalmazó Arisztotelész azt vallotta, hogy minden dolog vagy jó, vagy rossz. Filozófiája annyira jónak tűnt, hogy a klasszikus természettudomány egyik alappillérének tekinthetjük. A Fuzzy logika ezzel szemben egy ellentmondásokra épülő logikai szemlélet, alapjait a buddhista vallásban találhatjuk meg. Buddha filozófiája szerint minden dolog lehet egyszerre jó és rossz is. A műszaki alkotások fejlődése, a múlt században elindult rohamos léptekben fejlődő számítástechnika, a gondolkodni tudó gép kifejlesztése utáni vágy végül is kikényszerítette a Fuzzy logika kidolgozását.

A 17. ábra a bináris és a fuzzy logika közti különbséget szemlélteti. A bináris szemlélet diszkrét diagramját tekintve, egy funktió vagy részt vesz a megoldások képzésében, vagy nem. Ezzel szemben a fuzzy szemlélet sokkal több lehetőséget rejt oly módon, hogy a termékfunktciók mellé konkrét, 0 és 1 közé eső értékeket lehet rendelni, valamint meg kell határozni egy küszöbértéket (például a 0,5). Ekkor az ezen szint feletti értékkel rendelkező funktiók benne vannak a megoldást képező funktióhalmazban, míg a többi funktió nincs.



17. ábra. A bináris logika, valamint a fuzzy logika értéktartománya [35]



18. ábra. Kívánság-funkció nyíldiagram

DECIU, OSTROSI és FERNEY [35] korábban már említett módszere a konfiguratív termékcsaládok leírására, négy fuzzy modellből épül fel. Az általuk megfogalmazott teória teljes mértékben felhasználó orientált, emiatt a termékfunkciók F_t , valamint a felhasználói igények U halmaza csupán általánosan írható le. Ha a tervező ismeri a felhasználói igények U halmaza, valamint a termékfunkciók F_t halmaza közti $\tilde{R}_1$ fuzzy kapcsolatot, valamint a felhasználói igények U' fuzzy halmazát, a termékfunkciók F_t' fuzzy halmaza meghatározható. Ekkor egy új halmaz (F_t') jön létre, mely a felhasználói igények U' halmazának megfelelő funkciókat tartalmazza. A felhasználói igények U' fuzzy halmaza megmutatja, hogy a felhasználók által definiált egyes követelményeket a tervező mérnök hogyan értékeli, mennyire tartja fontosnak. Ezután a terméktervező meghatározza a felhasználói igények U' fuzzy halmaza és a termékfunkciók lehetséges F_t' fuzzy halmaza közti kapcsolatot az U' fuzzy halmaz elemei, vagyis a felhasználói követelmények alapján. A matematikai max-min művelet segítségével felírható a

$$F_t' = U' \circ \tilde{R}_1 \quad (12)$$

egyenlet. A $\mu_{F_t'}(f)$ tagsági függvény (12) egyenlet alapján

$$\mu_{F_t'}(f) = \mu_{U' \circ \tilde{R}_1}(f) = \max \min[\mu_{U'}(r), \mu_{\tilde{R}_1}(r, f)], \forall r \in U, \forall f \in F \quad (13)$$

egyenlet szerint írható fel, amit a 18. ábra szemléltet.

4. FUNKCIÓSTRUKTÚRÁK GENERÁLÁSÁNAK MÓDSZEREI

A következő fejezetrészekben egy olyan eljárás elvi alapjai kerülnek bemutatásra, mely a módszeres tervezés eszközszerével kombinálva nagy haszonnal alkalmazható a koncepcióképzés, vagyis az optimális termékváltozat alapjainak kidolgozása során, és emellett figyelembe veszi a felhasználói követelményeket is. Mindenek előtt tisztázni kell néhány fogalmat, melyek korábban már említésre kerültek, azonban a következő fejezetekben valamelyest módosított jelentéstartalommal bírnak majd.

Koncepcionális tervezés: a teljes tervezési folyamatnak (*műszaki életútnak*) az a szakasza, amely során a tervező mérnök az adott tervezési feladattal kapcsolatos ötleteit rendszerezi. Ez a tervezési fázis nem része a konstrukciós tervezésnek. Itt még nem rendelkezik sem formával, sem nagysággal, sem anyaggal a tervezendő műszaki alkotás. A koncepció (*termékötlet, megoldásváltozat*) csupán arra vonatkozó információt hordoz, hogy milyen főbb részegységekből épül fel, és azok között milyen logikai kapcsolatrendszer van.

Koncepció: termékötlet. A koncepcionális tervezési fázis végeredménye.

Funkcionális részegység: a termék egyes „alkatrészei”, melyek valamely fontosabb részfeladatot látnak el. A funkcionális részegység fizikai tartalommal nem bír, hozzájuk hatáselv a tervezésnek ebben a szakaszában még nem rendelhető.

Termékstruktúra: a funkcionális részegységekből alkotott egyes megoldásváltozatokat jelent.

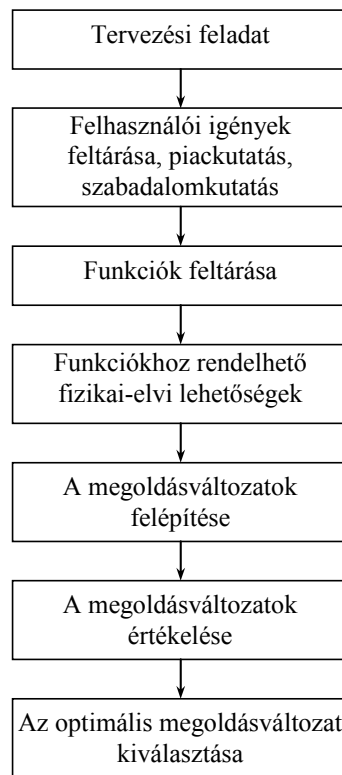
4.1. MÓDSZERTANI KRITIKÁK

A meglévő módszerek nagy hátránya az, hogy számítógépes adaptálásuk csak korlátokkal valósítható meg. A jelenlegi módszerek 19. ábra szerinti szemléltetése egy lényegesen leegyszerűsített közelítés. A hagyományos módszerek az egyes lépéseknél számos javaslatot tesznek az adott lépés megvalósítására, ami magával vonja az eljárások összetettségét. Az egyik feladat tehát az, hogy a hagyományos módszerek helyett, azok pozitív tulajdonságait felhasználva egy olyan tervezői folyamatot kell javasolni, mely pontosan megadott lépéseket követ.

A tradicionális módszerek másik nagy hátránya, hogy a különböző hatáselveket (*pl.: fizikai, kémiai, biológiai*) tartják szem előtt. Ez azért hátrányos, mivel így a tervező már a koncepcióképzés fázisában korlátozhatja a megoldási lehetőségeket. Ily módon, akaratán kívül kizárhat olyan megoldásokat, melyek a későbbi szelektálási szakasz során azért lehetnének értékesek, mert újszerű, vagy teljesen új megoldásokat eredményeznének. A másik feladat tehát az, hogy a megoldásváltozatokat a fizikai elvek figyelembevételével kell előállítani.

A meglévő módszerek harmadik –és talán legnagyobb– hiányossága az, hogy nem próbálják meg előállítani az összes lehetséges megoldást, ami a feltárt funkciók alapján előállítható volna, hiszen az emberi kapacitásokra –nem pedig a számítógépek által nyújtott lehetőségekre– épít. Ily módon fennáll annak a lehetősége, hogy néhány –vagy akár jelentős mennyiségű– el-

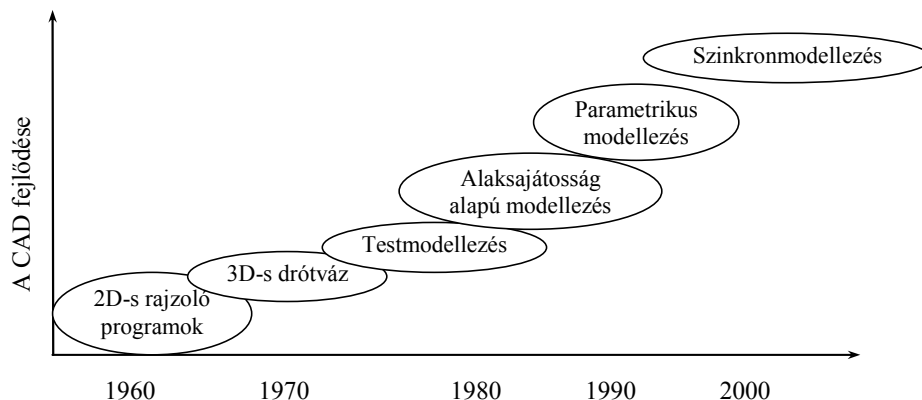
vileg helyes, esetleg teljesen új megoldás elkerüli a tervezőmérnök figyelmét. Ennek megfelelően a legnagyobb feladat, hogy megvalósuljon a feltárt funkcionális részegységekből felépíthető összes lehetséges megoldásváltozat előállíthatósága és kezelhetősége.



19. ábra. A koncepcionális tervezői fázis lépései, egyszerűsített folyamatábra

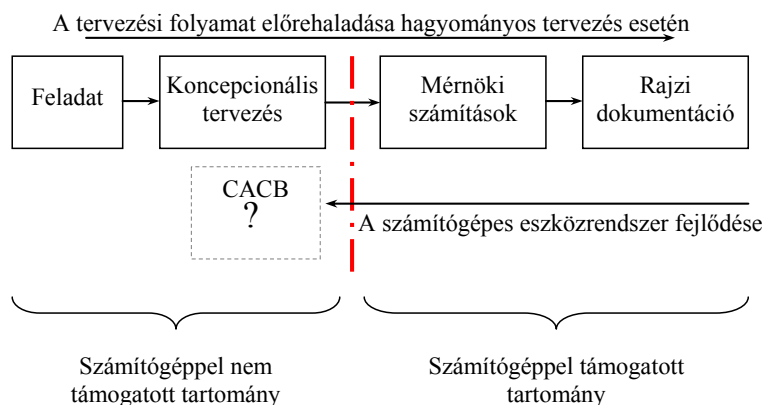
4.2. SZÁMÍTÓGÉPPEL SEGÍTETT MÓDSZERES KONCEPCIÓÉPÍTÉS (CACB)

A 3.1 fejezetben leszögezésre került, hogy a disszertáció a teljes tervezési folyamatnak csupán a koncepcionális részével foglalkozik. Ennek kapcsán tisztázni kell, melyek a tervezési folyamatnak azok a részfolyamatai, melyek segítésére már léteznek számítógépes alkalmazások.



20. ábra. A CAD fejlődése

A számítógéppel segített tervezés fejlődését a 20. ábra szemlélteti. A számítógéppel segített tervezés kialakulásának és fejlődésének története az 1950-es évek végéig nyúlik vissza; 1957-ben jelent meg az első CAD szoftver, a Sketchpad [41]. A korai CAD-s rendszerek csupán a számítógéppel segített rajzolást támogatták, a további modulok –*mint például a CAM, vagyis számítógéppel segített gyártás*– integrálása iránti igény csak később jelent meg. A CAD rendszerekbe integrált egyéb technológiákat, vagyis a CAxx technológiákat [60] a teljes tervezési folyamat vonatkozásában vizsgálva megállapítható, hogy a tervezési folyamat számítógépes támogatása a tervezési folyamat előrehaladásával ellentétes irányban fejlődött (21. ábra). A tervezés végső fázisának tekinthető dokumentációs szakaszban jelentek meg először a számítógépes technológiák, míg a tervezés alapját képező koncepcionális fázis számítógépes támogatása és annak automatizálása még napjainkban sem teljes körűen megoldott.



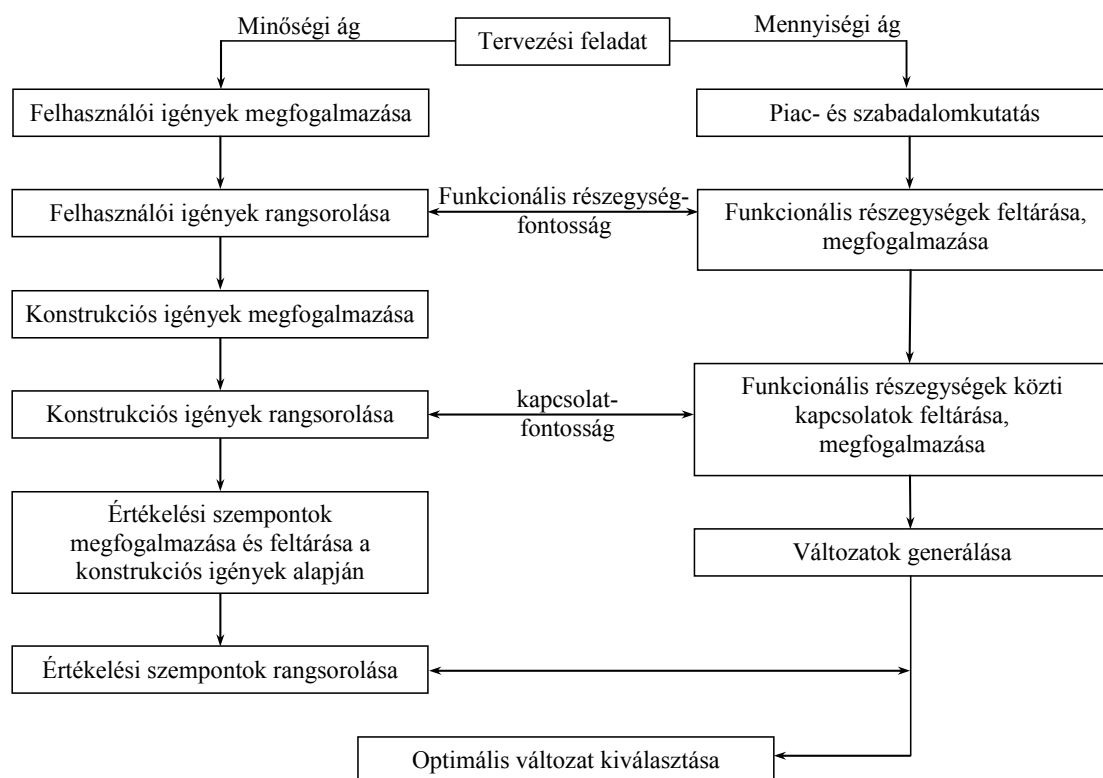
21. ábra. CACB (Computer Aided methodological Concept Building), javaslat a számítógépes eszközrendszer kiterjesztésére

Ahhoz, hogy a hagyományos értelemben vett koncepcionális tervezési szakasz számítógéppel a megfelelő mértékben támogatható legyen, olyan módszereket kell kidolgozni, melyek a hagyományos –*számítógéppel nem támogatott*– tervezésmódszertan előnyeit felhasználva számítógépre adaptálhatók és segítségükkel az optimális termékkonceptió kidolgozható.

A 2.3.2. fejezetben bemutatott hagyományos tervezési modellek elemzése, valamint a számítógépes lehetőségek tanulmányozása révén a disszertáció egy újszerű módszerre tesz javaslatot. A számítógéppel segített módszeres koncepcióépítés (*Computer Aided methodological Concept Building, vagyis CACB*) javasolt folyamatát a 22. ábra mutatja. Látható, hogy a koncepcionális tervezés két, egymással párhuzamos szálon fut. Az egyik a mennyiségi ág, amely elnevezés a folyamat során keletkező megoldásváltozatok számára, sokaságára utal. A minőségi ágon a minőségi tényezők, vagyis az igények összegyűjtése történik meg, melyek révén, a mennyiségi ágon keletkező megoldástér, megoldáshalmaz szűkíthető.

A tervezési feladat kidolgozása előtt a feladatot elemezni kell. Ennek eszközei a piackutatás, valamint a már szabadalmaztatott megoldások feltárása, elemzése, értékelése. Ezzel párhuzamosan azonban fel kell tární, és meg kell fogalmazni a felhasználói igényeket. A termékkel szemben a felhasználók által támasztott elvárásokat tervezői szemmel értékelni, rangsorolni kell, hiszen a koncepcióképzési folyamat végén előállított megoldások értékelemzési szempontjainak alapját ezek a kritériumok nyújtják. A piac- és szabadalomkutatás során fel kell tární a lehetséges funkcionális részegységeket. Ezekből generálhatók a termékstruktúrák, vagyis a megoldásváltozatok. A megoldásváltozatokat a terméktervező mérnök műszaki érték-

vizsgálatnak veti alá. Az értékelés során az értékelési szempontoknak legjobban megfelelő megoldás képezi a koncepcióképzési feladat megoldását.



22. ábra. A számítógéppel segített módszeres koncepcióépítés javasolt logikai lépései

4.2.1. Felhasználói igények

A felhasználói igények, valamint a termékkel szemben támasztott követelmények a tervezési munka kezdetén általában rendelkezésre állnak követelmény jegyzék formájában [35]. A követelmény jegyzék alapján a kritériumok összegyűjthetők, illetve tervezői szemmel rangsorolhatók. Ez a rangsorolás a fuzzy logika alapjaira épül, de felhasználja az értékelő szempontok sorba rendezésére a hagyományos módszeres tervezés során javasolt páronként történő összehasonlítási módszert is.

A kritériumok sorba rendezését a

$$\begin{matrix} & K_1 & K_2 & K_3 & \dots & K_n & \Sigma \\ \begin{matrix} K_1 \\ K_2 \\ K_3 \\ \dots \\ K_n \end{matrix} & \begin{bmatrix} 0 & k_{12} & k_{13} & \dots & k_{1n} \\ k_{21} & 0 & k_{23} & \dots & k_{2n} \\ k_{31} & k_{32} & 0 & \dots & k_{3n} \\ \dots & \dots & \dots & 0 & \dots \\ k_{n1} & k_{n2} & k_{n3} & \dots & 0 \end{bmatrix} & \begin{matrix} k_{1f} \\ k_{2f} \\ k_{3f} \\ \dots \\ k_{nf} \end{matrix} \end{matrix} \quad (14)$$

egyenlet egy, a

$$K := (k_{ij})_{n \times n} \quad (15)$$

egyenlettel is kifejezhető **K** kritérium mátrix-szal szemlélteti. A kritériumokat páronként kell egymással összehasonlítani. Azt kell megvizsgálni, hogy a sorokban lévő kritériumoktól az oszlopokban lévő kritériumok fontosabbak-e. A (14) és (15) szerinti mátrix k_{ij} elemei csak a 0 és az 1 értékeket vehetik fel. A mátrix alsó háromszöge a felső háromszögnek kiegészítője

$$1 - k_{ij} = k_{ji} \quad (16)$$

A kritérium mátrix fontos tulajdonságai:

- főátlója minden esetben csak nullákat tartalmazhat,
- a főátló feletti elemeket, ha hozzáadjuk a főátló alatti megfelelő elemekhez, 1-et kapunk (16).

Ennek megfelelően elegendő a főátló feletti elemeket megadni, a főátló alattiak kiszámíthatók. Ily módon egy hasonló négyzetes mátrix meghatározásához szükséges elemek száma egy matematikai sorral is kifejezhető:

$$z = n \cdot (n-1) - [(n-1) \cdot (n-1)] + [(n-2) \cdot (n-2)] - \dots \\ \dots + [(n-i) \cdot (n-i)] - \dots + \dots [(n-n) \cdot (n-n)] \quad (17)$$

A kritériumok páronkénti összehasonlításainak eredményét soronként összegezve adódik az egyes kritériumok fontossági mértéke.

$$k_{if} = \sum_{j=1}^n k_{ij} \quad (18)$$

Ily módon egy $\underline{k}_f$ kritérium fontossági vektor adódik

$$\underline{k}_f = [k_{1f} \quad k_{2f} \quad k_{3f} \quad \dots \quad k_{nf}] \quad (19)$$

amit az értékelő szempontok meghatározásánál figyelembe kell venni.

4.2.2. Megoldásváltozatok generálása

A szabadalomkutatás, valamint a piackutatás során fel kell tárni az ismert megoldásokban megtalálható funkcionális részegységeket, melyek a megoldások felépítése szempontjából fontosak. Adott esetben intuitív módon új részegységeket is meg kell fogalmazni. A változatok generálása többféle módon történhet –*például a bináris logika szerint, vagy véletlenszerű generálással*. Abban az esetben, ha a tervezőmérnök a funkcionális részegységek elemzése révén egy előzetes szelektálást kíván végrehajtani, a (14) és (15) egyenletek szerinti sorba rendezést kell megismételni a funkcionális részegységekre is annak megfelelően, hogy mennyire fontosak a termékkoncepcióban:

$$\begin{array}{cccccc} F_1 & F_2 & F_3 & \dots & F_n & \Sigma \\ \begin{array}{l} F_1 \\ F_2 \\ F_3 \\ \dots \\ F_n \end{array} & \left[\begin{array}{ccccc} 0 & f_{12} & f_{13} & \dots & f_{1n} \\ f_{21} & 0 & f_{23} & \dots & f_{2n} \\ f_{31} & f_{32} & 0 & \dots & f_{3n} \\ \dots & \dots & \dots & 0 & \dots \\ f_{n1} & f_{n2} & f_{n3} & \dots & 0 \end{array} \right] & \begin{array}{l} f_{1f} \\ f_{2f} \\ f_{3f} \\ \dots \\ f_{nf} \end{array} \end{array} \quad (20)$$

$$F := (f_{ij})_{n \times n} \quad (21)$$

$$1 - f_{ij} = f_{ji} \quad (22)$$

A sorokban lévő összegzések révén kialakul a funkcionális részegységeknek egy fontossági sorrendje:

$$f_{if} = \sum_{j=1}^n f_{ij} \quad (23)$$

amit a funkcionális részegységek fontossági vektora tartalmaz:

$$\underline{f}_f = [f_{1f} \quad f_{2f} \quad f_{3f} \quad \dots \quad f_{nf}] \quad (24)$$

Ekkor, az egyes funkcionális részegységek rangját képező értéket, a 0 és 1 közé eső nyitott intervallumon kell megadni, vagyis ekkor minden funkcionális részegység fuzzy értékkel bír.

Ezután a tervező egy 0 és 1 közé eső érték megadásával meghatározza a megoldásszintet is. A megoldásszint meghatározása azt jelenti, hogy egy adott számértéknél kisebb rangú funkcionális részegységeket ki kell venni a megoldásokat felépítő részegységek halmazából. Így a tervező számos megoldás-lehetőséget zár ki, viszont a műszaki szempontból legfontosabbnak tekintett funkcionális részegységek az értékes megoldások képzését segítik elő. Minél közelebb van a 0-hoz az adott számértékkel jelölt megoldásszint, annál több funkcionális részegységből épül fel a generált megoldás. A megoldásszint számértékkel történő kijelölése tehát nem a generálható megoldások minőségére utal, sokkal inkább egy mennyiségi szelekciót eredményez, ami a generálás időtartamát csökkentheti. A megoldásszint növelésével a termékbonyolultság csökken, így a ráfordítási költségek is alacsonyabbak lehetnek. Azonban az nem biztos, hogy így a legfontosabb felhasználói kritériumok is kielégülnek.

Ezután a változatokat képező funkcionális részegységek közti kapcsolatokat kell definiálni, vagyis szabályozni kell, hogy melyik részegység melyik részegységhez kapcsolódhat. Ekkor létrejön a C kapcsolati mátrix:

$$\begin{matrix} & C_1 & C_2 & C_3 & \dots & C_n \\ \begin{matrix} C_1 \\ C_2 \\ C_3 \\ \dots \\ C_n \end{matrix} & \begin{bmatrix} 0 & c_{12} & c_{13} & \dots & c_{1n} \\ c_{21} & 0 & c_{23} & \dots & c_{2n} \\ c_{31} & c_{32} & 0 & \dots & c_{3n} \\ \dots & \dots & \dots & 0 & \dots \\ c_{n1} & c_{n2} & c_{n3} & \dots & 0 \end{bmatrix} \end{matrix} \quad (25)$$

$$C := (c_{ij})_{n \times n} \quad (26)$$

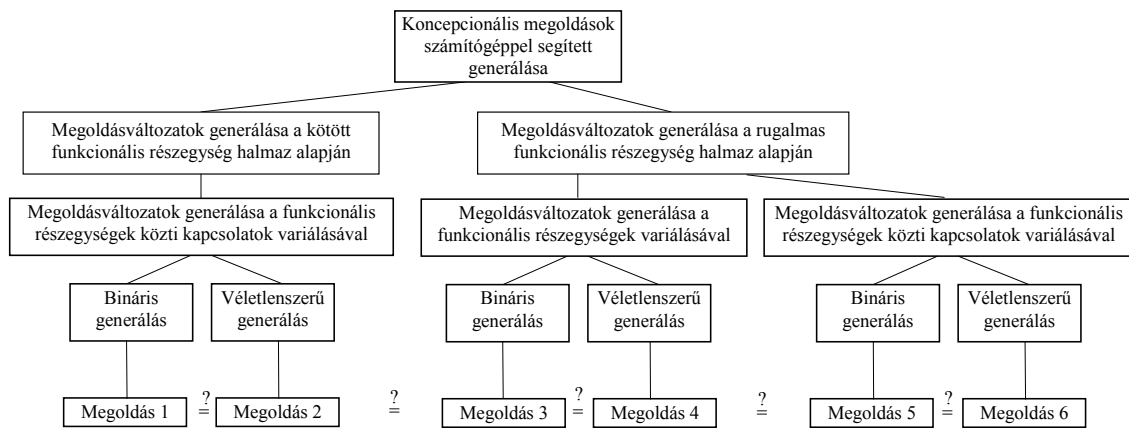
A mátrix elemei a 0, 1 és x karakterek lehetnek. Ezek írják le, hogy a struktúrában az adott részegységek közt van-e kapcsolat, vagy nincs. Ha a kapcsolat az x karakterrel van megfogalmazva, akkor a generálás során a program dönti el, hogy kapcsolat részt vesz-e a struktúrában. Ily módon a generálás során minden egyes megoldásstruktúrához különböző kapcsolati mátrix tartozik, tehát mindegy egyes struktúra leírható egy S struktúra mátrixszal

$$\begin{matrix} & S_1 & S_2 & S_3 & \dots & S_n \\ \begin{matrix} S_1 \\ S_2 \\ S_3 \\ \dots \\ S_n \end{matrix} & \begin{bmatrix} 0 & s_{12} & s_{13} & \dots & s_{1n} \\ s_{21} & 0 & s_{23} & \dots & s_{2n} \\ s_{31} & s_{32} & 0 & \dots & s_{3n} \\ \dots & \dots & \dots & 0 & \dots \\ s_{n1} & s_{n2} & s_{n3} & \dots & 0 \end{bmatrix} \end{matrix} \quad (27)$$

$$S := (s_{ij})_{n \times n} \quad (28)$$

amely mátrix már teljesen bináris felépítésű, vagyis csak 0 és 1 elemeket tartalmaz. Minden egyes struktúra jellemezhető továbbá egy-egy struktúra gráffal, melyek az egyes koncepciók különböző funkcionális részegységeinek egymás közti kapcsolatait határozzák meg. A gráfok csomópontjai a funkcionális részegységeket, élei az egyes részegységek közti kapcsolatokat jelentik. A struktúrák struktúra egyenletekkel is meghatározhatók, melyek a struktúra gráfok éleit határozzák meg, vagyis azokat a kapcsolatokat írják le, melyek az egyes megoldásváltozatok struktúráiban létező kapcsolatként vesznek részt. A struktúra egyenletek felépítése egyszerű: a struktúra mátrixban 1-el jelölt funkcionális részegységek közti kapcsolatokat tartalmazza. Erre egy példa:

$$F_1 - F_2, F_1 - F_i, F_j - F_i, F_j - F_n \quad (29)$$



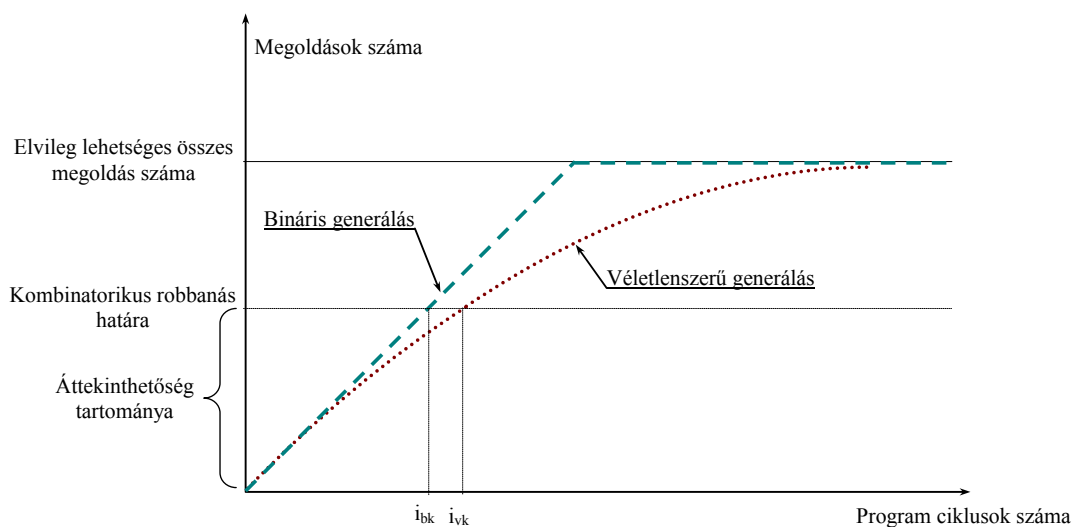
23. ábra. Megoldásváltozatok generálása

A változatok generálása alapvetően kétféle módon történhet, ahogyan azt a 23. ábra is szemlélteti; a „*kötött funkcionális részegység halmaz*”, valamint a „*rugalmas funkcionális részegység halmaz*” alapján. Vannak esetek, amikor a funkcionális részegységek fontosságának megadása mellőzhető. Ekkor a funkcionális részegységek halmaza nem változik, vagyis a generálás az úgynevezett „*kötött funkcionális részegység halmaz*” alapján történik. Ha a funkcionális részegységek halmazát csökkentjük a funkcionális részegységek fontosságának megadásával, valamint a megoldásszint meghatározásával, a funkcionális részegységek halmazának mérete megváltozik, vagyis a generálás a „*rugalmas funkcionális részegység halmaz*” alapján történik. A két módszer optimális alkalmazási területének feltárása és hatékonyság elemzése további vizsgálatot igényel. Vannak olyan termékek –*például egy gépkocsi visszapillantó tükrök*–, melyek esetében, ha meghatározunk egy alacsony megoldásszintet, akkor minden funkcionális részegység benne lehet a termékstruktúrában, viszont egy magasabb megoldásszint esetében csupán az alapvető működést nem befolyásoló funkcionális részegységek hiányoznak. Ilyen termékek esetében a megoldásszint szigorúsága nem befolyásolja a termék működését. Vannak azonban olyan termékek is –*például különféle típusú bolygóművek összekapcsolásából származó megoldás változatok*–, ahol a működőképes változatok generálásához minden egyes funkcionális részegységre szükség van. Ezért nevezhető a funkcionális részegységek közti kapcsolatok variálása szerint történő megoldás generálás a funkcionális részegységek totális kombinálásának, hiszen ha a funkcionális részegységek közti kapcsolatok előzetesen nem kerülnek definiálásra, akkor az összes lehetséges megoldás előállítható a bináris

generálás révén. A véletlenszerű generálással nem biztos, hogy előállítható az összes lehetséges megoldás.

A disszertációban kidolgozott módszer alapvetően kétféle elmélettel foglalkozik: a funkcionális részegységek variálása révén létrehozott megoldásváltozatokkal, illetve a funkcionális részegységek közti kapcsolatok variálásának elvével. A disszertáció mind a kétféle elméletre kétféle matematikai megoldást javasol: a bináris-, illetve a véletlenszerű generálás módszerét. Ahogyan azt az ábra is szemlélteti a különböző megoldás előállítási eljárások különböző eredményeket mutathatnak. A módszerek nem zárják ki annak lehetőségét, hogy az egyes eredmények adott esetben meg is egyezhetnek.

A véletlenszerű generálás esetén erre nincs szükség a funkcionális részegységek közti kapcsolatok definiálására, hiszen ebben az esetben pontosan az a lényeg, hogy a számítógép az összes lehetőség közül véletlenszerűen határoz meg megoldásokat. Ezeket azonban célszerű szigorú értékeléssel átvizsgálni, hogy működőképes megoldások-e. A bináris generálás során minden funkcionális részegység minden funkcionális részegységgel kapcsolódhat, ily módon a funkcionális részegységekből előállítható összes lehetséges megoldás biztosan előáll. Azonban a felhasználó le is köthet kapcsolatokat; ekkor kevesebb megoldása lesz a generálásnak. A különböző szemléletek szerint történő generálás különböző eredményeket hozhat. A módszerek hatékonyság-elemzése további vizsgálatot igényel.



24. ábra. A bináris és a véletlenszerű generálással elérhető megoldásváltozatok száma

A 24. ábra a bináris és a véletlenszerű generálással képzett megoldások várható darabszámát mutatja a generáló ciklusok függvényében a javasolt módszerek számítógépes támogatása mellett. Az ábrán látható, hogy a bináris generálással az összes lehetséges megoldás véges ciklusszám esetén is előállítható, míg a véletlenszerű generálás görbéje végtelenszámú programciklus esetén simul az elvileg lehetséges összes megoldás vízszintes vonalához. A megoldás-generálás bemenő adataitól függően előfordulhat, hogy a kombinatorikus robbanás határa alacsonyabb megoldásszámnál jelentkezik, mint az elvileg lehetséges összes megoldások száma. Ebben az esetben a megoldáshalmaz inhomogén lesz, mert a generálási tartomány kombinatorikus robbanás feletti megoldásai teljesen hiányozni fognak a megoldáshalmazból. Ebben az esetben kedvezőbb a véletlenszerű generálás alkalmazása, mert a kombinatorikus robbanás határáig előállított megoldások a teljes generálási tartományból kerülnek ki, ezért a

megoldáshalmaz homogén lesz. Azonos számú megoldás generálásakor i_{vk} mindig nagyobb, mint i_{bk} , tehát a véletlenszerű generálási módszer esetében mindig több programciklusra van szükség, mint a bináris módszernél.

4.2.3. Értékelési szakasz

Az értékelési szakaszban a generált megoldásváltozatokat kell értékelni, mely értékelés során egy E értékmátrix állítható elő:

$$E := (e_{ij})_{n \times n} \quad (30)$$

$$\begin{matrix} & E_1 & E_2 & E_3 & \dots & E_n & \Sigma \\ \begin{matrix} E_1 \\ E_2 \\ E_3 \\ \dots \\ E_n \end{matrix} & \begin{bmatrix} 0 & e_{12} & e_{13} & \dots & e_{1n} \\ e_{21} & 0 & e_{23} & \dots & e_{2n} \\ e_{31} & e_{32} & 0 & \dots & e_{3n} \\ \dots & \dots & \dots & 0 & \dots \\ e_{n1} & e_{n2} & e_{n3} & \dots & 0 \end{bmatrix} & \begin{matrix} e_{1f} \\ e_{2f} \\ e_{3f} \\ \dots \\ e_{nf} \end{matrix} \end{matrix} \quad (31)$$

$$1 - e_{ij} = e_{ji} \quad (32)$$

amely mátrix a felhasználói igények alapján kiválasztott értékelési szempontokat rangsorolja. Az értékelési szempontok páronkénti összehasonlításainak eredményét soronként összegezve az egyes kritériumok fontossági mértéke adódik:

$$e_{if} = \sum_{j=1}^n e_{ij} \quad (33)$$

melyeket vektorba írva az értékelő szempontok fontossági vektora állítható elő:

$$\underline{e}_f = [e_{1f} \quad e_{2f} \quad e_{3f} \quad \dots \quad e_{nf}] \quad (34)$$

5. táblázat. Súlyozott terméktulajdonságok módszere

	Sz ₁	Sz ₂	Sz ₃	...	Sz _m	Σ	q
V ₁	P _{min} ÷p _{max}	P _{min} ÷p _{max}	P _{min} ÷p _{max}		P _{min} ÷p _{max}		q ₁
sV ₁	s ₁ V ₁	s ₂ V ₁	s ₃ V ₁		s _n V ₁	ΣsV ₁	
V ₂	P _{min} ÷p _{max}	P _{min} ÷p _{max}	P _{min} ÷p _{max}		P _{min} ÷p _{max}		q ₂
sV ₂	s ₁ V ₂	s ₂ V ₂	s ₃ V ₂		s _n V ₂	ΣsV ₂	
V ₃	P _{min} ÷p _{max}	P _{min} ÷p _{max}	P _{min} ÷p _{max}		P _{min} ÷p _{max}		q ₃
sV ₃	s ₁ V ₃	s ₂ V ₃	s ₃ V ₃		s _n V ₃	ΣsV ₃	
...							
V _n	P _{min} ÷p _{max}	P _{min} ÷p _{max}	P _{min} ÷p _{max}		P _{min} ÷p _{max}		q _n
sV _n	s ₁ V _n	s ₂ V _n	s ₃ V _n		s _n V _n	ΣsV _n	

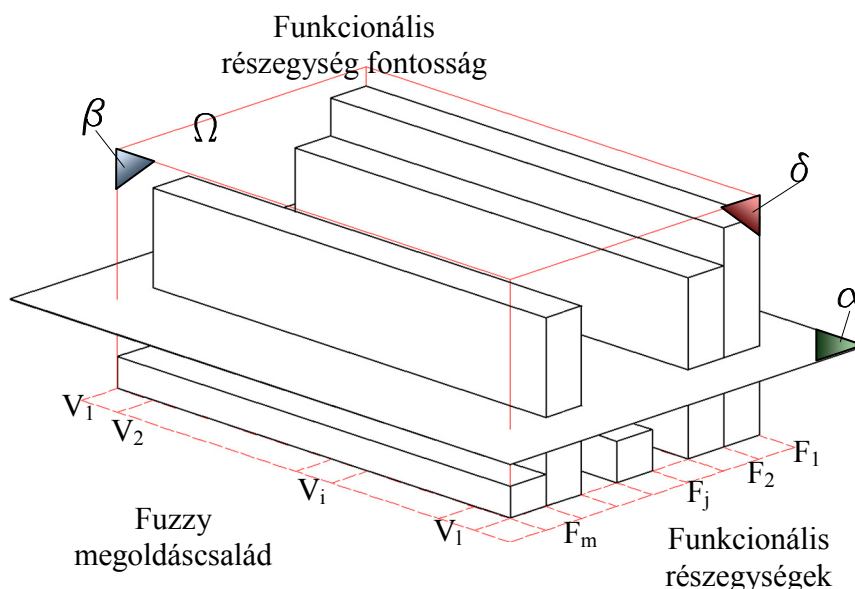
A rangsorolt értékelési szempontokat rangjuknak megfelelően súlyozó tényezővel kell ellátni. Ezután egy, a súlyozott terméktulajdonságok módszerét alkalmazva, egy 5. táblázat szerinti táblázatot kell elkészíteni. Az egyes változatokat 1÷5 (*a legrosszabb az 1-es, a legjobb az 5-ös, p_{min}÷p_{max}*) pontszámokkal kell értékelni, majd pedig az értékelési szempontok súlyozó té-

nyezőivel kell megszorozni. Ezután a súlyozott értékeket soronként összegezni kell, így kialakul az értékelt megoldások $\underline{q}$ vektorral jelölt sorrendje, amely a táblázat utolsó oszlopában található. A $\underline{q}$ vektor legmagasabb értékkel rendelkező q_i eleme adja meg az optimális megoldásváltozatot:

$$V_{opt} = \max \underline{q} \quad (35)$$

4.3. FUZZY MEGOLDÁSCSALÁD ÁBRÁZOLÁSA HÁROMDIMENZIÓS TÉRBEN

A fuzzy megoldáscsalád azokból a megoldásváltozatokból áll, melyek a rugalmas funkcióhalmaz szerinti generálás során elvileg helyes változatoknak bizonyultak. Descartes-féle koordináta-rendszerben a megoldáscsalád V elemei, a megoldáscsaládok generálása során alkalmazott F funkcionális részegységek és az egyes funkcionális részegységek f fontossága által meghatározott tér az Ω fuzzy megoldástér (25. ábra). Az Ω megoldástér megadható egy B bonyolultsági mátrix-szal és egy $\underline{f}_f$ funkcionális részegység fontossági vektorral.



25. ábra. Funkcionális részegységek fontossága, az egyes megoldásváltozatok bonyolultsága

Az Ω megoldástérrel 3 kitüntetett síkkal metszhetjük el. A megoldásváltozatok és a funkcionális részegységek által meghatározott síkkal párhuzamos síkok bármelyike az α fuzzy termék-bonyolultsági sík, mely megmutatja, hogy a funkcionális részegység fontosság különféle szintjein az egyes megoldásokban mely funkcionális részegységek szerepelnek. Azon funkcionális részegységek, melyek funkcionális részegység fontossága magasabb az α termék-bonyolultsági szintnél, résztvesznek az egyes megoldásokban. A sík függőleges mozgásával a termékek bonyolultsági foka szabályozható. A termék-bonyolultság különféle (α) szintjei a

$$B := (b_{ij})_{n \times m} \quad (36)$$

$$\begin{matrix} & F_1 & F_2 & F_3 & \dots & F_m \\ \begin{matrix} V_1 \\ V_2 \\ V_3 \\ \dots \\ V_n \end{matrix} & \begin{bmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1m} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2m} \\ b_{31} & b_{32} & b_{33} & \dots & b_{3m} \\ \dots & \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & b_{n3} & \dots & b_{nm} \end{bmatrix} \end{matrix} \quad (37)$$

termékbonyolultsági mátrixszal jellemezhetők.

Az Ω megoldásteret a megoldásváltozatok tengelyére merőleges síkkal elmeteszve a β fuzzy funkcionális részegység fontossági sík adódik, amely megmutatja, hogy a fuzzy megoldáscsalád egyes megoldásait alkotó funkcionális részegységek milyen fontosságértékkel bírnak az egyes megoldásokban.

Az Ω megoldásteret a β síkkal elmeteszve „prizmatikus” jellegű, mivel minden megoldás esetén ugyanazt a képet mutatja. Az egyes megoldások közti különbségek a funkcionális részegységek közti különböző kapcsolatokból adódnak, melyeket az S struktúra mátrixok írnak le. A funkcionális részegységek fontosságának β síkja a (22) szerinti $\underline{f}_f$ funkcionális részegység fontossági vektorral jellemezhető.

