SZENT ISTVÁN EGYETEM
GAZDÁLKODÁS- ÉS SZERVEZÉSTUDOMÁNYI DOKTORI ISKOLA

Fenntarthatósági kritériumok értelmezése játékelméleti modellek alkalmazásával

PhD Doktori értekezés

Dr. Fogarassy Csaba

Gödöllő 2014.

SZENT ISTVÁN EGYETEM

Egyetemi Doktori és Habilitációs Tanács

A doktori iskola megnevezése:	Gazdálkodás és Szervezéstudományi Doktori Iskola
Tudományága:	gazdálkodás- és szervezéstudományok
Vezetője:	Prof. Dr. Szűcs István egyetemi tanár, MTA doktora (DSc.) Gazdaság- és Társadalomtudományi Kar Közgazdaságtudományi, Jogi és Módszertani Intézet
Témavezető:	Prof. Dr. Szűcs István egyetemi tanár, MTA doktora (DSc.) Gazdaság- és Társadalomtudományi Kar Közgazdaságtudományi, Jogi és Módszertani Intézet Prof. Dr. Molnár Sándor egyetemi tanár, intézetigazgató, matematika tudományok (CSc.) Gépészmérnöki Kar Matematikai és Informatikai Intézet

Az iskolavezető jóváhagyása

A témavezetők jóváhagyása

A dolgozatot Fogarassy Tamás (1968-2013) informatikus, fizikus unokatestvérem emlékének ajánlom.

TARTALOMJEGYZÉK

JELÖLÉSEK ÉS RÖVIDÍTÉSEK JEGYZÉKE

AB *mátrixok*

x, y *vektorok*

$\langle x, y \rangle$ *vektorok skaláris szorzata*

$D(f)$ *az f függvény értelmezési tartománya*

$R(f)$ *az f függvény értékkészlete*

φ_k *kifizetőfüggvény (a k − dik célfüggvény)*

$\in$ *halmaz eleme*

L *a döntési változók értékeinek lehetséges halmaza*

x^* *egy játékos Nash − féle egyensúlypontja*

S_i *az i − dik játékos stratégiahalmaza*

S *szimultán stratégiahalmaz*

$S_1 \times S_2 \dots \dots \times S_n$ *stratégiahalmazok direkt szorzata*

Γ $(n: S_1 S_2, \dots \dots S_n; \varphi_1 \varphi_2 \dots \dots \varphi_n)$ *az* S_i *stratégiahalmazokkal és a* φ_i *kifizetőfügvényekkel definiált személyes játék*

X^T *az x vektor transzponáltja*

A^T *az A mátrix transzponáltja*

$u \succ w$ *az u eloszlás dominálja a w eloszlást*

$D_k(A)$ *az A halmaz k − szor folytonosan deriválható függvények osztálya*

∇f *az f függvény gradiense*

K_M ***az ember által létrehozott tőke***

K_H ***a humán tőke***

K_N ***a természeti tőke***

TGÉ ***teljes gazdasági érték***

FGÉ ***fenntarthatósági gazdasági érték***

SMART ***Simple Multi Attribute Ranking Technic***

NE ***Nash-egyensúly***

CIE ***Cleantech Incubation Europe***

BEVEZETÉS

Nagyon nehéz előre kalkulálni, megállapítani, hogy egy-egy beruházási programnak, fejlesztési beavatkozásnak milyen hosszú távú pozitív és negatív hatásai lesznek. Szerte a világon súlyos gond, hogy a környezet és a gazdaság (piac) állapotát egyaránt jelentősen rontják a felelőtlenül magvalósított környezetvédelmi célú, privát és „állami” beavatkozások. Európában se szeri se száma azoknak az innovatív energetikai beruházásoknak, hulladékgazdálkodási és vízkezelési programoknak, amelyek több kárt okoztak a társadalomnak, mintha nem is lettek volna. Magyarországon a környezetvédelmi jellegű közvetlen és közvetett támogatások évente sok ezermilliárd forintot tesznek ki, ezeknek a forrásoknak a rossz irányú felhasználása akár évtizedekre is kedvezőtlen irányba terelhet egy-egy gazdasági szektort. Ezért ezeknek a támogatásoknak a feltérképezése, a piaci mechanizmusok szempontjából is helyes irányba terelése, mielőbbi átstrukturálása hatalmas erőforrásokat szabadíthat fel a társadalom, a nemzetgazdaság számára, egyúttal hozzájárulhat a környezeti állapot, a munkaerőpiac, valamint a jóléti mutatók számottevő javításához. Annak érdekében, hogy ez említett fejlesztési stratégiák, beruházási programok valóban fenntartható gazdasági pályára állíthatók legyenek, olyan „építkezési módra” van szükség a tervezés a megvalósítás folyamatában, amely ezeket a káros folyamatokat, rossz tervezési irányokat eleve kizárja. Dolgozatomban erre a tervezési, vagy módszertani kérdésre keresek megoldásokat, melyeket egyrészt a Rubik Logika, másrészt a modern játékelméleti alkalmazásokra alapoztam.

A környezetileg/gazdaságilag káros támogatások egy jellemző formája, amikor a gazdaságpolitika nincs megfelelő információk birtokában arra vonatkozóan, hogy a fejlesztendő infrastruktúra (pl. energiatermelő rendszerek vagy regionális hulladékkezelők) milyen környezeti károkkal, negatív környezeti jelenségekkel járnak együtt. A környezetvédelmi célú fejlesztések gazdasági, piaci átláthatatlansága miatt, az „állami támogatásoktól mentes” szektorokkal összehasonlítva, ezeken a termékpályákon sokkal kedvezőtlenebb feltételek között valósulhat meg a piaci egyensúlyt célzó gazdasági fejlődés, melynek hiányában akár a túltermelés/túlkínálat, akár az alultermelés/alul kínálat sokkal jellemzőbb jelenségek.

Társadalmi szempontból tehát alapvető követelmény, hogy a beruházások ne csak a gazdasági mutatószámok alapján, hanem a munkaerő piaci és természeti erőforrás felhasználás szempontjából is életképes struktúrát mutassanak. A beruházásokat megelőző értékelési kritériumok, elsősorban az átmenti gazdaságok (EU tag volt szocialista országok) esetében sajnos a termékpályák instabilitása, valamint a folyton változó gazdasági szabályozás, instabil gazdasági környezet miatt, olyan kiszámíthatatlan tervezési környezetet jelentenek, melyre tervezési módszert nem, vagy csak nagyon körülményesen lehet alkotni. Ezért ha kockázatkerülő beruházási, fejlesztési programokkal találkozunk, akkor esetükben jellemző az egyedi törvényszerűségek mentén megfogalmazott projektmenedzsment és kivitelezés, illetve új gazdaságpolitikai termékek (pl. Policy Risk Insurance – Politikai Kockázatok elleni Biztosítás) bevezetésének gyakorlata.

Tovább nehezíti a projektekhez kapcsolódó erőforrások értékelését, hogy a gazdasági mérőszámok gyakorlatilag alkalmatlanok a tervezési folyamatokban való felhasználásra, mivel a fekete és szürkegazdaság jelenléte úgy eltorzítja, illetve a piaci mérőrendszeren kívülre tereli a folyamatokat befolyásoló feltételrendszert, hogy azok irreális gazdasági optimumokat, megtérülési időket jelölnek ki. A körülményeket ismerve, nem véletlen tehát, hogy a *United Nations Environment Programme*-ja vagy az *European Energy Centre* megújuló energia finanszírozási feltételrendszert vizsgáló programjai is azokat az extrém finanszírozási gyakorlatokat vizsgálják, amelyek a megújuló energiatermelés rendszereinek elterjedését gátolják. A fejlődő országokban jelenleg jellemző magas beruházási kockázat miatt, a banki projektfinanszírozás kerüli a 3-4 éven túli megtérülési időket, amelyek a megújuló energiahordozók esetében sem jelenthetik a fenntartható gazdasági fejlesztés alapját. A gazdaságpolitikai beavatkozások ezért a szabályozási vagy adózási rendszereken keresztül folyamatosan próbálják korrigálni a fejlesztés feltételrendszereit, de ezek rendszerint sokkal több kárt okoznak a környezetvédelemhez kötődő termékpályákon, mint amennyi előnyt jelentenek. Ennek következtében a klímavédelemhez vagy a megújuló energiatermeléshez kapcsolódóan, bebukott projektek tömegével találkozhatunk mind az európai, mind az amerikai gazdasági térségben. A lassabb, de biztonságos fejlesztések, ahol a megtérülési idő 8 éven túl realizálható, csak az állami beruházások esetében elképzelhető, viszont a tőkeforrások rugalmassága itt nem biztosítható a szükséges mértékben. A fenntartható fejlesztésekhez (low-carbon innovációkhoz) szükséges beruházási gyakorlat tehát alapvetően hiányzik a fejlesztési folyamatokból, viszont amit az előbbi áttekintésből is láthattunk, ennek megvalósulása nem csak a projektmenedzsment folyamatok tökéletesítésének függvénye.

A dolgozat fő célja, hogy a gazdasági és politikai instabilitás tudatában is fenntarthatóan tervezhetők legyenek a különböző környezetvédelmi célú vagy a klímaváltozást előnyösen befolyásoló beruházások. Ennek érdekében egy olyan projektfejlesztési gyakorlat, menedzsment rendszer, illetve tudományosan is értelmezhető, matematikai összefüggéseken alapuló modell kialakítása, amely újszerű megközelítésével, a fent említett problémákat országhatárok figyelembe vétele nélkül kezelni tudja.

Kutatásom fő céljának megvalósítása érdekében az alábbi részcélok megvalósítását tűztem ki, fogalmaztam meg kutatási programomban:

- A gazdasági érték és fenntarthatóság viszonyának klasszikus és új szemléletű bemutatása szakirodalmi feldolgozáson keresztül.
- Fenntarthatósági szempontok matematikai értelmezése, a fenntartható gazdasági egyensúly vagy vállalati stratégiák játékelméleti megközelítésének bemutatása, klasszikus egyensúlypontok keresésének értelmezése a nem-kooperatív játékelméleti megoldásokban.
- Játékelméleti módszerekkel történő konfliktus feloldás és kompromisszum keresés bemutatása, adaptálhatóságának vizsgálata a fenntartható kooperatív vállalati stratégiákban.
- A Rubik kocka felépítésének és kirakásának fenntarthatósági értelmezése, a fenntarthatóság és Rubik kirakási algoritmusok közötti kapcsolat vizsgálata.

A jelenlegi gazdasági környezetet terhelő piaci hibák, externális, azaz a paci feltételrendszeren kívül elhelyezkedő, de azt jelentősen befolyásoló feltételrendszer feltérképezése létfeltétele a fenntartható környezetvédelmi beruházások megvalósításának. Társadalmi elvárás, hogy a fenntartható gazdasági struktúrákat támogató fejlesztések externális hatásainak pontos definiálásával, azok tipizálásával, az extern hatások összesítésével olyan (matematikai függvényekkel is leírható) törvényszerűségek kerüljenek megfogalmazásra, melyek a fenntarthatóság jelenleg érvényes kritériumrendszerét meg tudják fogalmazni a piaci szereplők, politikai döntéshozók számára.

A fentiek figyelembe vételével, célkitűzésem megfogalmazásánál elsősorban az alábbi Hipotézisek igazolását tartottam szem előtt:

- H1: A környezetvédelmi célú vagy a klímaváltozást előnyösen befolyásoló beruházások életképességét, fenntarthatóságát befolyásoló projekt tulajdonságok kapcsolati rendszerét modellekkel le lehet írni.
- H2: Játékelméleti módszerekkel történő fenntarthatósági egyensúlykeresés során, az összehasonlításra kerülő tényezők egymás közötti kapcsolatának függvényszerű leírását el lehet végezni.
- H3: A többváltozós próbafüggvények alkalmasak azoknak a tulajdonság-csoportoknak a kiválasztására, amelyek a projekt sikeres megvalósítását dominánsan befolyásolják.

- H4: A 3x3x3 Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatában.

Feltételezem, hogy a Rubik kocka *„Layer by layer”* kirakási módszere, azaz a sorról sorra történő kirakása alkalmas a fenntarthatósági kritériumok modellezésére, vagyis egy-egy projektfejlesztési folyamat (pl. megújuló energetikai beruházások) során ezzel a módszerrel a fenntarthatósági kritériumok alapvetően betarthatók.

A Rubik kocka kirakási algoritmusokra vonatkozó hipotézis értelmében, az egymás mellett elforgatott kockák, azaz az egymásra befolyást gyakorló projekt tulajdonságok fenntartható kapcsolati rendszerét matematikailag is le tudjuk írni, így azok (pl. Nash-féle) egyensúlyi pontja, tehát játékelméleti modellekkel (véges játék, zérusösszegű játékok, oligopol játékok stb.) determinánsan meghatározhatók.

Jelen dolgozatban továbbá azt szeretném hitelt érdemlően bizonyítani, hogy a felvetett hipotézis, mely szerint a Rubik kocka kirakási algoritmusa, nevezetesen a *„Layer by layer” módszer*, alkalmas a projektfejlesztés folyamatának modellezésére. Ugyanakkor, megfelelő játékelméleti modellekkel reprezentálható az egyes projekttulajdonságok kapcsolati rendszere, így a különböző környezet és klímabarát beruházások mind a humánerőforrás tervezés, mind a környezeti feltételek megőrzése, javítása szempontjából is jól tervezhető, alacsony kockázatú gazdasági környezetben valósulhat meg.