Az Ω megoldásteret a funkcionális részegységek tengelyére merőleges síkkal elmeteszve a δ fuzzy funkcionális részegység előfordulás síkja adódik, amely megmutatja, hogy az egyes funkcionális részegységeket a fuzzy megoldáscsalád elemei tartalmazzák-e, és ha igen, azok milyen fontossággal bírnak. A δ sík a (22) szerinti $\underline{f}_f$ funkcionális részegység fontossági vektor valamely (24) szerinti elemével jellemezhető.

A bemutatott három sík egymáshoz viszonyított kapcsolatai a következőképpen fogalmazhatók meg:

- Az $\alpha - \beta$ síkok által definiált metszésvonalak meghatározzák, hogy a megoldáscsalád egy adott megoldása mely funkcionális részegységekből épül fel. A metszésvonal egy olyan vektorral definiálható, amelynek értékkészlete a 0 és az 1.
- Az $\alpha - \delta$ síkokkal párhuzamos síkok által meghatározott metszésvonalak a megoldáscsaládot képező megoldásokban előforduló adott funkcionális részegységek fontosságát mutatják meg. A metszésvonal a (22) szerinti $\underline{f}_f$ funkcionális részegység fontossági vektor valamely (24) szerinti eleme.
- A $\beta - \delta$ síkokkal párhuzamos síkok által meghatározott metszésvonalak a megoldáscsaládot képező megoldásokban előforduló adott funkcionális részegységek előfordulását mutatják meg bizonyos bonyolultsági fokon, vagyis α megoldásszinten. A metszés vonala egy olyan vektorral írható le, melynek minden eleme ugyanaz: vagy 0, vagy 1.
- A bemutatott három sík (a megoldás változatok bonyolultsági fokának síkja, a funkcionális részegység fontosság síkja, valamint a funkcionális részegység előfordulás síkja) metszéspontja a funkcionális részegység fontossági pont, mivel adott funkcionális részegység fontosságát határozza meg.

Az Ω megoldásteret jellemző tulajdonsága, hogy függőleges mérete állandó, maximális értéke az 1. Azonban az Ω megoldásteret mérete a megoldás generálás paramétereinek függvényében V és F irányban módosulhat, melyet a következő tényezők befolyásolnak:

- α fuzzy termékbonyolultsági szint változtatásával befolyásolni lehet a megoldás generálásban résztvevő funkcionális részegységek számát, mely kihat a megoldásváltozatok számára.
- a megoldásváltozatok száma attól is függ, hogy a kapcsolati mátrix milyen értékeket tartalmaz.

4.4. A JAVASOLT MÓDSZEREK SZÁMÍTÓGÉPI ADAPTÁLHATÓSÁGA

Meg kell jegyezni, hogy a bemutatott megoldásváltozat-generáló elvek kizárólag számítógépi alkalmazás mellett működhetnek hatékonyan. A módszerek arra tesznek javaslatot, hogy a feltárt funkcionális részegységekből hogyan lehet minél több lehetséges megoldást előállítani; az összes lehetséges megoldásváltozat előállítására azonban csak a funkcionális részegységek közti kapcsolatok bináris variálásának módszere alkalmas.

Megállapításra került, hogy ha túl sok a variálható kapcsolatok száma, célszerűbb a véletlenszerű generálást alkalmazni, hiszen így a teljes megoldástérről egy átfogó kép kapható. A bináris generálást akkor jó használni, ha az elvileg lehetséges összes megoldásváltozat előállítása várhatóan nem haladja meg az áttekintőképesség tartományát.

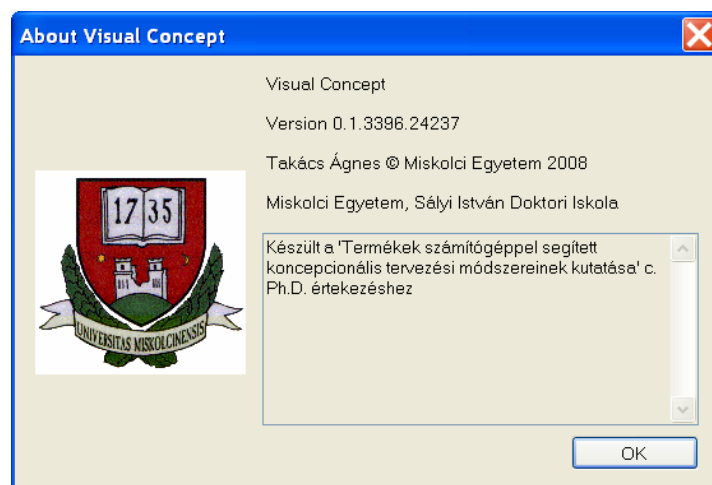
Összességében megállapítható, hogy a bemutatott módszerek lehetőséget adnak a módszeres tervezés algoritmizálására, a humán korlátok legyőzésére, a tervezési idő lerövidítésére, a terv várható minőségének javítása. A következő fejezetekben egy, az itt bemutatott elveket felhasználó számítógépi szoftver kerül bemutatásra, illetve ezen módszerek hatékonyságának vizsgálata esettanulmányok példáin keresztül.

5. KONCEPCIÓGENERÁLÓ SZOFTVER

A hagyományos tervezésmódszertan előző fejezetben összefoglalt hátrányai, hiányosságai alapján a bemutatott elméleteket felhasználva, illetve a javasolt új koncepció-tervezési módszer alapján egy olyan számítógépes szoftver került kidolgozásra, amely lehetővé teszi, hogy a tervező mérnök, a koncepcióképzés fázisában a piacon lévő termékek, valamint a szabadalmaztatott termékek elemzése révén feltárt ötleteket összefoglalja, és azokat számítógépi segítséggel szélesítse, újszerű termékvariációk képzése által. Az így létrejövő termék koncepciók természetesen tartalmazzák a már létező variációkat, a számítógépes szoftver azonban lehetőséget nyújt olyan változatok létrehozására is, amelyek korábban elkerülték a terméktervezők figyelmét. Ez a fejezet a kidolgozott szoftver elvi felépítését kívánja bemutatni. A program futtatása során elért eredményeket esettanulmányok segítségével a 6. fejezet, valamint a mellékletek kívánják szemléltetni.

5.1. A VISUALCONCEPT R1V0 FELÉPÍTÉSE

A Visual Concept program fejlesztése a Visual Basic .NET fejlesztő környezet segítségével történt, mely egy rendkívül hatékony, objektumorientált programnyelv. A Visual Concept elnevezésű szoftver kifejlesztésének célja a 4. fejezetben bemutatott módszer tesztelése, valamint a javasolt megoldás generálási módszerek hatékonyságvizsgálata volt. A program célja továbbá az, hogy a hagyományos tervezésmódszertani eljárások legnagyobb előnyét felhasználva lehetőséget nyújtson a tervezőmérnök számára, hogy adott bementi paraméterek mellett az összes lehetséges megoldás megoldásstruktúráját, gráfos elrendezésben, vizuálisan megjeleníthesse, és értékelhesse az egyes megoldásstruktúrák elvi helyességét.



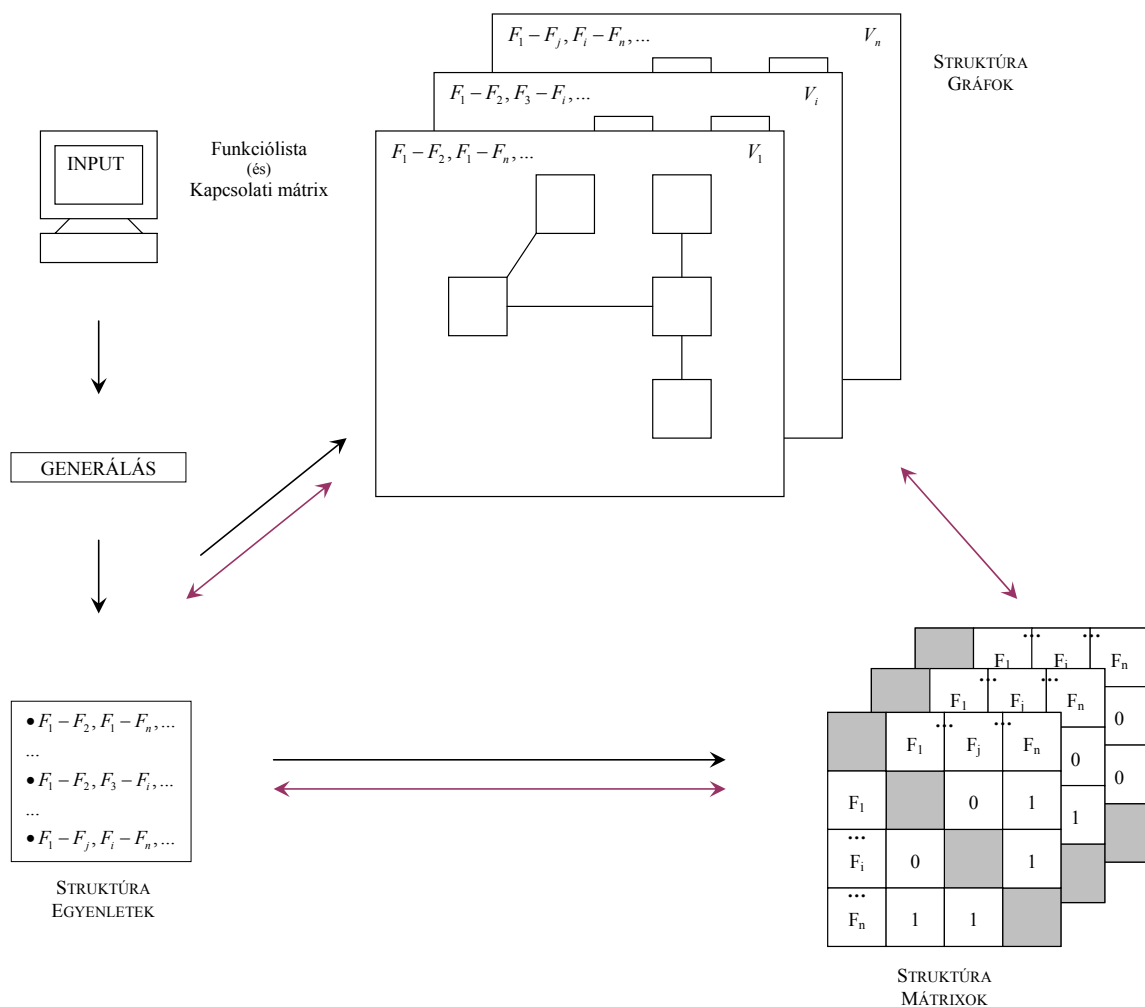
26. ábra. A Visual Concept névjegye

5.1.1. A szoftver felépítése

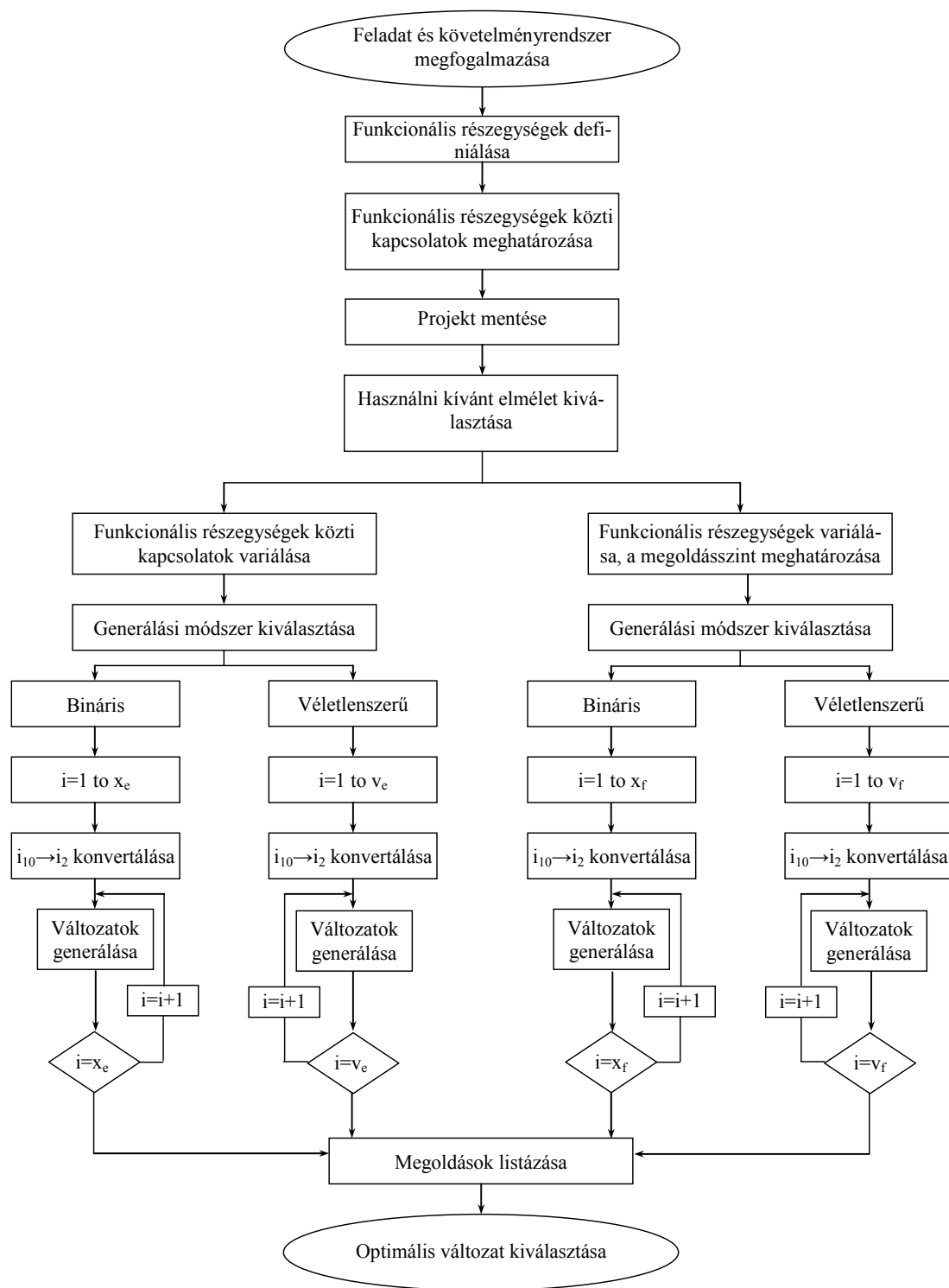
A 27. ábra a Visual Concept programcsomag funkcionális modelljét szemlélteti. A modell szerint a felhasználó betölti a programba a kívánt funkcionális részegységeket, összeállítja a funkcionális részegységek listáját. Ezenkívül megadhatja az általa ismert, fixnek tekinthető kapcsolatokat is az egyes funkcionális részegységek között, a kapcsolati mátrix kitöltésével. A program azon kapcsolatok variálásával generálja a megoldásváltozatokat, melyeket a felhasználó nem definiál előre. Ha egyik kapcsolat sincs meghatározva, a program előállíthatja a matematikailag lehetséges összes megoldásváltozatot.

Ezután a felhasználó kiválasztja a generálás módját. A program a generálás eredményét a megoldáslistában sorolja fel struktúra egyenletek formájában. Az egyes eredményeket kijelölve a program megmutatja a megoldások struktúra gráfjait, valamint a struktúra mátrixokat is. A struktúra mátrixok az egyes megoldások kapcsolatait foglalják magukba, ahogyan azt korábban a 4.2.2 fejezet már bemutatta.

Az ábrán kétirányú nyíllal jelölt folyamatok jelentősége abban rejlik, hogy a struktúra egyenletek, a struktúra gráfok, valamint a struktúra mátrixok egymásból képezhetők, ha az egyik ismert, akkor abból a másik kettő is felírható, előállítható.



27. ábra. A szoftver funkcionális modellje

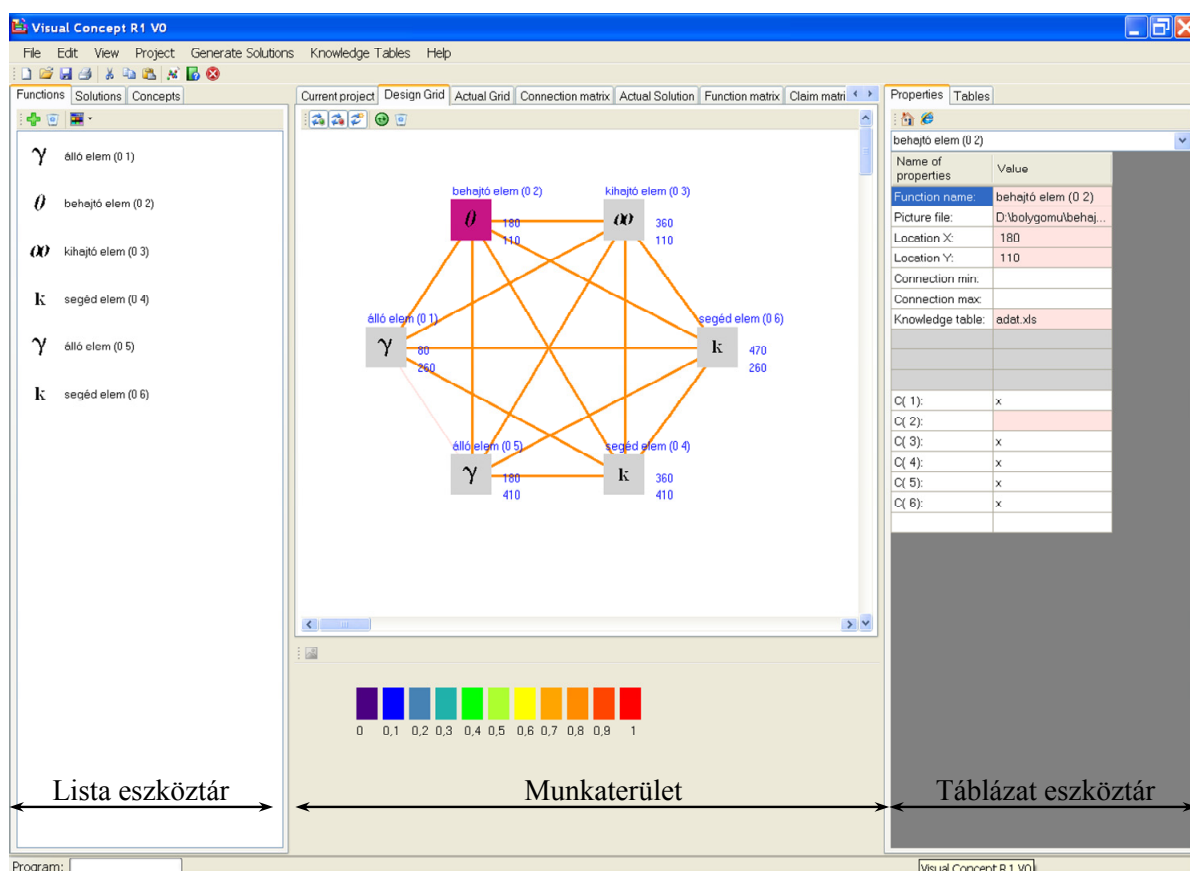


28. ábra. A szoftver egyszerűsített folyamatábrája

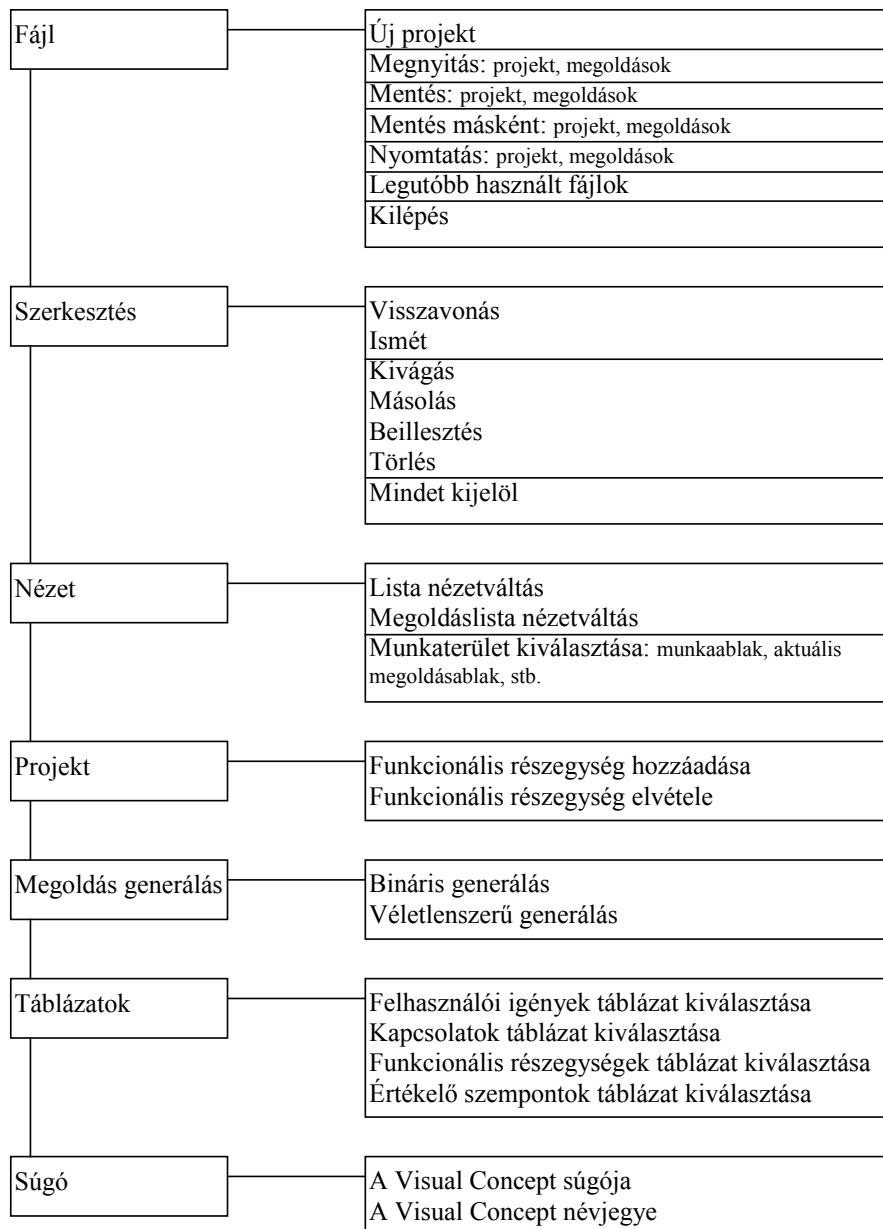
A 28. ábra a szoftver egyszerűsített folyamatábráját mutatja. A felhasználó a funkcionális részegységek megfogalmazása, és az azok közti kapcsolatok definiálása után az összeállított projektet elmentheti, majd a megfelelő generálás kiválasztása után a program elvégzi a megoldásváltozat előállítását. A különféle futtatásoknak megfelelően a ciklusváltozó is más határértékkel bír. Bináris generálás esetén az előre nem definiált kapcsolatok száma, véletlenszerű generálás esetén a felhasználó határozza meg a ciklusszámot.

5.1.2. A szoftver felhasználói felülete

A felhasználói felület részletes ismertetése előtt tisztázni kell a leírás során gyakran használni kívánt „projekt” fogalmat. A projekt egy *.prj kiterjesztésű szövegfájl, mely szekcionált struktúrájú, az első részében az egyes funkciókhoz tartozó tulajdonságok (*a funkció neve, a funkció ábrája, pozíciója a munkatérben, illetve a funkcióhoz rendelhető adattáblázat elérési útvonala*), második részében a funkciók közti kapcsolatokat definiáló **C** kapcsolati mátrix oszlopai találhatók. A projekt fájl feladata, hogy az egyes tervezési feladatokat meghatározó input adatok halmazát egyetlen fájlban lehessen tárolni. A projekt fájl átnevezésével, szerkesztésével a tervezést befolyásoló peremfeltételek gyorsan módosíthatók a program számára, így az input adatok megváltozásának a generált eredményekre gyakorolt hatása is elemezhető. Az 4.2. fejezetben bemutatott elméletekre épülő program a *VisualConcept* elnevezést kapta, mivel a cél egy jól átlátható, koncepciógeneráló szoftver kifejlesztése volt. A szoftver felhasználói felületét a 29. ábra szemlélteti, amely 3 fő részre tagozódik. Baloldalt a lista eszköztár látható a funkciók listájával (*Functions*), valamint a –*második fülön*– a generálás utáni eredmények listájával (*Solutions*), és –*a harmadik fülön*– a működőképes megoldások listájával (*Concepts*). A felület középső részén a program alkalmazása során összeállított projekt látható (*Design Grid*), illetve a megoldáslistából kiválasztott aktuális megoldás kapcsolati gráfja (*Actual Grid*), valamint a program használata során szükséges operátor mátrixok jeleníthetők meg. Jobboldalon a projekt éppen kijelölt elemének tulajdonságai tekinthetők meg (*Properties*), valamint a táblázat (*Tables*) fülön azok a táblázatok találhatók, melyekből az operátor mátrixokat állítja elő a program. A fejezet célja bemutatni a program részeit. A 30. ábra a szoftver menüstruktúráját szemlélteti.



29. ábra. A Visual Concept felhasználói felülete



30. ábra. A Visual Concept menüstruktúrája

A Lista eszköztár magába foglalja a funkcionális részegységek listáját, a futtatás utáni eredmények listáját, valamint a megoldások listáját. A funkcionális részegységek listája egy, a felhasználó által folyamatosan bővíthető/szűkíthető katalógus. A futtatási eredmények listáját a program automatikusan feltölti a generáló modul lefuttatása után. A lista megjelenítése módosítható. A megoldások listájának valamely elemét kiválasztva duplakattintással jeleníthető meg az aktuális megoldáspanelen a kiválasztott megoldás kapcsolati gráfja, illetve a hozzá tartozó kapcsolati mátrix.

A táblázatok eszköztár a program felhasználói felületének jobboldalán található. Az eszköztár két fő részre osztható, melyek az egyes funkcionális részegységekhez tartozó tulajdonság paramétereket, illetve az egyéb táblázatokat jelentik. A tulajdonság paraméterek azokat az alapjellemzőket foglalják össze, melyek alapján az egyes projektek elmenthetők, majd a későbbiek során újra betölthetők. Ilyen jellemzők például az adott funkcionális részegységhez tartozó

rendszerező táblázat elérési útvonala, vagy a funkcionális részegységet szemléltető képfájl elérési útvonala. A funkcionális részegységekhez tartozhat egy-egy rendszerező táblázat (*Roth-féle tervezői katalógus*), mely az adott funkcionális részegység megvalósításának lehetséges módjait foglalja össze. Ennek segítségével a kiválasztott koncepció egyes elemeinek kiválasztása a konstrukciós tervezés szakaszában egyszerűbbé tehető. Jellemző tulajdonságok továbbá a funkcionális részegységek helyzetének koordinátái, valamint az adott funkcionális részegység többi funkcionális részegységhez való kapcsolódási viszonya is megtekinthető a tulajdonság paraméterek közt.

A táblázatok azokat a táblákat foglalják magukba, melyek alapján az egyes projekteket jellemző projektmátrixok felépülnek (31. ábra):

- felhasználói igények táblázata (*Claims*),
- kapcsolatok táblázata (*Connection*),
- funkcionális részegységek táblázata (*Function*), valamint
- értékelő szempontok táblázata (*Value Criteria*).

A felhasználói igények táblázata a felhasználói igényeket tartalmazza, amelyeket a felhasználó fogalmaz meg. A felhasználó ebben a táblázatban adja meg az egyes igények fontosságát is, mely alapján a felhasználó igények (14) és (15) szerinti **K** mátrixa felépül. A legfontosabb igényeket az értékelő szempontok felsorolásánál szem előtt kell tartani, adott esetben változtatlanul át kell venni. Az értékelő szempontok és a felhasználói igények fontosságának megállapításánál is 0 és 1 közötti értékek írhatók a táblázatokba. A felhasználó által megadott megoldásszint feletti pontszámmal rendelkező felhasználói igényt feltétlenül figyelembe kell venni az értékelő szempontok összeállítása során. Az értékelő szempontokra adott pontszámok az értékelő szempontok esetén a súlyozó értéket fogják jelenteni.

Function Name	Sign	Connection Value
álló elem (0 1)	C(1):	x
behajtó ele...	C(2):	
kihajtó elem ...	C(3):	x
segéd elem ...	C(4):	x
álló elem (0 5)	C(5):	x
segéd elem ...	C(6):	x

Function Name	Sign	Function Value
álló elem (0 1)	F(1):	0
behajtó ele...	F(2):	1
kihajtó elem ...	F(3):	1
segéd elem ...	F(4):	0
álló elem (0 5)	F(5):	0
segéd elem ...	F(6):	1

Name of Criteria	Sign	Value
felhasználható	E(1):	0,2
helytakarékos	E(2):	0,4
növelhető munkafelület	E(3):	0,1
sok tárolóelem	E(4):	0,1
stabil	E(5):	0,1
masszív	E(6):	0,1

Name of Criteria	Sign	Value
felhasználható	E(1):	0,2
helytakarékos	E(2):	0,05
stabil	E(3):	0,3
sok tárolóelem	E(4):	0,05
szabványos alkatrés...	E(5):	0,4
	E(6):	

31. ábra. A táblázatok eszköztár táblázatai

A kapcsolatok táblázata az adott funkcionális részegység többi funkcionális részegységhez viszonyított kapcsolatát tartalmazza. A táblázatba beírható karakterek a 0, az 1, valamint az x. Ezek az úgynevezett kapcsolati értékek, a C kapcsolati mátrix ezek alapján az értékek alapján jön létre. A 0 azt jelenti, hogy két adott funkcionális részegység közt a kapcsolat nem lehetsé-

ges, az 1 azt jelenti, hogy a két funkcionális részegység közt a kapcsolat lehetséges. Az x esetében a számítógép generálás közben a 0-val és az 1-gyel is elvégzi a variálást, így képezve változatokat. A kapcsolatok táblázat alapján a **C** kapcsolati mátrix folyamatosan frissül.

A funkcionális részegységek táblázata két táblázatot foglal magába. Az egyik táblázatot a felhasználó tölti ki. Itt kell megadni, hogy az egyes funkcionális részegységek egymáshoz képest fontosabbak-e, így a 0 és az 1 karakterek adhatók meg. Ez alapján az **F** funkcionális részegység mátrix automatikusan frissül. A funkcionális részegységek táblázatokhoz tartozó másik táblázat ez után, a kész **F** funkcionális részegység mátrixból épül fel (4.2.2. fejezet). Első oszlopa az egyes funkcionális részegységek sorszámát, második oszlopa az egyes funkcionális részegységek értékeit tartalmazza, amit a program automatikusan kalkulál a felhasználó által meghatározott funkcionális részegység fontosság szerint. Ez alapján a táblázat harmadik oszlopában a felhasználó megadhatja az egyes funkcionális részegységeknek azt a 0 és 1 közötti fontossági mértékét, mellyel a program a változó funkcionális részegység halmaz szerint történő futtatáskor számolni fog.

A munkaablak (*design grid=tervezési háló*) az a terület, ahol a felhasználó a termékstruktúrához szükséges funkcionális részegységeket elhelyezi, a kapcsolati értékek megadása után a struktúra gráfot elrendezi. A munkaterület a szoftver felhasználói felületének közepén helyezkedik el. Itt vizuálisan is megtekinthető, hogy az egyes kapcsolatokat a felhasználó milyen módon kötötte le. *(Ha a két funkcionális részegység közti kapcsolatot 1-el értékelte, tehát meghatározta, hogy a két funkcionális részegység közt kapcsolat van, akkor ezt a munkaablakban egy zöld vonal jelöli a két funkcionális részegység közt. Ha a két funkcionális részegység közt nincs kapcsolat, vagyis a kapcsolatuk értéke 0, akkor a munkaablakban ezt egy halvány rózsaszín vonal jelzi a két funkcionális részegység közt. Ha a két funkcionális részegység közti kapcsolatot nem tudta meghatározni a felhasználó, vagyis x -el értékelte, akkor ez a munkaablakban egy narancssárga vonallal látszik).*

A megoldásablakban (*actual grid=aktuális háló*) megjeleníthetők a program által megfelelőnek értékelt megoldásváltozatok is. Az aktuális projekt (*Current project*) fölön az adott projekt neve adható meg, valamint a megoldásváltozatokra vonatkozó szűkítő szabályok adhatók meg, például minimálisan/maximálisan hány kapcsolatot enged meg a felhasználó egy struktúra gráfban.

A 4.2.2 fejezetben már megfogalmazásra került projektmátrixok az adott projektre vonatkozó mátrixok, melyeket a program automatikusan hozza létre a felhasználó által kitöltött táblázat eszköztár alapján. A program a következő projektmátrixokat állítja elő:

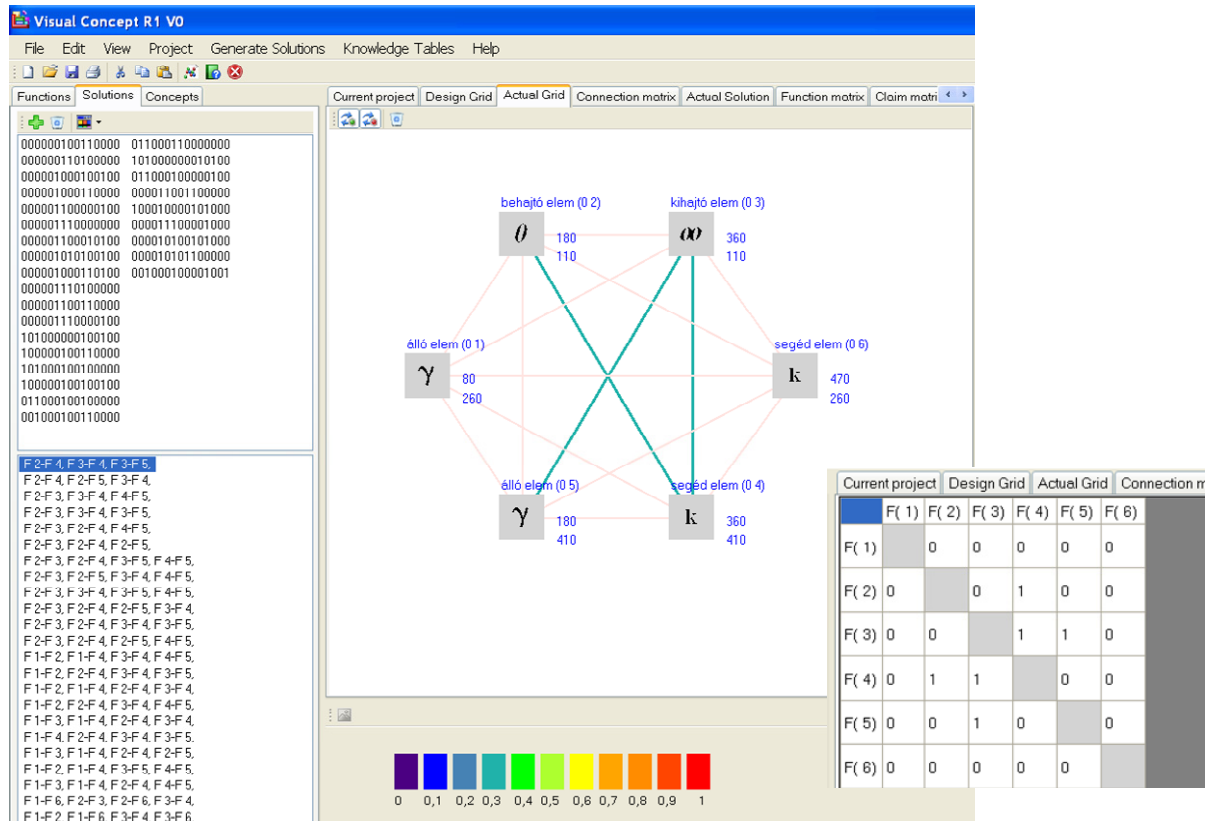
- általános kapcsolati mátrix (**C**),
- funkcionális részegység mátrix (**F**),
- aktuális kapcsolatmátrix, vagyis struktúra mátrix (**S**).

A **C** általános kapcsolati mátrix mátrixos formában mutatja meg, hogy az egyes funkcionális részegységek közt milyen kapcsolat van. A mátrix jellemző tulajdonsága, hogy főátlójába a program nem ír értéket, hiszen az egyes funkcionális részegységek önmagukkal való kapcsolata nem értelmezhető. További jellemző tulajdonság, hogy a főátló feletti háromszög megegyezik a főátló alatti háromszöggel.

Az **F** funkcionális részegység mátrix jellemző tulajdonsága, hogy a főátló ebben az esetben sem tartalmaz elemeket, minthogy a funkcionális részegységeket önmagukkal nincs értelme összehasonlítani. A mátrix főátlója feletti háromszögmátrix a főátló alatti párjának kiegészítője.

Az aktuális kapcsolatmátrix, vagy **S** struktúra mátrix a megoldáslistából kiválasztott megoldás kapcsolatait mutatja meg. A mátrix jellegzetessége, hogy az általános kapcsolati mátrixszal

ellentétben a benne előforduló karakterek csupán a 0 és az 1, minthogy a generálás során az x jelű kapcsolatokat a program mind kicserélte 1-re vagy 0-ra.



32. ábra. A megoldástér és egy generált megoldás a hozzá tartozó kapcsolatmátrix-szal

5.1.3. Megoldás generálás

A megoldás generálás alapvetően kétféle módon történhet, azonban ez négy lehetséges generálási módot jelent. Az egyik lehetséges eljárás, ha a funkcionális részegységek előre meghatározott, kötött halmaza szerint végzi a program a változatok keresését. Ekkor két lehetőség van: a bináris generálás –*melynek futtatási eredményére a 32. ábra mutat egy példát*–, illetve a véletlenszerű generálás. A másik lehetséges változat, amikor a program a megoldásváltozatokat változó funkcionális részegység halmaz szerint keresi, vagyis a funkcionális részegységek értékelése révén egy megadott határértéknél kevesebbrel bíró funkcionális részegységet eltávolítja a program a funkcionális részegység halmazból a megoldásgenerálás előtt. A tényleges megoldáskeresés ekkor is két lehetőség szerint történhet, a bináris,- vagy pedig a véletlenszerű generálás szerint.

Kötött funkcióhalmaz esetében a felhasználó beilleszti a munkatérbe a termékhez szükséges funkcionális részegységeket, a köztük lévő kapcsolatokat pedig a kapcsolatok táblázat kitöltésével határozza meg.

Ezután a felhasználó eldönti, hogy bináris, vagy véletlenszerű módon kívánja elvégezni a megoldás generálást. A bináris generálás során az összes lehetséges kapcsolatot megmutatja a program. A véletlenszerű generálás során azonban nem biztos, hogy az összes lehetséges megoldáshoz eljutunk. Ebben az esetben a program egy véletlen számsort generál, melynek megfelelően előfordulhat, hogy egyes megoldások „elkerülnek a program figyelmét”, más megoldások viszont többször előtérbe kerülnek. A véletlenszerű generálás során figyelembe

kell venni azt, hogy ennek a változat generálási módszernek csak akkor van értelme, ha az összes kapcsolatot a program határozza meg. Vagyis a kapcsolatok megfogalmazása során a felhasználónak minden kapcsolatot x-re kell felvennie. Annak ellenére, hogy a véletlenszerű generálás nem tárja fel az összes lehetséges megoldást, mégis van értelme foglalkozni vele, hiszen úgy is születhetnek érdekes és működőképes megoldások, hogy ha egyik kapcsolatot sem köti le a tervezőmérnök.

A változó funkcionális részegység halmaz alapján történő generálás annyiban tér el a kötött funkcionális részegység halmaz szerinti generálástól, hogy ebben az esetben a funkcionális részegységeket előzőleg értékeli, osztályozza a felhasználó, és csak azokkal a funkciókkal generál megoldásokat, melyek az osztályozás szerint nagyobb értékkel bírnak, mint egy 0 és 1 között választott határérték. Ekkor egy szűkebb funkcionális részegység halmazzal végzi el a generálást a program. Ebben az esetben is ugyanúgy kétféle módszer közül választhat a felhasználó. A különbség csupán az, hogy ha a funkcionális részegységekre adott értékeket növeli, vagy csökkenti a tervezőmérnök, különböző megoldásszinteket hoz létre, melyek mindegyike különböző megoldáshalmazzal bír. Nyilvánvaló, hogy ha a funkcionális részegységek nagyobb fontossági értéket kapnak, akkor több funkcionális részegység marad a funkcionális részegység halmazban, így változatosabb megoldások jönnek létre. Annak megfelelően, hogy egy adott funkcionális részegységet a felhasználó milyen fontosnak értékelt, háttérszíne a munkaterületen a 32. ábra szerinti színskála valamely színét veszi fel.

5.1.4. A program dokumentációs lehetőségei

A Visual Concept használata során a felhasználó több dolgot is elmenthet fájlba, melyek a következők:

- projekt,
- megoldáslista,
- koncepciólista.

A projekt tartalmazza egy-egy termék funkcionális részegységeit és azok *–felhasználó által definiált–* kapcsolatait, tulajdonságait. A projekt fájlba történő mentése teszi lehetővé, hogy ugyanaz a projekt egyszerűen betölthető legyen, és a futtatásokat azonos bemeneti paraméterek mellett, viszonylag gyorsan újra el lehessen végezni. A betöltött projekt módosítható, a javításokkal együtt újból elmenthető, átnevezhető. A fájl a programon kívül is módosítható. Ehhez azonban ismerni kell a projekt fájl felépítését, mely a program forráskódjában definiált projekt- és kapcsolati mátrixokból áll. A projektmátrix a következőket tartalmazza:

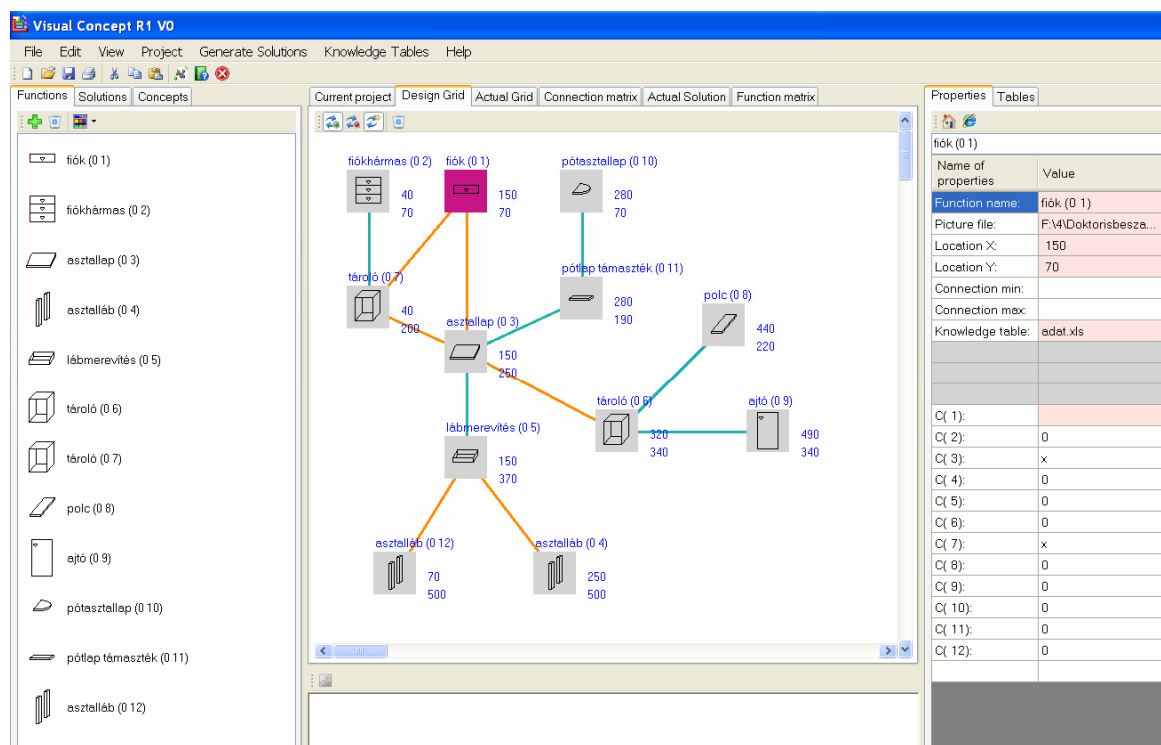
- az egyes funkcionális részegységek neve,
- az egyes funkcionális részegységekhez fájlból hozzárendelt kép elérési útvonala,
- az egyes funkcionális részegységeknek a munkatérben elfoglalt pozíciójának koordinátái,
- valamint az adott funkcionális részegységekhez rendelt rendszerező táblázat elérési útvonalát.

A megoldáslistába a generálás során létrejött megoldások közül a felhasználó által kiválasztott elvileg helyes, működőképes megoldások menthetők. Ha a felhasználó ezt a listát tölti be a programba, akkor a követelményjegyzék segítségével átnézve ezeket a megoldásokat, könnyen összeállítható az adott igényeket kielégítő megoldások listája, vagyis a koncepciólista, amely szintén fájlba menthető.

A szoftver használata során a felhasználó kinyomtathatja az adott projektet szöveges formátumban, valamint a megoldások listáját is.

6. ESETTANULMÁNY–HATÉKONYSÁGVIZSGÁLAT

A disszertáció kidolgozása során cél volt egy olyan számítógépes szoftver tesztelhető verziójának kifejlesztése, mely a disszertációban megfogalmazott elvek alapján működik. A program tesztelése több fázisban történt. Először olyan futtatásokat kellett végrehajtani, melyek révén a program hiányosságai feltárhatók, és megszüntethetők. A második fázisban a szoftver eredményességét elemző futtatások következtek. A program hatékonyságvizsgálatához két esettanulmány készült: egy egyszerű felépítésű termék megoldásváltozatainak generálása, valamint fogaskerék bolygóművek kapcsolásával, más módszerrel korábban már előállított megoldásváltozatok rekonstrukciója. Az esettanulmányok elvégzése során a cél a módszer hatékonyságának, a generálások eredményességének minél szélesebb körű vizsgálata volt.



33. ábra. A kiindulási projekt

Az első esettanulmány alapja egy olyan irodai munkahely koncepcióváltozatainak előállítása volt, mely növelhető munkafelülettel és tároló részekkel is rendelkezik. A feladat elvégzéséhez szükségeszerű egy követelményjegyzék összeállítása, mely a felhasználók és a gyártó termékkel szembeni elvárásait foglalja magába.

A második esettanulmány során a DÖBRÖCZÖNI, APRÓ, ANTAL és TERPLÁN [36] által létrehozott fogaskerék bolygómű változatoknak a Visual Concept programmal való rekonstruálására került sor.

6.1. ELSŐ ESETTANULMÁNY: IRODABÚTOR KONCEPCIÓK KÉPZÉSE

A 33. ábra a Visual Concept-be betöltött projekt kiindulási állapotát szemlélteti. Baloldalt látható a funkcionális részegységek listája, jobboldalt az F1 (*fiók*) funkcionális részegységhez tartozó tulajdonság ablak, a munkaablakban az összeállított struktúra gráf látható. A gráf zöld élei a biztos kapcsolatokat jelentik, a narancsszínű élek azokat a kapcsolatokat, melyekről a program rendelkezik bináris, vagy véletlenszerű megoldáskeresési módszerrel.

6.1.1. Követelményjegyzék

Az itt összefoglalt, a termékkel szemben elvárt követelmények, egy teljes igényjegyzéknek csupán azokat az elemeit tartalmazzák, amelyeket már a koncepció képzés fázisában is figyelembe kell venni. Ezeket a követelményeket a gyártók a felhasználók elvárásai, valamint a piacon népszerű hasonló termékek tulajdonságai alapján fogalmazzák meg úgy, hogy saját elvárásaikat is feltüntetik benne. A termékkel szembeni elvárások a teszt projekt során:

- Az asztal legyen falra szerelhető azért, hogy kis helyet foglaljon.
- Mivel a termék alapvetően kis helyet foglal, azonban gyakran van szükség több munkahelyre, így az asztal nővelhető munkafelülettel kell, hogy rendelkezzen
- Hogy minden kézreessen a felhasználó számára fontos, hogy az asztal számos tárolóelemmel rendelkezzen.
- A termék legyen stabil, masszív, hogy a felhasználó bármilyen munkát végezhesen rajta.
- Lényeges szempont továbbá, hogy a termék összességében véve keves alkatrészből álljon. Ez a gyártást olcsóbbá, az összeszerelést egyszerűbbé teszi.

6.1.2. Futási eredmények

6.1.2.1. Generálás a funkcionális részegységek közti kapcsolatok variálásával

A funkcionális részegységek közti kapcsolatok variálása azt jelenti, hogy az eljárás során a kapcsolati gráf élei kerülnek variálásra; a csomópontok, vagyis az egyes funkcionális részegységek közt alapvetően vagy van kapcsolat, vagy nincs, így a kapcsolatok értéke lehet 1 és 0. A funkcionális részegységek közti kapcsolatok értékéről a felhasználó dönt, így szűkíthető a program által előállítható megoldástér. Abban az esetben, ha egy-egy kapcsolatot nem lehet előre definiálni, mert felmerül az újszerű megoldások lehetősége, a program feladata létrehozni az ilyen megoldásváltozatokat. Lényeges megállapítás, hogy a kapcsolati gráf éleivel történő variálás során áttekinthetetlenül sok megoldásváltozat születhet, ha minden kapcsolat értékéről a program hoz döntést. Viszont minden egyes termékstruktúrában vannak olyan funkcionális részegységek, melyek vitathatatlanul összetartoznak, szétválasztásuk nem nyújt lehetőséget az újdonságra. Ily módon a kapcsolatoknak a program felhasználója által történő definiálása egyfajta szabálykészlet, szabályrendszer felállítását jelenti, mely már a megoldásváltozatok generálása előtt lehetővé teszi számos, biztosan rossz megoldás kiszűrését. A prog-

ram tesztelése során először hat olyan kapcsolat lett definiálva, melyek a program futása során kerülnek variálásra.

6.1.2.1.1 Bináris generálás

A bináris generálás a definiált peremfeltételek mellett az összes lehetőséget megmutatja. A kezdeti hat variálható kapcsolat a bináris generálás során a

$$V_{kb} = 2^e \quad (38)$$

szerint hatvannégy megoldást hozott. Ezek közül kellett kiválasztani előbb az elvileg helyes megoldásokat, majd a feltételeket, vagyis a követelményjegyzék pontjait kielégítő megoldásokra szűkíteni a listát.

Az elvileg helyes megoldások száma negyvenöt (*a Melléklet 9.1.1. fejezetében megtalálható mind a negyvenöt megoldás struktúragráfja a hozzájuk tartozó struktúra egyenletekkel*), azonban ezek a megoldások számos olyan lehetőséget foglalnak magukba, melyek az előzetesen összeállított követelményjegyzéket nem elégítik ki. A jegyzék feltételeinek végül hat megoldás felelt meg, melyek a Melléklet 9.1.2 fejezetében találhatók.

6.1.2.1.2 Véletlenszerű generálás

A véletlenszerű generálás fontos jellemzője, hogy annak ellenére, hogy a bemeneti paraméterek azonosak, több futtatás során nem ugyanazok az eredmények generálódnak. Nyilvánvalóan átfedések lesznek a véletlenszerű generálás futási eredményei közt, de pontosan ugyanaz a futási eredmény kis valószínűséggel lehetséges.

Másik fontos jellemző, hogy míg a bináris generálás során az adott bemeneti paraméterek mellett az összes lehetséges megoldás generálódik, a véletlenszerű generálás során nem, vagy nagyon kis valószínűséggel, illetve végtelen sok próbálkozás esetén jöhet létre az összes lehetséges megoldásstruktúra. Ennek oka az, hogy a program a futás során annyi véletlenszámot generál, ahány megoldása összesen lehet a futásnak.

Az esettanulmányban szereplő projektben, a véletlen próbálkozások száma száz. Mivel azonban a véletlenszámok közt ugyanaz a szám többször is előfordulhat, így nem meglepő, hogy a mellékletben szereplő megoldásokat eredményező futtatás esetében csupán harmincnégy különböző struktúrát generált a program.

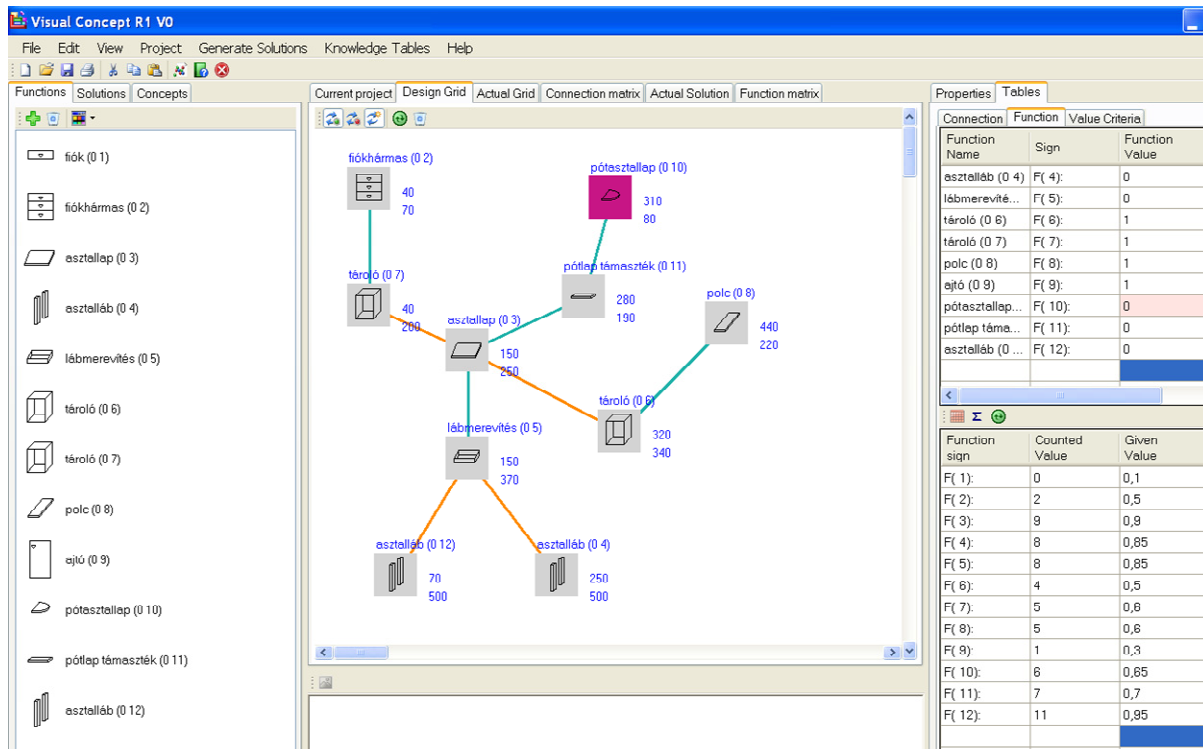
A Mellékletben (*9.1.1. fejezet*) szereplő megoldási struktúrák a funkcionális részegységek közti kapcsolatok szerint történő bináris generálás elvileg helyes eredményeit foglalják össze. Azok a megoldás változatok, melyeken keret van, a véletlenszerű generálás során is létrejöttek.

6.1.2.2. Generálás a funkcionális részegységek variálásával

A funkcionális részegységek variálásával történő megoldásváltozat generálás során, miután a struktúra gráf egyes csomópontjait összekötő kapcsolatok meghatározásra kerültek, a követelményjegyzék alapján el kell dönteni, az adott funkcionális részegységek sorrendjét. Ezt a program automatikusan elvégzi, hogy ha a felhasználó az egyes funkcionális részegység párok páronkénti fontosságát, mint bemeneti paramétert megadja. A program kiszámolja, hogy az egyes funkcionális részegységek külön-külön hány pontot kaptak az értékelés során. Ekkor a felhasználó a funkcionális részegység fontosságát egy 0 és 1 közé eső értékkel látja el. A program tesztelése során ez az érték 0,5-re lett felvéve. Ha valamely funkcionális részegység ettől kisebb értéket kap, akkor azt *–illetve a hozzá tartozó éleket–* ki kell vennie a struktúra gráfból. Ezáltal egy kisebb megoldástér keletkezik, amely alapvetően kevesebb megoldást

eredményez, mint a funkcionális részegységek közti kapcsolatok variálásával történő generálás.

A 34. ábra azt szemlélteti, hogy a kiindulási projekt hogyan módosult, miután a funkcionális részegység párok értékelésével az egyes funkcionális részegységekhez tartozó funkcionális részegység értékek definiálásra kerültek. A 33. ábra szerinti kiindulási projekttel összehasonlítva látható, hogy az új projekt két funkcionális részegységgel (*fiók*, *ajtó*), valamint a hozzájuk tartozó élekkel csökkent, vagyis a megoldástér kisebb lett.



34. ábra. A kiindulási projekt a struktúra gráf szűkítése után

6.1.2.2.1 Bináris generálás

A funkcionális részegységek variálása bináris módon hasonlóan történik, mint a funkcionális részegységek közti élek variálása során, azonban ebben az esetben azáltal, hogy a felhasználó a funkcionális részegységeket egymással összehasonlította és értékelte, a kevésbé fontosakat a struktúra gráfból eltávolította, csökkent a variálható élek száma. A megoldások száma ebben az esetben is a (40) szerinti képletből származtatható, azonban a variálható élek száma az esettanulmányban szereplő futtatás szerint négy. Ennek megfelelően a futás eredménye tizenhat megoldás, amelyből tizenöt elvileg helyesnek bizonyult, ezeket a Melléklet 9.2.1 fejezete foglalja össze. Ezeket a megoldásokat össze kell vetni a követelményjegyzékkel, és a jegyzékben felsorolt kritériumoknak legjobban megfelelő megoldásokat kell kiválasztani. A jegyzék feltevéleinek végül két megoldás felelt meg, melyek a Melléklet 9.2.2 fejezetében találhatók. Itt is látható, hogy a kerettel ellátott változat mind a kétféle futtatás során adódott, míg a keret nélküli változat csak a bináris generálás során.

6.1.2.2.2 Véletlenszerű generálás

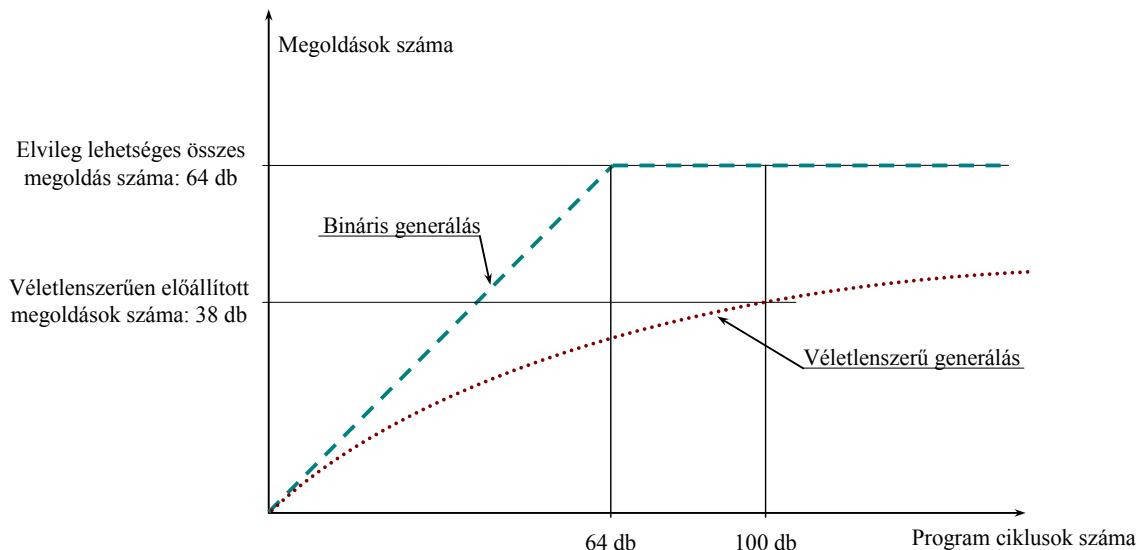
A véletlenszerű generálást ugyanazok a tulajdonságok jellemzik ebben az esetben is, mint a funkcionális részegységek közti élek variálása során. Az egyetlen különbség csupán az, hogy mivel kevesebb a variálható élek száma, ezért a program kevesebb véletlenszámot generál. A

futtatások során bebizonyosodott, hogy azonos számú próbálkozás esetén minél kisebb a variálható élek száma, annál jobban közelít a véletlenszerűen generált megoldások száma a binárisan generált megoldások számához. A véletlenszerű generálás módszerével végzett futtatás mindegyik eredménye megtalálható a binárisan generált megoldások közt. A Melléklet 9.2.1 fejezetében szereplő bekeretezett struktúrák a véletlenszerű generálás során is létrejöttek.

6.1.3. A futási eredmények tapasztalatai

A 35. ábra a funkcionális részegységek közti kapcsolatok szerint történő generálás eredményeit foglalja össze bináris és véletlenszerű futtatás esetén.

Az esettanulmányok során bebizonyosodott, hogy a kidolgozott szoftver alkalmas az adott bemeneti paraméterek mellett az összes lehetséges megoldásváltozatot létrehozni, a leggenerált megoldások kapcsolatainak vizuális megjelenítése révén lehetővé teszi az eredmények könnyű, és gyors kiértékelését, a helyesen működő megoldások kiválasztását.



35. ábra. A futási eredmények

Meg kell jegyezni, hogy a funkcionális részegységek variálása során a tervezőmérnök figyelmét elkerülhetik eredendően új ötletek, hiszen már a generálás előtt elvesz funkcionális részegységeket a struktúra gráfból. A funkcionális részegységek közti kapcsolatok variálásának módszere során is figyelembe kell venni a követelményeket, de ebben az esetben csak egyszer, a generálás végén. A funkcionális részegységek variálása során kétszer kell figyelembe venni a követelményjegyzéket, mert a funkcionális részegységek elvétele is még számos megoldást eredményez, amelyeket ki kell értékelni, és a követelményjegyzéknek meg kell feleltetni. Ebben az esetben a tervezőmérnök munkája könnyebb lehet, hiszen egyszerre kevesebb megoldást kell áttekinteni. A funkcionális részegységek közti kapcsolatok variálása során a tervezőmérnök egyszer végzi el a követelményjegyzéknek való megfeleltetést. Annak ellenére, hogy ily módon több megoldást kell megvizsgálnia, mégis ez a generálás a legteljesebb, hiszen előfordulhat, hogy a bemeneti paraméterek meghatározása során a funkcionális részegységek variálásának módszere esetében a tervezőmérnök tévesen határozta meg valamely funkcionális részegység pár fontossági viszonyát. Emiatt többszöri futtatás szükséges, és

javasolt mind a kétféle módszer együttes alkalmazása, hogy a tervezőmérnök minél szélesebb körben, minél nagyobb megoldástérben, minél alaposabb megoldáskutatást végezhesen.

6.2. MÁSODIK ESETTANULMÁNY: FOGASKERÉK BOLYGÓMŰVEK ELŐÁLLÍTÁSA

A javasolt eljárás és a kidolgozott program tesztelése során az is lényeges tényező volt, hogy bebizonyításra kerüljön, a javasolt eljárás nem csupán új termékek koncepcióinak generálására alkalmas, de képes arra is, hogy már meglévő termékeket/ termékcsaládot is le tudjon generálni. Az erre a célra kiválasztott termék egy hajtóműcsalád, a DÖBRÖCZÖNI, APRÓ, ANTAL és TERPLÁN [36] által előállított kb+kb kapcsolatú fogaskerék bolygóművek. Mivel ebben az esetben nem az volt a cél, hogy hányféle képpen állíthatók elő különféle termékek, hanem az, hogy a javasolt módszerrel megtalálhatók-e a szerzők által javasolt bolygóművek, ezért erre a célra a legalkalmasabb generálási mód a funkcionális részegységek közti kapcsolatok bináris variálása.

Az esettanulmány elvégzéséhez a szerzők által javasolt terminológiák és kinematikai ábrák alapján az alábbi funkcionális részegységek kerültek megfogalmazásra:

- behajtó elem (0)
- kihajtó elem (∞)
- álló elem 1 (γ)
- álló elem 2 (γ)
- segédelem 1 (k)
- segédelem 2 (k)

A segédelem külső nyomatékkal nem terhelt elem, így lehet a kar, a napkerék, vagy a gyűrűkerék. Az álló elem a ház, vagy valamely rögzített elem, mely minden esetben szabadságfok lekötését eredményezi.

A bináris variálás módszere szerint a Melléklet 9.3 fejezetében megtalálhatók a VisualConcept által generált koncepcióváltozatok. Mivel a szerzők által javasolt bolygómű-változatok kinematikai ábrája egy konkrétabb szerkezeti felépítésre utal, ezért az esettanulmány jól alátámasztja, hogy a koncepció a termékről egy átfogó kép, a későbbi konstrukciós kialakításhoz vezető első lépcső. A koncepció nem zár ki konkrét megoldásokat, csupán arra utal, hogy melyik alkotóelem melyik alkotóelemmel van kapcsolatban. A kapcsolat minőségére, fizikai törvényszerűségeire nem tesz utalást, így nem is zár ki eredendően jó, és újszerű megoldásokat. Az esettanulmány során előállított 27 koncepcionális megoldásból mind a 108 –a szerzők által, más előállított– konstrukció eredeztethető.

DÖBRÖCZÖNI, APRÓ, ANTAL és TERPLÁN az egyes bolygómű-változatokat az összekapcsolt bolygóművek (3 típus) relatív szögsebességviszony tartományai (9 hajtóműcsoport, 4 szögsebességtartomány), valamint a tartományokra jellemző belső áttétel szerint hozták létre ($3 \cdot 9 \cdot 4 = 108$). Belátható, hogy ez a fajta megoldásvariáció-előállítás a kinematika függvénye, amely már túllépi a koncepció fogalmát.

7. ÖSSZEFOGLALÁS

A 20. századi gyorsuló technikai fejlődés tette szükségessé a tervezési folyamat nagyobb egységeinek meghatározását, a koncepcióképzés különféle módszereinek kialakulását, egy újfajta tervezői szemlélet megszületését, mely szerint egy adott problémára nem csak egy megoldás létezik, hanem meglehetősen sok. Így a tervezőmérnök legfőbb küldetésének nem azt kell tekinteni, hogy a tervezési feladat egy lehetséges megoldását kidolgozza, hanem hogy az elvileg végtelen nagy megoldáshalmazban megtalálja az adott igényeknek legjobban megfelelő változatot.

A hagyományos tervezés módszertani eljárások tanulmányozása során számos irodalom került feldolgozásra, azonban ezek a munkák nem térnek ki a számítógépes alkalmazás lehetőségére, mert nagyrészüket még a számítógépes korszak előtt született. Az informatika fejlődése generálta azt az igényt, hogy nagyszámú koncepcióváltozat előállításával a tervezői munka határfoka javítható legyen, illetve a terv várható minősége és a tervezési idő jobban becsülhető legyen. A disszertáció célja, hogy *a klasszikus módszeres tervezési eljárások előnyeinek és hiányosságainak feltárásával, az előnyök alkalmazása, a hátrányok kiküszöbölése révén egy számítógépre adaptálható, új koncepcióképzési módszerre tegyen javaslatot, az elméleti háttér kidolgozásával és egy erre épülő programcsomag prototípusának létrehozásával.*

- A hagyományos módszerek nagy hátránya az, hogy számítógépes adaptálásuk csak korlátokkal valósítható meg. Ezek a módszerek az egyes lépéseknél számos javaslatot tesznek az adott logikai lépés megvalósítására, ami magával vonja az eljárások összetettségét. Az egyik feladat tehát az, hogy a hagyományos módszerek helyett, azok pozitív tulajdonságait felhasználva egy olyan tervezői folyamatot kell javasolni, mely pontosan megadott lépéseket követ.
- A tradicionális módszerek másik nagy hátránya, hogy a fizikai elveket tartják szem előtt. Ez azért hátrányos, mivel így a tervező már a koncepcióképzés fázisában korlátozhatja a megoldási lehetőségeket. Ily módon, akaratán kívül kizárhat olyan megoldásokat, melyek a későbbi szelektálási szakasz során azért lehetnének értékesek, mert újszerű, vagy teljesen új megoldásokat eredményeznének. A másik feladat tehát az, hogy a megoldásváltozatokat a fizikai elvek figyelembevétele nélkül kell előállítani.
- A meglévő módszerek harmadik –és talán legnagyobb– hiányossága az, hogy nem próbálják meg előállítani az összes lehetséges megoldást, ami a feltárt funkciók alapján előállítható volna, hiszen az emberi kapacitásokra *–nem pedig a számítógépek által nyújtott lehetőségekre–* építenek. Ily módon fennáll annak a lehetősége, hogy néhány –vagy akár jelentős mennyiségű– elvileg helyes, esetleg teljesen új megoldás elkerüljön a terméktervező mérnök figyelmét. Ennek megfelelően a legnagyobb feladat, hogy megvalósuljon a feltárt funkciók számítógépi kezelhetősége, valamint ennek révén az összes lehetséges megoldásváltozat előállíthatósága és kezelhetősége.

Az értekezés javaslatot tesz a felsorolt problémák megoldására egy olyan eljárás elvi alapjainak leírásával, mely ezeket a hibákat kiküszöböli, illetve egy *–teljes mértékben–* erre az eljárásra épülő számítógépes szoftver működését, esettanulmányait, azok eredményeit írja le.

A disszertáció szakít a hagyományosan funkció alapú eljárásokkal, és felveti annak lehetőségét, hogy a tervező funkcionális részegységek (*bizonyos funkciót ellátó szerkezeti egységek*) meghatározásával állítson elő termékstruktúrákat, így egy új leíró módszert vezet be a termékstruktúrák definiálására, mely már nem a fizikai elvek alapján, hanem a konstrukciós elemek logikai kapcsolatán alapszik.

A disszertációban egy termékstruktúra generáló módszer került kidolgozásra, mely a kötött-, illetve rugalmas funkcionális részegység halmazból az elvileg lehetséges valamennyi koncepcionális megoldást képes előállítani. Megállapításra került, hogy a kötött funkcionális részegység halmaz alapján több megoldás generálható.

A disszertáció javaslatot tesz továbbá a fuzzy-megoldáscsaládok háromdimenziós térben történő leírására, mely a rugalmas funkcionális részegység halmaz alapján történő generálás eredményeire alkalmazható.

Kidolgozásra került egy, a koncepcionális tervezés fázisában alkalmazható programrendszer. A Visual Concept program előre definiált funkcionális részegységek halmazán több módszer szerint képes a termékstruktúrák létrehozására. A programrendszer működőképes, tesztelhető prototípusa is elkészült.

Az esettanulmányok igazolták, hogy a számítógépes program a felvetett három probléma mindegyikét kiküszöböli. A kidolgozott eljárás elvi alapjai lehetővé tették egy olyan számítógépes szoftver kifejlesztését, mely segíti a terméktervező mérnököt abban, hogy a koncepcionális tervezési folyamatot elvégezve, valamely terméknek, a felhasználói igényeket legjobban kielégítő koncepcióját találja meg, az összes lehetséges változat közül, az adott peremfeltételek mellett.

7.1. AZ ÚJ TUDOMÁNYOS EREDMÉNYEK ÖSSZEFOGLALÁSA

Az értekezés új tudományos eredményeit az alábbi tézisek foglalják össze:

T1. A terméktervezés koncepcionális szakaszában alkalmazható új szemléletet vezettem be, amelynek lényege, hogy a koncepcióképzés folyamata egy minőségi és egy mennyiségi ágon megy végbe.

- Megállapítottam, hogy a mennyiségi ágon olyan matematikai módszereket kell alkalmazni, amelyek az áttekinthetőségi tartományon belül az elvileg lehetséges összes megoldást képesek előállítani (*a dolgozat a bináris logikát alkalmazza*). Ráműtattam, hogy ha az elvileg lehetséges megoldások száma nagyobb, mint az áttekinthetőségi tartomány, akkor olyan matematikai módszert kell alkalmazni, amely véletlenszerű kiválasztással biztosítja a szükséges mennyiségű generált megoldás megfelelő eloszlását a bemeneti paraméterek által meghatározott megoldástérben (*erre a dolgozat a véletlenszám generáláson alapuló elemkiválasztást használja*).

- Megállapítottam, hogy a minőségi ágon funkcionális részegység- és kapcsolatfontossági vizsgálattal korlátozni lehet a megoldásteret. Így biztosítható, hogy az elvileg lehetséges megoldások száma az áttekinthetőségi tartományon belül legyen.

T2. Termékstruktúra generáló módszereket dolgoztam ki, melyek kötött-, illetve rugalmas funkcionális részegység halmazból állítanak elő koncepcionális megoldásokat.

- Bevezettem a kötött- és a rugalmas funkcionális részegység halmaz, valamint a megoldásszint fogalmát, amellyel rugalmas funkcionális részegység halmaz esetén a megoldáskeresésben résztvevő funkcionális részegységek mennyisége szabályozható.
- Megállapítottam, hogy a rugalmas funkcionális részegység halmaz szerinti generálás esetén a funkciók fontosságának elemzése és a megoldásszint változtatása megoldáscsaládok létrehozásához vezet.

T3. Kidolgoztam a rugalmas funkcionális részegység halmaz szerinti generálás megoldásváltozatainak (*megoldáscsalád*) háromdimenziós térben történő fuzzy leírását.

- Descartes-féle koordinátarendszerben definiáltam az Ω fuzzy megoldásteret, valamint a megoldásteret meghatározó α fuzzy termékbonyolultsági-, β fuzzy funkcionális részegység fontossági-, valamint a δ fuzzy funkcionális részegység előfordulás síkokat.

T4. Kidolgoztam egy, a koncepcionális tervezés fázisában alkalmazható programrendszert. A Visual Concept program előre definiált funkcionális részegységek halmazán a beillesztett módszerek segítségével –*melyek a bináris számlálás és a véletlenszámon alapuló kiválasztás*– képes termékstruktúra változatok létrehozására. Elkészítettem a programrendszer működőképes, tesztelhető prototípusát.

- Esettanulmányokkal igazoltam, hogy a kidolgozott módszer, valamint a javasolt szoftver alkalmas termékkonceptió változatok előállítására.

7.2. TOVÁBBFEJLESZTÉSI IRÁNYOK, LEHETŐSÉGEK

A legfontosabb továbbfejlesztési feladat az, hogy a javasolt eljárásra kidolgozott szoftver a CAD rendszerekkel együtt tudjon dolgozni, az optimálisnak kiválasztott koncepciót olyan parametrikus geometriára alakítsa, melyen a később szükséges mérnöki számítások elvégezhetők legyenek.

A szoftver továbbfejlesztését a konkrét ipari megbízások nagymértékben elősegítenék. A teszt termékek csupán egy fő irányvonalat mutattak a szoftver felépítése során, azonban a termék specifikus paraméterek a fejlesztés alatt nem lettek figyelembe véve.

A továbbfejlesztési munkák végső célja a kidolgozott szoftver iCAD modullá fejlesztése.

7.3. KÖSZÖNETNYILVÁNÍTÁS

A disszertáció témáját képező tervezésmódszertan negyedéves gépészmérnök hallgatóként kezdett érdekelni. A tudományterületre témavezetőm, DR. KAMONDI LÁSZLÓ egyetemi docens vezetett be, iránymutatását ezúton szeretném megköszönni.

A doktori tanulmányokat a Miskolci Egyetemen a Sályi István Gépészeti Tudományok Doktori Iskolában a 2005/2006-os tanévben a Gépelemek Tanszékén nappali tagozatos doktoranduszként kezdtem. A képzés során számos olyan tantárgyat hallgathattam, amelyek a kutatá-

saimat meghatározták, befolyásolták. Ezért szeretném megköszönni DR. DÖBRÖCZÖNI ÁDÁM professzor úr, DR. KOVÁTS ATTILA, DR. PÉTER JÓZSEF, DR. SIPOSS ISTVÁN, DR. SZABÓ J. FERENC, valamint DR. SZENTE JÓZSEF tanár urak segítségét.

Köszönet illeti a Sályi István Doktori Iskola egykori és jelenlegi vezetőjét –DR. PÁCZELT ISTVÁN *professzor urat* és DR. TISZA MIKLÓS *professzor urat*–, akik észrevételeikkel, javaslataikkal támogatták munkámat.

Szeretném megköszönni továbbá JAN VOJNA, JIRI BARTÁK és ROMÁN CERMÁK kollégák (*West-Bohemian University, Pilzen, Csehország*) segítségét, baráti támogatásukat, hogy a disszertációval kapcsolatban számos konferencián, szakmai vitán való részvételemet elősegítették.

A legnagyobb köszönet Családom tagjait illeti, akik mindig támogattak és a disszertáció elkészítésében a legnagyobb segítséget nyújtották.

8. SUMMARY

The accelerated technical development of the 20th century made it necessary to determine the considerable steps of the design process, to develop the different methods of conceptual design, to polarise design schools the birth of a novel design aspect according to which there is not only one solution for a given problem, but there are significantly lot. So the main task of the designer-engineer is not to find a solution for the design task, but to find the optimal concept out of the infinitely big solution set.

A lot of literature was studied during the examination of design methods, but most of these works are not dealing with the possibility of the adaptation for computers, because they were worked out before the computer era. The development of the informatics generated the claim to improve the effect of the design work by producing a big number of concept variants, and that the expectable quality of the design and the design term should be better forecast. The aim of the PhD. thesis is to find the advantages and disadvantages of the classical design theories and with the application of the advantages and elimination of the disadvantages make a suggestion for a novel approach of design method in the field of product design that is adaptable for computer, and also to create a prototype of a programme package on the basis of the proposed method.

- A significant disadvantage of existing methods is that their adaptability to computers is limited. Traditional methods suggest several possibilities for the sequential steps realising the given logical step that implies their multiple character. Firstly a fix sequential design method has to be created instead of traditional methods, though utilizing their positive features.
- Keeping physical principles in view is the second great disadvantage of traditional methods. It is disadvantageous because the constructor may limit the solution possibilities in a very early phase of the design process. This way he may unwittingly eliminate solutions that could be really valuable in the period of selection, because they are novel or absolutely new concepts. This way the second task is to generate the solution variants, without paying any attention to physical principles.
- The third *–and maybe the greatest–* deficiency of existing methods is that they do not even try to generate all the possible solutions that could be made on the basis of the given functions, because they pay attention to the human capacity and not to the capacity of computers. So it can happen that some *–or significantly a great number of–* principally correct and absolutely new solutions escape the designer's eyes. That is why the most important task is to make it possible to generate and handle all the solutions that can be generated on the basis of the given functions.

This thesis suggests a solution of the above listed problems by defining the doctrinal basis of a method that eliminates these failures and it also shows the operation of a software *–that has been built absolutely on the basics of the suggested method–*, case studies of the software, and their results.

The dissertation introduces a new expositive method for define general function-structures, that are not underlying physical principles, but the logical connection of the constructional-functions.

A function-structure generator method has been developed in the thesis that is suitable for generating all the possible conceptual solution from the tied- and the flexible function-cluster. It has been determined that according to the tied function-cluster more solution can be generated and in the case of products the application of the flexible function-cluster is practical.

The dissertation propounds for describing fuzzy-solutionfamilies in a three-dimensional space that can be applied for the solutions of the generation on the basis of the flexible function-cluster.

A program-system has been developed that can be used in the phase of the conceptual design. The software –*Visual Concept*– is suitable for generating function-structures on the cluster of previously defined functions, according to several different methods. The operating, testable version of the program-system has been completed.

Case-studies were confirmed that the above given problems are eliminated by the developed software. The principle basics of the suggested method have been permitted the development of a software that helps the product designer to find the optimal concept of the given design task from all the possible solutions, along given conditions.

Táblázatok jegyzéke

1. táblázat. A tervezésmódszertan mérföldkövei.....	15
2. táblázat. Tervezői katalógusok	26
3. táblázat. A tervezési modellek jellemzői.....	34
4. táblázat. Struktúra mátrix (S).....	40
5. táblázat. Súlyozott terméktulajdonságok módszere	51

Ábrajegyzék

1. ábra. Tudomány-igény kereszt.....	12
2. ábra. Az intuíció SELYE alapján.....	21
3. ábra. Az intuitív tervezés modellje	22
4. ábra. A kognitív tervezési modell OSHUGA alapján [61].....	23
5. ábra. Az alaprendszer HANSEN alapján	24
6. ábra. KOLLER tervezői módszere	25
7. ábra. A teljes tervezési folyamat ROTH alapján.....	26
8. ábra. A TRIZ feladatmegoldási metódusa.....	27
9. ábra. Az ellentmondási mátrix részlete.....	28
10. ábra. A TRIZ szerkezeti felépítése	29
11. ábra. A termékfelépítési folyamat.....	31
12. ábra. Többszörös Fuzzy modell [35] alapján	32
13. ábra. A pszichológiai szemléletű tervezési modellek hierarchiája.....	34
14. ábra. A koncepcióképzés folyamata [85]	36
15. ábra. Morfológiai mátrix.....	38
16. ábra. Megoldásfa.....	39
17. ábra. A bináris logika, valamint a fuzzy logika értéktartománya [35].....	41
18. ábra. Kívánság-funkció nyíldiagram	42
19. ábra. A koncepcionális tervezői fázis lépései, egyszerűsített folyamatábra	44
20. ábra. A CAD fejlődése.....	44
21. ábra. CACB (Computer Aided methodological Concept Building), javaslat a számítógépes eszközrendszer kiterjesztésére	45
22. ábra. A számítógéppel segített módszeres koncepcióépítés javasolt logikai lépései.....	46
23. ábra. Megoldásváltozatok generálása	49
24. ábra. A bináris és a véletlenszerű generálással elérhető megoldásváltozatok száma	50
25. ábra. Funkcionális részegységek fontossága, az egyes megoldásváltozatok bonyolultsága ..	52
26. ábra. A Visual Concept névjegye	55
27. ábra. A szoftver funkcionális modellje.....	56
28. ábra. A szoftver egyszerűsített folyamatábrája	57
29. ábra. A Visual Concept felhasználói felülete	58
30. ábra. A Visual Concept menüstruktúrája.....	59
31. ábra. A táblázatok eszköztár táblázatai.....	60
32. ábra. A megoldástér és egy generált megoldás a hozzá tartozó kapcsolatmátrix-szal.....	62
33. ábra. A kiindulási projekt.....	64
34. ábra. A kiindulási projekt a struktúra gráf szűkítése után	67
35. ábra. A futási eredmények	68

FELHASZNÁLT IRODALOM

- [1] ALEXANDER, C.: Notes on the Synthesis of Forms, ISBN 0 674 62751 2, Harvard University Press, Cambridge, Massachusetts, 1964.
- [2] ÁLMOS, A.; GYÖRI, S.; HORVÁTH, G.; VÁRKONYINÉ, K. A.: Genetikus Algoritmusok, ISBN 963 9326 453, Typotex Kiadó, Budapest, 2002.
- [3] ALTSHULLER, G. S.; SHAPIRO, R.V.: About a technology of creativity, Questions of Psychology, No.6., 1956.
- [4] ALTSHULLER, G. S.: Creativity as an Exact Science—The Theory of the Solution of Inventive Problems, ISBN 0 677 21230 5, Gordon and Breach Science Publishers, New York, 1984.
- [5] ANDREASEN, M. M.: Design Methodology?, Journal of Engineering Design, Vol 2, Issues:4, 1991.
- [6] ARCHER, L. B.: A view of the Nature of design research, In: Design: Science: Method, Edited by R. Jacques and J. Powell, ISBN 0 86103 047 8, Westbury House, Surrey, 1981.
- [7] BACH, C.: Die Maschinenelemente, Arnold Bergsträsser Verlagsbuchhandlung, Stuttgart, 1880.
- [8] BEITZ, W.; GROTE, K.-H.: Dubbel–Taschenbuch für den Maschinenbau, ISBN 3 540 62467 8, 19. kiadás, Springer Verlag, Berlin, 1997.
- [9] BERCSEY, T.; HORVÁTH, I.: A korszerű géptervezés feltételei, módszerei és eszközrendszere, GÉP, Vol. 37., No 11., 1985.
- [10] BERCSEY, T.: A tervezéelmélet és a CAD oktatása, GÉP, Vol. 40., No 10., 1988.
- [11] BERCSEY, T.: A gépszerkesztés tudományának fejlődése és jellemző vonásai. GÉP Vol. 42., No. 2., 1990.
- [12] BERCSEY, T.; VAJNA, S.: Ein autogenetischer Ansatz für die Konstruktionstheorie. Teil I. CAD-CAM Report, Vol. 13., Nr. 2. 1994., Teil II. CAD-CAM Report, Vol. 13., Nr. 3. 1994.
- [13] BERCSEY, T.; LŐRINCZ, S.: A terméktervezés megújulás: az új alapelvek és integrált módszerek, Gyártástechnológia, No 7-8., 1996.
- [14] BERCSEY, T.; DÖBRÖCZÖNI, Á.; DUPCSÁK, ZS.; HORÁK, P.; KAMONDI, L.; KELEMEN, T.; PÉTER, J.; TÓTH, J.: Terméktervezés és fejlesztés, PHARE TDQM, Budapest, 1997.
- [15] BERCSEY, T.; DÖBRÖCZÖNI, Á.; DUPCSÁK, ZS.; HORÁK, P.; KAMONDI, L.; PÉTER, J.; SCHOLTZ, P.: Új termék kifejlesztése és bevezetése, a piacra vitel ideje és az azt meghatározó tényezők, PHARE TDQM, Miskolc, 1997.
- [16] BINIEK, G.: Konstruktionssystematik. Feingerätetechnik, 1952.