1. SZAKIRODALMI ÁTTEKINTÉS

A részletes szakirodalom feldolgozás során – a hazai és nemzetközi szakirodalmakat – három témakör köré csoportosítottam:

- A fenntarthatósági dilemmák és tolerancia kérdések a gazdasági mérőrendszerekben.
- Egyensúlykeresés nem-kooperatív játékelméleti modellekkel, kooperatív stratégiák a vállalati stratégiaalkotásban.
- A Rubik kocka struktúrája és kirakásához kapcsolódó matematikai algoritmusok.

A fenntarthatósági dilemmák között az erős és gyenge fenntarthatóság definiálása, az ökológiai közgazdaságtan, valamint a környezetgazdaságtan fenntartható mérőszámainak, illetve a teljes gazdasági érték új szempontú megközelítésének kérdése kerül bemutatásra. Ugyanakkor rávilágítok többek között arra is, hogy miért nem igazak azok a gazdasági elméletek, melyek a környezetgazdaságtan és az ökológiai közgazdaságtant gyakorlatilag egységes rendszernek tekintik.

A játékelméleti fejezetekben azok a hazai és nemzetközi játékelméleti tételek sorát tekintjük át, amelyek a fenntarthatóság és a vállalati együttműködés, a kooperatív stratégiák kialakításának területeit érintik. Kiemelt témaként mutatom be a Nash-egyensúlyt, mely ugyan már 20 évvel ezelőtt került leírásra, de alkalmazásának és hasznosságának megítélése nagyon eltérő. Nash az optimalizálási játékokat kooperatív és nem kooperatív játékokra osztotta fel, elmélete szerint a kooperatív játék olyan játék, melyben a játékosok között a gyakorlati együttműködés egyszerűen kikényszerítődik. Álláspontja szerint nem kooperatív játékokról csak akkor beszélünk, amikor a játékosok között nem lehet egyességet kikényszeríteni. A Nash-egyensúly esetében az egyes játékosok stratégiái a legjobb válaszok más játékosok stratégiáira, így egyik játékos sem akar ebből az állapotból kimozdulni, más új, esetleg kooperatív stratégiát választani.

A szakirodalom feldolgozás harmadik egysége alapvetően a Rubik kocka mítoszról szól. Rubik kocka, mint a Világ egyik legjelentősebb magyar gyökerekkel rendelkező játéka, még ma is fogalom és matematikai rejtély egyben. A hármas szám misztikus jelentése végigkísérte a kocka sikertörténetét. Sokak szerint a kocka olyan kapcsolatokat szimbolizál, amelyek az embert a természettel, a természetes léttel kötik össze. A Rubik kockáról szóló leírásban a kirakási módszerek, az „Isteni Szám", illetve a kirakási folyamat matematika értelmezése kerül bemutatásra.

1.1. Fenntarthatósági dilemmák és tolerancia kérdések

A fenntarthatóság értelmezésére a közgazdaságtanban rendkívül eltérő megközelítésekkel találkozhatunk. Közismert dilemma a fenntarthatósági célrendszerek/stratégiák megfogalmazásakor, hogy a *gyenge fenntarthatóság*, avagy az *erős fenntarthatóság* elméleti összefüggéseit alkalmazzuk az egyes esetekben.

1.1.1. Erős és gyenge fenntarthatóság értelmezése

Kerekes Sándor (2005) „A fenntarthatóság közgazdaságtani értelmezése" című tanulmányában rendkívül alaposan és jól árnyékoltan végig követhető a különböző, környezeti szempontból jelentős közgazdaságtani felfogások fejlődése. Jelen doktori disszertációban viszont arra törekedtem, hogy elsősorban a különböző erőforrások egymásra hatásának rendszerében mutassam be a fenntarthatósági koncepció közgazdasági rendszerbe ágyazását, mely révén világos képet kapunk arra vonatkozóan, hogy a környezeti kérdések kezelése piaci eszközökkel is lehetséges, ha abban a piaci hibák nem halmozódnak.

A gondolatmenet alapján tehát az az indirekt feltételezés körvonalazódik, hogy a piacgazdasági alapelveken nyugvó fejlődés önmagában is fenntartható fejlődési struktúrát jelent (túltermelés és túl kereslet mentes környezetet), mely azonban az indirekt globális hatások, valamint az ezeket kompenzálni kívánó direkt kormányzati, állami beavatkozások okán, a pareto optimális egyensúlytól igen távoli egyensúlyi pontban mozog, a megjelenő piaci hibák miatt, a fenntarthatósági ponttól meglehetősen távol. A kívánatos piaci egyensúly, a fenntarthatóság definiálása azonban nem egyértelmű, főleg ha ezt a folyamatot gazdaságpolitikai érdekek is vezérlik (Spash, 2011).

A probléma és megoldáskeresés nem új keletű, mert Pearce és Atkinson már 1993-ban, az globális gazdasági fejlődés korai szakaszában megfogalmazzák és Hicks–Page–Hartwick–Solow-szabálynak nevezik azt az általánosan megfogalmazott követelményt, ami az úgynevezett gyenge fenntarthatósági kritérium feltételeit fejezi ki.

A Hicks–Page–Hartwick–Solow-szabály képlettel:

$$dK/dt = K = d(K_M + K_H + K_N)/dt \geq 0$$

$$\text{ahol: } K = K_M + K_H + K_N$$

Pearce és Atkinson három tőketípust különböztetnek meg:

- K_M - az ember által létrehozott (vagy újratermelhető) mesterséges tőke (utak, ipari létesítmények, gyárak, lakóházak stb.),
- K_H - a humán tőke (a felhalmozott tudás és tapasztalat),
- K_N - a természeti tőke, amit igen tágan értelmeznek, magában foglalja a természeti erőforrásokat (ásványok, termőföld stb.), az élet fenntartásához nélkülözhetetlen egyéb természeti javakat is, mint például a biodiverzitás, a szennyezést asszimiláló kapacitás stb.

Pearce és Atkinson szerint (1995) amennyiben elfogadjuk a neoklasszikus közgazdaságtan azon alapfeltevését, hogy a tőkejavak egymással korlátlanul helyettesíthetők, akkor a gyenge fenntarthatóságot a következő képlettel fejezhetjük ki:

$$\frac{dK}{dt} = \frac{d(K_M + K_H + K_N)}{dt} \geq 0$$

Közgazdasági értelemben tehát a gyenge fenntarthatóság akkor áll fenn, ha a társadalom rendelkezésére álló tőke javak összértéke időben nem csökken.

Miután a tőke a megtakarítások és az értékcsökkenés különbségeként határozható meg, és a fenti három tőkeelem közül a humán tőke értékcsökkenése nullának tekinthető (első közelítésként elfogadva, hogy az emberiség által felhalmozott tudás és tapasztalat nem „kopik”, nem avul el), a gyenge fenntarthatósági kritérium (Z - fenntarthatósági érték) Pearce és Atkinson szerint a következő képlettel írható le (Kerekes, 2005 alapján):

$$Z = \frac{S}{Y} - \frac{\delta_M * K_M}{Y} - \frac{\delta_N * K_N}{Y}$$

Ahol S megtakarítás, Y Bruttó Nemzeti Termék (GNP), a delta M (δ_M) és delta N (δ_N) az ember alkotta és a természeti tőke amortizációs rátái. Amennyiben nem engedjük meg a tőkeelemek közti helyettesítést, akkor Pearce–Atkinson (1993) szerint az erős fenntarthatósági kritériumhoz jutunk. A szigorú fenntarthatóság teljesülésének feltétele, hogy a természeti tőke értéke időben ne csökkenjen:

$$\frac{\delta_N * K_N}{Y} \geq 0$$

Pearce és Atkinson (1993) szerint egy gazdaság akkor fenntartható, ha a megtakarítások nagyobbak, mint az elértéktelenedés, értékcsökkenés mértéke az ember által létrehozott mesterséges és a természeti tőke vonatkozásában (1. táblázat). Az 1. táblázatból nyomon követhető, hogy a megtakarítások és a GNP hányadosaként értelmezett erőforrás összérték akkor pozitív, azaz fenntartható, ha alapvetően a természeti tőke értékcsökkenése kisebb, mint az újratermelhető mesterséges tőke értékcsökkenése, vagy az adott évben létrehozott erőforráskészlet nagysága jelentősen meghaladja a vizsgált két erőforrás ($K \geq K_M + K_N$) amortizációjának összértékét.

A természettudósok és ökológusok a tőkeelemek helyettesíthetőségét és így a gyenge fenntarthatóságot nem fogadják el, sőt a szigorú fenntarthatósággal is problémáik vannak, hiszen a természeti tőkén belüli kompenzációkat ez utóbbi is megengedi. Az ökológiai közgazdászok zöme a szigorú fenntarthatósággal kapcsolatban kiköti, hogy a természetben nem szabad irreverzibilis változásokat (pl. fajok kipusztulása, diverzitás csökkenés stb.) előidézni (Pearce-Atkinson, 1995).

1. táblázat: Fenntartható és fenntarthatatlan gazdasági rendszerek (1992-es adatok alapján)

Gyenge fenntarthatóság képlete	*S/Y*	-	δ_M/Y	-	δ_N/Y	=	***Z***
Fenntartható gazdasági szerkezet							
Brazília	*20*		*7*		*10*		*+3*
Finnország	*28*		*15*		*2*		*+11*
Németország	*26*		*12*		*4*		*+10*
Magyarország	*26*		*10*		*5*		*+11*
Japán	*33*		*14*		*2*		*+17*
USA	*18*		*12*		*4*		*+2*
Marginális gazdasági szerkezet							
Mexikó	*24*		*12*		*12*		*0*
Philippines	*15*		*11*		*4*		*0*
Fenntarthatatlan gazdasági szerkezet							
Etiópia	*3*		*1*		*9*		*-7*
Indonézia	*20*		*5*		*17*		*-2*
Madagaszkár	*8*		*1*		*16*		*-9*
Nigéria	*15*		*3*		*17*		*-5*
Mali	*-4*		*4*		*6*		*-14*

Megjegyzés:
S megtakarítás, Y Bruttó Nemzeti Termék (GNP), a delta M (δ_M) és delta N (δ_N) az ember alkotta és a természeti tőke amortizációs rátái, Z – fenntarthatósági érték

Forrás: Pearce-Atkinson: Are national economies sustainable? Measuring Sustaibanle Development, 1993

A természeti és mesterséges tőke viszonya határozza meg alapvetően a gyenge és erős fenntarthatóság közötti különbséget. A gyenge fenntarthatóság elmélete azt mondja ki, hogy a természeti és mesterséges tőke egymással helyettesíthető viszonyban áll. A fenntarthatósági kritérium akkor is teljesül, ha a két tőketípus együttes értéke nem csökken. Ez a kritérium teljesül, ha a természeti erőforrások pusztulása során, legalább ugyanabban az értékben mesterséges tőkeforrások jönnek létre (Gowdy-Erickson, 2005).

Az erős fenntarthatóság elmélete viszont határozottan kimondja, hogy a természeti tőke mesterséges tőkével nem, vagy csak nagyon kismértékben helyettesíthető, így ez a kritérium abszolút fenntarthatósági korlátot hoz létre, amelynek egy meghatározott szintjét meg kell őrizni a fenntarthatóság érdekében (Málovics-Bajmócy, 2009).

1.1.2. Ökológiai közgazdaságtan versus környezetgazdaságtan

Turner és társai szerint (1996) szerint a gazdasági rendszerek és az ökológiai rendszer viszonyával kapcsolatos vita nagyon jó lehetőség a közgazdászok számára, hogy az optimista és a pesszimista elméletek, közgazdasági, gazdasági stratégiák egymással ütköztethetők legyenek benne. Az alapvetően növekedésorientált, techno-optimista környezetgazdaságtan, valamint a stabil (egyensúlyi) méretorientált, techno-pesszimista ökológiai közgazdaságtani megközelítések különböző fenntarthatósági felfogást jelenítenek meg az elméletekben.

Az alapvető elméletek abban megegyeznek, hogy a környezetgazdaságtan és az ökológiai közgazdaságtan gyakorlatilag egységesek, a kiindulási alap az, hogy a természeti tőke mindenféle gazdasági tevékenység alapja, így ezen irányzatokban a fenntarthatósággal kapcsolatos véleménykülönbségeket elsősorban nem a gyenge, vagy az erős fenntarthatósághoz kapcsolódó, a mesterséges és természeti tőke viszonyával kapcsolatos vita határozza meg (Spash, 1999). A különbséget inkább a környezet-gazdaságtani és ökológiai közgazdaságtani megközelítés viszonyrendszerében kell keresni (Turner et al., 1993).

A környezetgazdaságtan vizsgálódásainak alapját azok a makrogazdasági mutatószámok (GDP, GNP stb.) adják, amelyek a neoklasszikus közgazdaságtanban is jól ismertek, és hibáikkal együtt mutatják számunkra a jólét változásának jellemzőit. Ezzel ellentétben az ökológiai közgazdaságtan egy problémaközpontú megközelítést alkalmaz a megismerés során, amely sokszor közgazdsági szempontból nem is igazán konkretizálható rendszerjellemzőt jelent. Az ökológiai közgazdaságtan esetében a fenntarthatóság megállapítása szempontjából fontos a többi társadalom- és természettudományi ismeret mélyebb integrálása (Korten, 2011).