- [17] BIRKHOFFER, H.: Analyse und Synthese der Funktionen technischer Produkte, Dissertatiton, TU Braunschweig, 1980.
- [18] BROWN, D. C.: Defining Configuring, AI EDAM, Vol. 12., Cambridge, 1998.
- [19] CROCE, B.: Guide to Aesthetics, fordította: Patric Romanell, ISBN 0 87220 304 2, Hackett Publishing Company inc., Indianapolis, 1995
- [20] CROSS, N.: History of design methodology, Behavioral and social Sciences, Vol. 71., Kluwer Academic Publishers, Netherlands, 1993.
- [21] CROSS, N.: Engineering Design Methods–Strategies for Product Design, ISBN 0 471 94228 6, John Wiley & Sons, Chichester, 1994.
- [22] CROSS, N.: Developments in Design Methodology, ISBN 0 471 10248 2, John Wiley & Sons, Chichester, 1984.
- [23] CROSS, N.; CROSS, A.: Expertise in Engineering Design, Research in Engineering Design, pp.: 141-148, Springer-Verlag London, 1998.
- [24] CROSS, N.: Natural intelligence in design, Design Studies, Vol. 20., No. 1., pp.: 25-39, 1999.
- [25] CROSS, N.: Design Research: A Disciplined Conversation, Design Issues, Vol. 15., No. 2., pp.: 5-10, 1999.
- [26] CROSS, N.: Designerly Ways of Knowing: Design Discipline versus Design Science, Design Issues, Vol. 17., No. 3., pp.: 49-55, 2001.
- [27] CROSS, N.: Can a Machine Design?, Design Issues, Vol. 17., No. 4., pp.: 44-50, 2001.
- [28] CROSS, N.; DORST, K.: Creativity in the design process: co-evolution of problem-solution, Design Studies, Vol. 22., No. 5., pp.: 425-437, 2001.
- [29] CROSS, N.: Design as a Discipline, The Inter-disciplinary Design Quandary Conference, De Montfort University, 2002. február 13.
- [30] CROSS, N.: The Expertise of Exceptional Designers, Expertise in Design, ISBN 0 9751533 0 7, Creativity and Cognition Press, University of Technology, Sydney, Australia, pp.: 23-35, 2003.
- [31] CROSS, N.: Expertise in design: an overview, Design Studies, Vol. 25., No. 5., pp.: 427-441, 2004.
- [32] CROSS, N.; KRUGER, C.: Solution driven versus problem driven design: strategies and outcomes, Design Studies, Vol. 27., No. 5., pp.: 527-548, 2006.
- [33] CROSS, N.: Forty years of design research, Design Studies, Vol. 28., No. 1., pp.: 1-4, 2007.
- [34] CSERESNYÉS, Á.; DELI, L.; FARKAS, J.; HORÁNYI, I.; KOCSIS, J.; CSEH, I.; OSMAN, P.; PAPP, O.; SEREGI, F.; SZÁNTÓ, B.: Innováció, ISBN 963 10 7128 6, Műszaki Könyvkiadó, Budapest, 1987.
- [35] DECIU, E. R.; OSTROSI, E.; FERNEY, M.; GHEORGHE, M.: Configurable product design using multiple fuzzy models, Journal of Engineering Design, ISSN 0954-4828, Vol. 16., No. 2., Taylor and Francis, 2005.
- [36] DÖBRÖCZÖNI, Á.; TERPLÁN, Z.; APRÓ, F.; ANTAL, M.: Fogaskerék-bolygóművek, ISBN 963-10-2208-0, Műszaki Könyvkiadó, Budapest, 1979.
- [37] EDELMANN-NUSSER, J.; HELLER, M.; CLEMENT, S.; VAJNA, S.; JORDAN, A.: Lightweight design optimization of a Bow Riser in Olympic Archery applying evolutionary computing, European Journal of Sport Science, ISSN 1536-7290, Vol. 4., Issue 3., Taylor and Francis, 2004.

- [38] EDER, W. E.: Design Engineering–Not Just Applied Science, CDEN-CCEE Conference, Winnipeg, 2007.
- [39] EDER, W. E.: Designing as an Educational Discipline, Engineering Education, Vol. 15., Nr. 1., 1999.
- [40] EDER, W. E.: Case Study in Design Engineering, CDEN-RCCI Conference, Toronto, 2006.
- [41] ENCARNACAO, J.; SCHLECHTENDAHL, E. G.: CAD, számítógéppel segített tervezés, ISBN 963 10 7195 2, Műszaki Könyvkiadó, Budapest, 1987.
- [42] ENDREI, W.: Műszaki Mendemondák, Műszaki Könyvkiadó, Budapest, 1985.
- [43] ERKENS, A.: Beiträge zur Konstruktionserziehung, Z. VDI 72., pp.:17-21. 1928.
- [44] FEKETE, I.; GREGORITS T.; NAGY S.: Bevezetés a mesterséges intelligenciába, ISBN 963 576 015 9, Budapest, 1990.
- [45] FRENCH, M. J.: Conceptual Design for Engineers, ISBN 0 85072 155 5, The Design Council, London, 1985.
- [46] FURKA, Á.: Forradalom a gyógyszerkutatásban, Mindentudás Egyeteme, 2003.
- [47] FURKA, Á.: A kombinatorikus kémia, Természet világa, Vol. 131, 2000.
- [48] FURKA, Á.: Visszapillantás a kombinatorikus kémia kialakulásának korai időszakára, Kémikusok Lapja, Vol. 56, 2001.
- [49] GASPARI, W.: Editorial: Contributions to Design Science, Design Methods and Theories, Vol 24, Issue 2, 1990.
- [50] GASPARI, W.: On the General Theory (Praxeology) of Design, Design Methods and Theories, Vol 24, Issue 2, 1990.
- [51] GLYNN, S.: Science and Perception as Design, Design Studies, Vol 6, Issue 3, 1985.
- [52] GRANT, D. P.: Design Methodology and Design Methods, Design Methods and Theories, Vol 13, Issue 1, 1979.
- [53] GREGORY, S. A.: The Design Method, Butterworth Press, London, 1966
- [54] GROH, W.: Die Technische Zeichnung, VEB Verlag Technik, Berlin, 1968.
- [55] HANSEN, F.: Konstruktionssystematic– Grundlagen für eine allgemeine Konstruktionslehre, ETO 621.002.2, VEB Verlag Technik, Berlin, 1965.
- [56] HEGEDŰS, J.: Értékelemzés a termékfejlesztésben, ISBN 9631057844, Műszaki Könyvkiadó, Budapest, 1983.
- [57] HENTSCHEL, C.: Methodisches Erfinden mit TRIZ zur Steigerung der Innovationsfähigkeit, Konstruktion, Springer-Verlag, pp.: 75-78., 2004.
- [58] HOLZNER, S.: Visual Basic .NET Fekete Könyv, ISBN 963 202 842 2 Ö, Perfact-Pro Kft., Budapest, 2002.
- [59] HORVÁTH, GY.: A tartalmas gondolkodás, ISBN 963 17 7314 0, Tankönyvkiadó, Budapest, 1984.
- [60] HORVÁTH, I.; JUHÁSZ I.: Számítógéppel segített gépészeti tervezés, ISBN 963 16 1051 9, Műszaki Könyvkiadó, Budapest, 1996.
- [61] HORVÁTH, P.: Mesterséges intelligencia módszerek alkalmazása a szerszámgéptervezésben, Kandidátusi értekezés, Miskolc, 1994.
- [62] HUBKA, V.: Theorie der Maschinensysteme– Grundlagen einer wissenschaftlichen Konstruktionslehre, ISBN 3 540 06122 3, Springer-Verlag, Berlin, 1973

- [63] HUBKA, V.: Theorie der Konstruktionsprozesse– Analyse der Konstruktionstätigkeit, ISBN 3 540 07767 7, Springer-Verlag, Berlin, 1976
- [64] HUBKA, V.; EDER, W. E.: Einführung in die Konstruktionswissenschaft, ISBN 3 540 54832 7, Springer-Verlag, Berlin, 1992
- [65] IVÁNYI, A. SZ: Értékelemzés, Tankönyvkiadó, Budapest, 1985. (Egyetemi jegyzet)
- [66] IVÁNYI, A. SZ: Az értékelemzés módszertana, Tankönyvkiadó, Budapest, 1987. (Egyetemi jegyzet)
- [67] IVÁNYI, A. SZ: Értékelemzés racionalizálás a gyártástervezésben, ISBN 963 221 617 2, Közgazdasági és jogi könyvkiadó, Budapest, 1985.
- [68] KAMONDI, L.: Tervezéselmélet, Phare HU0008-02, Miskolc 2003.
- [69] KESSELRING, F.: Die starke Konstruktion, 1942.
- [70] KESSELRING, F.: Bewertung von Konstruktionen, 1951.
- [71] KESSELRING, F.: Technische Kompositionslehre, Springer-Verlag, Berlin, 1954.
- [72] KOLLER, R.: Konstruktionsmethode für den Maschinen-, Geräte- und Apparatebau, Springer-Verlag, Berlin, 1979.
- [73] KOLLER, R.: Konstruktionslehre für den Maschinenbau–Grundlagen zur Neu- und Weiterentwicklung technischer Produkte, Springer-Verlag, Berlin, 1994.
- [74] KOTLER, P.: Marketing menedzsment, ISBN 963 16 3026 9, Műszaki Könyvkiadó, Budapest, 2001.
- [75] LEVY, R.: Science, Technology and Design, Design Studies, Vol 6, Issue 2, 1985.
- [76] LEYER, A.: Maschinenkonstruktionslehre, Birkhauser Verlag, Stuttgart, 1963-1971.
- [77] LINDE, H.; HILL, B.: Erfolgreich erfinden –Widerspruchsorientierte Innovationstrategie für Entwickler und Konstrukteure, Hoppenstedt Technik Tabellen Verlag, Darmstadt, 1993.
- [78] LIPÓTH, A.: Megmunkáló központ konstrukciós változatok módszeres előállítása és értékelése, kandidátusi értekezés, Budapest, 1993.
- [79] MATOUSEK, R.: Konstruktionslehre des allgemeinen Maschinenbaus, Springer Verlag, Berlin, 1957.
- [80] MILES, L. D.: Értékelemzés, ISBN , Közgazdasági és Jogi Könyvkiadó, Budapest, 1973
- [81] MITTAL, S.; FRAYMAN, F.: Towards a Generic Model of Configuration Tasks, IJCAI, Vol 2., 1989.
- [82] NAGY, T.; GAULT, D.; NAGY, M.: Így készül a szakértői rendszer, ISBN 963 026 112, Novotrade, 1988.
- [83] NIEMANN, G.: Maschinenelemente, Springer Verlag, Berlin, 1950.
- [84] OTTO, K.; WOOD, K.: Product Design– Techniques in Reverse Engineering and New Product Development, ISBN 0 13 021271 7, Prentice Hall, 2001.
- [85] PAHL, G.; BEITZ W.: Konstruktionslehre– Handbuch für Studium und Praxis, ISBN 963 10 3796 7, Springer-Verlag, Berlin, 1981., 2003., 2005., 2007.
- [86] PATURI, F. L.: A technika krónikája, ISBN 963 7836 44 6, Officina Nova Könyv- és Lapkiadó Kft., 1991.
- [87] PÓLYA, GY.: A gondolkodás iskolája–hogyan oldjunk meg feladatokat?, ISBN 963 7803 75 0, Akkord Kiadó, 2000.

- [88] PÓLYA, GY.: Matematikai módszerek a természettudományban, ISBN 963 281 321 9, Gondolat Kiadó, 1984.
- [89] PUGH, S.: Total design– Integrated methods for successful product engineering, ISBN 0 201 41639 5, Addison-Wesley Publishers Ltd., 1991.
- [90] REULEAUX, F.: Der Consturcteur, Friedrich Vieweg und Sohn, Braunschweig, 1865.
- [91] REDTENBACHER, F.: Prinzipien der Mechanik und Maschinenbaues, Friedrich Bassermann, Mannheim, 1852.
- [92] RODENACKER, W. G.: Methodisches Konstruieren, 2. kiadás, ISBN 3 540 07513 5, Springer Verlag, Berlin, 1976
- [93] ROTH, K.: Konstruieren mit Konstruktionskatalogen, ISBN 963 10 7246 0, VEB Verlag Technik, Berlin, 1982.
- [94] RULE, J. T.; WATTS, E. F.: Engineering Graphics, McGraw-Hill Book Company Inc., London, 1951.
- [95] SCHÖN, D. A.: Problems, frames and perspectives on designing, Design Studies, Vol 5, Issue 3, 1984.
- [96] SCHÖN, D. A.: Designing: rules, types and worlds, Design Studies, Vol 9, Issue 3, 1988.
- [97] SELYE, J.: Álomtól a felfedezésig – Egy tudós vallomásai, Akadémiai Kiadó, Budapest, 1967.
- [98] SHIGLEY, J. E.: Machine Design, McGraw-Hill Book Company, Inc., London, 1956.
- [99] SIMON, H. A.: The Sciences of the Artificial, MIT Press, Cambridge, 1969.
- [100] SUH, N. P.: Axiomatic Design: Advances and Applications, ISBN 0 19 513466 4, Oxford University Press, 2001.
- [101] SZAKÁLY, D.: Innováció- és technológia menedzsment, ISBN 963 9103 89 6, Bíbor Kiadó, Miskolc, 2002.
- [102] TAJNAFŐI, J.: Mechanizmusok származtatáselméletének alapjai és hatása a kreatív gondolkodásra, Akadémiai doktori értekezés, Miskolc, 1991.
- [103] TAJNAFŐI, J.: Szerszámgéptervezés I., Nemzeti Tankönyvkiadó, Budapest 1973.
- [104] TAJNAFŐI, J.: Szerszámgéptervezés II., Nemzeti Tankönyvkiadó, Budapest 1990.
- [105] TAKÁCS, GY.: Szerszámgépek strukturális tervezése grafikus adatbázisokkal, egyetemi doktori értekezés, Miskolc, 1996.
- [106] TIIHONEN, J.; SOININGEN, T.; MÄNNISTÖ, T.; SULONEN, R.: State-of-the-practice in product configuration–a survey of 10 cases in the Finnish industry, Knowledge Intensive CAD, Vol. 1, Chapman and Hall, 1996.
- [107] TIIHONEN, J. ET AL: Modeling configurable product families, 4th WDK Workshop on Product Structuring, 1998.
- [108] USHER, J. M.; ROY U.; PARSAEI H. R.: Integrated Product and Process Development – Methods, Tools, and Technologies, ISBN 0 471 15597 7, John Wiley & Sons, Chichester, 1998.
- [109] VAJNA, S.; BERCSEY, T.; CLEMENT, S.; JORDAN, A.; MACK, P.: Autogenetische Konstruktionstheorie ein Beitrag für eine erweiterte Konstruktionstheorie, Konstruktion, Springer-Verlag, 2004. március 3., 71-77. oldal
- [110] VITRUVIUS: Tíz könyv az építészetről, ISBN 963 336 159 1, Budapest, 1988.
- [111] VDI 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, 1993.

- [112] VDI 2222 Blatt 1: Konstruktionsmethodik – Methodisches Entwickeln von Lösungsprinzipien, 1997.
- [113] VDI 2222 Blatt 2: Konstruktionsmethodik; Erstellung und Anwendung von Konstruktionskatalogen, 1982.
- [114] VDI 2223: Methodisches Entwerfen technischer Produkte, 2004
- [115] VDI 2225 Blatt 3: Konstruktionsmethodik - Technisch-wirtschaftliches Konstruieren – Technisch-wirtschaftliche Bewertung, 1998.
- [116] WILLEM, R. A.: Design and Science, Design Studies, Vol 11, Issues 1, 1990.
- [117] WÖGERBAUER, H.: Die Technik des Konstruierens, München, Berlin, 1943.
- [118] YOSHIKAWA, H.: General Design Theory as a Formal Theory of Design, Intelligent CAD I: Proceedings of the IFIP TC5 Wg5.2 Workshop 1987, ISBN: 0444874747, Elsevier Science & Technology Books, 1989.
- [119] ZADEH, L. A.: Fuzzy Sets, Information and Control, Vol. 8., 1965.
- [120] ZADEH, L. A.: Outline of a New Approach to the Analysis of Complex Systems and Decision Processes, IEEE Transactions on Systems, Man, Cybernetics, Vol. 3., 1973.
- [121] ZADEH, L. A.: A fuzzy-algorithmic approach to the definition of complex or imprecise concepts, Intl. J. Man-Machine Studies, Vol. 8., 1976.
- [122] ZWICKY, F.: Morphological Astronomy, The Observatory, Vol. 68., 1948.

PUBLIKÁCIÓK AZ ÉRTEKEZÉS TÉMAJÁBAN

Külföldi folyóiratban megjelent lektorált szakkikk

- [TA 1] TAKÁCS, Á.- KAMONDI, L.: Design Science- A neuvel approach for the product design, Advanced Engineering, International Journal, ISSN 1846-5900, pp 303-314., 2008. No. 2.

Magyar nyelvű folyóiratban megjelent lektorált szakkikk

- [TA 2] TAKÁCS, Á.: A koncepcionális tervezés módszerei és irányelvei, algoritmikus megoldási lehetőségei, GÉP, ISSN 0016-8572, 2007/5-6, LVIII. évfolyam, pp. 62-70
- [TA 3] TAKÁCS, Á.- KAMONDI, L.: Számítógéppel segített koncepcionális terméktervezési lehetőségek, Műszaki szemle, ISSN 1454-0746, 2008/különszám, pp. 379-382

Tudományos közlemény, lektorált, magyar nyelvű konferencia kiadványban

- [TA 4] TAKÁCS, Á.- KAMONDI, L.: Számítógépre adaptált koncepcionális tervezési módszer, Géptervezők és Termékfejlesztők XXI. Országos Szemináriuma, 2005. november 10.-11., GÉP, ISSN 0016-8572, 2005/9-10, LVI. évfolyam, pp. 178-181
- [TA 5] TAKÁCS, Á.- SZABADOS, V.- KAMONDI, L.: Tervezési irányelvek és katalógusok a módszeres géptervezésben, Géptervezők és Termékfejlesztők XXII. Országos Szemináriuma, Miskolc, 2006. november 9.-10., GÉP, ISSN 0016-8572, 2006/9-10, LVII. évfolyam, pp. 181-185
- [TA 6] TAKÁCS, Á.- KAMONDI, L.: Tervezésmódszertan egy kicsit másképp, Géptervezők és Termékfejlesztők XXIII. Országos Szemináriuma, Miskolc, 2007. november 15.-16., GÉP, ISSN 0016-8572, 2007/10-11, LVIII. évfolyam, pp. 133-137

Tudományos közlemény, idegen nyelvű konferencia kiadványban

- [TA 7] TAKÁCS, Á.- KAMONDI, L.: Computer adaptierte Konstruktionsmethode für die konzeptionelle Konstruktion eines Laufbands, 3. Gemeinsames Kolloquium Konstruktionstechnik 2005, Magdeburg, Németország, 2005. június
- [TA 8] TAKÁCS, Á.- KAMONDI, L.: Computer aided morphological design, CADAM 2005. International Conference on Computer Aided Design and Manufacturing, Supetar, Horvátország, 2005. szeptember

- [TA 9] TAKÁCS, Á.- KAMONDI, L.: Synthesis of the intuitive and the discursive designer schools, PhD2005, 3rd International PhD Conference on Mechanical Engineering, Srni, Csehország, 2005 november 7-9.
- [TA 10] TAKÁCS, Á.- KAMONDI, L.: Functions and function-structures in the product design, microCAD 2006., Miskolc, Magyarország, 2006. március 16-17.
- [TA 11] TAKÁCS, Á.- KAMONDI, L.- PÉTER, J.: Browsing machine elements aided by a knowledge-based computer system, PhD2006, 4th International PhD Conference on Mechanical Engineering, Pilsen, Csehország, 2006. szeptember 11.-13.
- [TA 12] TAKÁCS, Á.- KAMONDI, L.: Algorithm-based morphological design procedures, microCAD2007, International Scientific Conference, Miskolc, 2007. március 22.-23.
- [TA 13] TAKÁCS, Á.- KAMONDI, L.: Conceptual product design theories adaptable for computers, 2nd Fatigue Symposium konferencia kiadványa, pp. 102-109, ISBN 978-3-902544-0, 2nd Fatigue Symposium, Leoben, Ausztria, 2008. április 23.-24.
- [TA 14] TAKÁCS, Á.- KAMONDI, L.: Conceptual design- or not?, 49th KCMS conference, ISBN 978-80-7043-718-6, pp 261-264, Srni, Csehország, 2008. 09. 08-10.

Tudományos közlemény, magyar nyelvű konferencia kiadványban

- [TA 15] TAKÁCS, Á.: Az intuitív és a diszkurzív tervezőrendszerek összehasonlítása, Doktoranduszok Fóruma, 2005. november 9.
- [TA 16] TAKÁCS, Á.- KAMONDI, L.: A genetikus algoritmusok, OGÉT2006, Nemzetközi Gépész Találkozó, Marosvásárhely, Románia, 2006 április 27-30.
- [TA 17] TAKÁCS, Á.: Gépelemek kiválasztásának támogatása tudásalapú számítógépes rendszer segítségével, Doktoranduszok Fóruma, 2006. november 9.

Szakmai tudományos előadás idegen nyelven

- [TA 18] TAKÁCS, Á.: Computer aided morphological design, CADAM 2005. International Conference on Computer Aided Design and Manufacturing, Supetar, Horvátország, 2005. szeptember
- [TA 19] TAKÁCS, Á.: Synthesis of the intuitive and the discursive designer schools, PhD2005, 3rd International PhD Conference on Mechanical Engineering, Srni, Csehország, 2005 november 7-9.
- [TA 20] TAKÁCS, Á.: Functions and function-structures in the product design, microCAD 2006., Miskolc, Magyarország, 2006. március 16-17.
- [TA 21] TAKÁCS, Á.: Browsing machine elements aided by a knowledge-based computer system, PhD2006, 4th International PhD Conference on Mechanical Engineering, Pilsen, Csehország, 2006. szeptember 11.-13.
- [TA 22] TAKÁCS, Á.: Algorithm-based morphological design procedures, microCAD2007, International Scientific Conference, Miskolc, 2007. március 22.-23.
- [TA 23] TAKÁCS, Á.: Conceptual design- or not?, 49th KCMS conference, Srni, Csehország, 2008. 09. 08-10.
- [TA 24] TAKÁCS, Á.: Design Science- A new approach for the product design, CADAM 2008 conference, Krk, Horvátország, 2008. 09. 16-20.

Szakmai tudományos előadás magyar nyelven

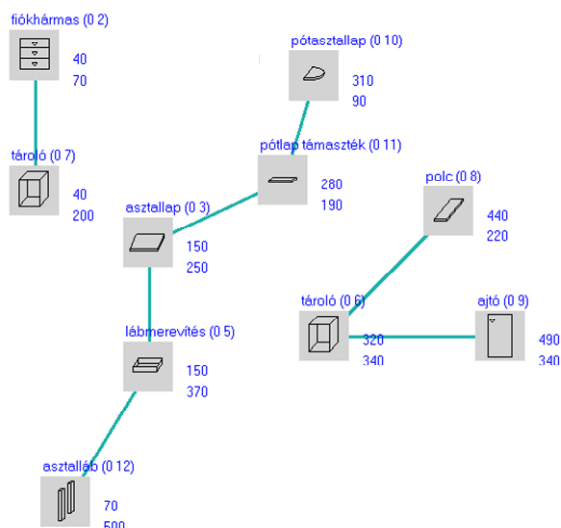
- [TA 25]TAKÁCS, Á.: Az intuitív és a diszkurzív tervezőrendszerek összehasonlítása, Doktoranduszok Fóruma, 2005. november 9.
- [TA 26]TAKÁCS, Á.: Számítógépre adaptált koncepcionális tervezési módszer, Géptervezők és Termékfejlesztők XXI. Országos Szemináriuma, 2005. november 10.-11.
- [TA 27]TAKÁCS, Á.: A genetikusan algoritmusok, OGÉT2006, XIV. Nemzetközi Gépész Találkozó, Marosvásárhely, Románia, 2006. április 27.-30.
- [TA 28]TAKÁCS, Á.: Gépelemek kiválasztásának támogatása tudásalapú számítógépes rendszer segítségével, Doktoranduszok Fóruma, 2006. november 9.
- [TA 29]TAKÁCS, Á.: Tervezési irányelvek és katalógusok a módszeres géptervezésben, Géptervezők és Termékfejlesztők XXII. Országos Szemináriuma, Miskolc, 2006. november 9.-10.
- [TA 30]TAKÁCS, Á.: Tervezésmódszertan egy kicsit másképp, Géptervezők és Termékfejlesztők XXIII. Országos Szemináriuma, Miskolc, 2007. november 15.-16.
- [TA 31]TAKÁCS, Á.: Számítógéppel segített koncepcionális terméktervezési lehetőségek, OGÉT 2008, XVI. Nemzetközi Gépész Találkozó, Brassó, Románia, 2008. május 1.-4.

9. MELLÉKLETEK

9.1. FUNKCIÓK KÖZTI KAPCSOLATOK SZERINTI GENERÁLÁS

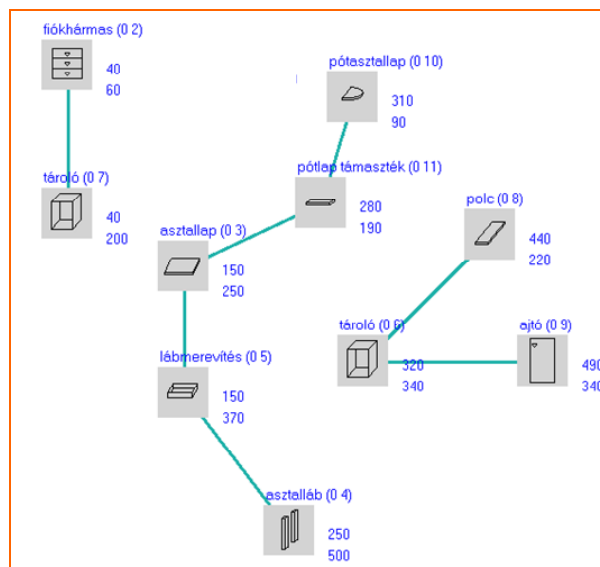
9.1.1. Elvileg helyes megoldások

S01



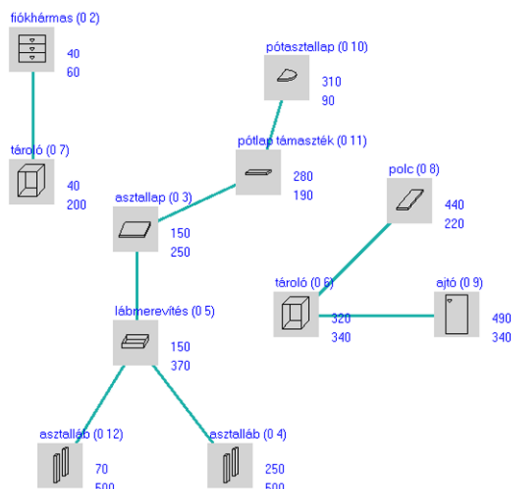
Struktúra egyenlet: F2-F7, F3-F5, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

S02



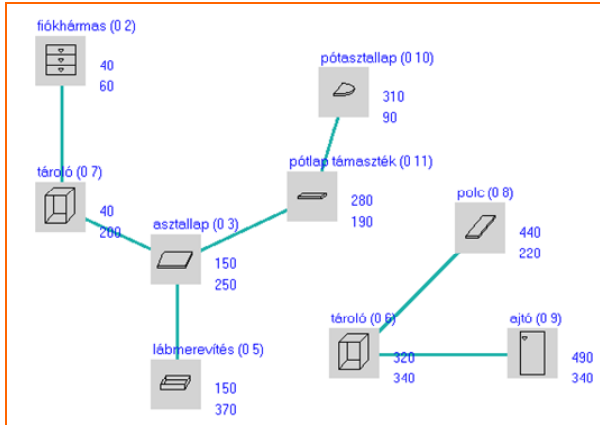
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S03



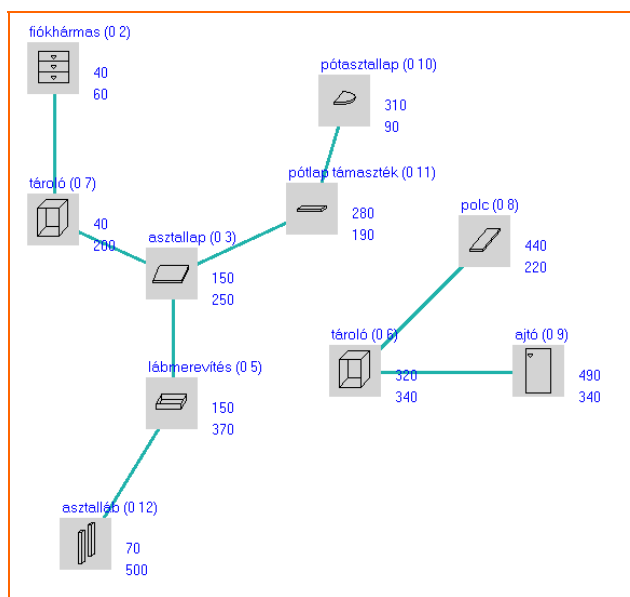
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S04



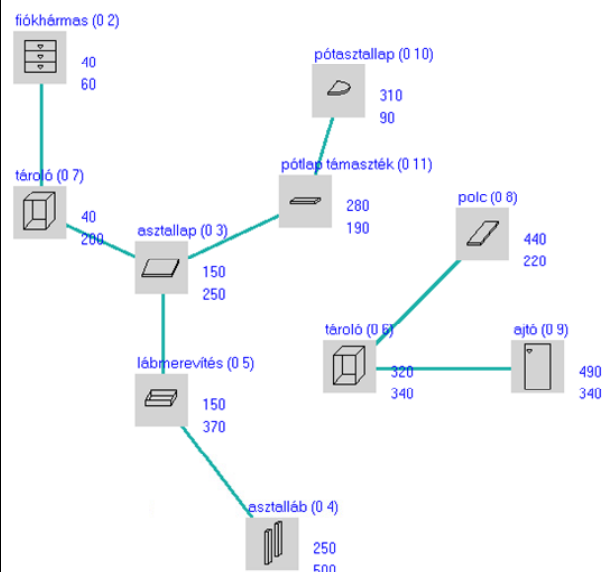
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

S05



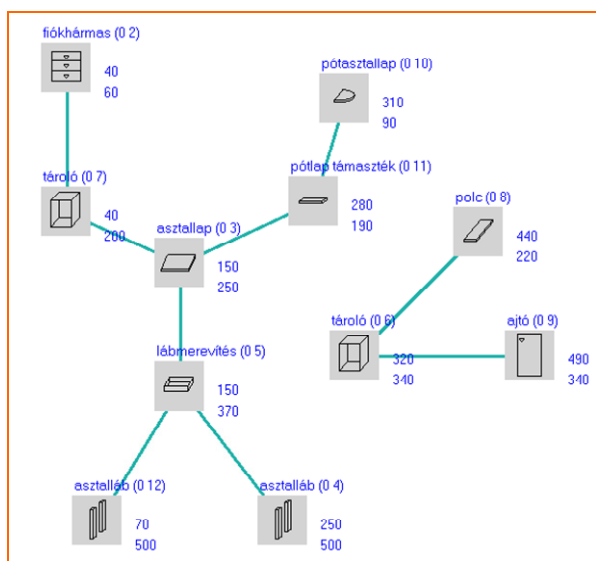
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

Struktúra: 06



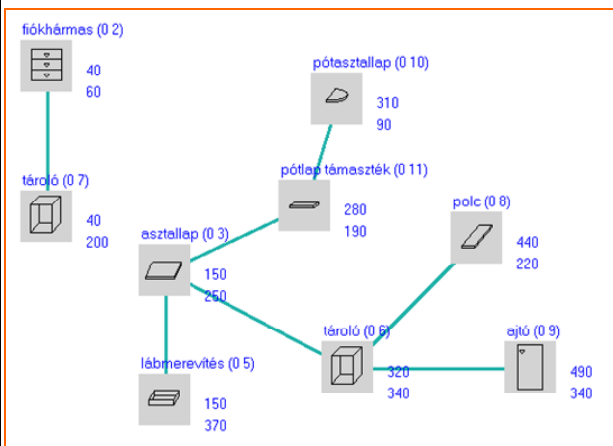
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S07



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

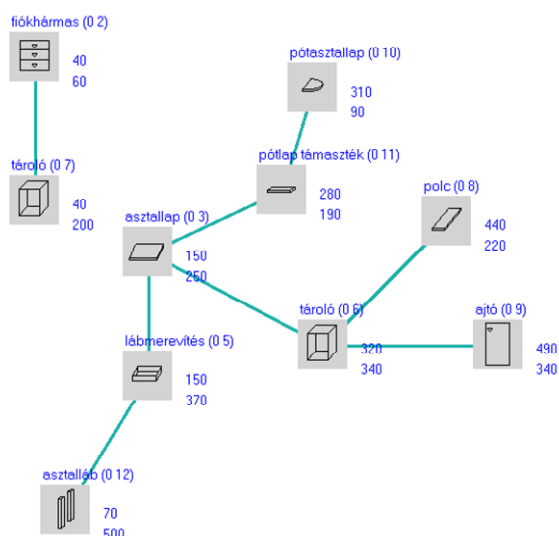
S08



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

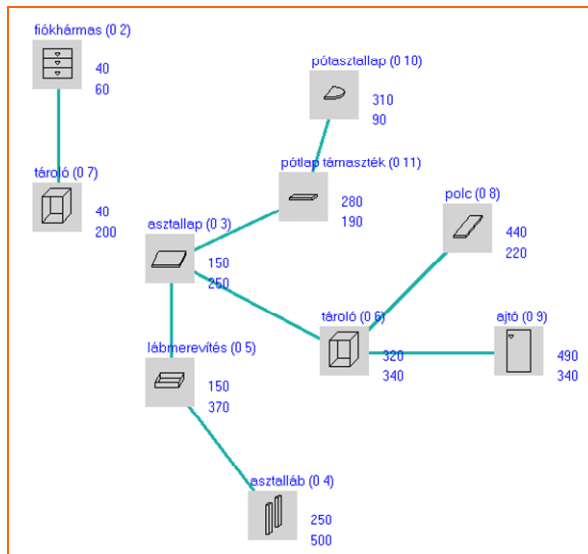
Az S01, S02, és S03 jelű megoldások nem feleltek meg a követelményeknek, mert mindkét tárolójuk az asztal egységtől elszeparáltan, mobil egységként jelenik meg. A funkcióösszevonást, mint gazdasági lépést mellőzik, mert lábakkal is el vannak látva ezek a megoldások. Az S05, S06 és S07 jelű megoldások szintén falra szerelhetőségi szempontból vannak túlhatározva. A követelményeknek való minél precízebb megfelelés szempontjából elég lenne egy pár hagyományos láb, vagy egy azokat helyettesítő tároló egység.

S09



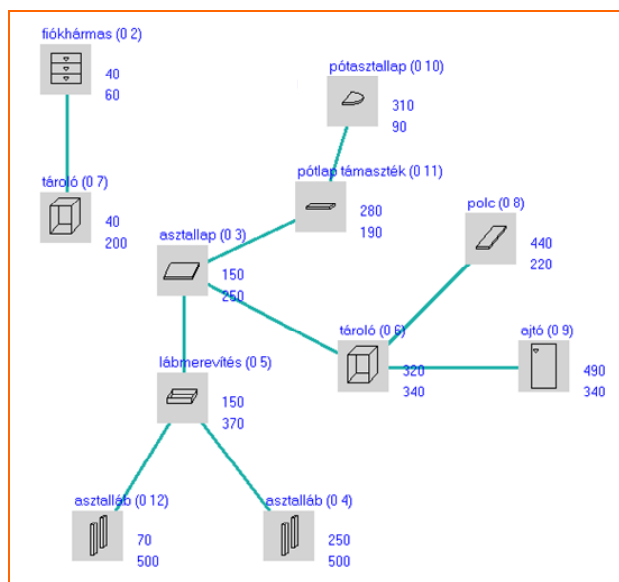
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

S10



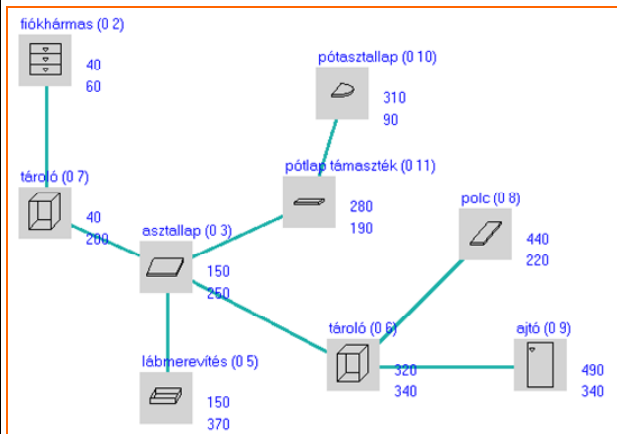
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S11



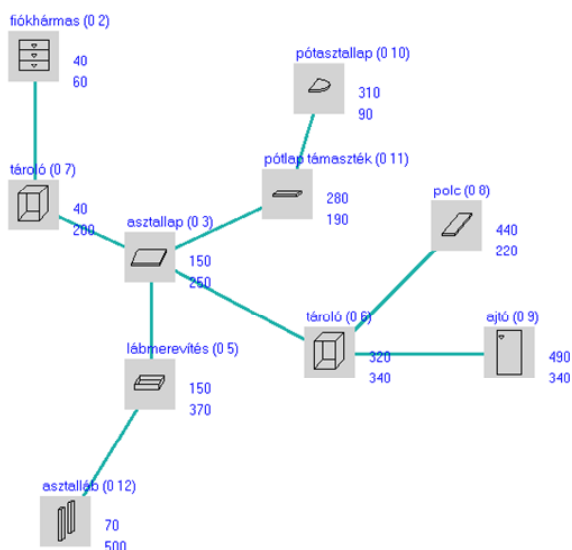
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S12



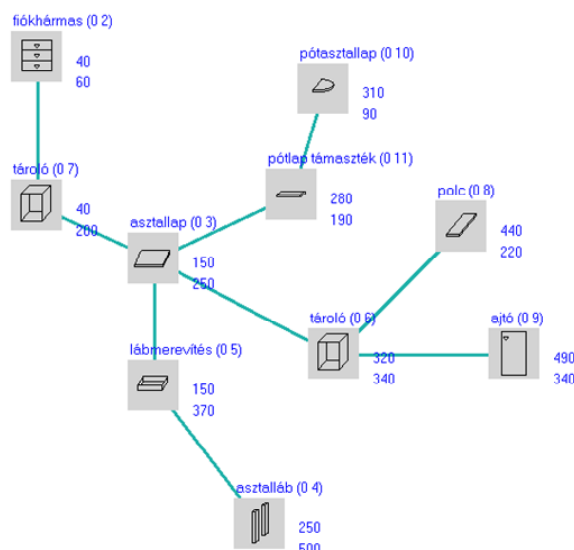
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

S13



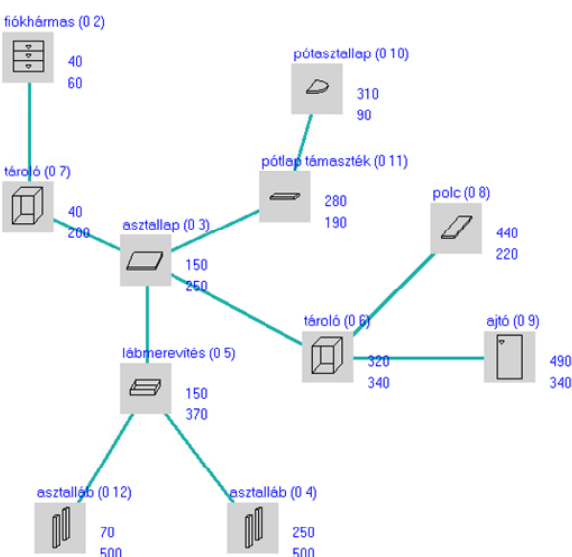
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

S14



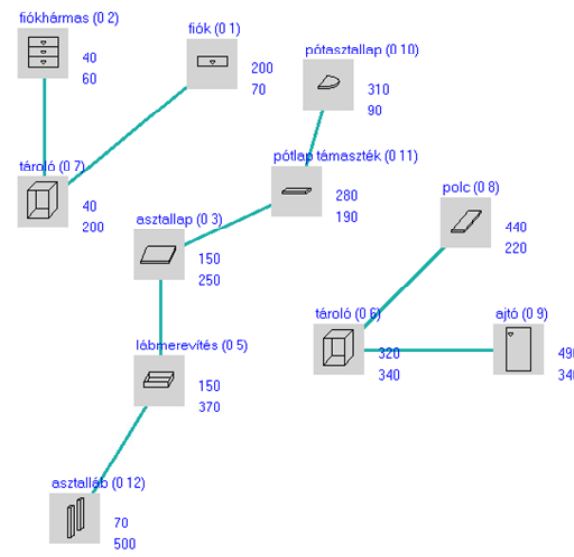
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S15



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

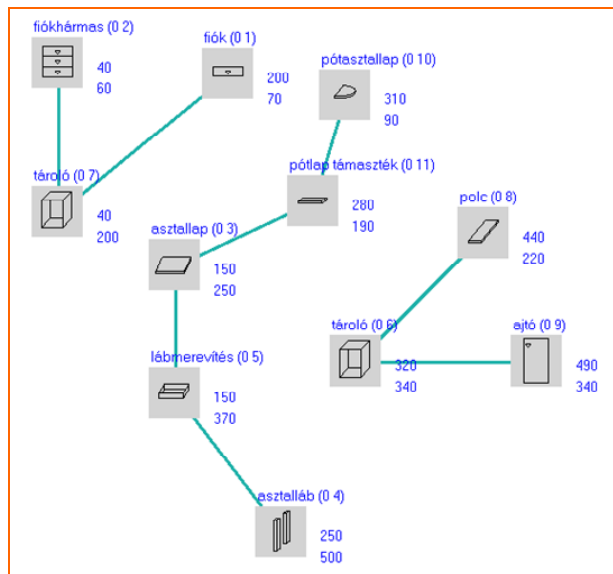
S16



Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

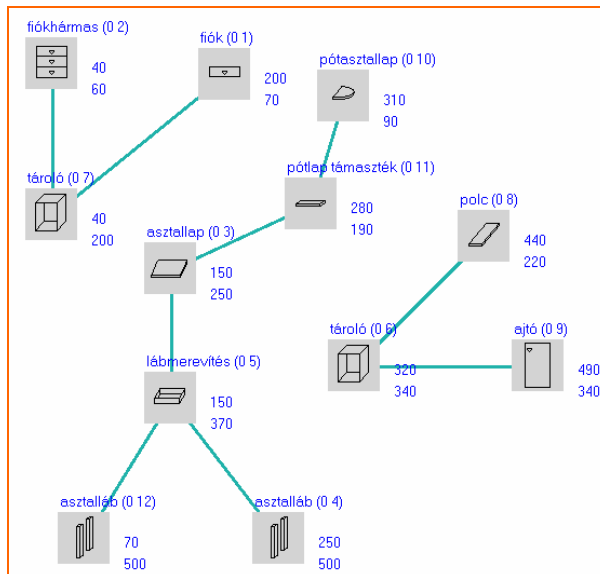
Az S09, S10, S11, S12, S13, S14, S15 és S16 megoldások túl sok alátámasztási lehetőséget tartalmaznak.