Az ökológiai közgazdaságtan nagyon sok esetben fenntartással kezeli a környezetgazdaságtan leegyszerűsítő nézeteit és problémamegoldó javaslatait. A környezeti problémák okait a piaci hibáknál, piaci elégtelenségi problémánál (az externális hatások halmozódása, nem megfelelő internalizálása) sokkal mélyebben rejlőnek látja, és gyökeres intézményi változásokat szeretne elérni a fenntarthatóság felé haladás érdekében (Málovics-Bajmócy, 2009).

Az ökológiai közgazdaságtan a fenntarthatóság eredeti koncepciója mentén körvonalazódik, mely szerint erősen kötődik három fő kérdéskörhöz: az etika, a gazdaság és a gazdasági érték kapcsolati rendszerének meghatározásához.

Ennek értelmében tehetjük fel a központivá váló kérdéseket: Mi is a gazdaság? Hol vannak az erőforrásrendszerek határai? Lehet-e társadalmilag és/vagy környezetileg felelős gazdasági magatartásról beszélni? Milyen értékrendszer alapján hozzák meg a gazdasági döntéseket a gazdasági és politikai döntéshozók? Melyik vizsgálati rendszer mit hajlandó tolerálni és mit nem?

1.1.3. Teljes gazdasági érték és a fenntarthatósági gazdasági érték viszonya

Az elmúlt évtizedekben, a közgazdaságtanban, ezen belül a környezetgazdaságtanban jelentős fejlődés történt a természeti környezet gazdasági értékének osztályozása terén. Az értékelés alapja az értékelő, az ember és az értékelt jószág között fennálló hagyományos kapcsolat átformálása volt. Az emberek egyre nagyobb értéket tulajdonítanak az egyes nem pontosan definiálható jószágoknak, így a környezeti javaknak is (Korten, 2013). Ezen értékek aggregátumát felfoghatjuk az ún. teljes gazdasági érték fogalmaként. A teljes gazdasági értéket (TGÉ) több összetevőre bonthatjuk, melyben a két fő elemet a használattal összefüggő, illetve azzal nem összefüggő értékkomponensek jelentik.

A teljes gazdasági érték tehát:

TGÉ = Használattal összefüggő értékek + Használattól független értékek

A teljes gazdasági értékelés (TGÉ) rendszere, mely alapvetően a környezeti értékek számbavételét is egy korszerű módon kezelő közgazdaságtani adaptáció, mégsem tudja megfelelően kezelni a pénz időértékéhez kapcsolódó erőforrás transzformációk kérdését. Ezen segít egy ma még kevéssé ismert megközelítés, melyet Molnár 2005-ben fenntartható gazdasági értéknek nevezett el (Molnár, F. 2005).

A fenntarthatósági gazdasági érték (rövidítve: FGÉ) a teljes gazdasági érték (rövidítve: TGÉ) hagyományos koncepciójához képest kiemelten épít a jövőképekre, illetve a visszacsatolások és egymásra-hatások megjelenítésére, mivel ezek központi kérdések a fenntarthatóság biztosítása szempontjából. A fenntarthatósági gazdasági érték koncepcióban a gazdaság csupán származtatott módon jelenik meg, nem úgy, mint a hagyományos teljes gazdasági érték koncepciónál. A FGÉ koncepció sokkal hangsúlyosabban veszi figyelembe a pénz időértékével kapcsolatos kérdéseket, azaz nem feledkezik meg a későbbi generációkról, illetve a javak megőrzéséhez kapcsolódó értékrészekről sem (Molnár F., 2005). A fenntartható gazdasági érték tehát egy olyan értékelési mód, amely képes – a lokális információk figyelembe vételével, azokat célirányosan felhasználva – integráltan bemutatni a természeti tőke mellett a társadalmi és a technikai tőke-elemek változását is, melyet a teljes gazdasági érték csak igen korlátozottan valósít meg (Kiss, 2004).

A gazdasági egyensúlyok keresése tehát alapfeltétele a fenntarthatósági kritériumok teljesülésének, így az ehhez kapcsolódó kutatások, modellkísérletek elsődleges preferenciát kaptak az elmúlt évek során. Az egyensúlypontok keresése (egyfajta fenntarthatósági gazdasági értékként definiálva), a játékelméleten alapuló összefüggések pontos feltárása tehát a legnépszerűbb kutatási területek közé tartozik. Annak érdekében, hogy többek között a megújuló energetikai beruházások esetében is egzakt képet kapjunk a fenntarthatóság valós kritériumrendszeréről, a piaci környezetet befolyásoló egyensúlyi körülményekről, a kooperatív, azaz az együttműködésen, illetve a nem-kooperatív játékelméleti összefüggések alapos ismeretét nem nélkülözhetjük (Fogarassy et al., 2007). A következő játékelméleti fejezetekben azt kívánom bemutatni, hogy a gazdasági döntéseket befolyásoló piaci tényezők, hogyan hatnak a fenntarthatóságot jelentő „gazdasági egyensúlypontok” meghatározására, azok változására.

1.2. Nem-kooperatív játékok elmélete

A közgazdaságtan alapvetően azt állítja, hogy a racionális magatartás a következetes preferenciákon alapszik. Ha egy adott személy preferenciái kielégítenek néhány alapvető igényt, konzisztencia-kritériumot, akkor ezeket a preferenciákat egy jól definiált hasznosságfüggvénynek is megfeleltethetjük. Így a racionális magatartás a hasznosságfüggvény maximalizálásának, azaz hasznosság-maximalizálásnak tekinthető. Ebből az következik, hogy a racionális magatartásnak ezt a definícióját a hasznosságelméletnek nevezhetjük, mondja Harsányi (1995).

Ugyanakkor Harsányi (1995) azt is kimondja, hogy rengeteg bizonyíték van arra, hogy ez a feltevés nem teljesül, mégis a közgazdaságtan azon a feltevésen nyugszik, hogy az emberek preferenciái teljesen következetesek. A legtöbb közgazdász úgy tekint erre, mint hasznos leegyszerűsítő feltevésre, úgy érvelve, hogy az a közgazdaságtudomány, melyet erre a feltevésre építünk, általában meglehetősen jó, ha nem is teljesen megfelelő előrejelzéseket ad a gazdasági rendszer működéséről.

A gazdaságpolitika vitát folytat arról, hogy a társadalom magatartását hogyan és miként befolyásolják a gazdasági élet történései. Annak bizonyítására, hogy ez a mindennapi élet velejárója, tényszerű bizonyíték az, hogy az elmúlt évtizedekben például az átlagos családméret a legtöbb országban jelentősen redukálódott. Ennek oka, hogy a nagycsalád gazdasági előnyei lényegesen csökkentek, ezzel szemben a költségei nagymértékben megnövekedtek az urbanizációs hatás és az intenzív technológiai fejlődés miatt. A kisebb családmodell preferenciája, a 3-4 fős családi életközösség fogyasztási szokásai, életvitele így fontos szerepet játszott az optimumkeresés folyamatában (Hobson, 2012).

Annak érdekében, hogy ne csak a gazdasági mérőszámmal jellemezhető választási szempontok, hanem a valódi életminőséget befolyásoló tényezők is egy-egy család választási szokásaihoz, azaz pl. a vásárlásaihoz kapcsolódó kritériumrendszeréhez kapcsolódjon, több, egymástól független tulajdonsághoz is kapcsolható összehasonlítási módszer kialakítására kell törekednünk (Vincze, 2009). Ha döntéseink során egy rendszert vagy objektumot egyszerre több tulajdonság alapján kell megítélnünk, illetve ezek a tulajdonságok jórészt ellentmondásos tulajdonsághalmazt jelentenek, az ehhez kapcsolódó rendezési elvek bonyolult összefüggéseket mutatnak, akkor a matematikai modellezést, illetve ezek szoftveralkalmazásait hívhatjuk segítségül. A többcélú rendszerek vizsgálatát tehát célszerű úgy kezdeni, hogy kijelöljük a legfontosabb kritériumokat. Több lehetőség optimalizálásakor el kell fogadnunk azt a tényt, hogy Axelrod (2000) szerint nem tudunk minden figyelembe vett tulajdonság szerint egyszerre optimalizálni. Ennek az az egyszerű oka, hogy rendszerint nem ugyanazon alternatíva mellett lesz valamennyi tulajdonság-reprezentáló célfüggvény optimális.

A kritériumrendszer kijelöléséről általánosságban is elmondhatjuk, hogy nem tudunk minden egyes tulajdonságot figyelembe venni, amely a rendszerünket, vagy a vizsgált objektum működését befolyásolja. Ezért a számunkra lényeges tulajdonságok közül ki kell választanunk azokat, amelyek érdemesek a további vizsgálódásra. Az egyes tulajdonságoknak teljesen függetleneknek kell lenniük egymástól. Ez azért nagyon fontos a kiválasztás során, hogy az egyes kritériumok között ne legyenek átfedések, ugyanis ezek a „kereszttulajdonságok" felesleges vizsgálatokat, időveszteséget okozhatnak az analízis során.

A többcélú optimalizálási feladatok során tehát a következő teendőink lehetnek annak érdekében, hogy a modellalkotás folyamata, a probléma megoldás feltételrendszere körvonalazódjon (Forgó et al., 2005):

1. Kritériumrendszer megadása (főbb tulajdonsághalmazok meghatározása)
2. A tulajdonsághalmazok függetlenségi vizsgálata (tulajdonság átfedések elkerülése)
3. Döntési változók, paraméterek megadása a tulajdonsághalmazon belül (determinisztikus vagy sztochasztikus, azaz valamilyen valószínűségi szinttel teljesülő tulajdonságok megjelölése)
4. Korlátozó feltételek megadása a halmazra vonatkozóan (halmazok felállítása)

5. Lehetséges kritériumok kijelölése a feltételrendszerhez a halmazon belüli célfüggvények megadása (célfüggvények száma véges)
6. A célfüggvények maximumának keresése

A ***többcélú programozási feladat*** általános alakja Molnár et al. (2010a) szerint a következő alakban írható fel (ha feltesszük, hogy $D\ (\varphi_k\ ) = L$ és tetszőleges $X \in L$ esetén $\varphi_k\ (X)$ valós szám):

$$\boldsymbol{X \in L}$$

$$\boldsymbol{\varphi_k\ (X) \rightarrow max \qquad (k=1, 2, \ldots.., n)}$$

X - a döntési változók rendszere
L - a döntési változók értékeinek lehetséges halmaza
$D\ (\varphi_k)$ – a függvény döntési tartománya
n – a célfüggvények száma
φ_k = k - adik célfüggvény, azaz a kifizetőfüggvény

Azonban meg kell jegyeznünk, hogy a többcélú problémamegoldás iránti igény nem a jelenkor követelménye, mivel a racionális döntésekhez kapcsolódó viselkedések függvényszerű törvényszerűségeit Neumann János már 1928-ban papírra vetette, illetve Oskar Morgensternnel írt közös művében, melynek címe: „Játékelmélet és gazdasági viselkedés”, már 1944-ben pontosan leírta (Neumann –Morgenstern, 2007).

1.2.1. Egyensúlypontok keresése a nem-kooperatív játékelméletben

A játékelmélet alapvetően a többcélú problémák megoldásával, az úgynevezett stratégiai játékokkal foglakozik. A játékelmélet a matematika egyik interdiszciplináris jellegű ága, amely főként azzal a kérdéssel próbál megbirkózni, hogy mi a racionális viselkedés olyan helyzetekben, ahol minden résztvevő döntéseinek eredményét befolyásolja a többiek lehetséges választása. Egy problémát, vagy probléma megoldást stratégiai játéknak nevezünk, ha a döntéshozóknak a játék kimenetelére a fennálló feltételek és szabályok keretei között, befolyásuk lehet (Mező, 2011a).

A játékelméleti megoldásokban mindig azt feltételezzük, hogy a játék kimenetelét minden játékos számára egy célfüggvénnyel, vagyis az előbbiekben már említett kifizető függvénnyel jellemezhetjük. Az egyes játékosok (szereplők) részére pedig a minél nagyobb kifizető-függvény érték jelenti a játék előnyösebb kimenetelét. A játékosok döntéseit, azaz a döntések kimenetelét a végső eredményre, nevezzük a játékos stratégiájának. A játékelméleti megoldások két- vagy többszemélyes megoldásait ismerjük. A játékelméleti megoldás nem kooperatív, ha problémamegoldás közben a játékosok vagy szereplők versengenek egymással, illetve logikusan következik, kooperatív a játék, ha a játékosok között kialakul az együttműködés. A játékelméletben kiemelten fontos az egyensúlypontok keresése. Ha az egyensúlykeresés arra irányul, hogy az összes játékos összes stratégiájának figyelembe vétele esetén, egyik játékosnak sem származik előnye abból, ha stratégiáján változtat, amíg a többi játékos azonos módon játszik tovább, Nash-egyensúlynak nevezzük (Szidarovszky-Molnár, 1986).

A Nash-egyensúly elmélete John Nash-től származik, akit az elmélet kidolgozásáért Harsányi Jánossal egy időben, pontosan 20 évvel ezelőtt Nobel-díjjal is kitüntettek. Nash egyensúlyi elméletén keresztül megmutatta azt, hogy minden véges játéknak van legalább egy egyensúlyi pontja.

Nash az optimalizálási játékokat kooperatív és nem kooperatív játékokra osztotta fel, elmélete szerint a kooperatív játék olyan játék, melyben a játékosok között a gyakorlati együttműködés egyszerűen kikényszerítődik. Álláspontja szerint nem kooperatív játékokról csak akkor beszélünk, amikor a játékosok között nem lehet egyességet kikényszeríteni. Egy nem-kooperatív játék a játékosok adott stratégiái esetén akkor és csak akkor nevezhető stabilnak, ha létezik az a bizonyos Nash-egyensúly. A Nash-egyensúly esetében az egyes játékosok stratégiái a legjobb válaszok más játékosok stratégiáira, így egyik játékos sem akar ebből az állapotból kimozdulni, más stratégiát választani. A játék nem lesz stabil akkor, ha nincs meg a Nash-egyensúlyi pont, mert ebben az esetben mindig van legalább egy olyan játékos, akinek a stratégiája nem a legjobb választ jelenti az adott helyzetben, ezért érdekelt lesz abban, hogy új stratégiát keressen magának (Harsányi, 1995).

Amint az már korábban leírtam, kooperatív játékok esetén az egyensúlyi helyzet lehet stabil akkor is, ha a stratégiai kombinációk valamelyike nincs összhangban a többi stratégiával, mert a stratégia együttműködés előbb-utóbb kikényszerítődik. A gazdasági életben azonban leginkább olyan egymás mellett futó, egymást figyelembe nem vevő stratégiaalkotással is találkozhatunk, amely a többcélú döntési folyamatot, vagy a döntési optimum kialakítását nem veszik figyelembe (Molnár-Kelecsényi, 2009). Az európai vagy gazdaságpolitikai gyakorlatból jó példa erre a környezetvédelmi, vagy megújuló energetikai fejlesztésekre vonatkozó stratégiák zöme, mivel itt gyakran találkozunk ellentétes irányú stratégiaalkotással.

Alaptétel, hogy a környezetvédelmi célú fejlesztések ellentétes irányúak a gazdaságfejlesztési prioritásrendszerrel (pl. az üvegházgáz csökkentést, fosszilis energia felhasználást célzó program az energiafelhasználás minimalizálását célozza, a másik a szennyező energiahordozó növelését. Jó példa erre az EU két vezető stratégiája: EU Low-carbon Roadmap 2050 vs. Nuclear Power in France (2014). Franciaországban jelenleg 58 db nukleáris erőmű üzemel, és további fejlesztések folynak, míg Németországban most döntöttek arról, hogy 2020-re mindegyiket (most 8 db üzemel) leszerelik.

A kooperatív játékok esetén még akkor is stabil lehet a választott stratégia, ha egy stratégia-kombináció nem Nash-egyensúly, amennyiben a játékosok egyezségre jutnak, hogy ezt a stratégia-kombinációt választják. A Nash-egyensúly egyensúlypont keresési dilemmájának bemutatása során – nem-kooperatív játékok esetére - alapvetően Mező István (2011b) Játékelmélet című tanulmányára támaszkodom, amennyiben ettől eltérek, azt külön jelzem a leírás során.

Definíció1:

A Nash-egyensúlypontra vonatkozó *Definíció* szerint:

Egy $J = (n, S, (\varphi_i)_{i=1}^{n})$ n személyes játék egyensúlyi pontján vagy a stratégiáján olyan $(x_1^*, \ldots\ldots, x_n^*) \in S$ pontot (stratégiai *n*-est) értünk, melyre

$$\varphi_i(x_1^*, \ldots, x_{i-1,}^* x_i^*, x_{i+1,}^*) \geq \varphi_i(x_1^*, \ldots, x_{i-1,}^* x_i, x_{i+1,}^*) \qquad (1.1)$$

teljesül minden $i = 1, \ldots\ldots, n$ játékosra. Az egyensúlyi pontot tehát *Nash-féle equilibrumnak* nevezzük. [rövidítve lehet: $\varphi_k(x_1^*,\ldots, x_{k,}^* \ldots\ x_{n,}^*) \geq \varphi_k(x_1^*, \ldots, x_k, \ldots x_{n,}^*)\ ahol\ k = 1,2,\ldots n$]

Ha az 1.1. egyenlősége szigorú, akkor szigorú egyensúlyról beszélünk.

Ha mást nem mondunk, egyensúlyi pont alatt nem szigorú egyensúlyi pontot értünk.

Szemléletesen, az i –dik játékos saját kifizetését akkor tudja maximalizálni, ha az egyensúlyi stratégiáját, azaz x_i^* -ot játssza, feltéve, hogy a többi játékos is ugyanígy tesz. Az egyensúlyi állapot meghatározásához szükségünk lesz a következő fogalmakra:

Definíció2:

Egy $n-$ személyes J játékot konstans összegűnek nevezzük, ha a játékos által elért nyeremények és veszteségek végösszege egy állandó c szám, az összes lehetséges stratégia mellett.

Képletszerűen:

$$\sum_{i=0}^{n} \varphi_i(x) = c\ (x\ \in S).$$

Ha $c = 0$, a játékot *zérusösszegűnek* nevezzük.

A kétszemélyes, zérusösszegű játékok alkalmasak arra, hogy jobban tudjuk szemléltetni az egyensúlyi pont fogalmát. Alapul véve egy $(x_1^*, x_2^*) \epsilon\ S$ egyensúlyi pontot, (1.1) alapján

$$\varphi_1(x_1^*, x_2^*) \geq \varphi_1(x_1, x_2^*) \text{ minden } x_1 \in S_1 \text{ mellett} \qquad (1.2)$$

és

$$\varphi_2(x_1^*, x_2^*) \geq \varphi_2(x_1^*, x_2) \text{ minden } x_2 \in S_2 \text{ esetén.}$$

A játék zérusösszegű, így

$$\varphi_1(x_1, x_2) + \varphi_2(x_1, x_2) = 0\ , \varphi_2(x_1, x_2) = -\ \varphi_2(x_1, x_2),$$

és a második egyenlőtlenség így alakul:

$$-\varphi_1\ (x_1^*, x_2^*) \geq -\varphi_1\ (x_1^*, x_2).$$

Átírva a képletet:

$$\varphi_1\ (x_1^*, x_2^*) \geq \varphi_1\ (x_1^*, x_2).$$

Figyelembe véve (1.2) egyenlőtlenséget is,

$$\varphi_1\ (x_1, x_2^*) \leq \varphi_1\ (x_1^*, x_2^*) \leq \varphi_1\ (x_1^*, x_2).$$

adódik.

Ez az egyenlőtlenség-rendszer azt mondja, hogy ha az (x_1^*, x_2^*) egyensúlyi pontból az első játékos egyoldalúan kilép valamilyen x_i^*-tól különböző stratégiát választva, a kifizető függvény csak kisebb vagy egyenlő lehet. Ha a második játékos lép ki, az első játékos kifizető függvénye nagyobb vagy egyenlő lesz – mivel a játék zérusösszegű, ez azt jelenti, hogy az ő „fizetsége” pedig nem növekedhet.

Definíció3:

Legyen adott két játék, melyek csak kifizető-függvényekben különböznek:

$$J = (n, S, (\varphi_i)_{i=1}^n) és\ J^{'} = \left(n, S, (\varphi_i^{'})_{i=1}^n\right).$$

$J - t$ és $J^{'} - t$ stratégiailag ekvivalensnek nevezzük, ha van olyan pozitív szám, és vannak olyan $b_i - számok\ i = 1, \ldots., n)$, hogy

$\varphi_i^{'}(x) = a\varphi_i(x) + b_i$ minden $x\ \in S\ és\ i = 1, \ldots., n$ esetén.

A következő tétel kifejezi azt az egyébként intuíciónk alapján is világos tényt, hogy stratégiailag ekvivalens játékokat egyformán kell játszani.

Tétel1:

Stratégiailag ekvivalens játékok egyensúlyi pontjai megegyeznek.

Bizonyítás: Legyenek adottak $J = (n, S, (\varphi_i)_{i=1}^{n})$ és $J^{'} = \left(n, S, (\varphi_i^{'})_{i=1}^{n}\right)$ stratégiailag ekvivalens játékok, és legyen $(x_1^*, \ldots., x_n^*) \in S$ egy egyensúlyi pontja $J - nek.$ Az egyensúlyi pont (1.1) definiciója értrelmében az $i - dik$ játékos minden $x_i \in S_i$ stratégiája mellett

$$\varphi_i(x_1^*, \ldots.., x_{i-1}^*, x_i, x_{i+1}^*, \ldots. x_n^*) \leq\ \varphi_i(x_1^*, \ldots.., x_{i-1}^*, x_i, x_{i+1}^*, \ldots. x_n^*)$$

Mivel $J^{'}$ - stratégialilag ekvivalens $J -$ vel, így adott $a\ > 0\ és\ b_i$ konstansok mellett

$$a\varphi_i^{'}(x_1^*, \ldots., x_{i-1}^*, x_i, x_{i+1}^*\ \ldots.., x_n^*) + b_i \leq$$

$$\leq a\varphi_i^{'}(x_1^*, \ldots., x_{i-1}^*, x_i, x_{i+1}^*\ \ldots.., x_n^*) + b_i.$$

Amiből a pozitív előjele miatt következik, hogy

$$\varphi_i^{'}(x_1^*, \ldots., x_{i-1}^*, x_i, x_{i+1}^*\ \ldots.., x_n^*) \leq \varphi_i^{'}(x_1^*, \ldots., x_{i-1}^*, x_i, x_{i+1}^*\ \ldots.., x_n^*)$$

Ez pedig pontosan azt jelenti, hogy $(x_1^*, \ldots. x_n^*)$ egyensúlyi pontja $J^{'}$ -nek.

Tétel2:

Minden konstans összegű játékhoz található olyan zérusösszegű játék, amely vele stratégiailag ekvivalens és így az egyensúlyi pontjai egybeesnek.

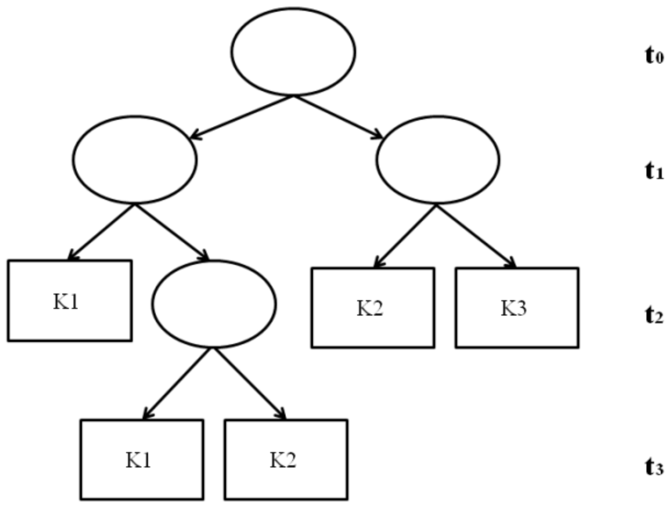
Bizonyítás: A konstans összeget egyszerűen vonjuk ki a kifizetőfüggvények értékéből. Így minden kimenetel zésusösszegű lesz és az új játék stratégiailag ekvivalens marad a régivel. Az előző tétel miatt ezért egyensúlyi pontjaik is egybeesnek. Ha konstans összegű játékoknak tekintünk, az előző tétel miatt elegendő azokon belül csak a zérusösszegűekkel foglalkoznunk. Egy kétszemélyes játék akkor zérusösszegű, ha az egyik játékos minden esetben annyit nyer, amennyit a másik veszít (vagy fordítva). Ekkor tehát

$$\varphi_2(x_1, x_2) = -\ \varphi_1(x_1, x_2)$$

a fenti függvényt alkalmazhatjuk és nem is szoktunk kifizetőfüggvényt használni, egyszerűen csak $\varphi -$ jelölést használjuk pl. $\varphi_1$helyett.

1.2.2. Véges játékok elméleti összefüggései

Ha egy n – személyes játék $S_k\ (k = 1,2, \ldots. . n)$ stratégiahalmazai végesek, akkor a játékot végesnek mondjuk. A véges játékok tipikusan lehetnek kétszemélyes és n - személyes, azaz kettőnél több szereplős játékok. A véges játékok függvényszerű kapcsolati rendszerében egyértelműsíthető, hogy a többcélú problémák megoldásakor a stratégia halmazai, azaz a meghatározott paraméterek, vagy kritériumcsoportok véges számúak, de mindenképpen nagyobbak kettőnél. Mivel az általunk feszegetett problémakör, vagyis a környezetvédelemhez, megújuló energetikai fejlesztésekhez kapcsolódó beruházások, projektfejlesztési döntésfolyamata n - személyes $(n\ \geq 2)$, ezért a több személyes játékok viselkedését célszerű elemeznünk (Szidarovszky et. al 2013). Ugyanakkor feltételezzük, hogy a játék nem egy időpillanat alatt játszódik le, hanem előre rögzített $t_0, t_1, t_2, \ldots.$ időpontokban, egy előre megadott sorrend szerint csak egyetlen játékos változtathatja meg a rendszer állapotát. Ezt az állapotot egy fagráffal ábrázolhatjuk (lásd a 1. ábrán). Ez az optimumkeresési folyamat jelentős mértékben hasonlít a 3x3x3-as Rubik kocka megoldási alternatíváihoz, ugyanis a kocka sorról-sorra („layer by layer" módszer) történő kirakása esetén kevertségének, azaz a rendezetlenségi állapotának függvényében döntjük el, hogy melyik legrövidebb megoldási kombinációsort (mélységi szintet) választjuk a rendezettség fokának növeléséhez, azaz a kocka kirakásához (Nagy G., 2008).