S17



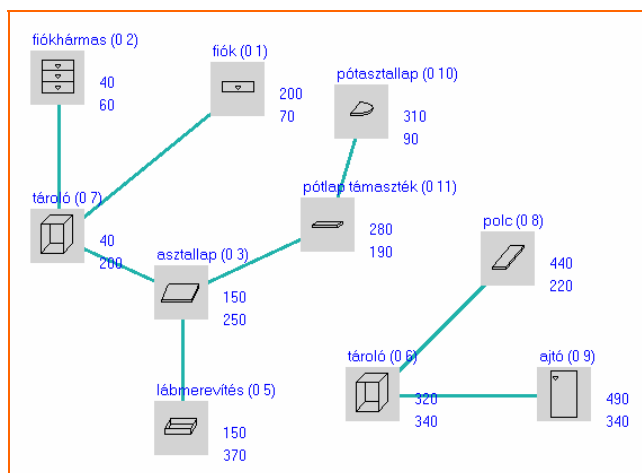
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S18



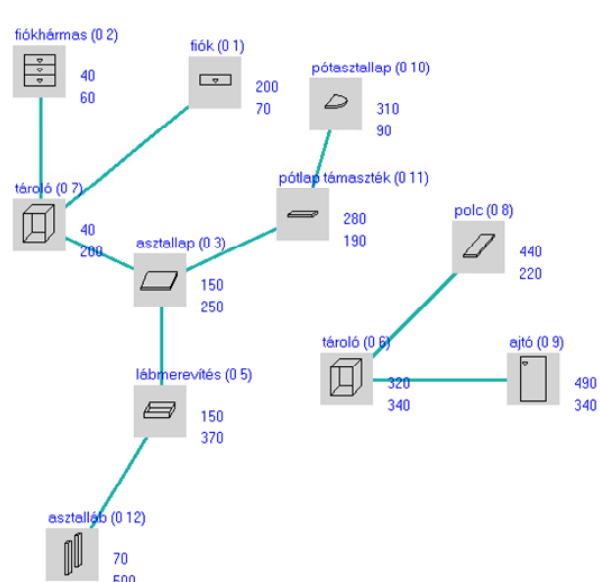
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S19



Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

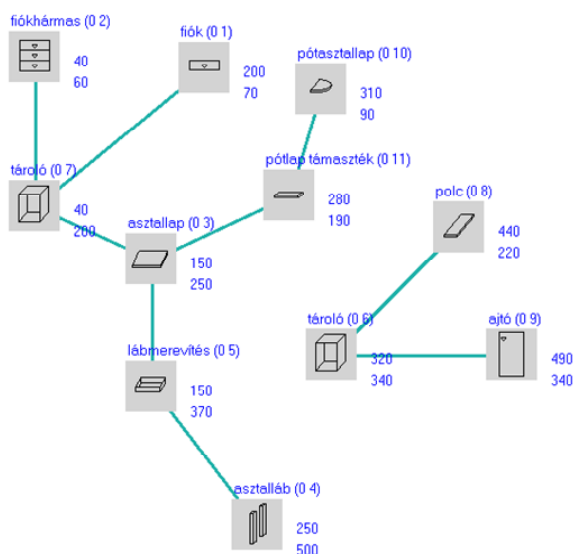
S20



Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

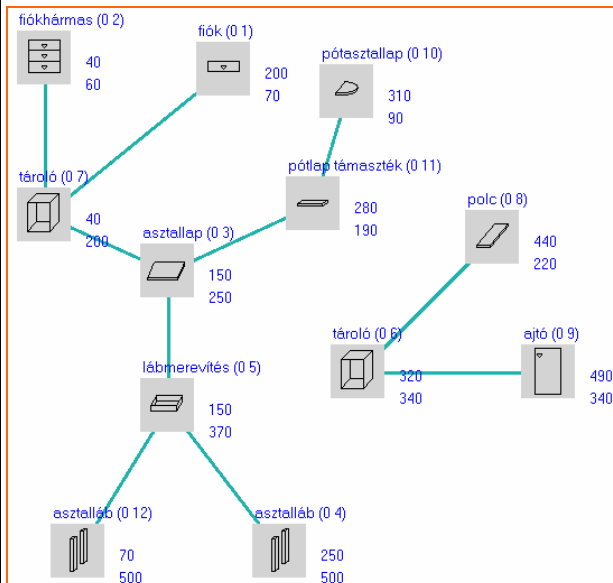
Az S17 és S18 esetében a funkció-összevonás feltétele nem teljesül. Az S18 esetében egy különálló (*nem falra szerelhető*) változatról van szó. Az S20 sok alátámasztási pontot tartalmaz.

S21



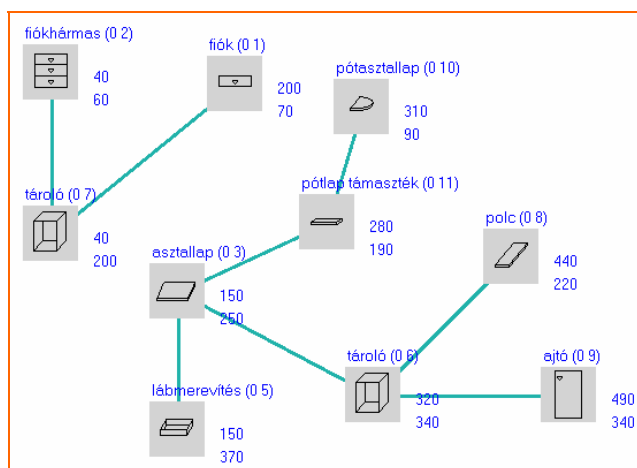
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S22



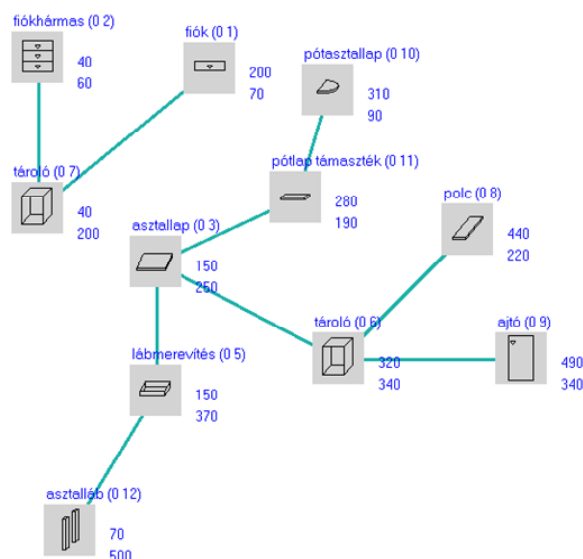
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S23



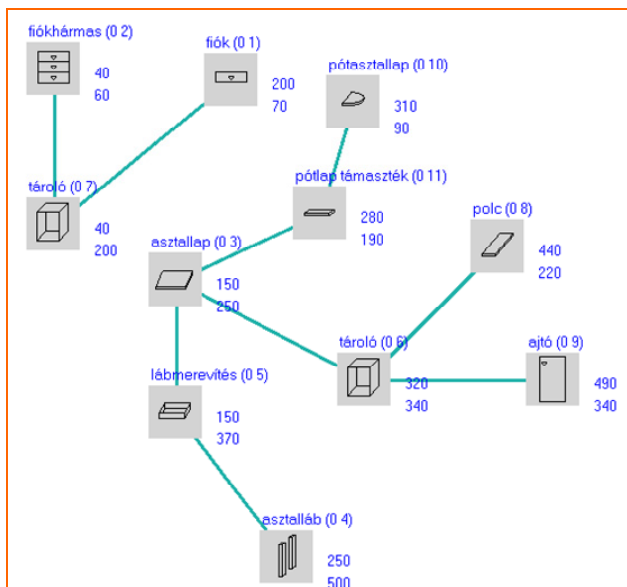
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F6-F9, F10-F11

S24



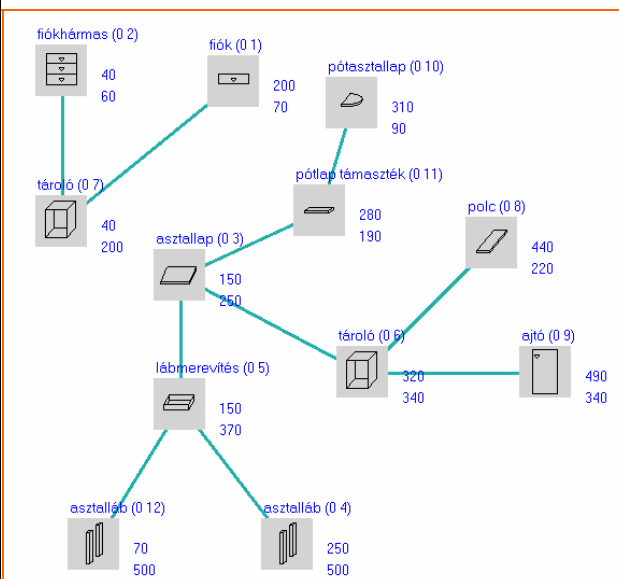
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

S25



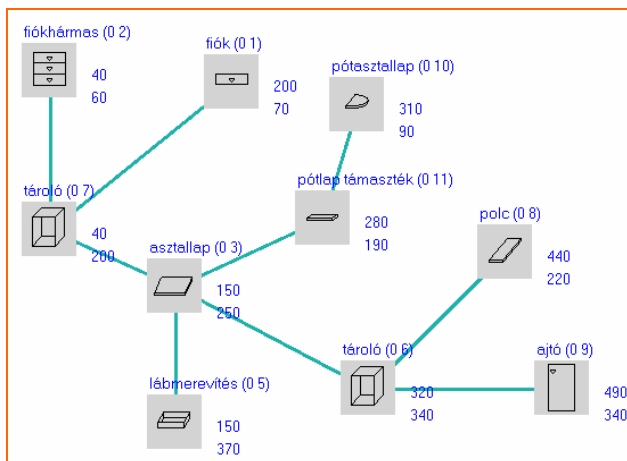
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S26



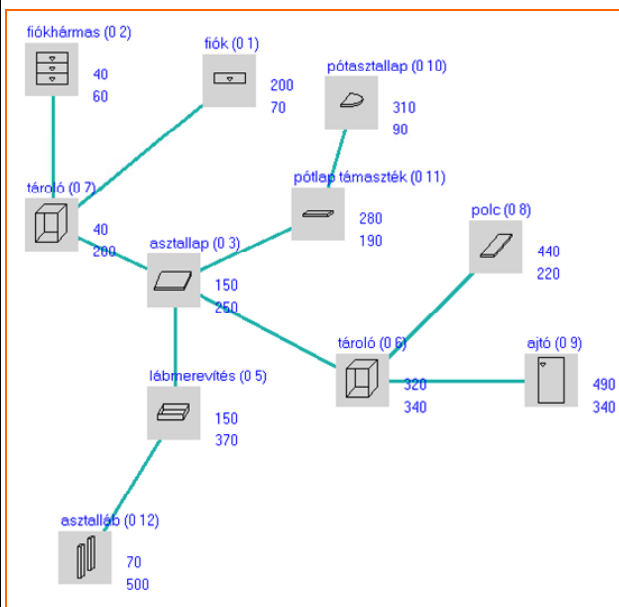
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S27



Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

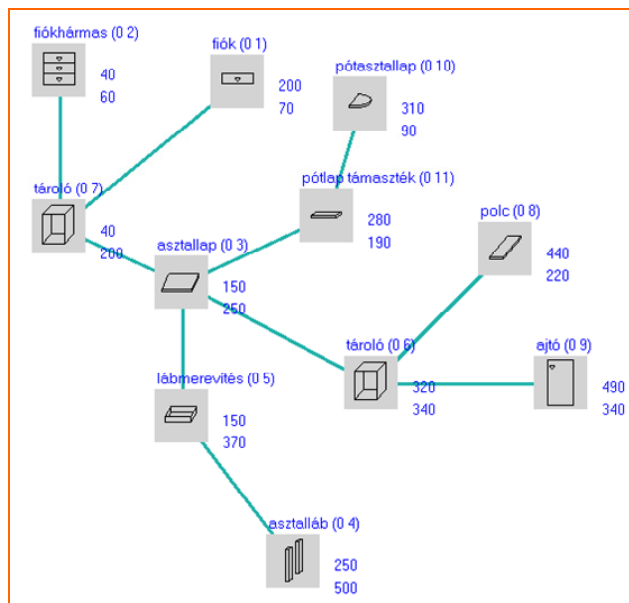
S28



Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

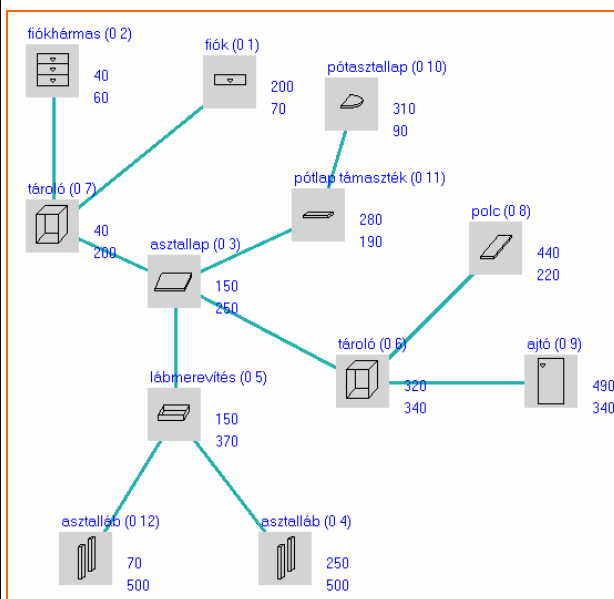
Az S21, S22, S24, S25, S26, S28, S29 és S30 sok alátámasztási pontot tartalmaz. Az S27 szintén túl sok alátámasztási pontot tartalmaz, mindemellett ha az egyik tároló mobil lenne, a követelményjegyzékben felsorolt elvárásoknak maximálisan eleget tenné.

S29



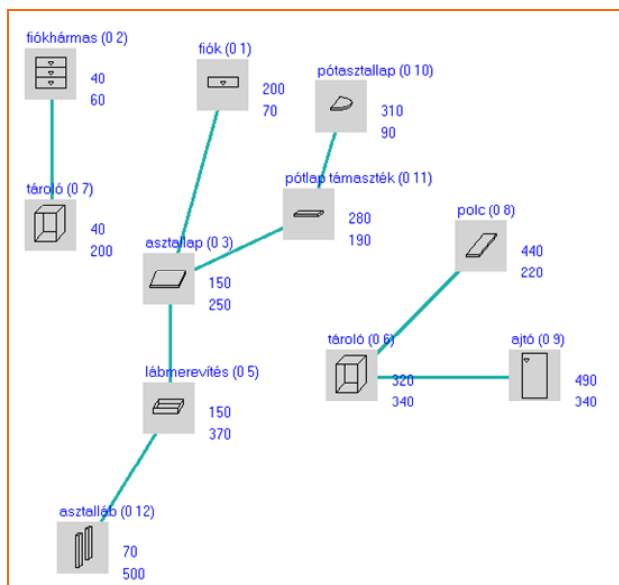
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S30



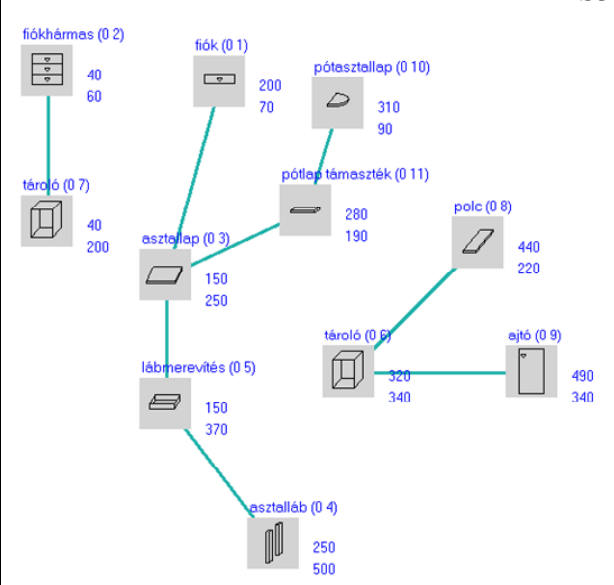
Struktúra egyenlet: F1-F7, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S31



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

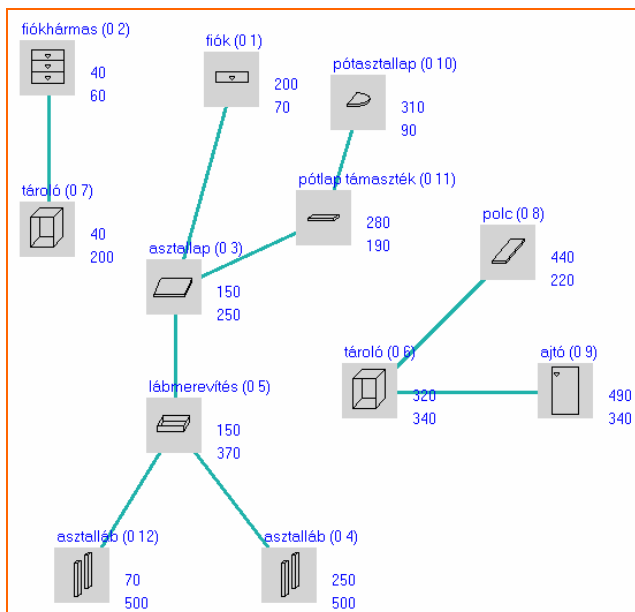
S32



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

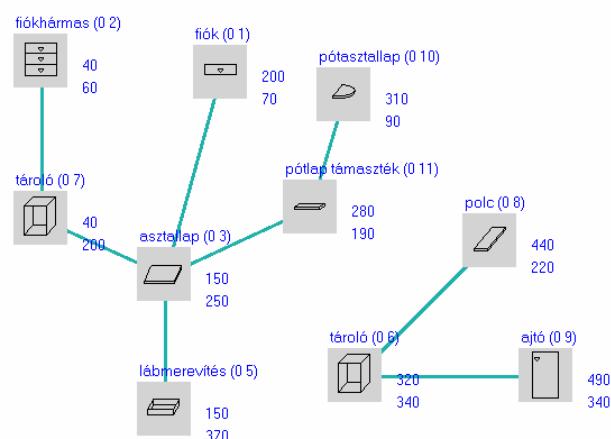
Az S31 és S32 a különálló tárolók miatt nem elégíti ki a funkció-összevonás feltételét.

S33



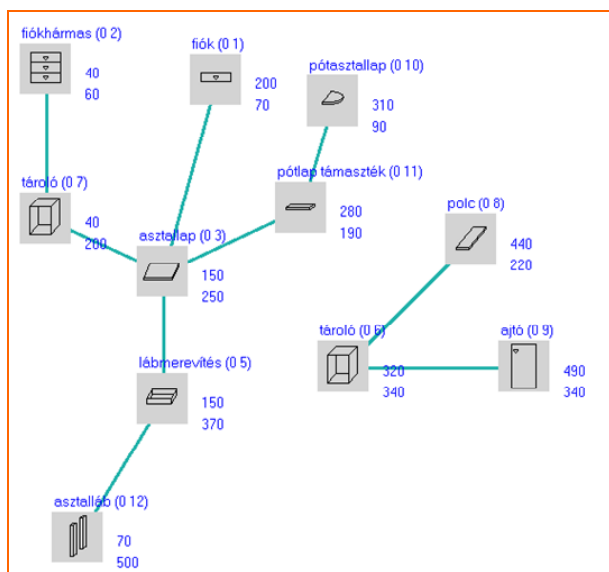
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S34



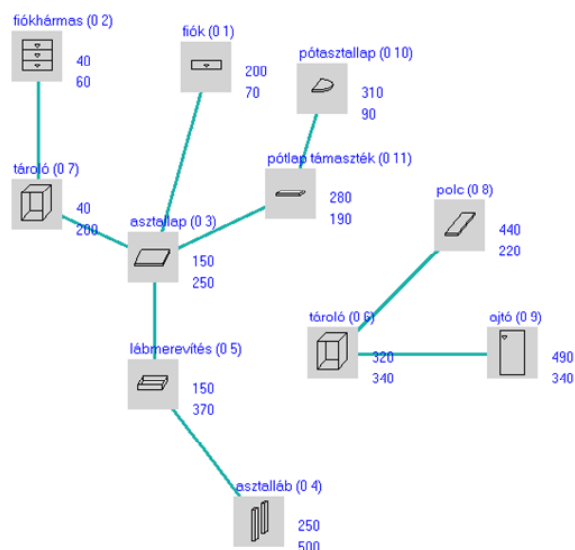
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

S35



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

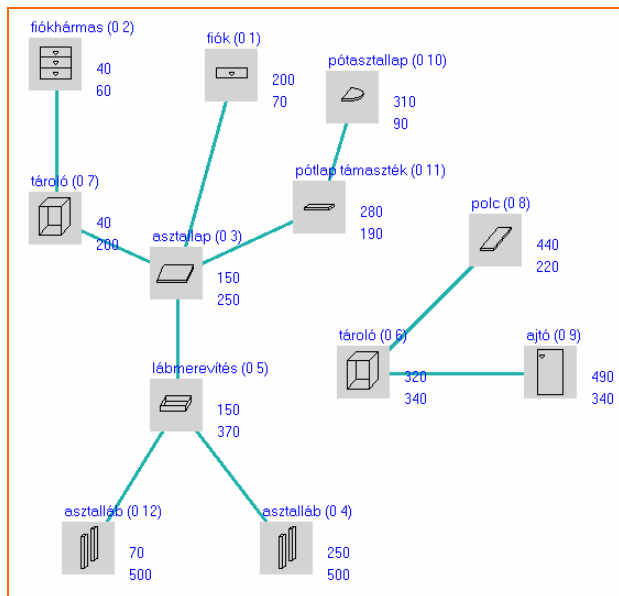
S36



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

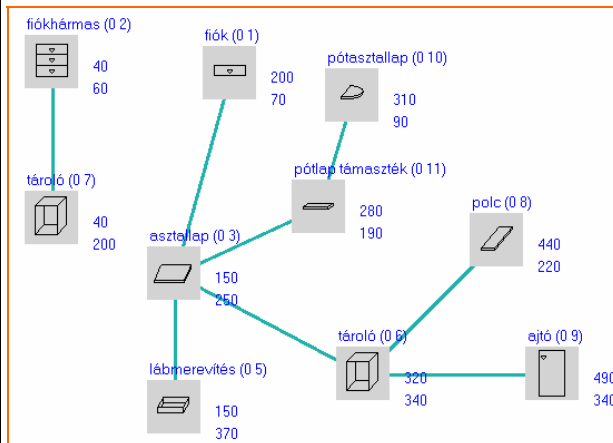
Az S33 egy különálló asztal is lehet, mert a hagyományos asztalokra jellemzően ez a termékstruktúra is 2 pár lábat tartalmaz. A Struktúra további hiányossága, hogy a tárolók külön vannak az asztal többi részétől, így nem helyettesíthetik a lábakat. Az S35 és S36 sok alátámasztási pontot tartalmaz, így a funkció-összevonás ebben az esetben sem érvényesül.

S37



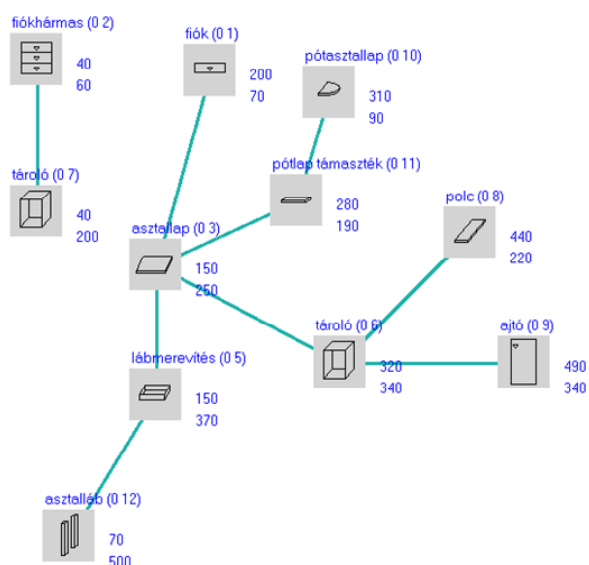
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S38



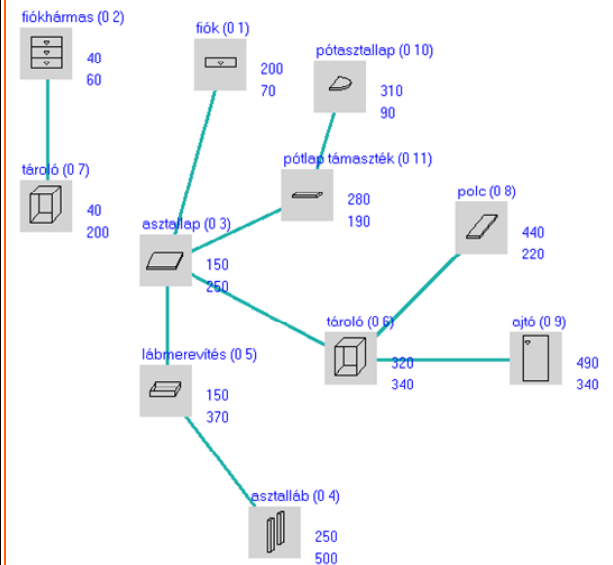
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F6-F9, F10-F11

S39



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

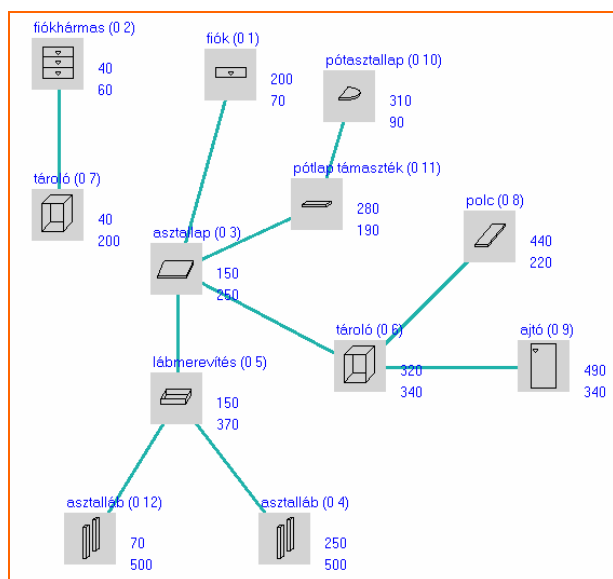
S40



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

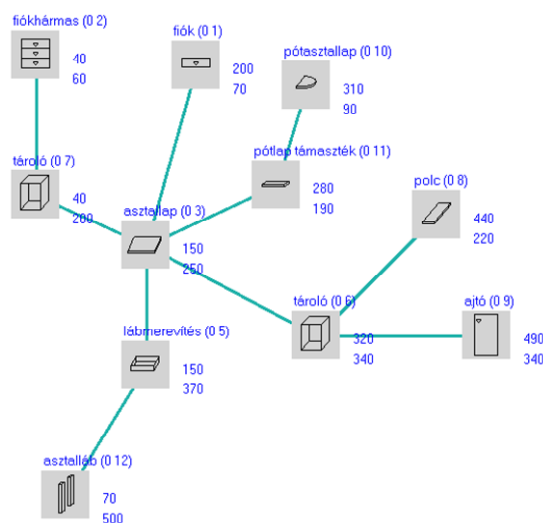
A Struktúra 37, 39, 40, 41, 42, 43 és 44 sok alátámasztási pontot tartalmaz. A Struktúra 45 alapvetően egy, a követelményeket kielégítő megoldás lenne, de mivel mindkét tárolóegység fix, ezért lett elvetve.

S41



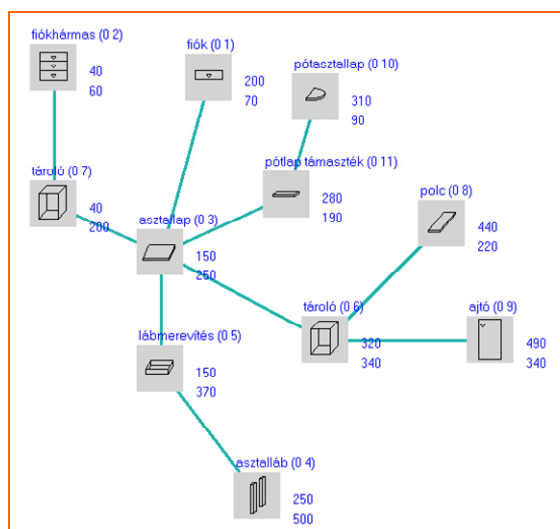
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S43



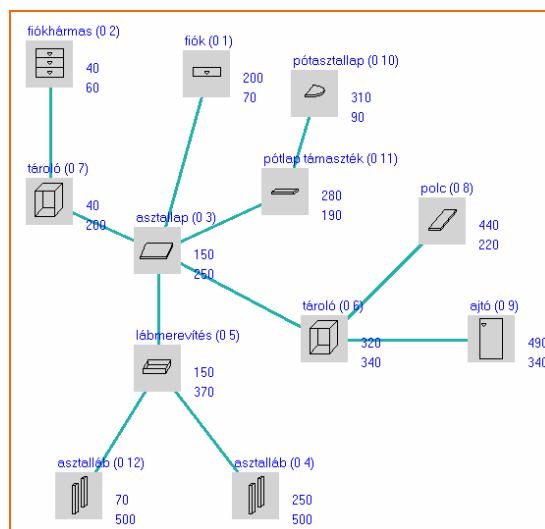
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F5-F12, F6-F8, F6-F9, F10-F11

S44



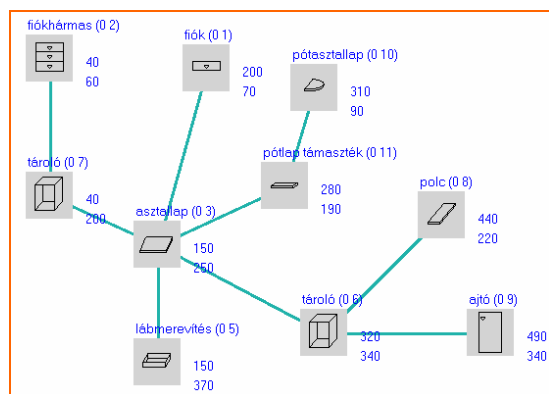
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F6-F8, F6-F9, F10-F11

S45



Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F6-F9, F10-F11

S42



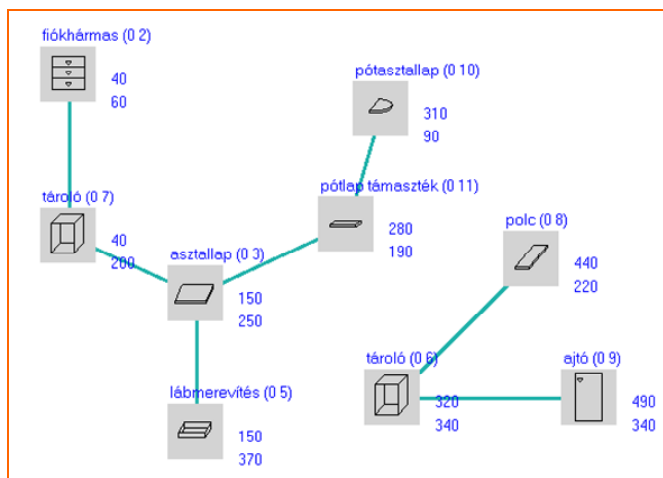
Struktúra egyenlet: F1-F3, F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

A fejezetrészen szereplő megoldási struktúrák a funkciók közti kapcsolatok szerint történő bináris generálás elvileg helyes eredményei. Azok a megoldás változatok, melyeken keret van, a véletlenszerű generálás során jöttek létre.

9.1.2. A követelményeknek megfelelő megoldások

S04

Struktúra gráf:

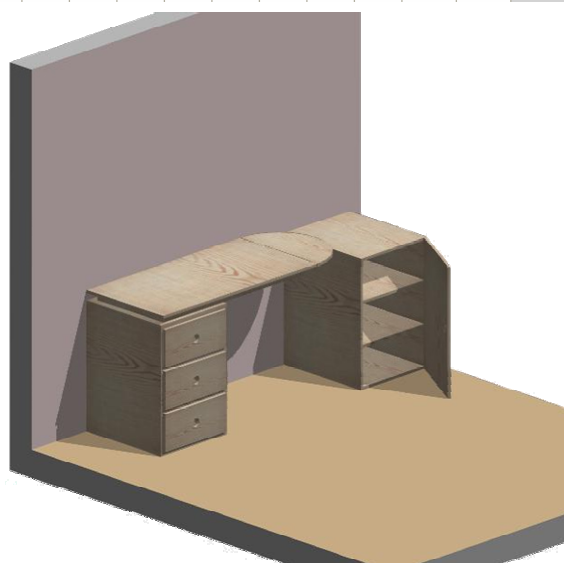


Struktúra egyenlet:

Struktúra mátrix

F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

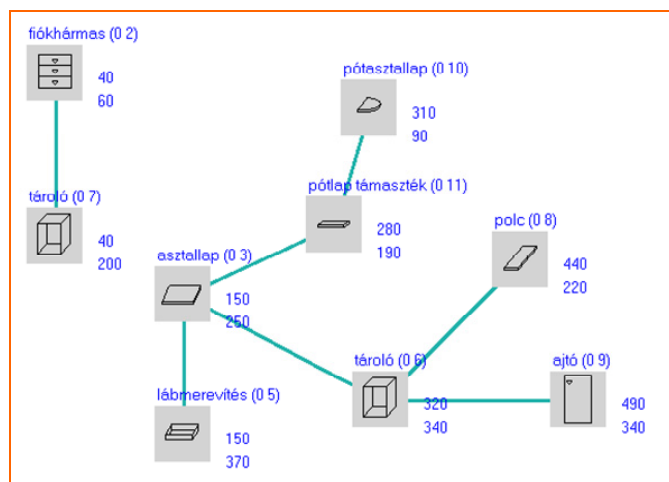
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	0	0	0	0	0	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	0	0		0	1	0	1	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	0	0	0		0	1	1	0	0	0
F(7)	0	1	1	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	



Ez a megoldás egy falraszerelhető változat. Az alátámasztást a fiókos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A polcos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatja az asztaltól. A megoldás növelhető asztalfelülettel rendelkezik.

S08

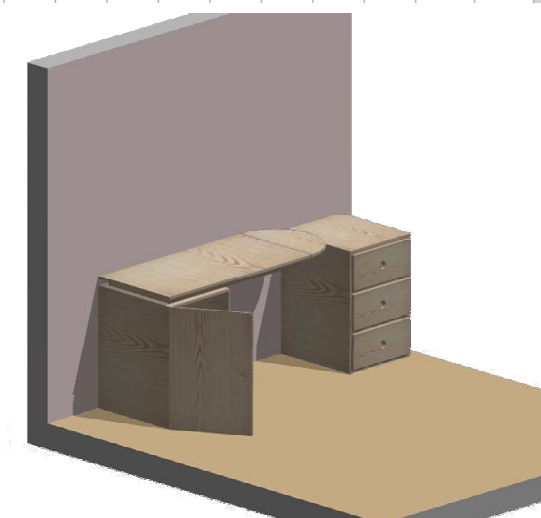
Struktúra gráf:



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

Struktúra mátrix

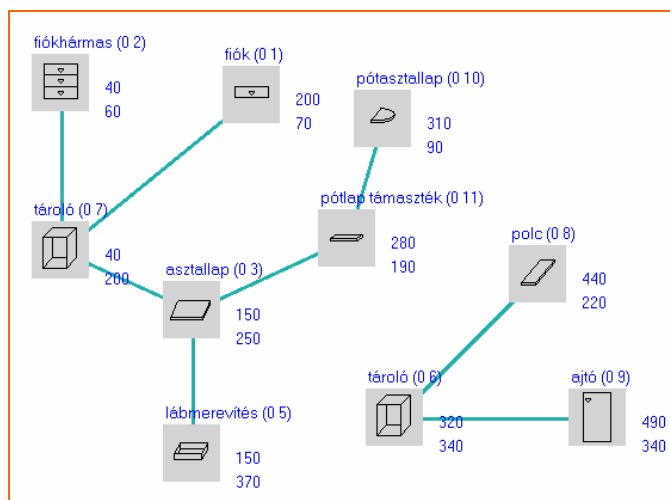
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	0	0	0	0	0	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	0	0		0	1	1	0	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	1	0	0		0	1	1	0	0	0
F(7)	0	1	0	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	



Ez a megoldás szintén egy falraszerelhető változat. Az alátámasztást a polcos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A fiókos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatj az asztaltól. A megoldás növelhető asztalfelülettel rendelkezik.

S19

Struktúra gráf:

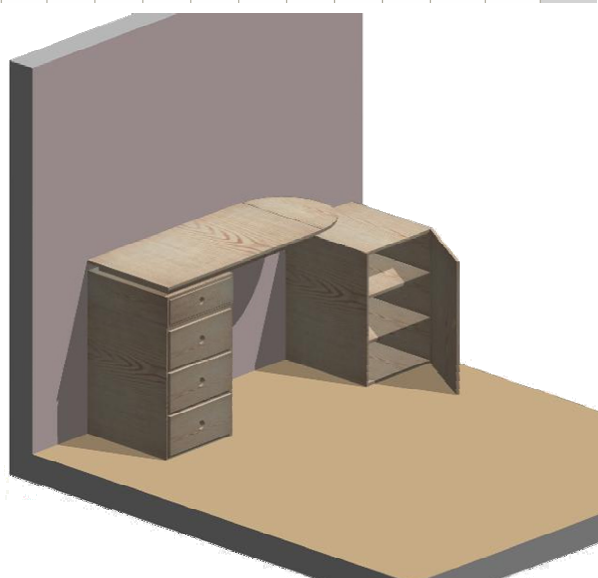


Struktúra egyenlet:

Struktúra mátrix

F1-F7, F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

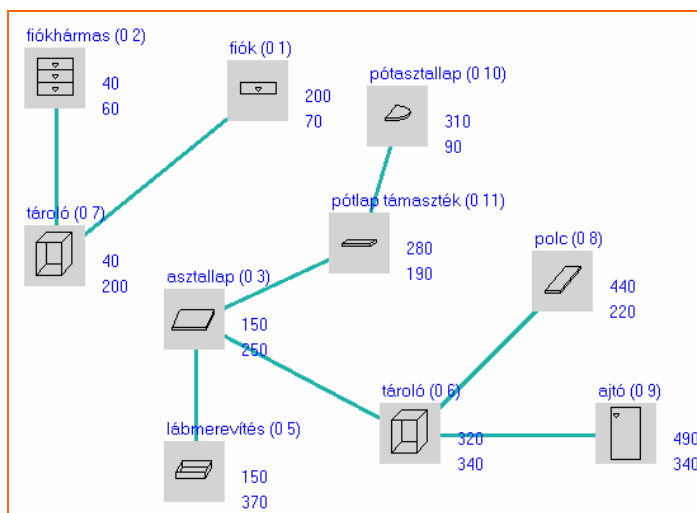
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	0	0	0	0	1	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	0	0		0	1	0	1	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	0	0	0		0	1	1	0	0	0
F(7)	1	1	1	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	



Ez a megoldás a Struktúra: 04-től annyiban tér el, hogy a fiókos tároló egy újabb fiókkal lett kibővíve. Az alátámasztást ebben az esetben is a fiókos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A polcos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatj az asztaltól. A megoldás növelhető asztalfelülettel rendelkezik.