1.ábra: Döntési fagráf az előre rögzített időpontokkal (t0, t1, t2, t3) és kombinációsorokkal (K1, K2, K3)

Forrás: saját szerkesztés (MIEA, 2005 alapján)

A játékosok által alkotott rendszer állapota, azaz a többcélú problémák megoldásai egy problémafával, vagyis egy fagráffal ábrázolhatók, ugyanakkor Forgó et al. (1999) alapján tegyük fel, hogy:

a) a fának van egy kiindulópontja (a t_0 időponthoz tartozó állapot, ami a $t_1, t_2 \ldots\ldots, n$ -felé tart),
b) a kiindulóponthoz és minden további elágazási ponthoz egy előre megadott szabály szerint egy-egy játékos van rendelve, aki az ebből a csúcsból kiinduló véges sok él közül választ, és a rendszer állapotát áthelyezi ennek az élnek a kezdőpontjából a végpontba,
c) minden játékos minden $t_k (k \geq 0)$ időpillanatban ismeri a játék eddigi lefutását és tudja annak pillanatnyi állapotát,
d) a fagráf minden egyes végpontjában mindegyik játékos számára adott egy-egy kifizető függvény érték, amelyet a játékosok ismernek.

Tegyük fel továbbá, hogy a fagráf bizonyos csúcspontjaihoz nincs játékos rendelve, az innen való továbbhaladás előre adott eloszlások szerint véletlenül történik, azaz itt a játékosoknak nincs döntésük, illetve van, de csak szimbolikusan. Az egyöntetűség kedvéért rendeljünk az ilyen csúcsokhoz is egy-egy játékost, az ő döntésük azonban csak formális. Ekkor a játék kifizető függvényeit természetesen a véletlen eloszlások szerint vett várható értékeikkel kell helyettesítenünk. Az itt leírt játékot véges fagráffal ábrázolható, teljes információs játéknak nevezzük (Molnár-Szidarovszky, 1995; Molnár 1994).

Tétel 3:

Minden véges fagráffal ábrázolható teljes információs játék rendelkezik legalább egy egyensúlyponttal.

Bizonyítás: Jelölje $\boldsymbol{I}$ a fagráf kiindulási pontját, valamint $V_1, V_2, \ldots\ldots V_M$ a végpontjait. Esetünkben a fagráf körmenetes, avagy összefüggő gráf, $\boldsymbol{I}$ – ből minden $V_k$végpontba pontosan egy út vezet. A leghosszabb ilyen út hosszát a fagráf hosszának nevezzük. A tétel bizonyítását Molnár és Szidarovszky (2011a) a játékot ábrázoló fagráf hosszára vonatkozó teljes indukcióval végezzük.

Ha $h(F) = 0$, akkor a fagráf csak kezdőpontból áll. Ez azt jelenti, hogy a játékosoknak egyetlen stratégiájuk van. Ekkor nyilvánvalóan ez az egyetlen egyensúlyi pont. Tegyük fel ez után, hogy $h(F) \geq 1$. Jelölje m a kezdőpontból kiinduló élek számát, és legyenek $U_1, U_2, \ldots\ldots, U_m$ a kezdőpontból kiinduló élek végpontjai. Jelölje továbbá $F_k(k = 1,2, \ldots, m)$ az U_k kezdőpontú, az I csúcsot nem tartalmazó maximális részgráfot. Nyilvánvaló, hogy $F_k$végpontjai F végpontjai közül valók. Jelölje ezután $\Gamma_k = \left(n; S_1^{(k)} \ldots, S_n^{(k)}; \varphi_1^{(k)} \ldots, \varphi_n^{(k)}\right)$ az eredeti játék F_k -ra való leszűkítését, azaz F_k csúcspontjaihoz az $F - ben$ hozzájuk rendelt játékosokat rendeljük, végpontjaihoz pedig az $F - ben$ hozzájuk tartozó kifizetőfüggvény-értékeket választjuk.

Nyilvánvaló ekkor, hogy $h(F_k) < h(F)$, így indukciós feltevésünk értelmében Γ_k játékoknak létezik $x^{(k)*} = (x_1^{(k)*}, \ldots\ldots x_n^{(k)*})$ egyensúlypontja. Tegyük fel, hogy az F gráfhoz tartozó játékban a kezdőponthoz az i_0 játékost rendeltük.

a) Tegyük fel, hogy az i_0 - dik játékos szabadon dönthet a kezdőpontból kiinduló m irány között. Ekkor nyilvánvalóan $i \neq i_0$ esetén $S_{i_0} = X_{k=1}^{m} S_i^{(k)}$

$$S_{i_0} = X_{k=1}^{m} S_i^{(k)} \; x \; \{1,2, \ldots \mathrm{m}\}$$

Tehát az i_0 – dik játékosnak az F_k gráfban való továbbhaladáson kívül döntenie kell a kezdőpontból való elindulásról is. Jelölje $\varphi_{i_0}\left(x^{(k)*}\right)$ számok legnagyobbjának indexét, ekkor nyilvánvalóan a Γ_k $(1 \leq k \leq m)$ játékok $x^{(k)*}$ egyensúlypontjaiból, az I kezdőpontból az U_{k0} -ba való lépéssel történő kiegészítéssel nyert stratégia-n-es az F gráf adott játék egyensúlypontját szolgáltatja.

b) Tegyük fel, hogy az i_0 – dik játékos a kezdőpontból egy előre adott $p_1, p_2, \ldots\ldots, p_m$ valószínűség eloszlás szerint véletlenül választ a játék elindítását illetően. Ekkor Γ_k játékok $x^{(k)*}$ egyensúlypontjainak direk szorzata adja az eredeti játék egyensúlypontját. Ehhez bizonyításul feltételezzük, hogy $k = 1,2, \ldots, m$ és $i = 1,2, \ldots, n$ esetén

$$\varphi_i^{(k)} x^{(k)*} \geq \varphi_i^{(k)} (x_1^{(k)*}, x_i^{(k)*} \ldots\ldots x_n^{(k)*})$$

tetszőleges $x_i^{(k)} \in S_i^{(k)}$ mellett, és ezt az egyenlőtlenséget p_k -val beszorozva és $k = 1,2, \ldots, m$ esetére összeadva nyerjük az egyensúlypontot definiáló egyenlőtlenséget az eredeti játékra vonatkozóan.

A tétel bizonyítását Molnár-Szidarovszky (2011b) alapján végeztem, akik hangsúlyozták a bizonyítás során, hogy a többcélú problémamegoldás, elsősorban kisebb feladatok megoldására, azaz rövid fagráfok kialakítására alkalmas. Itt zérus hosszúságú fagráffal ábrázolható játékok adódnak, amelyek esetén az egyetlen pontjukban való maradás jelenti az egyetlen egyensúlypontot. Nagyobb feladatok esetén azonban a rövid fagráfok száma annyira megnövekedhet, hogy az egyensúlypontok keresése nem válik lehetségessé. Ennek megállapítása próbálgatással látható be.

1.2.3. Folytonos játékok - egy és több egyensúlypontos játékelméleti modellek

1.2.3.1. Játékok egyetlen egyensúlyponttal

Játékok egyensúlypontjainak egyértelműségét például úgy igazolhatjuk, hogy az egyensúly problémával ekvivalens valamelyik fixpont-probléma egyértelmű megoldását bizonyítjuk be. Az igazolás során alapvetően Szidarovszky logikai levezetését követem, amennyiben ettől eltérek, azt külön jelzem.

Egyváltozós fixpont probléma általában az $f(x) = x$ egyenlet megoldását jelenti. Jól ismert, ha **f** csökkenő függvénye x-nek, akkor nem lehet 1-nél több megoldás. Próbáljuk meg általánosítani ezt a monotonitási feltételt a többdimenziós esetre. Legyen tehát **f** egy vektor változós, vektor értékű függvény, $\mathbf{f} : R^n \longrightarrow R^n$. A következő példa jól mutatja, hogy a komponensenkénti szigorú monotonitás sem garantál egyértelműséget (Szidarovszky, 1978a).

Példa. Tekintsük az

$$x = - x - 2 y$$

$$y = - 2x - y$$

fixpont problémát, amely az

$$\boldsymbol{f}(x, y)= \begin{pmatrix} -x-2y \\ -2x-y \end{pmatrix}$$

leképzéssel adott. Mindkét komponens szigorúan csökkenő függvény mindkét változójában és mégis végtelen sok fixpont létezik:

$$y = -x.$$

Kimutatjuk viszont, hogy egy másik típusú monotonitás már elegendő fixpontok egyértelműségének biztosításához.

Definíció. Legyen $D \subseteq R^n$ egy konvex halmaz. Egy *f:* $D \rightarrow R^n$ függvényt m*onotonnak* mondunk, ha tetszőleges $x, y \in D$ mellett

$$(x-y)^T(f(x)-f(y)) \geq \mathbf{0}.$$

*Az **f*** függvényt *szigorúan monotonnak* mondjuk, ha tetszőleges $x, y \in D$ *és* $x \neq y$ *esetén*

$$(x-y)^T(f(x)-f(y)) > \mathbf{0}.$$

Könnyen kimutathatjuk, hogyha **- f** monoton, akkor a leképzésnek nem lehet két különböző fixpontja. Tegyük fel ezzel ellentétben, hogy **x** és **y** egyaránt fixpont. Ekkor

$$\mathbf{0} < (x - y)^T(x - y) = (x - y)^T(f(x) - f(y))$$

$$= -(x-y)^T((-f(x))-(-f(y))) \leq \mathbf{0},$$

nyilvánvaló ellentmondás.

A leképzések monotonitását könnyen ellenőrizhetjük a következő tétel alapján:

Tétel. Legyen $D \subseteq R^n$ egy konvex halmaz és az ***f***: $D \rightarrow R^n$ függvény folytonosan differenciálható. Jelölje $\boldsymbol{J(x)}$ *az* ***f*** Jacobi-mátrixát az $\boldsymbol{x}$ helyen.

a) Ha $J(\mathbf{x}) + J^T(\boldsymbol{x})$ pozitív szemidefinit minden $\mathbf{x} \in D$ mellett, akkor **f** monoton,
b) Ha $J(\mathbf{x}) + J^T(\mathbf{x})$ pozitív definit minden $\mathbf{x} \in D$ mellett, akkor **f** szigorúan monoton.

Bizonyítás.
Vezessük be rögzített $\mathbf{x}, \mathbf{y} \in D$ mellett a

$$\mathbf{g}(t) = \mathbf{f}(\mathbf{y} + t(\mathbf{x} - \mathbf{y}))$$

skalárváltozós függvényt. Nyilvánvalóan

$$g(0) = f(y) \text{ és } g(l) = f(x),$$

így

$$f(x) - f(y) = \int_0^1 J(y + \mathbf{t}(x - y))(x - y)\boldsymbol{dt}.$$

Szorozzuk be mindkét oldalt balról az $(\mathbf{x} - \mathbf{y})^T$ sorvektorral, akkor a következőt kapjuk:

$$(\mathbf{x} - \mathbf{y})^T (\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})) = \int_{\mathbf{0}}^{\mathbf{1}} (\boldsymbol{x} - y)^{\boldsymbol{T}} J(y + t(x - y))(x - y)dt$$

$$= \frac{1}{2} \int_0^1 (\boldsymbol{x} - \boldsymbol{y})T \;\{J(y + t(x - y)) + J^T(y + t(x - y))\}(x - y)\,dt$$

A legutóbbi lépésben felhasználtuk, hogy tetszőleges $u \in D\ R^n$ vektor és $n \times n$ típusú J mátrix esetében

$$\mathbf{u}^T \mathbf{J} \mathbf{u} = \mathbf{u}^T \mathbf{J}^T \mathbf{u},$$

hiszen mindkét oldal skalár és egymás transzponáltja. Ha $\mathbf{J} + \mathbf{J}^T$ pozitív szemidefinit, akkor a jobboldal nem negatív, és ha pozitív definit, akkor pozitív. A fenti eredmények közvetlen alkalmazásához a leképezések pontos ismerete szükséges, ami nem mindig lehetséges, mert optimum problémák megoldását igénylik. Éppen ezért különleges jelentőséggel bírnak olyan eredmények, amelyek közvetlenül a stratégiahalmazok és a kifizető függvények tulajdonságára épülnek (Molnár-Szidarovszky, 2011c; Mészáros, 2005).

1.2.3.2. Kétszemélyes folytonos játék – biomassza-készletgazdálkodás

Vegyünk példaként egy Tüzép telepet, amely egy Fakitermelő és Kereskedő (FK) vállalattal áll kapcsolatban tűzifa (biomassza) beszerzés, ellátás tekintetében. Az egyszerűség kedvéért csak egyfajta tűzifa, azaz egyfajta árucikk beszerzésére szorítkozom a Tüzép telep részéről. Feltételezésünk szerint a Tüzép telepre érkező vásárlói igények nagysága exponenciális eloszlást követ, tehát az időegység alatt beérkező igények sűrűségfüggvénye (Molnár-Szidarovszky, 1995):

$$g(x) = \alpha e^{-\alpha x} \quad (\alpha > 0, x > 0).$$

A példa megértéséhez a következő jelöléseket vezetjük be: a_1 legyen a Tüzépnek egységnyi áru, azaz tűzifa eladásából származó tiszta nyeresége, ha az igényt a saját raktárából elégíti ki. a_2 legyen a Tüzépnek egységnyi áru, azaz tűzifa utánrendeléssel beszerzett áru eladásából származó tiszta nyeresége ($a_2 < a_1$). A b_1 legyen a Tüzép egységnyi raktározási költsége. A b_2 legyen a fakitermelő és kereskedő (FK) egységnyi raktározási költsége.

A Tüzép és a FK stratégiája egy-egy számmal, a két raktárkészlet nagyságával jellemezhető. Ha az y a Tüzép, a z pedig a Fakitermelő és Kereskedő raktárkészleteit jelöli, akkor a Tüzép várható nyeresége:

$$f_1(y,z) = a_1 \int_0^y xg\,(x)dx + \int_y^{y+z} [a_1 y + a_2(x-y)]g(x)dx + \int_{y+z}^{\infty} (a_1 y + a_2 z)g(x)dx - b_1 y$$

képlettel oldható meg. A képlet első tagja arra utal, hogy ha az igény y –nál kisebb akkor a vevői igényt teljes egészében saját raktárkészletről tudja kielégíteni, ekkor a_1 egységnyi haszonhoz jut. A második tag az $y + z \geq x \geq y$ esetnek felel meg, ekkor a Tüzép y mennyiségű igényt saját készletből, a fennmaradó $x - y$ igény teljesítését pedig utánrendelésből elégíti ki. A képlet harmadik tagja az $x > y + z$ esetre vonatkozik, akkor a Tüzép nem tud minden jelentkező igényt kielégíteni, ezért: y egységnyit saját raktárkészletből, z egységnyit pedig utánrendelésből, az FK raktárkészletéről megrendelve elégít ki. A képlet jobb oldalának utolsó tagja a raktározási költséget jelenti. Ugyanezen összefüggés alapján mutatható ki az FK vállalat várható profitja is:

$$f_2(y,z) = a_3 \int_y^{y+z} (x-y)g\,(x)dx + \int_{y+z}^{\infty} a_3\, zg(x)dx - b_2 z$$

Ezzel tehát egy kétszemélyes folytonos játékot definiáltam. Ha feltételezzük, hogy y megengedett értékei y_1 , $y_2, \dots \dots y_{m1}$ illetve z megengedett értékei pedig z , $z_2, \dots \dots z_{m2}$, valamint kevert stratégiákat is megengedünk, akkor bimátrix játékot nyerünk (Molnár et al., 2010b):

$$A = f_1((y_i, z_j))_{i,j=1}^{m1,m2}$$

$$B = f_2((y_i, z_j))_{i,j=1}^{m1,m2}$$

1.2.3.3. Háromszemélyes folytonos játék – ásványi anyagok kitermelése

A háromszemélyes folytonos játékok megoldási optimumait Molnár S. (2010) Környezetinformatikai esettanulmányain keresztül, a bányászat és környezetvédelmi szempontok figyelembe vételével történő példán keresztül mutatom be. A példában egy többegyensúlyos (három egyensúlyi pont) megoldást követhetünk nyomon, melynek játékelméleti megoldása során a kifizető függvények maximálására törekszünk mind a három szempont vonatkozásában.

Az ásványi anyagok kitermelése nemcsak a bányászati terület közvetlen környezetében, hanem az azt körülvevő nagyobb térségben, ökológiailag egységes területen is komoly hatásokat vált, válthat ki. Általában az ásványi anyagok a felszín alatti vízszint alatt találhatók, így a bányászati tevékenység tervezéséhez megfelelő vízvédelmi stratégiai kialakítása is szükséges, annak érdekében, hogy a kitermelés megvalósítható legyen. A vízrendszerben történő változások azonban nemcsak a bányászati tevékenység közvetlen közelében, hanem az ivóvízellátó rendszer vonatkozásában is stratégiaalkotásra késztet bennünket, mivel a bányászati vízstratégia hatással lesz az ivóvíz és ipari vízellátást jelentő karsztvíz készletre is. A karsztvizek ugyanakkor fontos szerepet játszanak a termál és hőforrások megfelelő utánpótlásának biztosításában is, melyre jelentős gyógyturizmus épül. A bányászati tevékenységhez, azaz az ásványi nyersanyagok kitermeléséhez tehát figyelembe kell vennünk a bányászati tevékenység direkt vízszabályozását, az ivóvíz és egyéb vízellátás regionális stratégiai célrendszerét, valamint a hőforrások vízutánpótlásának biztosításához szükséges vízpótlás fenntartását a gyógyturizmus zavartalan vízellátása érdekében. A fentiek értelmében három különböző stratégia programot fogalmazhatunk meg a közös nagy stratégiaalkotás folyamatában:

A. A bányászati tevékenységhez kapcsolódó vízvédelem
B. Ivóvízbázis védelmi feladatok biztosítása
C. Karsztvíz-utánpótlás biztosítása a termálvíz ellátáshoz

A fenti célok matematikai megfogalmazása nem egyszerű feladat, illetve a matematikai értelmezést követő játékelméleti kifizető függvények felírása is speciális feltételrendszerek beállítása után valósulhat csak meg. A matematikai értelmezéshez vezessük be a következő jelöléseket: legyen x_i az i-dik bányából felszínre hozott vízmennyiség, illetve annak éves költsége. Az xL_i az i -dik bányában fakadó vízbetörések hozama. Legyen az xD_i az i –dik bányában vízvédelmi célból elveszett vízhozam. Legyen az xG_i az i –dik bánya esetén eltömített vízmennyiség. Legyen a $b_i(x_i)$ az x_i vízkivétel (beruházási+üzemi) költsége az i –dik bányában. Legyen $L_i(xL_i)$ az xL_i vízbetörésekből adódó gazdasági veszteség az i –dik bányában és $d_i(xD_i)$ vízvédelmi célból elveszett víz vízvédelmi költsége (beruházás+üzemi) az i –dik bányában. A $g_i(xG_i)$ tömítés költsége (beruházás+üzemi) az i –dik bányában. Legyen v_{ik} az i –dik bányából a k –adik visszatáplálási pontba szállított vízmennyiség. Az r legyen a mesterséges visszatáplálási helyek száma, legyen $a_{ik}\,(V_{ik})$ a v_{ik} vízmennyiség szállításának költsége. Legyen m a vízigényhelyek száma, legyen y_{ij} bányából vagy más vízkivételi helyről (i) a j -dik vízkivételi helyre szállított mennyiség. Legyen $s_{ij}(y_{ij})$ az y_{ij} vízmennyiség szállításának költsége. Legyen $n = n_1 + n_2$ a bányák (n_1) és más vízkivételi helyek (n_2) együttes száma. Legyen d_j a j -dik vízigényhely vízigénye. Legyen a b hévizek felszín alatti eredeti szintű utánpótlása. Legyen A_i az i -dik bányában előforduló vízbetörések száma.

A modell példában x_i , xL_i , xD_i, xG_i, v_{ik} , y_{ij} változók szerepelnek döntési változóként. Értékük a megadható feltételi összefüggések következtében nem független, így az i –dik bánya költségfüggvényét könnyedén felírhatjuk a következőképpen:

$$f_{1i} = b_i(x_i) + L_i(xL_i) + d_i xD_i + g_i xG_i + \sum_k a_{ik}\,(v)_{ik} \ .$$

A változóknak eleget kell tenniük a következő feltételeknek:

$$xL_i + xD_i = x_i$$

azaz a kivett vízhozam megegyezik a fejtésben és a vízvédelmi célból elvezetett vízhozam összegével. A bányavíz A_i összhozama a kivett x_i és az előtömített xG_i vízmennyiség összege, azaz

$$A_i = x_i + xG_i$$

Ugyanígy

$$\sum_k v_{ik} \le x_i \ ,$$

azaz nem szállíthatunk el több vizet, mint amennyi a rendelkezésünkre áll. Az első célfüggvény a bányászati cél, az összes bánya együttes költsége:

$$\sum_k f_{1i} \to min.$$

A vízellátási cél jelenti, hogy a regionális vízgazdálkodás során a vízigények kielégítése a lehető legkisebb költséggel történjék. A lehetséges földalatti vízkivételi helyeket (beleértve a bányákat is), továbbá a vízigény helyeket egy hálózat pontjainak tekintjük (Salazar et al, 2010).

Ekkor a vízellátási cél a következő módon fejezhető ki:

$$f_2 = \sum_{i=1}^{n} \sum_{j=1}^{m} S_{ij}(y_{ij}) \to min;$$

A következő feltételeknek kell még teljesülni:

$$\sum_{i=1}^{n} = d_j \qquad (1 = 1, 2,.., m);$$
$$\sum_{j=1}^{m} y_{ij} + \sum_{k=1}^{r} v_{ik} \leq x_i \qquad (i=1, 2,..., n_1).$$

A környezetvédelmi célt teljes mértékben kielégítjük akkor, ha olyan $\{x_i\}$, $\{y_{ij}\}$, $\{v_{ik}\}$ döntéseket hozunk, amelyek együttes hatásával fenntartjuk a hévizek előírt utánpótlási mennyiségét. Így a környezetvédelmi célt a következő függvénnyel jellemezhetjük:

$$f_3=b-h(...., x_{i....}y_{ij.....}, v_{ij.....}) \rightarrow min$$

A h függvénnyel határozzuk meg a rendszert elhagyó felszín alatti vízmennyiséget i döntési változók függvényében. Ha csak a környezetvédelmi célt kellene figyelembe vennünk, akkor a döntési változókat úgy választanánk meg, hogy $h = b$ teljesüljön. Mindenképpen feltesszük természetes korlátozó feltételként a modellben, hogy

$$h(..., x_i ..., y_{ij} ...., v_{ik}.....) \leq b.$$

Három lehetséges mód áll rendelkezésre a h függvény becslésére. Molnár et al.(2010c) szerint mindhárom módszer a rendszer szimulációs egy-egy modelljére épül, amely a parciális differenciálegyenlet - rendszerrel történő modellezésből áll. Ez az egyenletrendszer írja le a felszín alatti vízmozgásokat, ennek segítségével az előzőekben elemzett $\{x_i\}, \{y_{jj}\}, \{v_{ik}\}$ döntési változók függvényében, ezek rögzített értékei mellett a h függvény könnyen kiszámítható.

1. A rendszer szimulációs modelljének segítségével kiszámíthatjuk h egy – egy értékét a döntési alternatívák értékének elég sok kombinációjára. A táblázatos modell csak elvileg lenne alkalmazható a megoldásra, a gyakorlatban viszont olyan számítástechnikai nehézségeket okoz, ami miatt nem foglalkozhatunk az alternatívával.

2. Megoldás lehet, ha a regressziós egyenletet használjuk f_3-ban h becslésére, mert szimulációs modell segítségével többváltozós lineáris regressziós függvényt készítünk.

3. A h becslésre ismét regressziós egyenletet használunk, de észlelési vagy tapasztalati adatok alapján. Ez a megoldás jó lehet, ha ismert rendszerről van szó, amikor a hatékonyabb továbbfejlesztés módját akarjuk megállapitani. Induló rendszer esetében viszont az észlelési adtok nem jellemzik a teljes rendszert.

A fentiek értelmében tehát esetünkben a második változatot, azaz a regressziós egyenletet használtuk. Vegyük észre, hogy a döntési változók három nagy csoportba bonthatók. Az $\boldsymbol{u}_1 = (x_i, xL_i, xD_i, xG_i)$ változók a bányászati célra, $\boldsymbol{u}_2 = (y_{ij})$ a vízellátási célra $\boldsymbol{u}_3 = (v_{ik})$ pedig környezetvédelmi célra vonatkoznak. A három célt kifizető függvényeknek, az $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, $\boldsymbol{u}_3$, vektorokat pedig stratégia vektoroknak tekintve egy három személyes játékot definiálhatunk. Feladatunk tehát matematikailag egy háromszemélyes játékkal írható le, ahol $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, $\boldsymbol{u}_3$, a megfelelő stratégiavektorok, $\boldsymbol{u} = (\boldsymbol{u}_i)_{i=1}^{3}$ a szimultán stratégia vektor:

$$\varphi_i(\boldsymbol{u}) = \varphi_i(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3,) = c^T_{i1}\,\boldsymbol{u}_1 + \boldsymbol{c}^T_{i2} + \boldsymbol{u}_2 + \boldsymbol{c}^T_{i3} + \boldsymbol{u}_3 = \boldsymbol{c}^T_i + \boldsymbol{u}$$

a célfüggvények a stratégiavektorok.

$$\boldsymbol{A}_1\,\boldsymbol{u}_1 + \boldsymbol{A}_2\boldsymbol{u}_2 + \boldsymbol{A}_3\boldsymbol{u}_3 \geq \boldsymbol{b}$$

alakú feltételeknek tesznek eleget. Itt az együtthatók az előző modellegyütthatókból keletkező vektorok és mátrixok.

A játékelméletben a kifizető függvények maximumára szoktak törekedni, így esetünkben célszerű kifizető függvényeknek az eredeti célfüggvények (-1) –szereseit választanunk.