S23

Struktúra gráf:

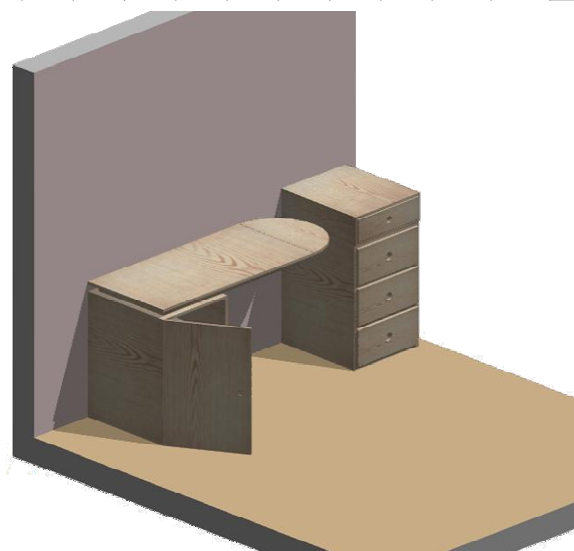


Struktúra egyenlet:

Struktúra mátrix

F1-F7, F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F6-F9, F10-F11

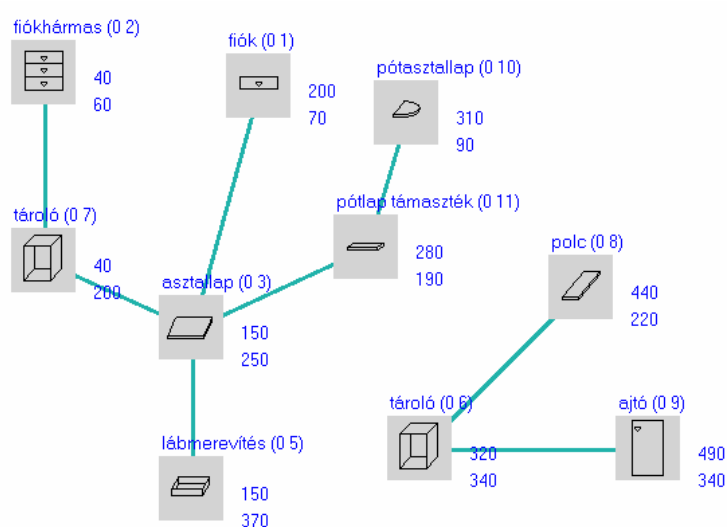
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	0	0	0	0	1	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	0	0		0	1	1	0	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	1	0	0		0	1	1	0	0	0
F(7)	1	1	0	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	



Ez a megoldás szintén egy falraszerelhető változat. Az alátámasztást a polcos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A fiókos tároló mobil egységet alkot, és még egy fiókkal lett kibővítvé. A megoldás növelhető asztalfelülettel rendelkezik.

S34

Struktúra gráf:



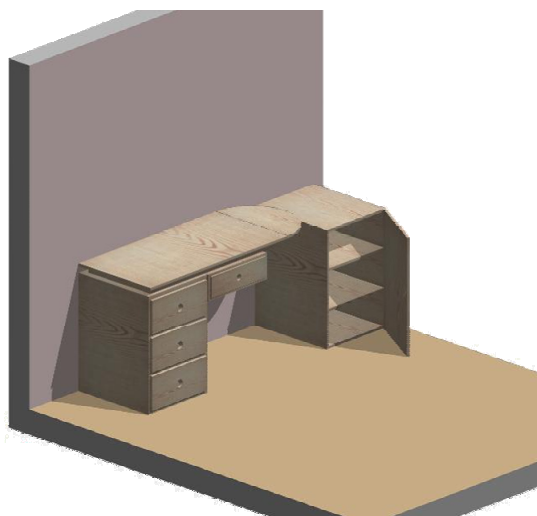
Struktúra egyenlet:

Struktúra mátrix

F1-F3, F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F6-F9, F10-F11

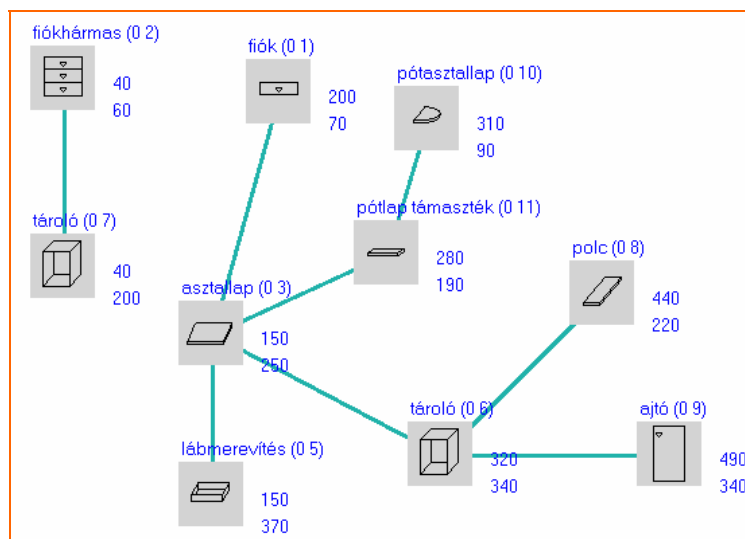
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	1	0	0	0	0	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	1	0		0	1	0	1	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	0	0	0		0	1	1	0	0	0
F(7)	0	1	1	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	

Ez a megoldás szintén egy falraszerelhető változat. Az alátámasztást a fiókos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A polcos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatj az asztaltól. A megoldás növelhető asztalfelülettel rendelkeznek. Ezen kívül még egy fiók tartozik hozzá, tároló egységtől függetlenül, az asztallaphoz csatlakoztatva.



S38

Struktúra gráf:



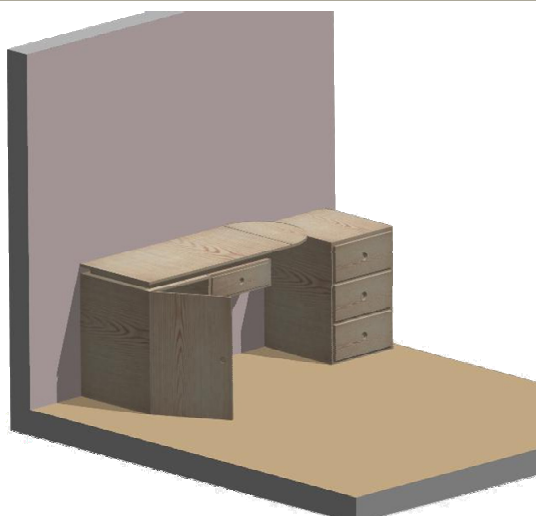
Struktúra egyenlet:

Struktúra mátrix

F1-F3, F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F6-F9, F10-F11

	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		0	1	0	0	0	0	0	0	0	0	0
F(2)	0		0	0	0	0	1	0	0	0	0	0
F(3)	1	0		0	1	1	0	0	0	0	1	0
F(4)	0	0	0		0	0	0	0	0	0	0	0
F(5)	0	0	1	0		0	0	0	0	0	0	0
F(6)	0	0	1	0	0		0	1	1	0	0	0
F(7)	0	1	0	0	0	0		0	0	0	0	0
F(8)	0	0	0	0	0	1	0		0	0	0	0
F(9)	0	0	0	0	0	1	0	0		0	0	0
F(10)	0	0	0	0	0	0	0	0	0		1	0
F(11)	0	0	1	0	0	0	0	0	0	1		0
F(12)	0	0	0	0	0	0	0	0	0	0	0	

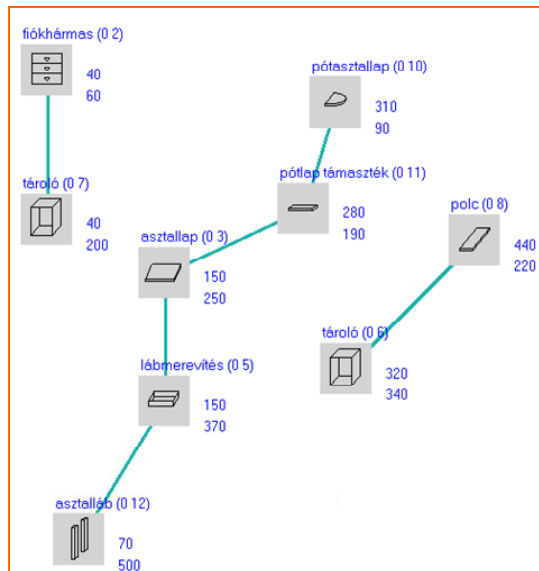
Ez a megoldás is egy falraszerelhető változat. Az alátámasztást a polcos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A fiókos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatja az asztaltól. A megoldás növelhető asztalfelülettel, továbbá egy asztallaphoz csatlakoztatott fiókkal rendelkezik.



9.2. FUNKCIÓK SZERINTI GENERÁLÁS

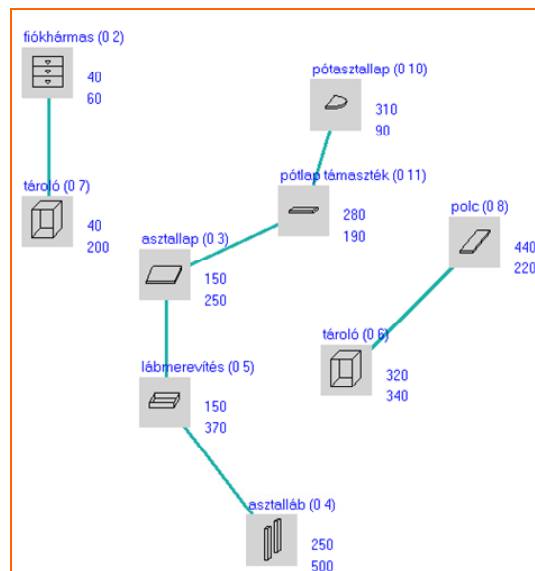
9.2.1. Elvileg helyes megoldások

S01



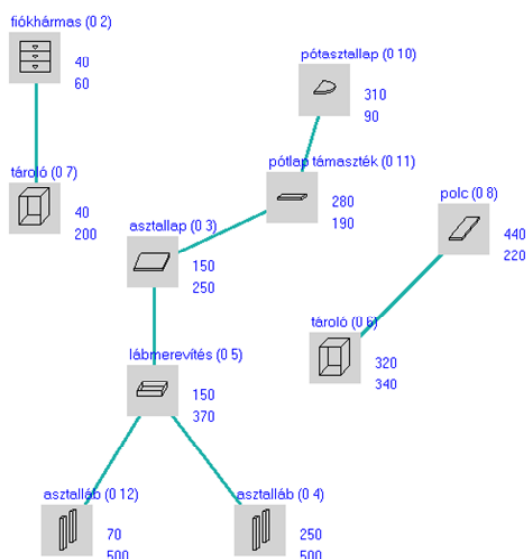
Struktúra egyenlet: F2-F7, F3-F5, F3-F11, F5-F12, F6-F8, F10-F11

S02



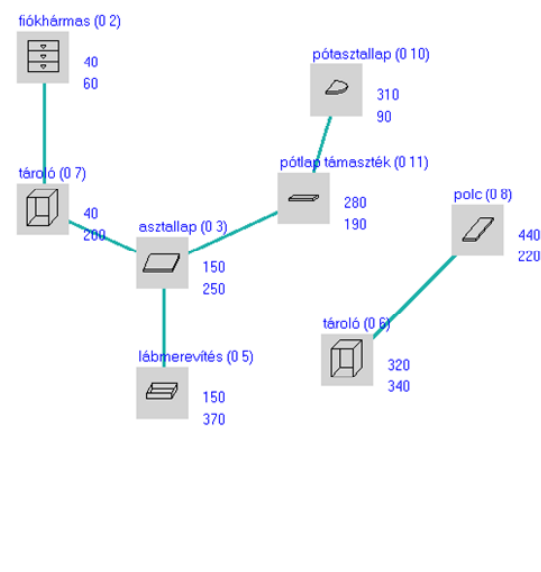
Struktúra egyenlet: F2-F7, F3-F5, F3-F11, F4-F5, F6-F8, F10-F11

S03



Struktúra egyenlet: F2-F7, F3-F5, F3-F11, F4-F5, F5-F12, F6-F8, F10-F11

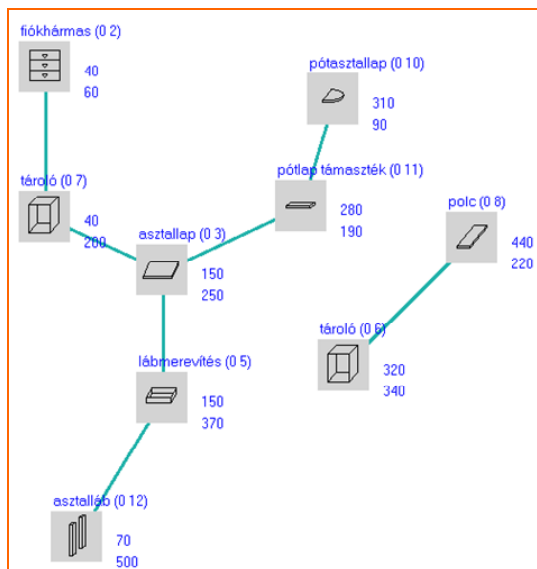
S04



Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F10-F11

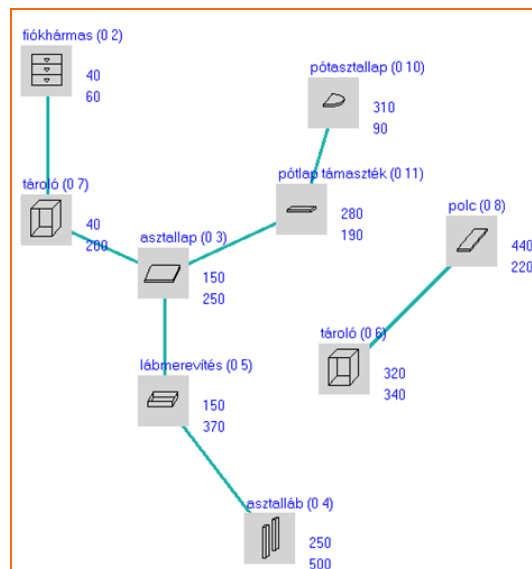
Az S01 és S02 esetében a tárolók az asztaltól függetlenül, külön egységként vesznek részt a struktúrában, ezáltal a követelményjegyzék pontjai alapján megfogalmazott funkcióösszevonás feltételét nem teljesítik. Az S03 két pár lábat tartalmaz, így a struktúra egy önálló, falra nem szerelhető változatot képez. Ez pedig a falra szerelhetőségi elvárást nem teljesíti.

S05



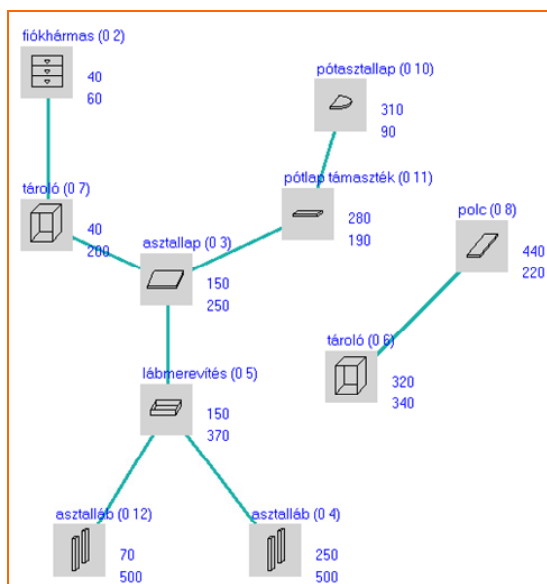
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F5-F12, F6-F8, F10-F11

S06



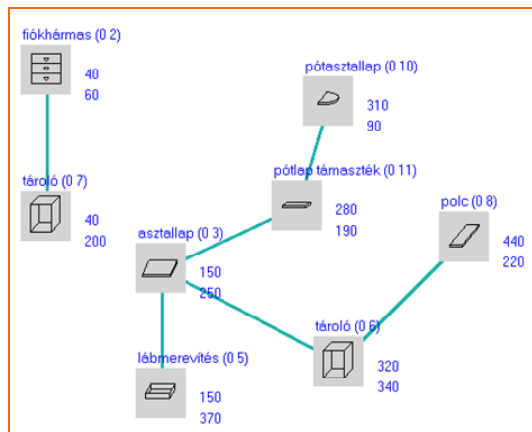
Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F6-F8, F10-F11

S07



Struktúra egyenlet: F2-F7, F3-F5, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F10-F11

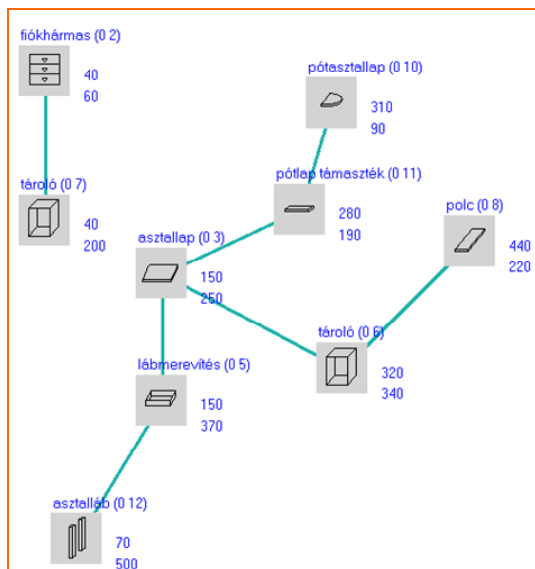
S08



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F10-F11

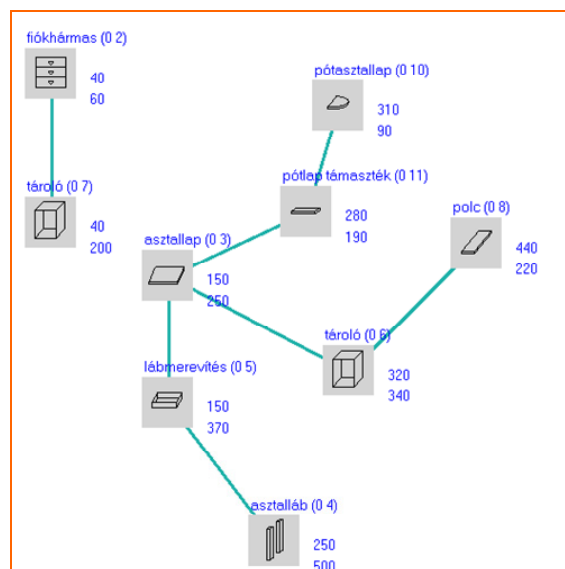
Az S05 és 06 nem elégíti ki a funkció-összevonás feltételét, mert lábbal és fixen beépített tárolóval is rendelkeznek. Az S07 sok alátámasztási pontot tartalmaz.

S09



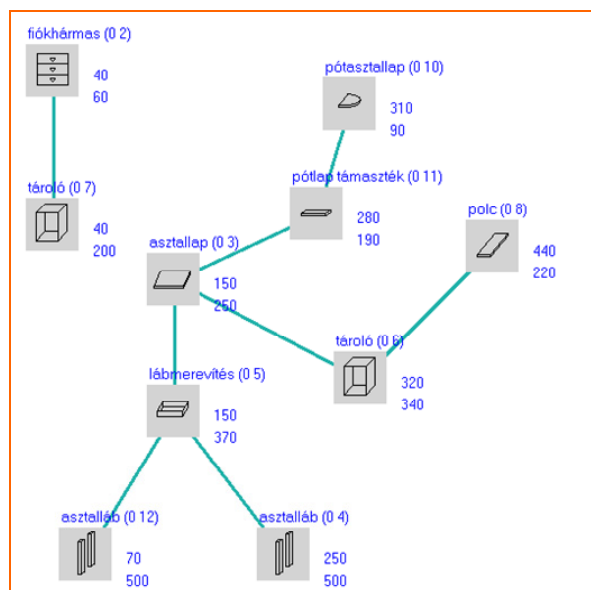
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F5-F12, F6-F8, F10-F11

S10



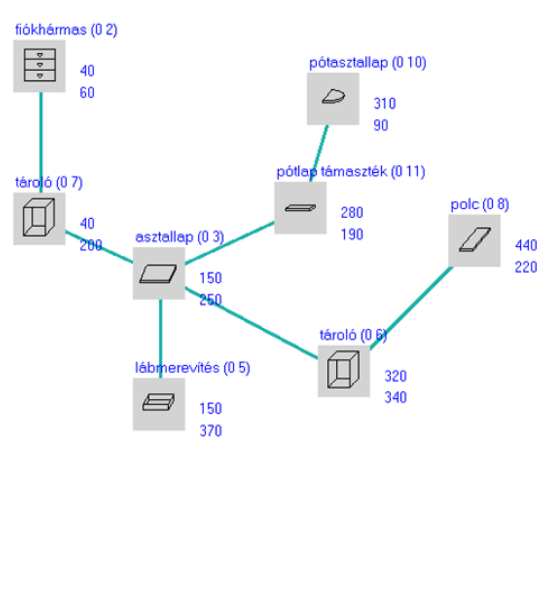
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F6-F8, F10-F11

S11



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F11, F4-F5, F5-F12, F6-F8, F10-F11

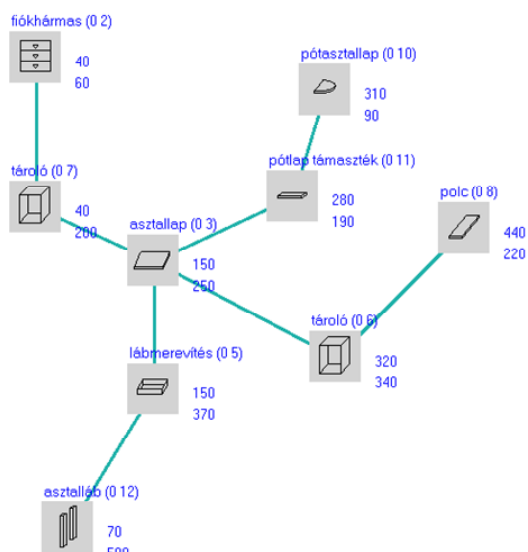
S12



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F6-F8, F10-F11

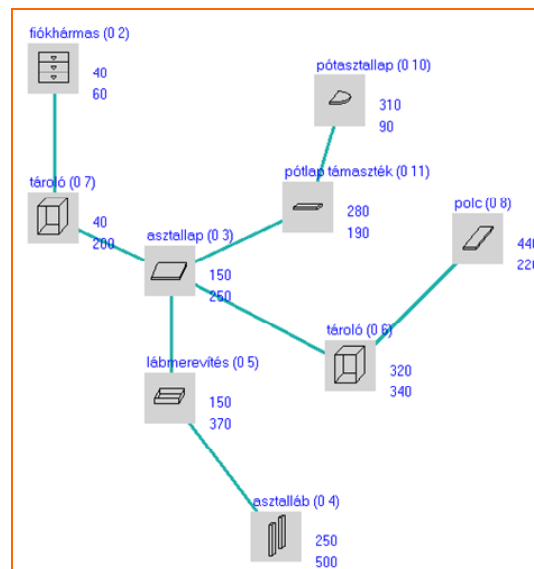
Az S09, S10 és S11 nem elégíti ki a funkció-összevonás feltételét, mert lábbal és fixen beépített tárolóval is rendelkeznek. Az S11 két pár lábat is tartalmaz, így önálló asztalt képez, nem elégíti ki a falra szerelhetőség és a kevés alkatrész kritériumait. Az S12 alapvetően kielégíti a követelményrendszerben feltüntetett elvárásokat, azonban a helytakarékoság kritériumait nem maximálisan. A követelményjegyzék alapján jó, ha az egyik tároló mozgatható, mert ha a felhasználó elteszi, akkor így hely nyerhető a munkához.

S13



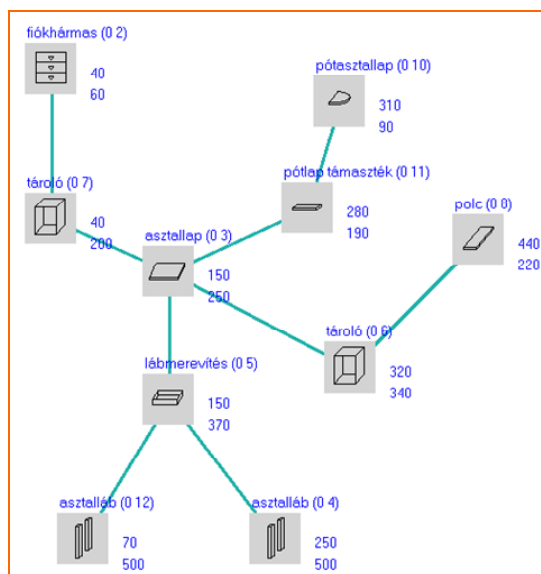
Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F5-F12, F6-F8, F10-F11

S14



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F6-F8, F10-F11

S15



Struktúra egyenlet: F2-F7, F3-F5, F3-F6, F3-F7, F3-F11, F4-F5, F5-F12, F6-F8, F10-F11

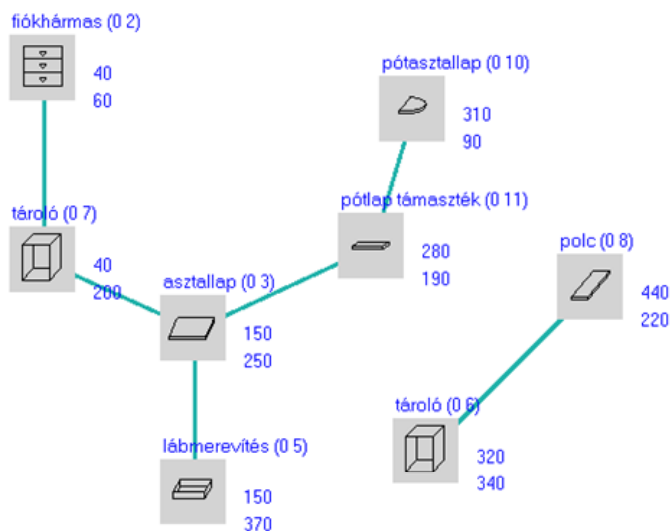
Az S13, S14 és S15 túl sok alátámasztási pontot tartalmaz, ezáltal a követelményjegyzék szempontjaihoz mérten szerkezetük túl bonyolult.

A fejezetrészben szereplő megoldási struktúrák a funkciók szerint történő bináris generálás elvileg helyes eredményei. Azok a megoldás változatok, melyeken keret van, a véletlenszerű generálás során jöttek létre.

9.2.2. A követelményeknek megfelelő megoldások

S04

Struktúra gráf:

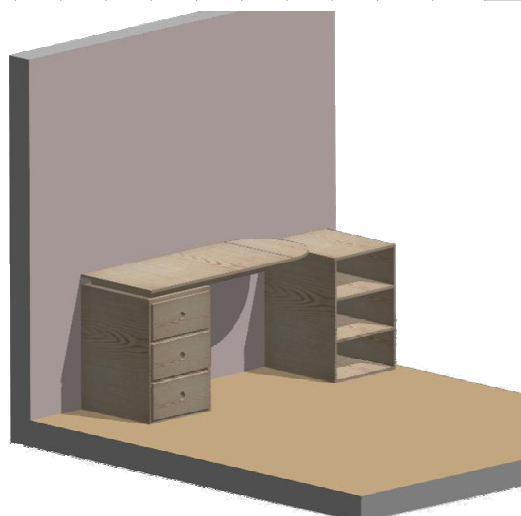


Struktúra egyenlet:
Struktúra mátrix

F2-F7, F3-F5, F3-F7, F3-F11, F6-F8, F10-F11

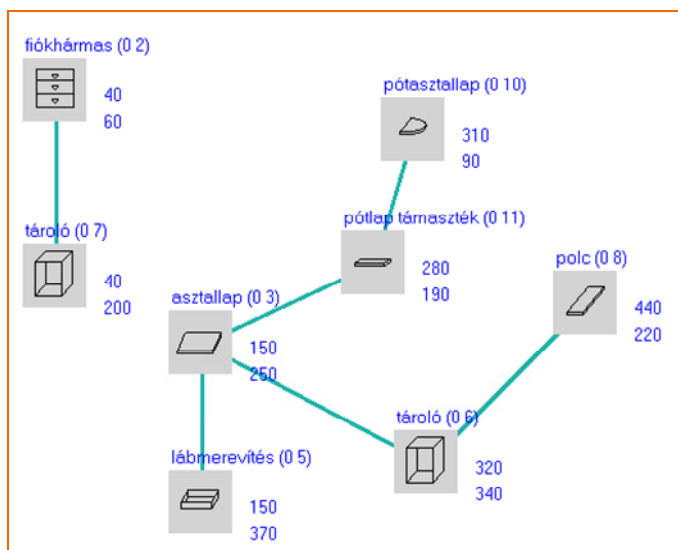
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		z	z	z	z	z	z	z	z	z	z	z
F(2)	z		0	0	0	0	1	0	z	0	0	0
F(3)	z	0		0	1	0	1	0	z	0	1	0
F(4)	z	0	0		0	0	0	0	z	0	0	0
F(5)	z	0	1	0		0	0	0	z	0	0	0
F(6)	z	0	0	0	0		0	1	z	0	0	0
F(7)	z	1	1	0	0	0		0	z	0	0	0
F(8)	z	0	0	0	0	1	0		z	0	0	0
F(9)	z	z	z	z	z	z	z	z		z	z	z
F(10)	z	0	0	0	0	0	0	0	z		1	0
F(11)	z	0	1	0	0	0	0	0	z	1		0
F(12)	z	0	0	0	0	0	0	0	z	0	0	

Ez a megoldás egy falra szerelhető változat. Az alátámasztást a fiókos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A polcos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatja az asztaltól. A megoldás növelhető asztalfelülettel rendelkezik.



S08

Struktúra gráf:

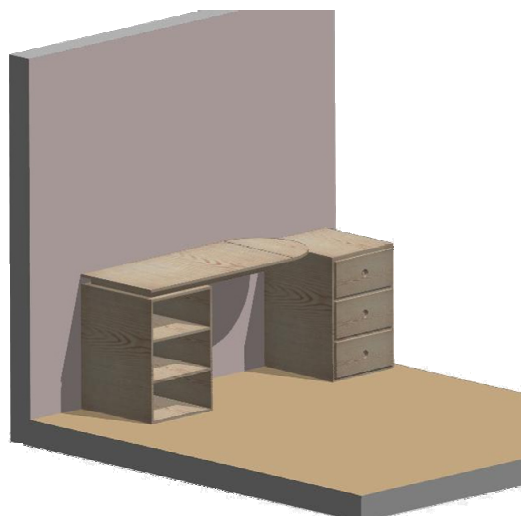


Struktúra egyenlet:

Struktúra mátrix

F2-F7, F3-F5, F3-F6, F3-F11, F6-F8, F10-F11

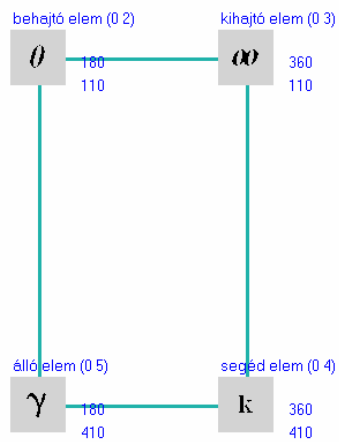
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)	F(7)	F(8)	F(9)	F(10)	F(11)	F(12)
F(1)		z	z	z	z	z	z	z	z	z	z	z
F(2)	z		0	0	0	0	1	0	z	0	0	0
F(3)	z	0		0	1	1	0	0	z	0	1	0
F(4)	z	0	0		0	0	0	0	z	0	0	0
F(5)	z	0	1	0		0	0	0	z	0	0	0
F(6)	z	0	1	0	0		0	1	z	0	0	0
F(7)	z	1	0	0	0	0		0	z	0	0	0
F(8)	z	0	0	0	0	1	0		z	0	0	0
F(9)	z	z	z	z	z	z	z	z		z	z	z
F(10)	z	0	0	0	0	0	0	0	z		1	0
F(11)	z	0	1	0	0	0	0	0	z	1		0
F(12)	z	0	0	0	0	0	0	0	z	0	0	



Ez a megoldás is egy falra szerelhető változat. Az alátámasztást a polcos tároló oldja meg, mely fixen van rögzítve az asztallaphoz. A fiókos tároló mobil egységet alkot, ha a felhasználónak éppen nincs szüksége a benne tárolt eszközökre, akkor el is mozdíthatja az asztaltól. A megoldás növelhető asztalfelülettel rendelkezik.

9.3. BINÁRISAN GENERÁLT BOLYGÓMŰ VÁLTOZATOK

S01



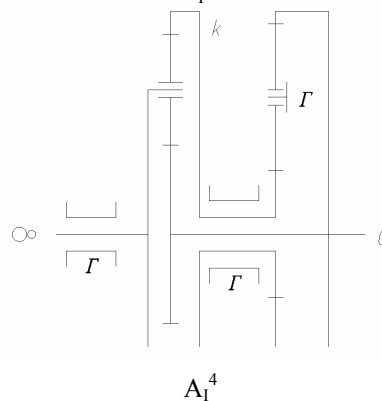
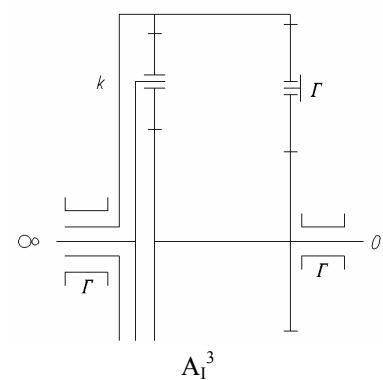
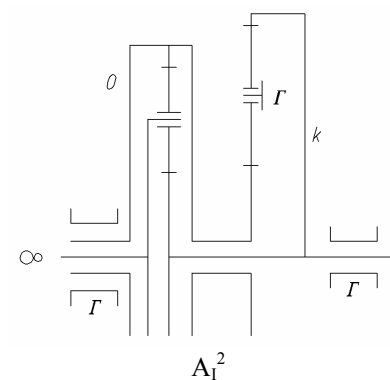
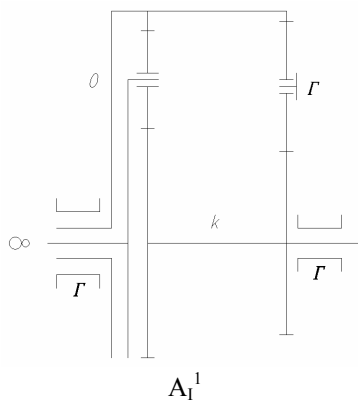
Struktúra gráf

Struktúra egyenlet: $F2-F3, F2-F5, F3-F4, F4-F5$

Struktúra mátrix:

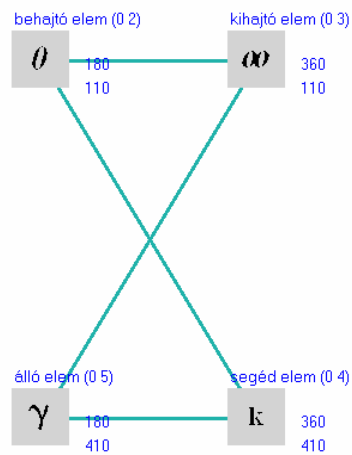
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	0	1	0
F(3)	0	1		1	0	0
F(4)	0	0	1		1	0
F(5)	0	1	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló elem, a másik bolygómű karja a kihajtó elem.

S02



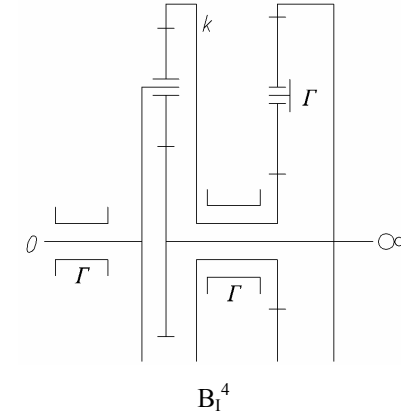
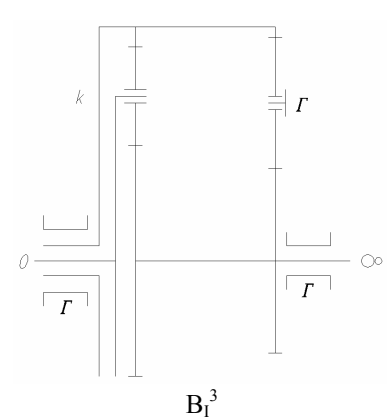
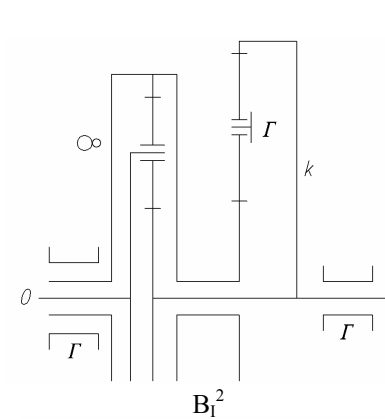
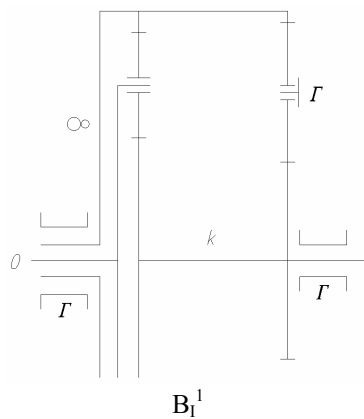
Struktúra gráf

Struktúra egyenlet: F2-F3, F2-F4, F3-F5, F4-F5

Struktúra mátrix:

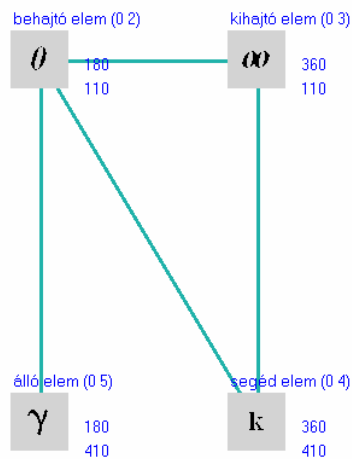
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	0	0
F(3)	0	1		0	1	0
F(4)	0	1	0		1	0
F(5)	0	0	1	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló elem, a másiké a behajtó elem.

S03



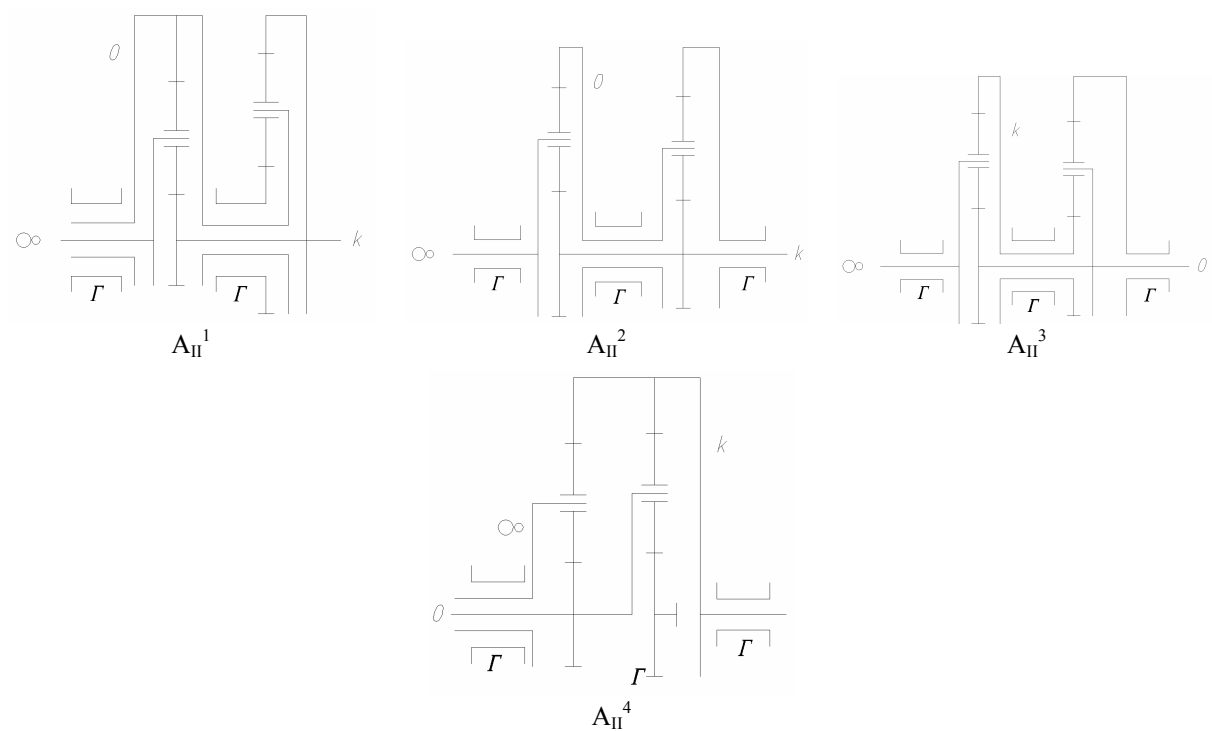
Struktúra gráf

Struktúra egyenlet: $F2-F3, F2-F4, F2-F5, F3-F4$

Struktúra mátrix:

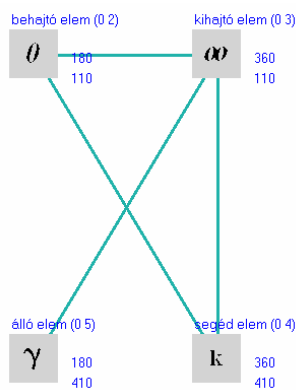
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	1	0
F(3)	0	1		1	0	0
F(4)	0	1	1		0	0
F(5)	0	1	0	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja a kihajtó elem, a másik bolygómű karja a behajtó elem. A behajtó elem kapcsolt alapelem.