Jelölje S az $\boldsymbol{A}_1 \boldsymbol{u}_1 + \boldsymbol{A}_2\boldsymbol{u}_2 + \boldsymbol{A}_3\boldsymbol{u}_3 \geq \boldsymbol{b}$ feltételrendszernek eleget tevő $\boldsymbol{u} = (\boldsymbol{u}_i)_{i=1}^3$ vektorok halmazát. Ekkor S –t szimultán stratégiahalmazának kell felfognunk. Az S halmazzal és a φ_i kifizetőfüggvényekkel rendelkező játék egyensúlypontján olyan $u^* = (u_i^*) \in S$ vektort értünk, amelyre tetszőleges S –beli

$$u^{(1)} = \begin{pmatrix} u_1 \\ u_2^* \\ u_3^* \end{pmatrix} \qquad u^{(2)} = \begin{pmatrix} u_1^* \\ u_2 \\ u_3^* \end{pmatrix} \qquad u^{(3)} = \begin{pmatrix} u_1^* \\ u_2^* \\ u_3 \end{pmatrix}$$

vektorok esetén: $\varphi_k(u^{(k)}) \leq \varphi_k(u^*)$

A $\varphi_k(u^{(k)}) \leq \varphi_k(u^*)$ összefüggés, mint tudjuk, szemléletesen azt jelenti, hogy egyik célfüggvény értéke sem növelhető meg a hozzá tartozó stratégia egyoldalú megváltoztatásával, azaz az $\mathbf{u}^*_k$ stratégia vektor φ_k –nak maximumhelye, feltéve, hogy a többi $\mathbf{u}^*_i$ $(i \neq k)$ egyensúlystratégiát rögzítjük. Minthogy az $\boldsymbol{A}_1 \boldsymbol{u}_1 + \boldsymbol{A}_2\boldsymbol{u}_2 + \boldsymbol{A}_3\boldsymbol{u}_3 \geq \boldsymbol{b}$ feltételrendszer és a célfüggvények lineárisak, a Kuhn – Tucker – féle elmélet alapján **u** akkor és csak akkor egyensúlypont, ha teljesül a következő feltételrendszer:

$$\left.\begin{aligned} c_k^T + v_k^T A_k &= 0^T \\ A_1u_1 + A_2u_2 + A_3u_3 &\geq b \\ A_1u_1 + A_2u_2 + A_3u_3 &\geq b \\ v_k &= 0 \end{aligned}\right\} \qquad (k = 1, 2, 3,),$$

ahol $\mathbf{V}_k$ alkalmas vektor. A harmadik feltétel bal oldala mindenképpen nem negatív, így a probléma ekvivalens a problémával, amely a célfüggvény miatt kvadratikus programozási feladatot jelent az $\mathbf{u}_k$ döntési és a $\mathbf{v}_k$ új változókra.

$$\left.\begin{aligned} \boldsymbol{v}_k &\geq \boldsymbol{0} \\ \boldsymbol{A}_k^T\boldsymbol{v}_k &= -\boldsymbol{c}_k \end{aligned}\right\} \qquad \mathbf{(k = 1, 2, 3)}$$

$$\boldsymbol{A}_1\boldsymbol{u}_1 + \boldsymbol{A}_2\boldsymbol{u}_2 + \boldsymbol{A}_3\boldsymbol{u}_3 \geq \boldsymbol{b}$$

$$\sum_{k=1}^{3} \boldsymbol{v}_k^T \left(\boldsymbol{A}_1\boldsymbol{u}_1 + \boldsymbol{A}_2\boldsymbol{u}_2 + \boldsymbol{A}_3\boldsymbol{u}_3 - \boldsymbol{b}\right) \rightarrow \mathbf{min}.$$

Egyszerű átalakításokkal a feladat célfüggvényét a következő alakra is hozhatjuk:

Ha bevezetjük a

$$H = \begin{pmatrix} 0 & A_2 & A_3 \\ A_1 & 0 & A_3 \\ A_1 & A_2 & 0 \end{pmatrix}, c = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}, b = \begin{pmatrix} b \\ b \\ b \end{pmatrix}, u = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}, v = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}.$$

hipermátrixot és vektorokat, akkor a függvény egy egyszerűbb

$$\boldsymbol{v}^T\boldsymbol{H}\boldsymbol{u} - \boldsymbol{c}^T\boldsymbol{u} - \boldsymbol{b}^T\boldsymbol{v}$$

alakban is felírható.

A módon tehát eredeti feladatunkat egy kvadratikus programozási feladatra redukáltuk, amelynek megoldására már egyszerűbben alkalmazható módszerek állnak rendelkezésünkre.

Számításaink során három bányát vizsgáltunk meg, azaz i = 1, 2, 3. Ezekre $A_1 = 60 m^3$/ perc, $A_2 = 150\ m^3$/ perc, $A_3 = 100\ m^3$ / perc, amelyeket éves adatokból kapunk. Az éves egységnyi beruházási és üzemköltségeket az 2. táblázat adja meg (104 Ft / m^3/ perc). Feltételezzük tehát, hogy a szóban forgó függvények valamennyien lineárisak.

2. táblázat: A különböző bányák hozamának éves adatai (Ft/m^3/perc)

bi, Li, di, gi	VÍZKIVÉTEL			VÍZVÉDELEM		TÖMÖRÍTÉS	
	beruházás	üzem	veszteség	beruházás	üzem	beruházás	üzem
1. bánya	28,5	73,5	940	nem realizálható		12,0	42,0
2. bánya	32,0	116	1200	35,0	129,0	8,5	25,5
3. bánya	100,0	184	1420	45,0	147,0	22,3	34,7

Forrás: saját szerkesztés Molnár S. (2010c) alapján

A három bányával együtt összesen hat vízkivételi helyet tekintettünk, és hét vízigény helyet vizsgáltunk meg. A vízszállítás éves beruházási és üzemköltségeit az 3. táblázatban találhatjuk meg. A költségeket itt is 104 Ft / m^3 / perc -ben adjuk meg, mint korábban. Minden relációban két költséget szerepeltetünk, az első a beruházási, a második pedig az üzemelési egységköltséget jelenti, így az S_{ij} függvények is lineárisak.

Az egyes vízigényhelyek igényei rendre (m^3 / perc) 33, 14, 83, 14, 83 és 28. Két visszatáplálási helyet veszünk figyelembe, ezek költségét (az első a beruházást, a második pedig az üzemköltséget jelenti) az 4. táblázatban találhatjuk meg (Ft / m^3/ perc).

3. táblázat: Vízkivételi helyek vizsgálata (Ft / m^3 / perc)

S_{ij}	1		2		3		4		5		6		7	
1	6	3	7	8	9	7	11	8	14	8	15	7	1	7
2	9	10	5	4	5	2	8	4	10	4	10	4	14	6
3	9	11	5	2	7	3	10	6	10	6	14	8	18	9
4	261	45	257	40	255	39	262	43	262	43	264	45	268	46
5	272	45	269	43	262	40	259	30	259	30	253	37	256	40
6	264	42	258	32	261	35	258	32	258	32	262	32	265	34

Forrás: saját szerkesztés Molnár et al. (2010c) alapján

4. táblázat: Visszatáplálási helyek költségei (Ft /m³/ perc)

a_{ik}	1. VISSZATÁPLÁLÁSI HELY		2. VISSZATÁPLÁLÁSI HELY	
1.bánya	25	0	45	53
2.bánya	68	112	72	78
3.bánya	63	94	50	32

Forrás: saját szerkesztés Molnár et al. (2010c) alapján

A harmadik célfüggvényben b = 30 m^3 / perc és a regresszióval kapott h függvény alakja a következő:

$$h = 30{,}5 - 0{,}021x_1 - 0{,}012x_2 - 0{,}014x_3 - 0{,}006\sum_i y_{i4} - 0{,}0021\sum_i y_{i5} - 0{,}0042\sum_i y_{i6} + 0{,}07\sum_i v_{i1} + 0{,}14\sum_i v_{i2}.$$

A leírt modellt a fenti kvadratikus programozási feladat segítségével tudjuk megoldani. Az optimális döntési változókat az 5. és 6. táblázatok mutatják részleteiben. Az első táblázatban az optimális x_i, xL_i, xD_i, xG_i, v_{i1}, és v_{i2} értékek találhatók $i = 1,2,3$ esetén, a másodikban pedig az y_{ij} ($1 \leq i \leq 6, 1 \leq j \leq 7$) értékeket adjuk meg. Az 5. táblázatból a konkrét y_{ij} értékek alapján az is kiolvasható, hogy összesen kilenc hely között történik vízszállítás az optimális program szerint.

5. táblázat: Optimális döntési változók a vízkivételi helyeken (3 helyszínen)

i	x_i	xL_i	xD_i	xG_i	v_{i1}	v_{i2}
1	33	33	0	27	0	0
2	33	0	130	0	0	0
3	150	0	100	0	0	25,16

Forrás: saját szerkesztés Molnár et al. (2010c) alapján

6. táblázat: Döntési változók a vízkivételi helyeken (6 helyszínen)

i/j	1	2	3	4	5	6	7
1	33	0	0	0	0	0	0
2	0	0	25	14	0	83	28
3	0	14	58	0	2,84	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	11,6	0	0
6	0	0	0	0	0	0	0

Forrás: saját szerkesztés Molnár et al., (2010c) alapján

Megjegyzés: Az egymással nem egyező eredmények nem ellentmondást jelentenek, hanem a többcélú programozási feladatok esetében állandóan felmerülő elvi problémát illusztrálják, miszerint az optimális megoldás nagymértékben függ az alkalmazott módszer megválasztásától. A különböző módszerek más és más optimum pontokat jelölhetnek ki. Ez pedig a döntéshozó az elvi hozzáállásának mikéntjét tükrözi.

1.3. Kooperatív játékok elmélete

A játékelméleti megoldásokat a piaci szereplők, elsősorban a piacon jól definiálható vállalatok jól tudják alkalmazni már a jelenlegi gyakorlatban is. A játékelmélet matematika, így a megfelelő stratégiák kiválasztása számokkal alátámasztva, egzakt módon történhet. Ami jelenleg még nem jellemzi általánosan a stratégiai célok kiválasztását, az a kooperatív cél vagy stratégiai választás gyakorlata. A fenntarthatósági koncepciók egyértelműsítik, hogy termelési/fogyasztási célrendszer akkor tervezhető hosszú távon, ha az erőforrások felhasználása tervezett módon, összehangoltan folyik. Az ilyen típusú együttműködés vagy kooperáció nélkül nem valósulhat meg fenntartható növekedés sem. A kooperatív jelző, s az a tény, hogy a piaci szereplők koalíciókat alkotnak, új megközelítést ad a játékelméleti megközelítésnek is. Ez nem feltétlenül jelenti azt, hogy a játékosok érdekei azonosak, pl. a költségelosztás esetében ez egyértelmű lehet, de az együttműködés során megadott közös feltételrendszerek betartása kötelező (Hardin, 1968).

A kooperatív stratégiák során a játékosok vagy piaci szereplők természetes törekvése, hogy függetlenségük teljes vagy részleges feladásával hasznukat növeljék (Solymosi, 2009). Ily módon a játékosok egyes csoportja, vagy minden szereplő kooperál egymással és esetlegesen koalíciókat vagy koalíciót alkot. Az együttműködés természetes feltétele az, hogy az együttműködésben részt vevő játékosok vagy piaci szereplők haszna nagyobb legyen, mint azoké, akik ebben nem vesznek részt. A cél ez esetben nem az egyéni hasznok növelése lesz, hanem a kooperáció vagy együttműködés együttes hasznának a maximalizálása. Ez a törekvés a fenntarthatósági vagy a fenntartható növekedés, azaz a közgazdasági (gyenge) fenntarthatóság kritériumait teljesítik (Molnár-Kelecsényi, 2009).

A *kooperatív játék*ot a következő fogalmakkal határozzunk meg. $N = \{1, \ldots\ldots, n\}$ a *játékosok halmaza*, amelynek egy tetszőleges S részhalmazát közismert szóval *koalíció*nak nevezzük: $S \subseteq N$. Legyen S a részhalmazok, azaz a lehetséges koalíciók halmaza. Az N alaphalmazt *teljes koalíció*nak nevezzük (Szidarovszky, 1978b).

1.3.1. Konfliktus-feloldási módszerek

A konfliktus-feloldási módszerek a kooperatív játékelméleti megoldások egyik népszerű családját jelentik. Ezek közül kiemelhetjük a Nash axiomatikus megoldásrendszerét, amely axióma halmazok megadásával biztosította, hogy a megoldás mindig a *Pareto-vonalon* legyen. A Kálai-Smorodinsky megoldás pedig a konfliktushelyzet legrosszabb kimeneteli pontjának meghatározásával megadja a minimálisan elérhető, vagy a konfliktus megoldásaként megadható utolsó lehetséges pontot, vagyis a még elfogadható legrosszabb kimenetelét (Molnar-Szidarovszky, 1994).