S04



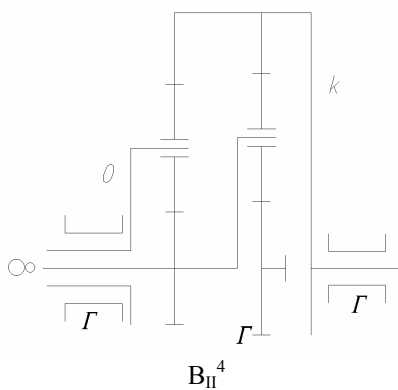
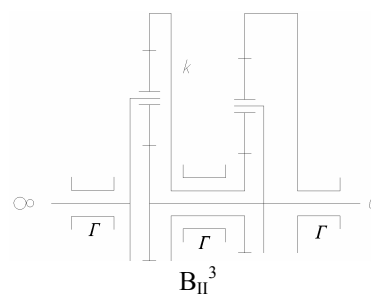
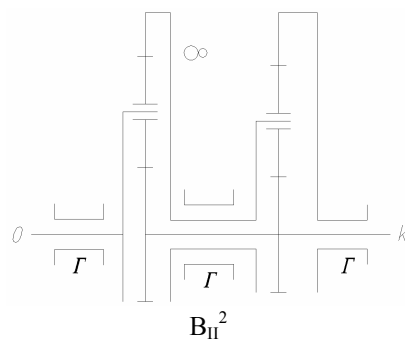
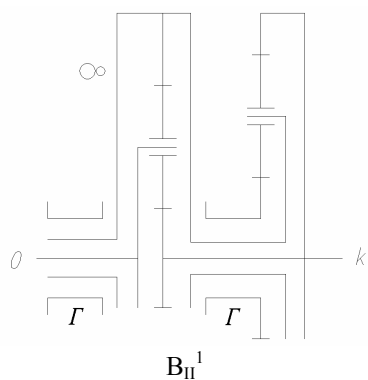
Struktúra gráf

Struktúra egyenlet: $F2-F3, F2-F4, F3-F4, F3-F5$

Struktúra mátrix:

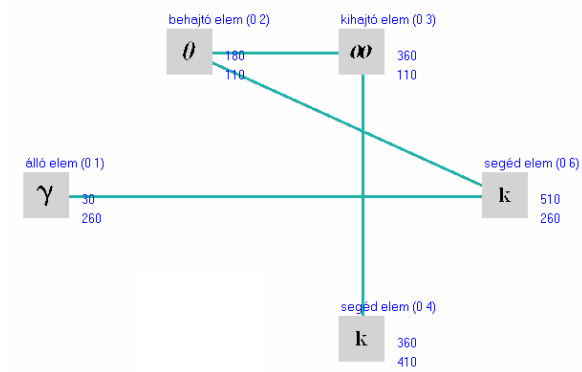
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	0	0
F(3)	0	1		1	1	0
F(4)	0	1	1		0	0
F(5)	0	0	1	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja a behajtó elem, a másik bolygómű karja a kihajtó elem. A kihajtó elem kapcsolt alapelem.

S05



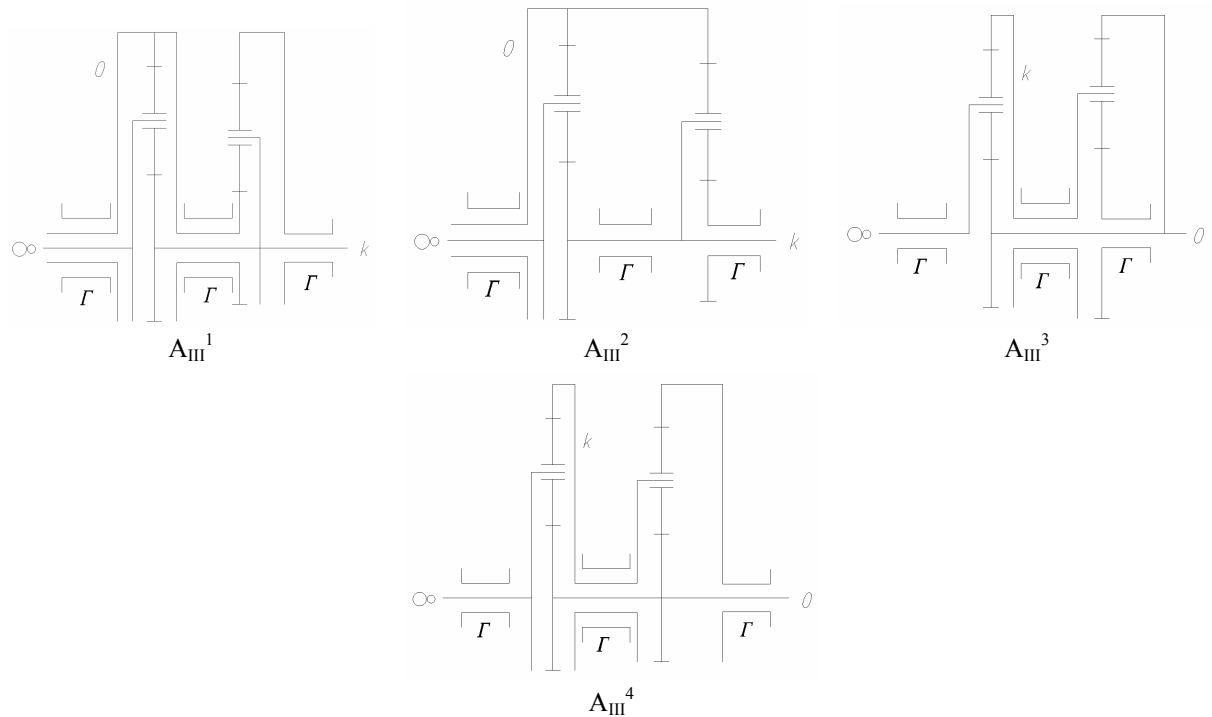
Struktúra gráf

Struktúra egyenlet: F1-F6, F2-F3, F2-F6, F3-F4

Struktúra mátrix:

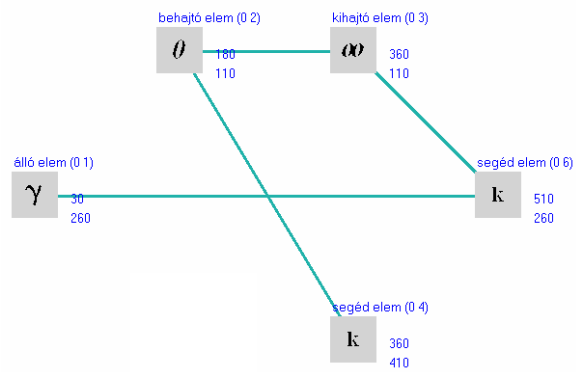
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	1
F(2)	0		1	0	0	1
F(3)	0	1		1	0	0
F(4)	0	0	1		0	0
F(5)	0	0	0	0		0
F(6)	1	1	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja a kihajtó elem, a másiké a kapcsolt segédelem.

S06



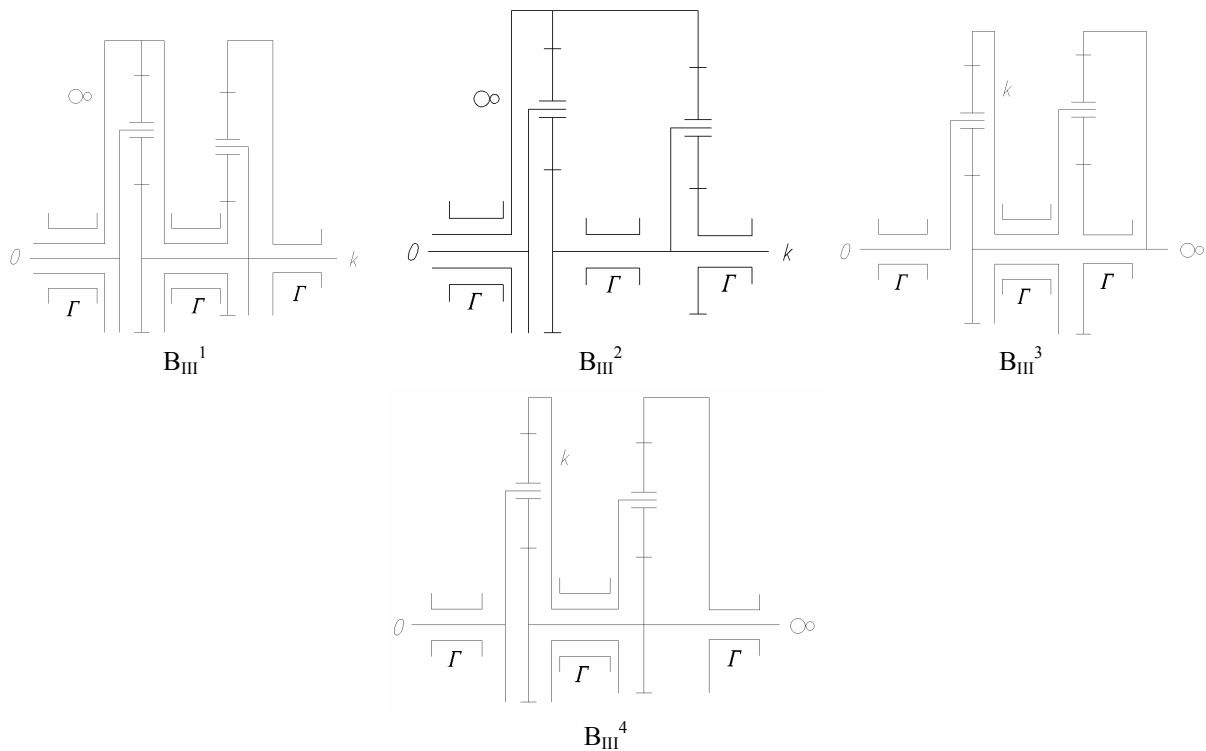
Struktúra gráf

Struktúra egyenlet: F1-F6, F2-F3, F2-F4, F3-F6

Struktúra mátrix:

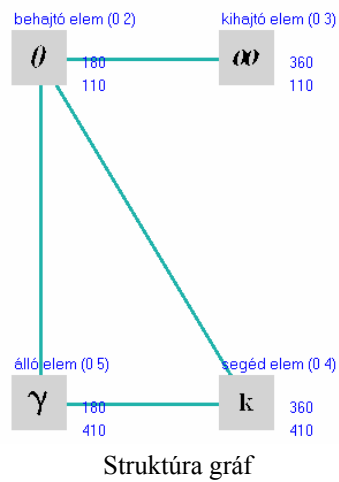
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	1
F(2)	0		1	1	0	0
F(3)	0	1		0	0	1
F(4)	0	1	0		0	0
F(5)	0	0	0	0		0
F(6)	1	0	1	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja kapcsolt segédelem, a másik bolygómű karja a behajtó elem.

S07

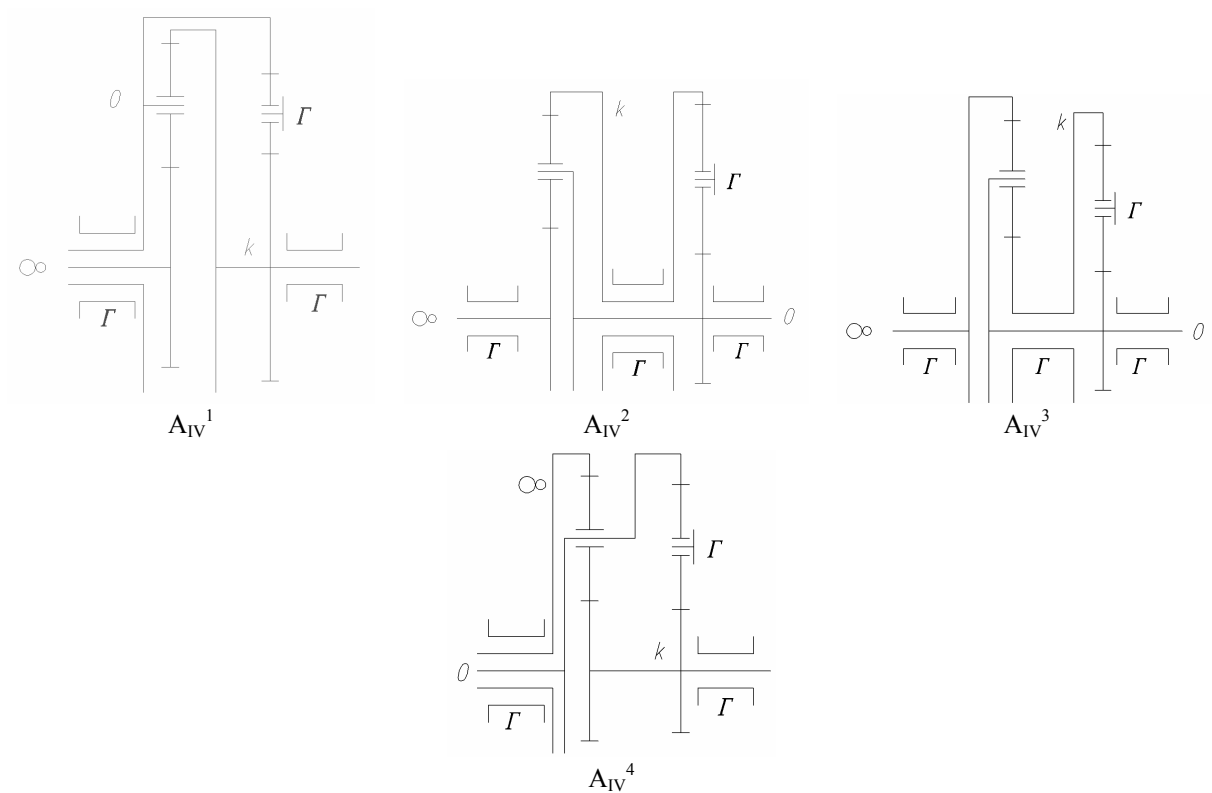


Struktúra egyenlet: F2-F3, F2-F4, F2-F5, F4-F5

Struktúra mátrix:

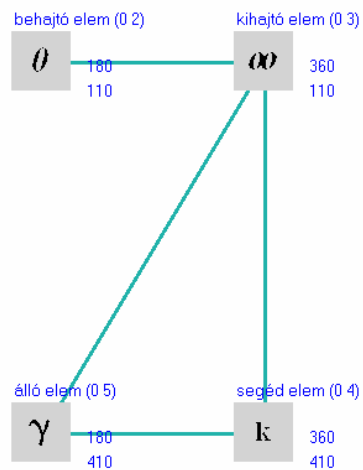
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	1	0
F(3)	0	1		0	0	0
F(4)	0	1	0		1	0
F(5)	0	1	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló alapelem, a másik bolygómű karja a behajtó elem. A behajtó elem kapcsolt elem.

S08



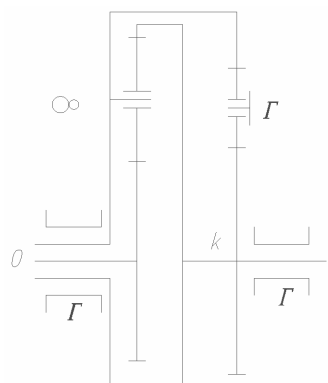
Struktúra gráf

Struktúra egyenlet: F2-F3, F3-F4, F3-F5, F4-F5

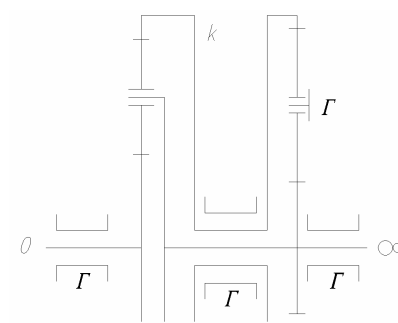
Struktúra mátrix:

	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	0	0	0
F(3)	0	1		1	1	0
F(4)	0	0	1		1	0
F(5)	0	0	1	1		0
F(6)	0	0	0	0	0	

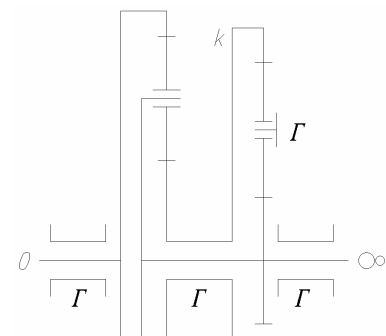
Kinematikai ábrák:



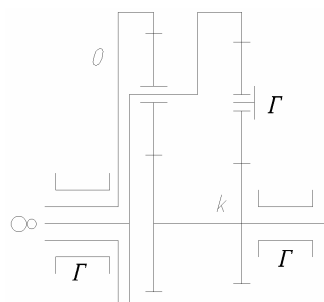
B_{IV}^1



B_{IV}^2



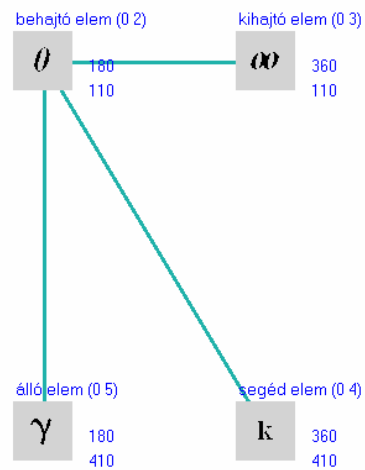
B_{IV}^3



B_{IV}^4

Az egyik bolygómű karja álló alapelem, a másik bolygómű karja a kihajtó elem. A kihajtó-elem kapcsolt alapelem.

S09



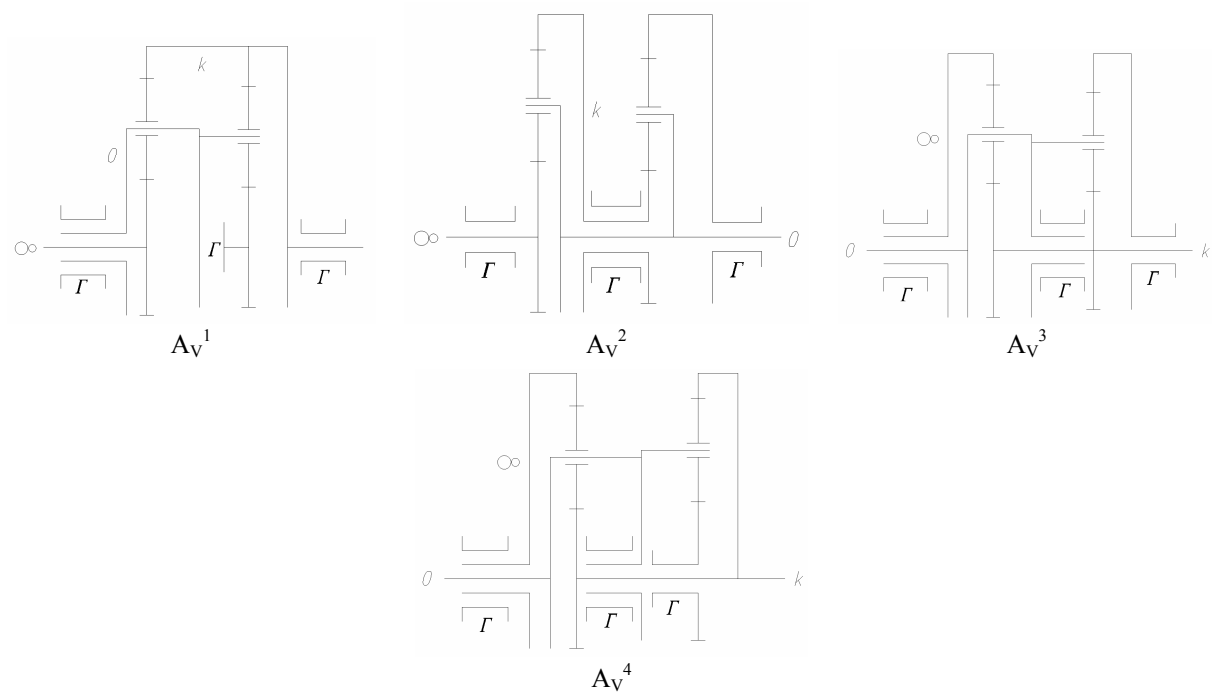
Struktúra gráf

Struktúra egyenlet: F2-F3, F2-F4, F2-F5

Struktúra mátrix:

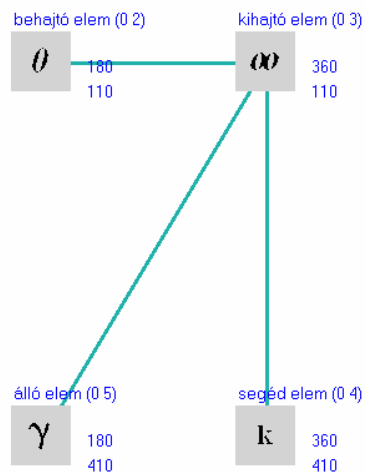
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	1	0
F(3)	0	1		0	0	0
F(4)	0	1	0		0	0
F(5)	0	1	0	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja egymással van összekapcsolva, így mind a két bolygómű karja a behajtó elem.

S10



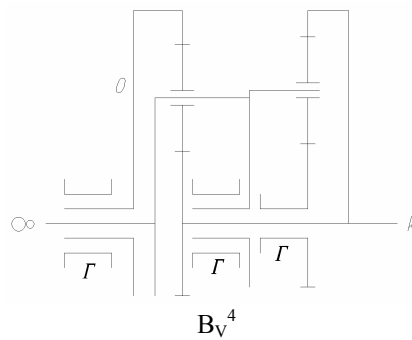
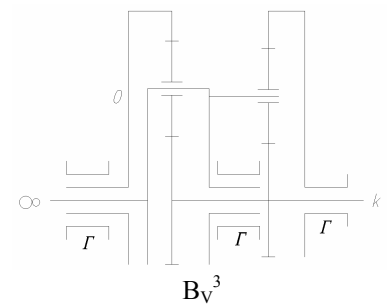
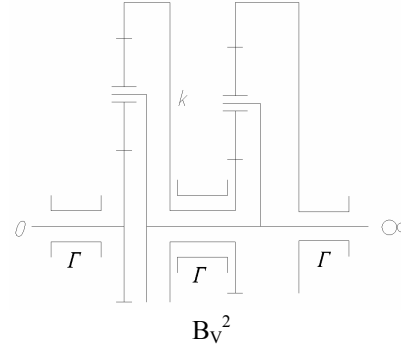
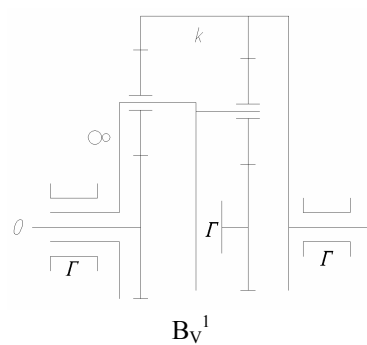
Struktúra gráf

Struktúra egyenlet: F2-F3, F3-F4, F3-F5

Struktúra mátrix:

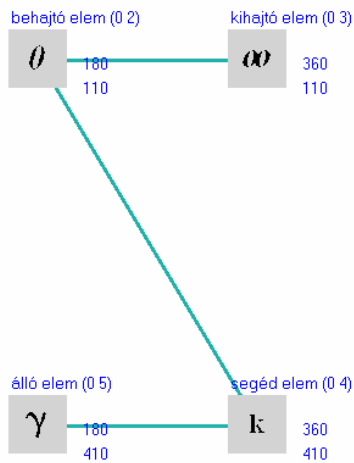
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	0	0	0
F(3)	0	1		1	1	0
F(4)	0	0	1		0	0
F(5)	0	0	1	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja egymással van összekapcsolva, így mind a két bolygómű karja a kihajtó elem.

S11



Struktúra gráf

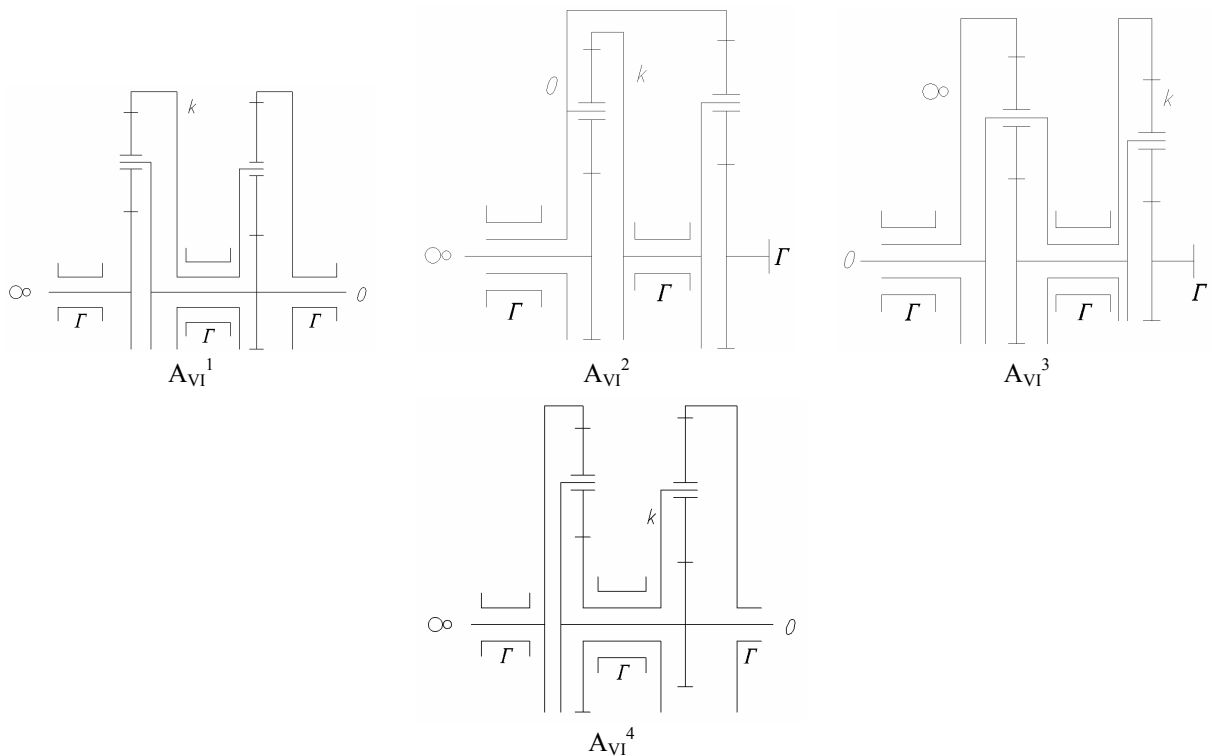
Struktúra egyenlet:

F2-F3, F2-F4, F4-F5

Struktúra mátrix:

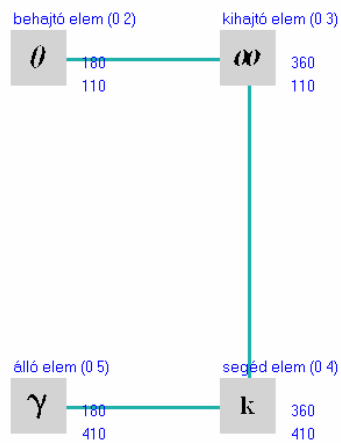
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	1	0	0
F(3)	0	1		0	0	0
F(4)	0	1	0		1	0
F(5)	0	0	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja kapcsolt alapelem; az egyik bolygómű karja a behajtó elem, a másiké a segédelem.

S12



Struktúra gráf

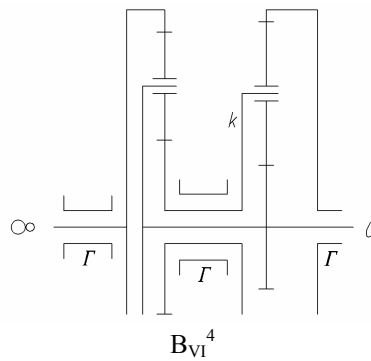
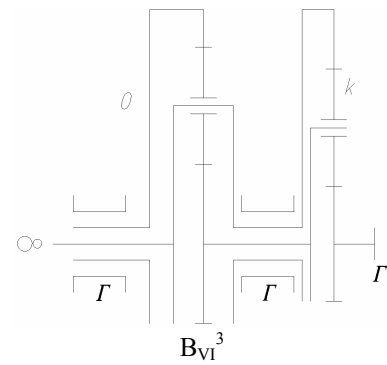
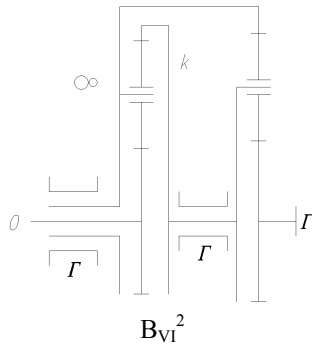
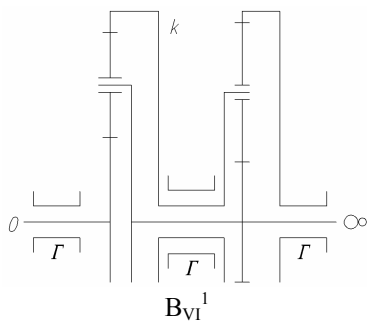
Struktúra egyenlet:

F2-F3, F3-F4, F4-F5

Struktúra mátrix:

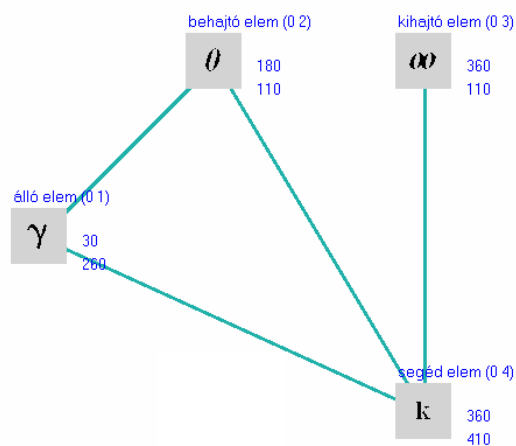
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		1	0	0	0
F(3)	0	1		1	0	0
F(4)	0	0	1		1	0
F(5)	0	0	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómu karja kapcsolt elem, az egyik a kihajtó elem, a másik a segédelem.

S13



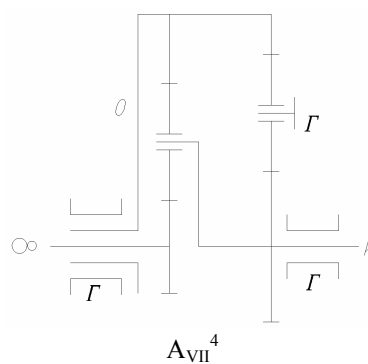
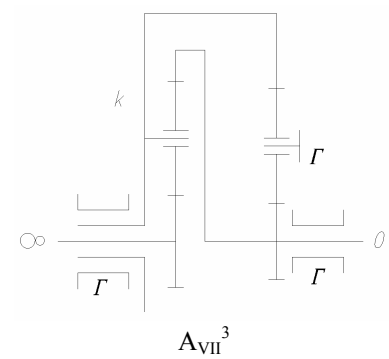
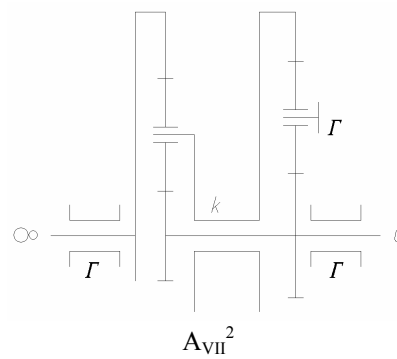
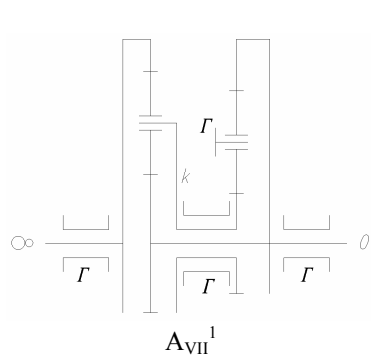
Struktúra gráf

Struktúra egyenlet: $F1-F2, F1-F4, F2-F4, F3-F4$

Struktúra mátrix:

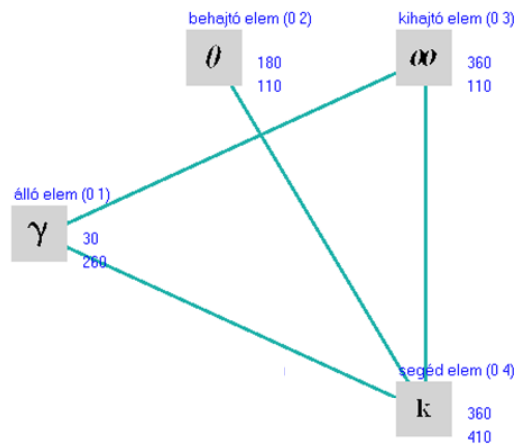
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	1	0	0
F(2)	1		0	1	0	0
F(3)	0	0		1	0	0
F(4)	1	1	1		0	0
F(5)	0	0	0	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló alapelem, a másiké a segédelem.

S14



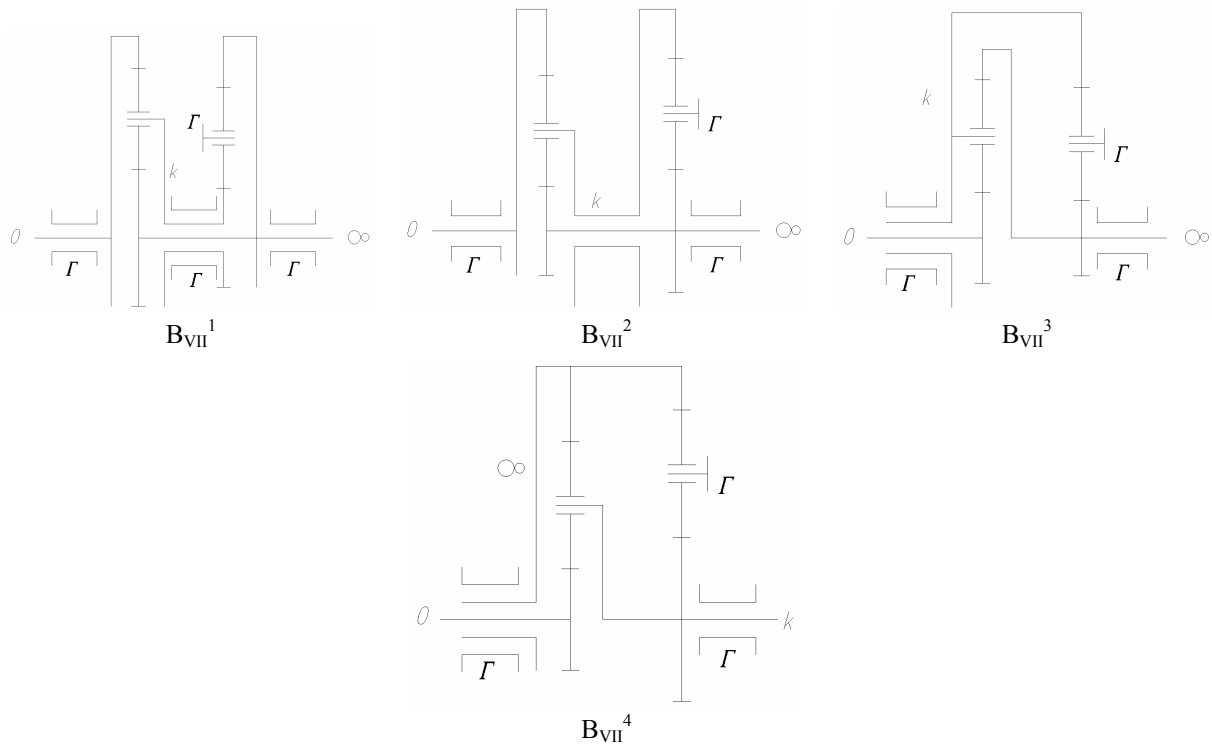
Struktúra gráf

Struktúra egyenlet: $F1-F3, F1-F4, F2-F4, F3-F4$

Struktúra mátrix:

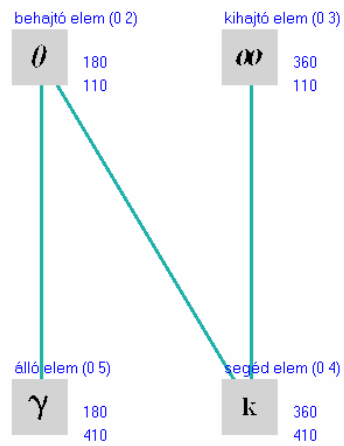
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	1	1	0	0
F(2)	0		0	1	0	0
F(3)	1	0		1	0	0
F(4)	1	1	1		0	0
F(5)	0	0	0	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló alapelem, a másiké kapcsolt segédelem.

S15



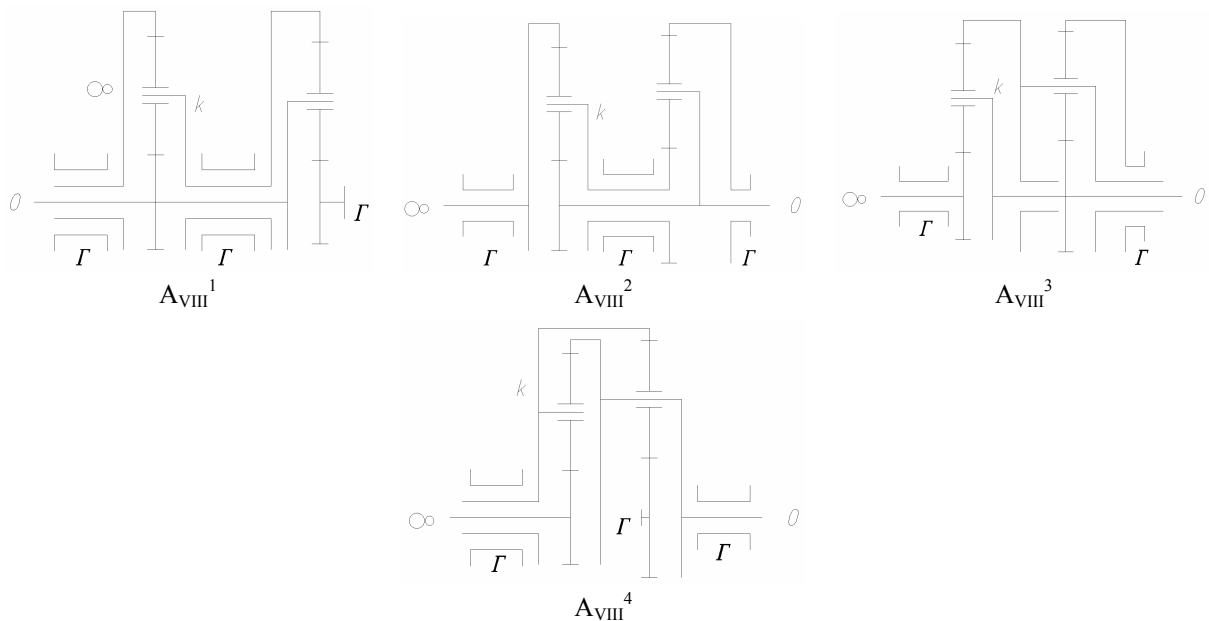
Struktúra gráf

Struktúra egyenlet: $F_2-F_4, F_2-F_5, F_3-F_4$

Struktúra mátrix:

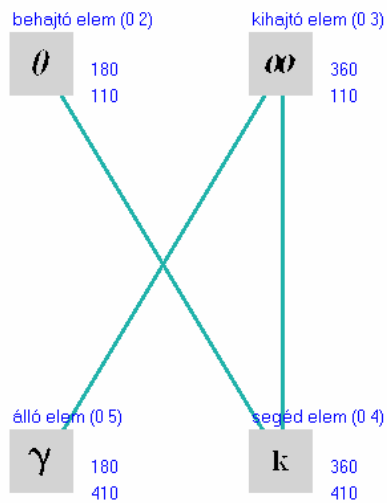
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		0	1	1	0
F(3)	0	0		1	0	0
F(4)	0	1	1		0	0
F(5)	0	1	0	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja kapcsolt elem, az egyszabadságfokúként működő hajtómű karja a behajtó elem, a másik hajtómű karja a segédelem.

S16



Struktúra gráf

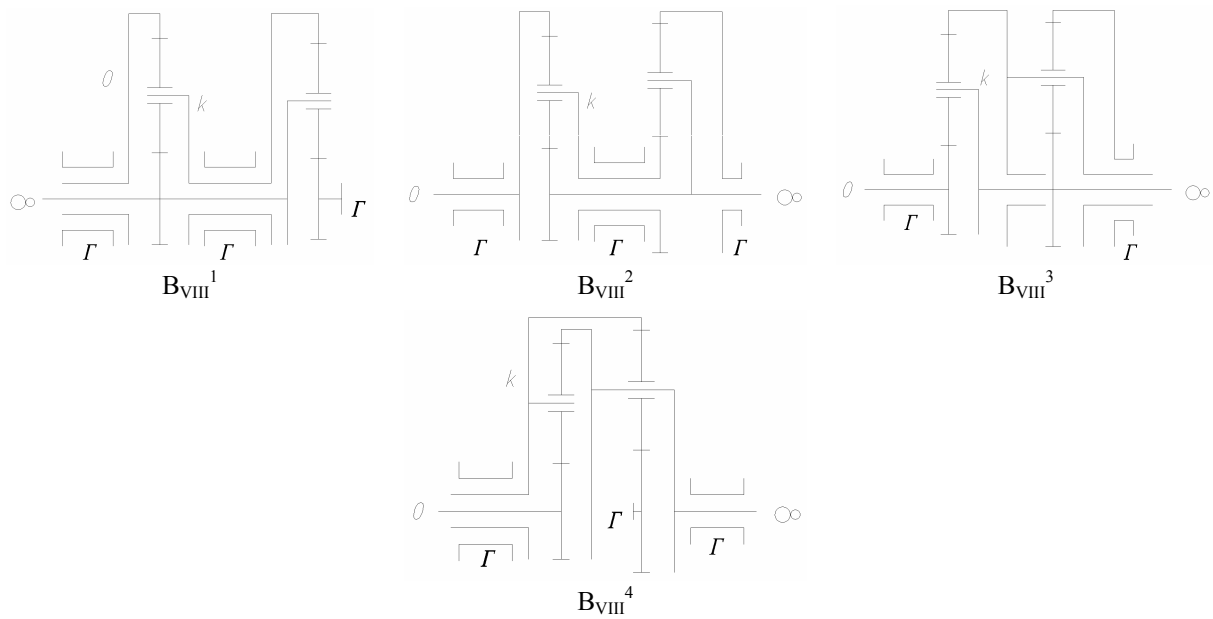
Struktúra egyenlet:

F2-F4, F3-F4, F3-F5

Struktúra mátrix:

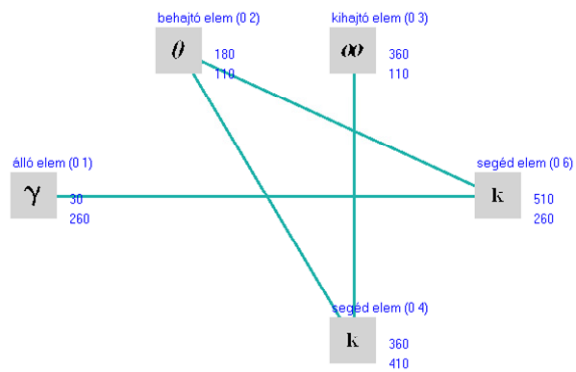
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	0
F(2)	0		0	1	0	0
F(3)	0	0		1	1	0
F(4)	0	1	1		0	0
F(5)	0	0	1	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja kapcsolt elem, az egyszabadságfokúként működő hajtómű karja a kihajtó elem, a másik hajtómű karja a segédelem.

S17



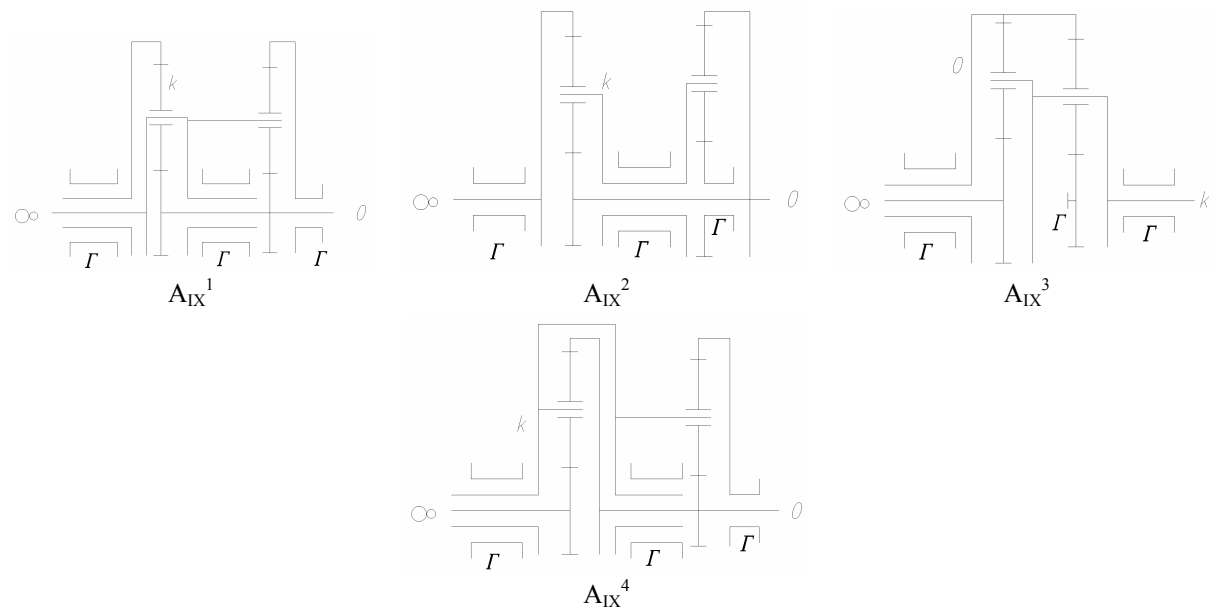
Struktúra gráf

Struktúra egyenlet: $F1-F6, F2-F4, F2-F6, F3-F4$

Struktúra mátrix:

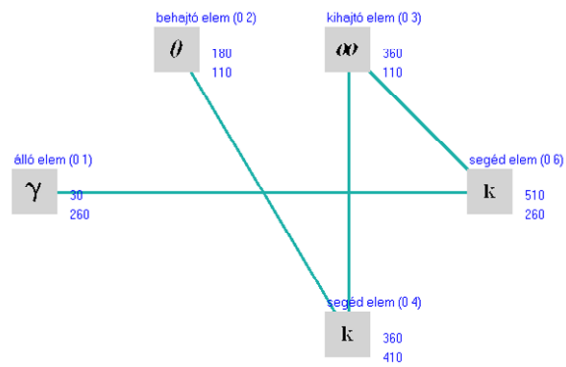
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	1
F(2)	0		0	1	0	1
F(3)	0	0		1	0	0
F(4)	0	1	1		0	0
F(5)	0	0	0	0		0
F(6)	1	1	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja egymással van összekapcsolva, így mind a két bolygómű karja a kihajtó elem. A másik kapcsolt elem a behajtó elem.

S18



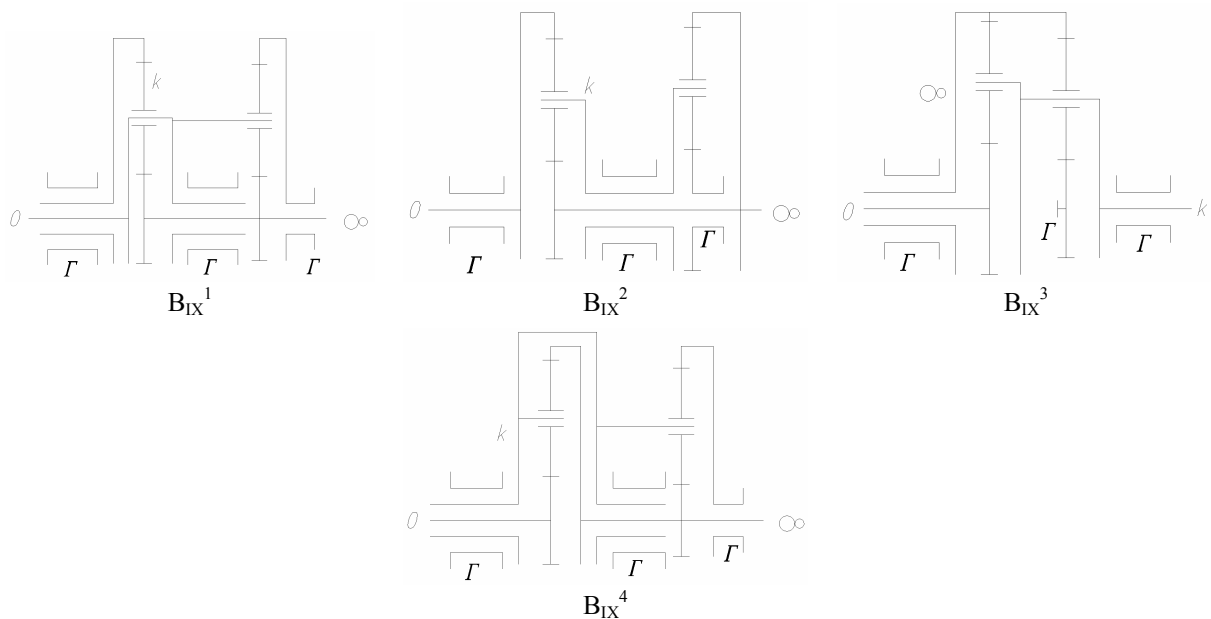
Struktúra gráf

Struktúra egyenlet: $F1-F6, F2-F4, F3-F4, F3-F6$

Struktúra mátrix:

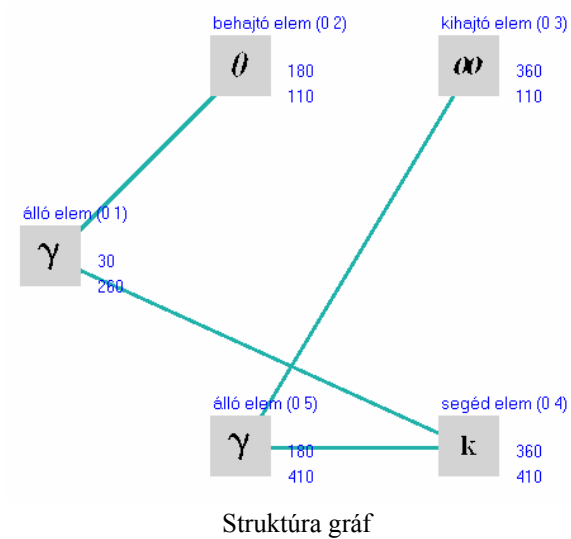
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	0	0	1
F(2)	0		0	1	0	0
F(3)	0	0		1	0	1
F(4)	0	1	1		0	0
F(5)	0	0	0	0		0
F(6)	1	0	1	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja egymással van összekapcsolva, így mind a két bolygómű karja a behajtó elem. A másik kapcsolt elem a kihajtó elem.

S19

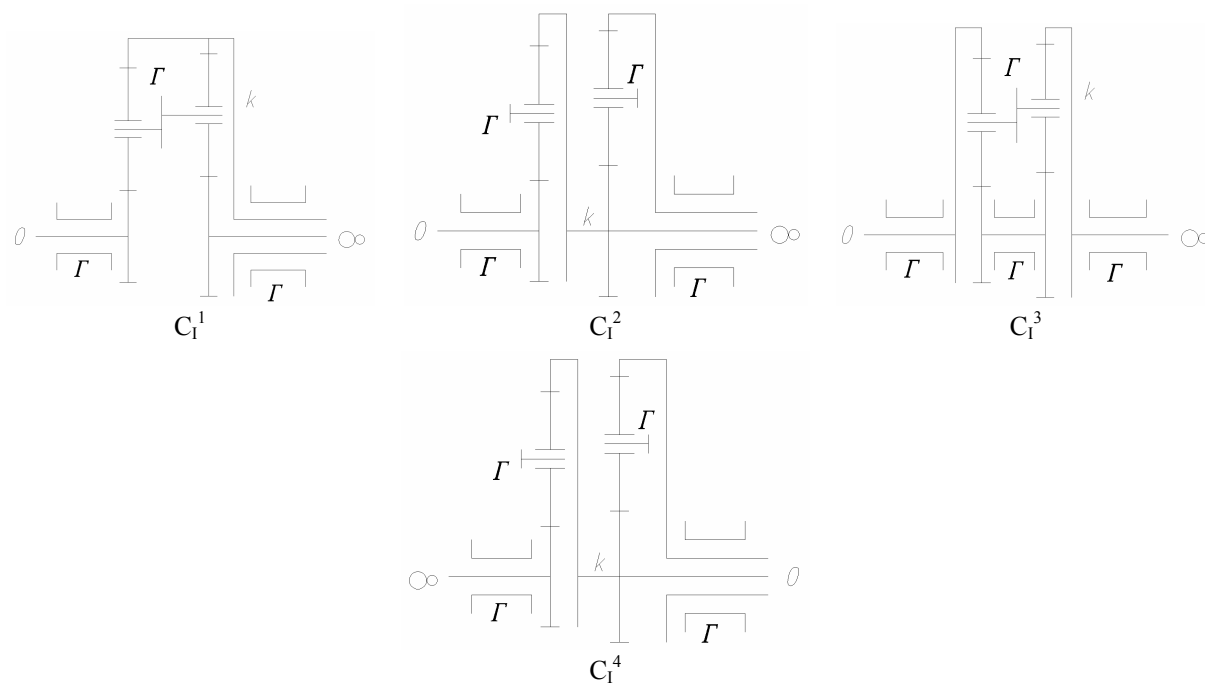


Struktúra egyenlet: $F1-F2, F1-F4, F3-F5, F4-F5$

Struktúra mátrix:

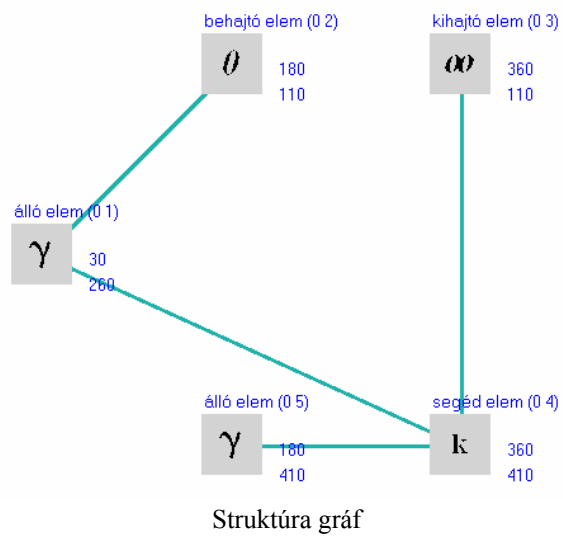
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	1	0	0
F(2)	1		0	0	0	0
F(3)	0	0		0	1	0
F(4)	1	0	0		1	0
F(5)	0	0	1	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómű karja egymással van összekapcsolva, és álló alapelem. A másik kapcsolt elem a segédelem.

S20

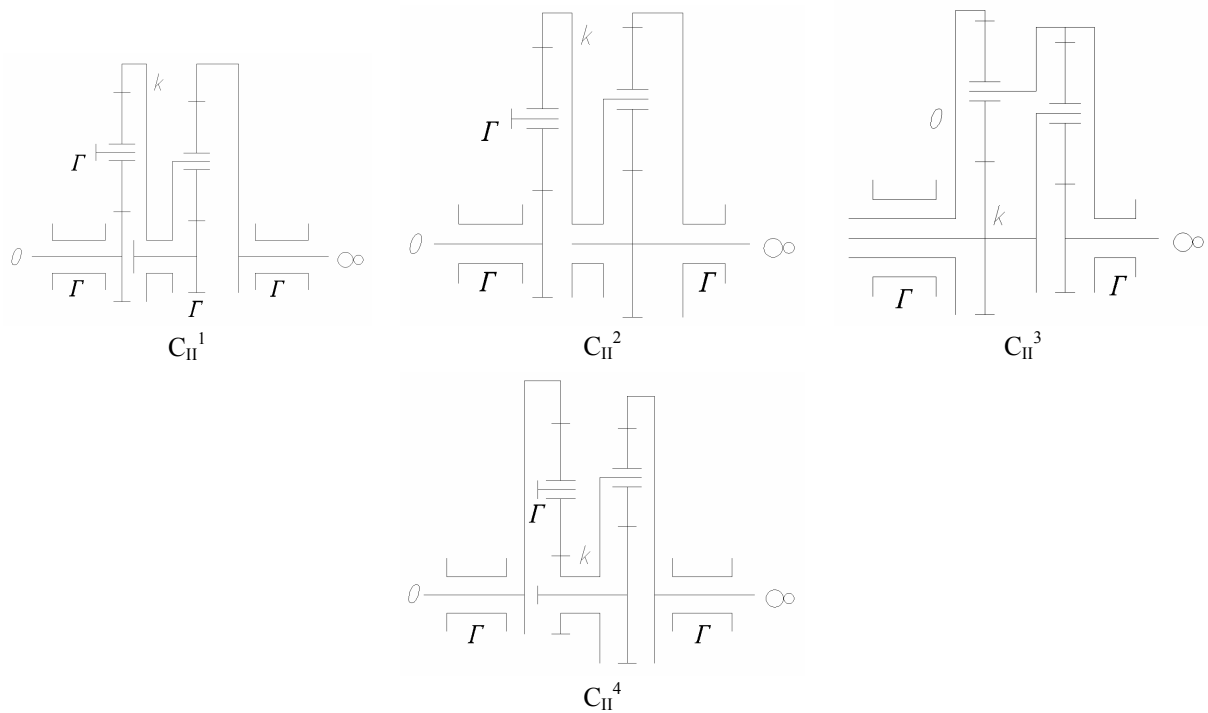


Struktúra egyenlet: $F1-F2, F1-F4, F3-F4, F4-F5$

Struktúra mátrix:

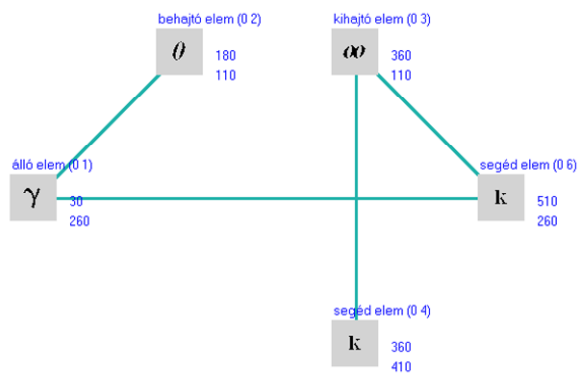
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	1	0	0
F(2)	1		0	0	0	0
F(3)	0	0		1	0	0
F(4)	1	0	1		1	0
F(5)	0	0	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja álló alapelem, a másik bolygómű karja a kapcsolt segédelem.

S21



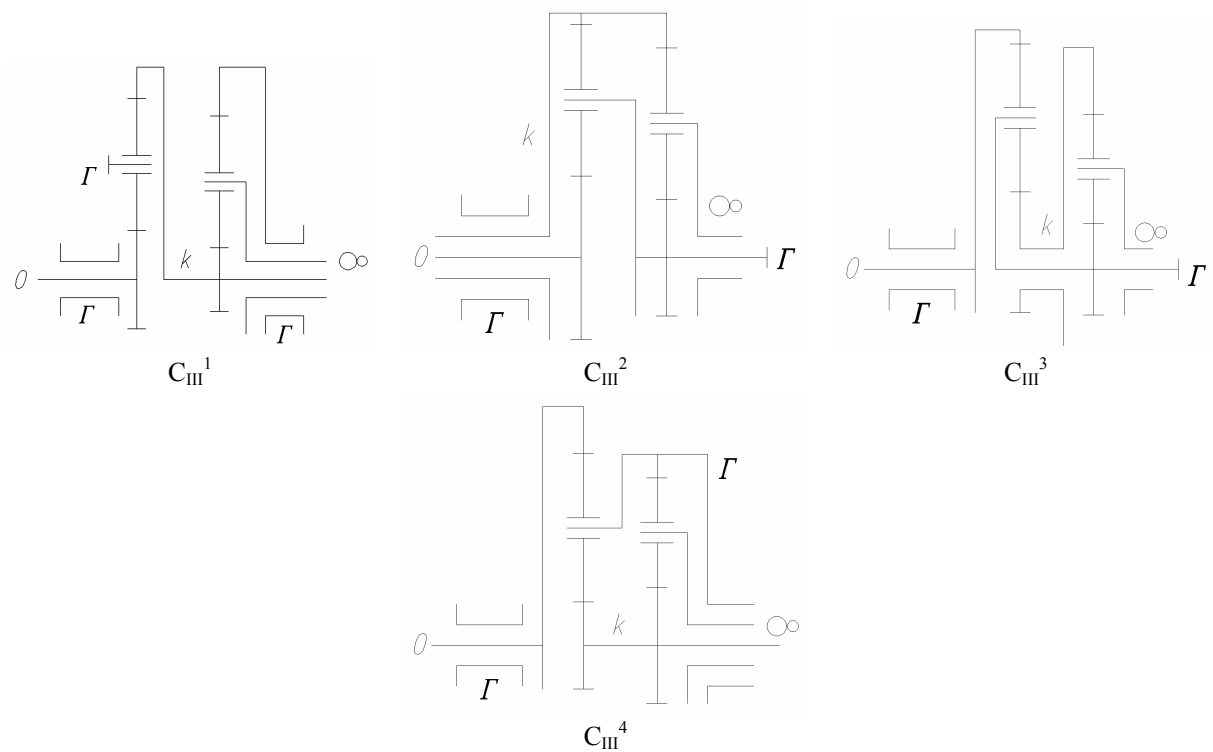
Struktúra gráf

Struktúra egyenlet: F1-F2, F1-F6, F3-F4, F3-F6

Struktúra mátrix:

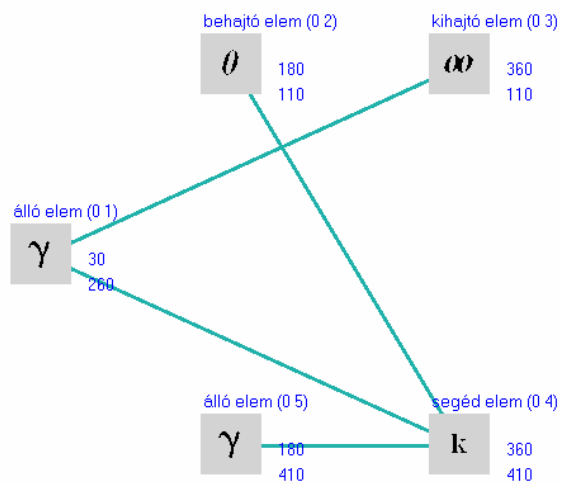
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	0	0	1
F(2)	1		0	0	0	0
F(3)	0	0		1	0	1
F(4)	0	0	1		0	0
F(5)	0	0	0	0		0
F(6)	1	0	1	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja a kimenő elem, a másik bolygómű karja az álló alapelem.

S22



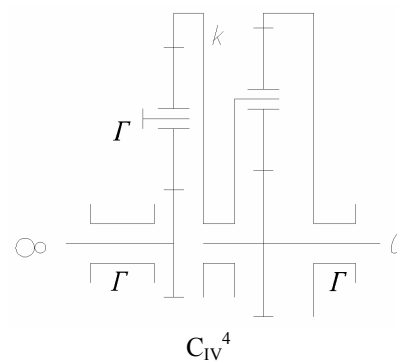
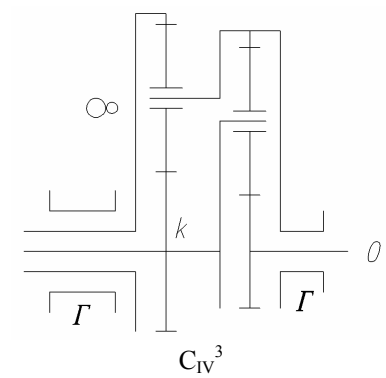
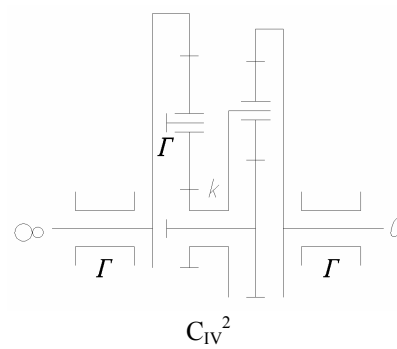
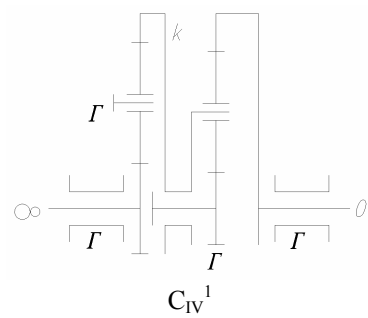
Struktúra gráf

Struktúra egyenlet: $F1-F3, F1-F4, F2-F4, F4-F5$

Struktúra mátrix:

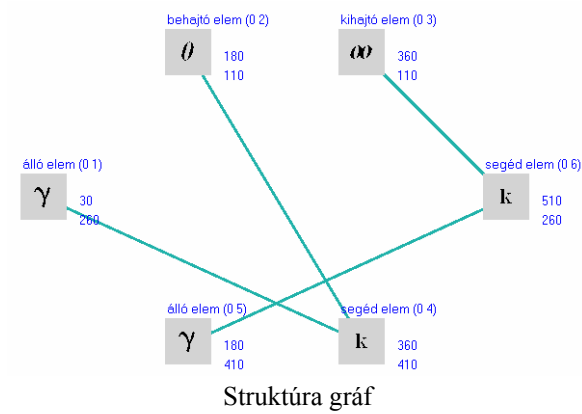
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	1	1	0	0
F(2)	0		0	1	0	0
F(3)	1	0		0	0	0
F(4)	1	1	0		1	0
F(5)	0	0	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Mind a két bolygómu karja kapcsolt; az egyiké álló elem, a másiké a segédelem.

S23

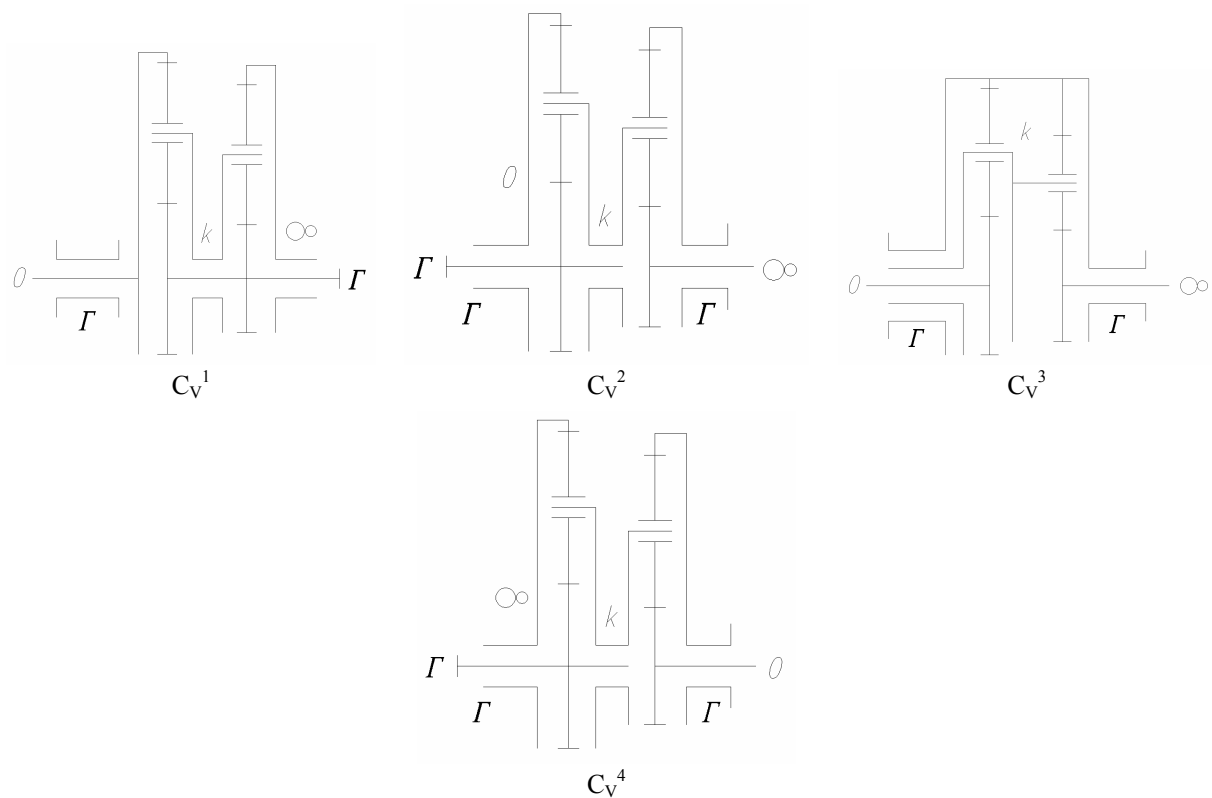


Struktúra egyenlet: F1-F4, F2-F4, F3-F6, F5-F6

Struktúra mátrix:

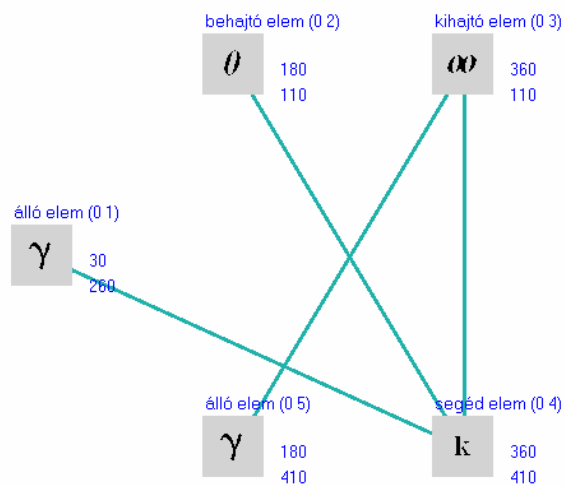
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	1	0	0
F(2)	0		0	1	0	0
F(3)	0	0		0	0	1
F(4)	1	1	0		0	0
F(5)	0	0	0	0		1
F(6)	0	0	1	0	1	

Kinematikai ábrák:



Mind a két hajtómű karja egymással van összekapcsolva, és mind a kettő segédelem.

S24



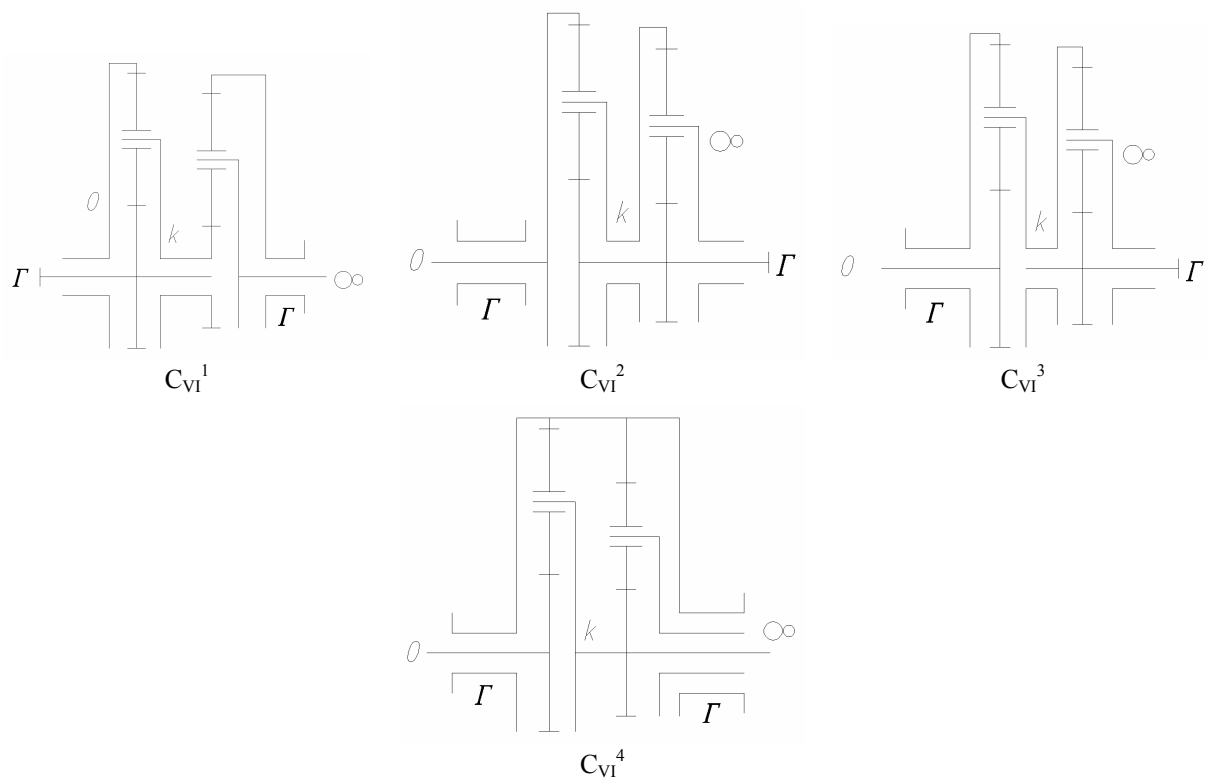
Struktúra gráf

Struktúra egyenlet: $F1-F4, F2-F4, F3-F4, F3-F5$

Struktúra mátrix:

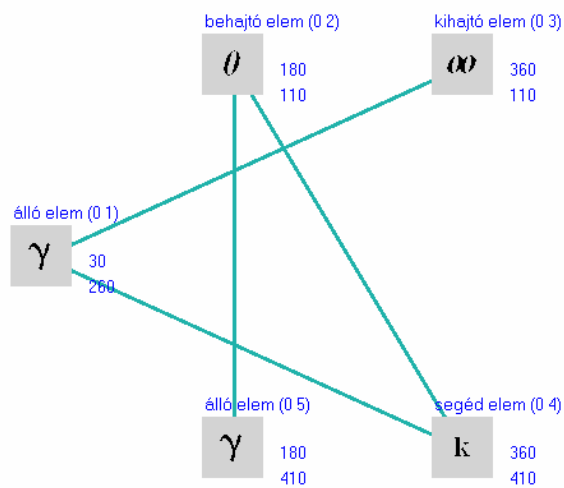
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	0	1	0	0
F(2)	0		0	1	0	0
F(3)	0	0		1	1	0
F(4)	1	1	1		0	0
F(5)	0	0	1	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja kapcsolt segédelem, a másiké a kihajtó elem.

S25



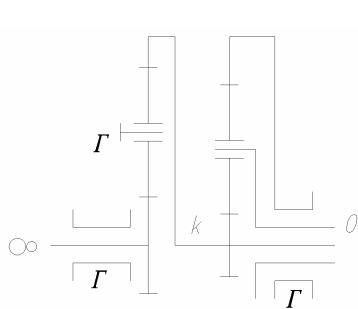
Struktúra gráf

Struktúra egyenlet: $F1-F3, F1-F4, F2-F4, F2-F5$

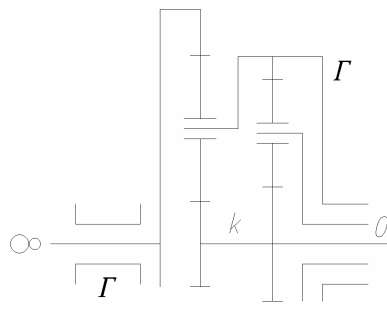
Struktúra mátrix:

	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		0	1	1	0	0
F(2)	0		0	1	1	0
F(3)	1	0		0	0	0
F(4)	1	1	0		0	0
F(5)	0	1	0	0		0
F(6)	0	0	0	0	0	

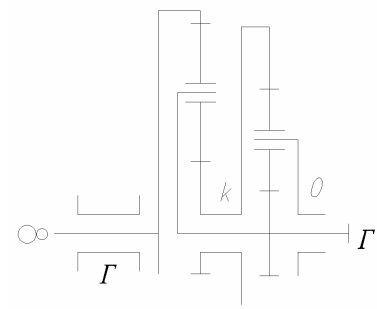
Kinematikai ábrák:



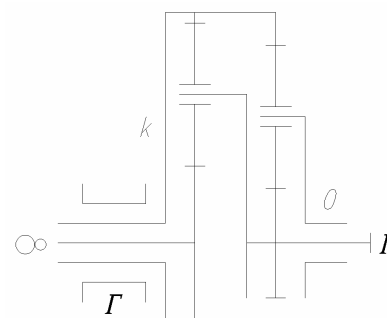
C_{VII}^1



C_{VII}^2



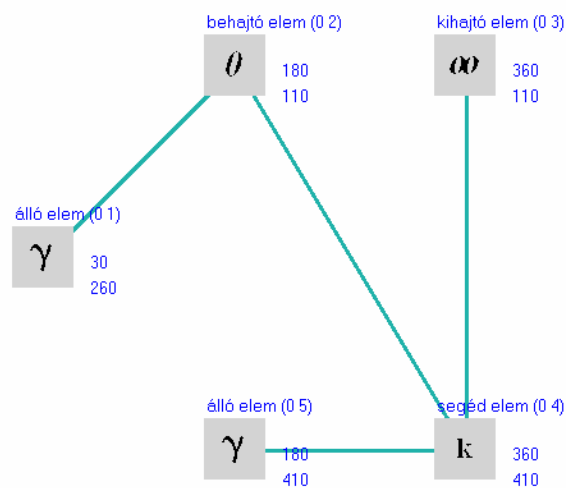
C_{VII}^3



C_{VII}^4

Az egyik bolygómű karja kapcsolt, és álló elem, a másik hajtómű karja a behajtó elem.

S26



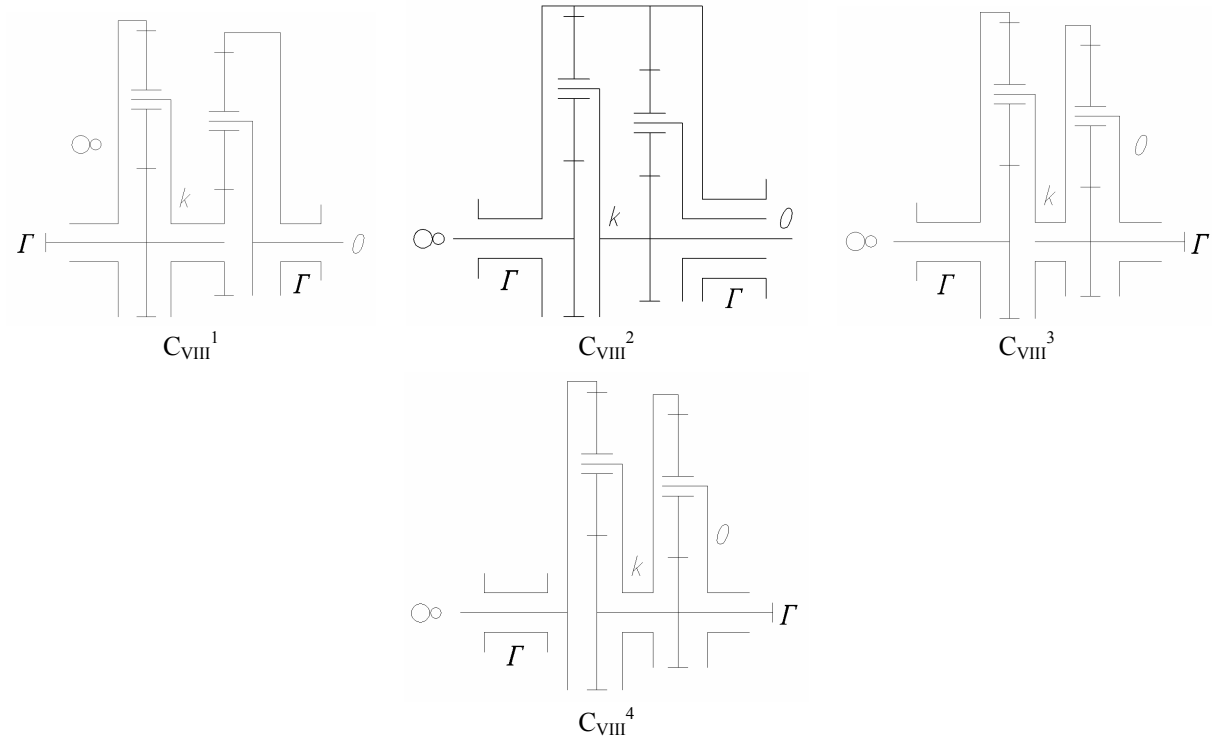
Struktúra gráf

Struktúra egyenlet: $F1-F2, F2-F4, F3-F4, F4-F5$

Struktúra mátrix:

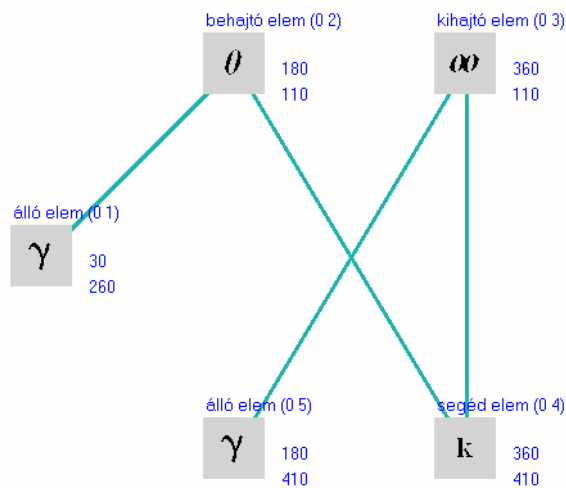
	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	0	0	0
F(2)	1		0	1	0	0
F(3)	0	0		1	0	0
F(4)	0	1	1		1	0
F(5)	0	0	0	1		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómű karja kapcsolt segédelem, a másik hajtómű karja a behajtó elem.

S27



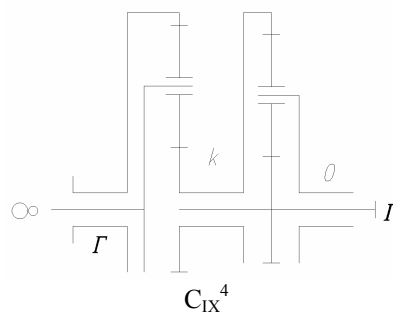
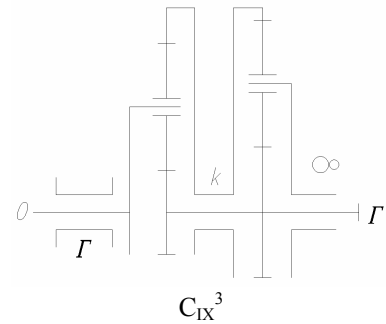
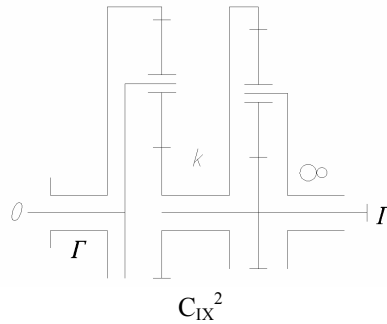
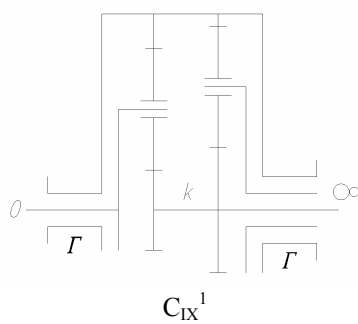
Struktúra gráf

Struktúra egyenlet: $F1-F2, F2-F4, F3-F4, F3-F5$

Struktúra mátrix:

	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
F(1)		1	0	0	0	0
F(2)	1		0	1	0	0
F(3)	0	0		1	1	0
F(4)	0	1	1		0	0
F(5)	0	0	1	0		0
F(6)	0	0	0	0	0	

Kinematikai ábrák:



Az egyik bolygómu karja a behajtó elem, a másiké a kihajtó elem. A két kar nem kapcsolt elem. Az egyik kapcsolt elem az álló elem, a másik a segédelem.