A konfliktus-feloldási módszer bemutatását egy kétszemélyes esettel vezetem be. A példában jelölje S_1 és S_2 a játékosok stratégia halmazát φ_1 *és* φ_2 a két kifizetőfüggvényt. A lehetséges kifizetések halmaza így 2-dimenziós lesz és következőképpen írható fel:

$$H = \{\varphi_1(x, y), \varphi_2(x, y) \,|(x, y) \in S_1 \times S_2\}$$

Ebben az esetben is, mint mindig, mindkét játékos a kifizetésének maximalizálására törekszik, de természetesen kifizetése függ a másik játékos stratégiájától és általános szabály, hogy az egyik játékos kifizetésének növelésével a másik csökkeni fog (Nowak-May, 1992). A feladat tehát az, hogy olyan megoldást találjunk, amely mindkét játékos számára elfogadható megoldást jelent. Minden megoldás előtt fel kell tennünk azt, ha nem jön létre megállapodás, akkor mindkét játékos alacsonyabb kifizetést, vagy büntetést kap.

Általános jelölések:

$$\boldsymbol{f}_* = (f_{1*}, f_{2*})$$

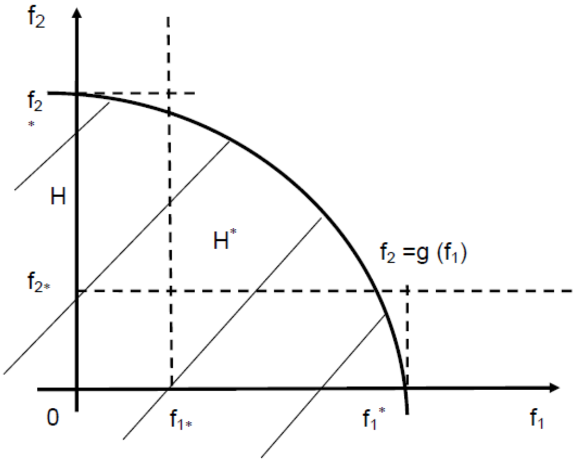
ez lesz a kifizetési vektorunk, amelynél feltételezzük, hogy létezik olyan $(f_1, f_2) \in H$, amely esetén $f_1 > f_{1*}$ és $f_2 > f_{2*}$. A konfliktus matematikailag $(H, \boldsymbol{f}_*)$ párral definiált. A 2. ábrán ezt a párt definiáltuk. Feltesszük továbbá, hogy a H halmaz zárt, konvex és korlátos, azaz

$$(f_1, f_2) \in H \text{ és } \bar{f_1} \leq f_1, \bar{f_2} \leq f_2$$

esetén szükséges $(\bar{f_1}, \bar{f_2}) \in H$, valamint mindkét koordinátájában korlátos, azaz

$$sup\ \{f_i | (f_1, f_2) \in H\} < \infty$$

$$i = 1{,}2 \quad \text{esetén.}$$



2.ábra: A konfliktushelyzet ábrázolása
Forrás: Molnár-Szidarovszky, 2011c alapján

Feltesszük továbbá, hogy a H határvonala egy $f_2 = g(f_x)$ függvény gráfja, amely szigorúan csökken f_1-ben és konkáv. A $g-$ függvény gráfját Pareto-vonalnak szokták nevezni, tehát a fenntarthatósághoz kapcsolódó optimum kritériumok kielégítésének feltétel itt teljesülhet. A játék és megoldásfeltételek között figyelembe kell venni, hogy racionális játékos nem fogad el olyan megállapodást, amely rosszabb kifizetést jelentene, mint a megállapodás nélküli kifizetés.

A lehetséges kifizetési halmazt így leszűkíthetjük a következőképpen:

$$H^* = \{f_1, f_2 | f_1 \geq f_{1*}, f_2 \geq f_{2*}, (f_1, f_2 \in H\}$$

1.3.2. Oligopol játékok modellje

A oligopol játékelméleti megoldások a legnépszerűbbek a gazdasági döntési folyamatok modellezésre. Az oligopol játékelméleti megoldások egyaránt alkalmazhatók kooperatív és nem kooperatív stratégia esetén is, én azonban a dolgozat vonatkozásában, a fenntarthatósági szempontrendszer kiemelt jelentőségére való tekintettel, a hasznosságfüggvény fenntartható maximalizálásának értelmezéséhez, a kooperatív modellt kívánom bemutatni.

A közgazdaság-tudománynak a közönséges értelemben vett hasznosságfüggvény alkalmazása esetén két problémája merülhetett fel. Az első probléma az volt, hogy nemcsak egy hasznosságfüggvény jellemez egy fogyasztót, hanem végtelen sok, és ezek ekvivalensek egymással. A második nehézség a bizonytalanság melletti választásnál vetődött föl. Az oligopol játékoknál alkalmazott megoldások a feltételek rögzítésével és a kooperatív stratégiák alkalmazásával, a legnagyobb eséllyel maximalizálhatják az egyes játékosok hasznosságfüggvényét (Simonovics, 2003; Ichiishi, 1983).

A műszaki (környezetvédelmi) feladatok széles skálájában találkozhatunk olyan konkrét problémákkal, amelyek matematikai modellje oligopol játékokra redukálható. A sokszereplős játékelmélet a legkülönbözőbb problémák elemzésére használható (Szilágyi, 2005). A következő példában úgy próbálom bemutatni az oligopol kooperatív stratégiára épülő optimumkeresés folyamatát, hogy egy termékre koncentrálunk a piaci folyamatban (ez lehetne például zöldáram az energiapiaci gyakorlatban), és figyelembe vesszük azokat a piaci játékosokat, piaci csoportokat (termelő, szállító, szabályozó, kivitelező stb.), amelyek hatással vannak a termék árának változására, az egyensúlyi pont kialakulására.

A kooperatív játékok esetében a függetlenség feladása a játékosok esetében a hasznok növelését kell, hogy eredményezze (Simonovics, 2003).

Általános meghatározások a kooperatív játékok esetében:

φ_k- kifizető-függvény

S_k- stratégiahalmazok , $(x_1^*, \ldots . x_n^*)$ – stratégai tényezők

n – valamilyen pozitív egész szám

k - S_k valamilyen halmaza

feltétel $k = 1,2,3 \ldots, n$ esetén $x_k^* \in S_k$, és tetszőleges $x_k \in S_k$mellett

$$\varphi_k(x_1^*, \ldots, x_n^*) \geq \varphi_k(x_1^*, \ldots., x_k^*, \ldots., x_n^*).$$

Az egyenlőtlenség szóban úgy fogalmazható meg, hogy a k -dik játékos $(1 \leq k \leq N)$ számára az egyensúlystratégia maximális stratégia annak feltételezésével, hogy a többi játékos a megfelelő egyensúlystratégiát válassza. A kooperatív játék koalíciót hoz létre, ahol a koalíció nyereséget mindig tud biztosítani magának a koalíción kívüli szereplőkkel szemben.

Kooperatív játék maximuma, a koalíció haszna:

tegyük fel, hogy a játékosok egy M csoportja a k elemű

$$M = \{i_1, i_2, \ldots, i_k\}$$

koalíciót hozza létre. Tekintsük ezután az

$$X = x_{j=1}^k S_{ij} \text{ és } Y = x_{l \neq i_j} S_l$$

stratégia halmazokkal és a

$$\tilde{\varphi}_1(x) = \sum_{j=1}^k \varphi_{i_j}(x), \qquad \tilde{\varphi}_2(x) = -\tilde{\varphi}_1(x)$$

kifizető-függvényekkel adott kétszemélyes játékot. Nyilvánvaló, hogy a $\tilde{\varphi}_1$ függvény „maximin” értéke, azaz

$$v(M) = \ max_{x_{i_j}} \ min_{x_{l_{(l \neq ij)}}} \tilde{\varphi}_1(x) \quad \text{és} \ v(\emptyset) = 0$$

mennyiség – amiben létezik - csak az M halmaztól függ. A koalíció feltételezi, hogy a koalícióba nem tartozó játékosok szeretnék a koalíció hasznát minimalizálni.

Példa az oligopol problémakezelésre:

Az egyszerűség kedvéért tegyük fel, hogy $M = 1, i_1 =. \ldots = i_n = 1,$ azaz egyetlen terméket vizsgálunk, és minden csoport egyetlen tagból áll. Az ismert maximin függvényre hivatkozva, f

differenciálható a $[0, \xi]$ intervallumba. Legyen $I = \{i_1, \dots . i_r\} \subset \{1, \dots, N\}$ egy koalíció (Molnár – Szidarovszky, 1994).

Ekkor $v(I)$ a következő képlettel írható fel.

$$v(I) = max_{x_i} min_{x_j} \sum_{i \notin I} \varphi_i(\boldsymbol{x}) \quad \text{ahol max} \rightarrow i \in I \text{ és min} \rightarrow j \notin I \text{ -nek.}$$

A $v(I)$ érték kiszámításához először a

$$\min_{x_j} \varphi_i(\boldsymbol{x}) = \psi_I(x_{i_1} \dots ., x_{i_r}) \quad \text{ahol } j \notin I$$

mennyiséget kell meghatároznunk.

Két esetet különböztethetünk meg a problémamegoldás során:

1) esetben ha $\sum_{j \notin I} L_j \geq \xi$, akkor nyilvánvalóan

$$\psi_1(x_{i1} \dots ., x_{ir}) = -\sum_{i \in I} K_i(x_i)$$

2) esetben ha $\sum_{j \notin I} L_j < \xi$, akkor nyilvánvalóan

$$\psi_1(x_{i1} \dots ., x_{ir}) = \left(\sum_{i \in I} x_i\right) f\left(\sum_{j \notin I} L_j + \sum_{i \in I} x_i\right) - \sum_{i \in I} K_i(x_i)$$

Az első esetben a ψ_1 függvény maximumhelye a K_i függvények növekedése alapján $x_{i1} =, \dots = x_{ir} = 0$, így ekkor a

$$v(I) = -\sum_{i \in I} K_i(0)$$

A második esetben vezessük be a $L_I = \sum_{i \in I} L_i$ és $\tilde{L}_I = \sum_{j \notin I} L_j$ jelölést.

Ekkor a fenti képlet szerint:

$$\psi_1(x_{i1} \dots ., x_{ir}) = s_I f(\tilde{L}_I + s_I) - \sum_{i \in I} K_i(x_i) \text{ ahol } s_1 = \sum_{i \in I} x_i$$

A fenti egyenleteket programozási feladattal oldhatjuk meg. A numerikus megoldáshoz a dinamikus programozás módszerét alkalmazhatjuk (Molnár–Szidarovszky, 1994; Simonovics, 2003).

1.3.3. Egyenlő kompromisszumok módszere

A környezeti problémák megoldásának megkeresését, amit azt már a fejezet bevezetőjében is említettem, többféleképpen is megvalósíthatjuk, az optimum pont, kimeneti pont vagy megoldás így a Pareto-vonalon több helyen is kialakulhat. Az alkalmazásra kerülő módszer függ az optimalizálást végző személy attitűdjétől, meggyőződésétől, intuíciójától, a feldolgozásra kerülő probléma jellegétől (Axelrod, 1984).

Az alkalmazható konfliktus megoldási módszerek közül jelen esetben a egyenlő kompromisszumok módszerét választottam a játékelméleti megoldás bemutatására, mely megoldás jellemzője, hogy két szereplő vagy játékos esetén a módszer feltételezi, hogy a két játékos egyforma sebességgel, folyamatosan csökkenti igényét egészen addig, amíg lehetséges megoldás adódik. A módszer jellemzője, hogy rendszerint egy megoldáspont van, így annak megadásával a legoptimálisabb feltételrendszer körvonalazható a játékosok számára (Forgó et al., 2005). Tehát az első megoldást (ami vélhetően az legjobb mindkettőjüknek) fogadják el a konfliktus megoldásaként vagy kimeneti pontként. Ha mindkét játékos a lehetséges legnagyobb kifizetést szeretné elérni, akkor az (f_1^*, f_2^*) pont a közös kívánság, ami nem lehetséges.

A probléma felírása tehát a folyamatosan és együtt csökkentett igények megfogalmazása esetén, ha (f_1, f_2) a megoldás, akkor a két játékos rendre $f_1^* - f_1$ $és$ $f_2^* - f_2$ engedményt adott, így egyenlő engedmények esetén:

$$f_1^* - f_1 = f_2^* - g(f_1)$$

Az f_1-re átírva az egyenletet

$$f_1 - g(f_1) + (f_2^* - f_1^*) = 0$$

ebben az alakban látjuk, hogy a baloldal szigorúan növekszik, $f_1 = f_{1*}$ esetén $f_{1*} - f_2^* + (f_2^* - f_1^*) < 0 és$ $f_1 = f_1^*$ esetén pedig $f_1^* - f_{2*} + (f_2^* - f_1^*) > 0$, így pontosan egy megoldása létezik a problémának (Molnár-Szidarovszky, 1995).

Hipotézisem az, hogy az előbbi fejezetekben bemutatott nem-kooperatív és kooperatív játékelméleti megoldások alkalmasak a fenntarthatósági kritériumok matematikai megfogalmazására, mivel olyan egyensúlypontok meghatározását teszik lehetővé a gazdasági, termelési, stratégiaalkotási és tervezési folyamatok során, melyek a hosszú távú fenntartható erőforrás-felhasználást, valamint káros gazdaságfejlesztési folyamatok elkerülését alapozzák meg egyértelműen.

A játékelméleti megközelítés alkalmazása során a Rubik kocka „*Layer by layer módszerhez”*, azaz sorról sorra történő kirakásához kapcsolódunk, melynek egyedi sajátossága, hogy az egymásra épülő logikai soron jól tudjuk modellezni a projektfejlesztés folyamatát, illetve a fejlesztés során egymásra ható tényezők egymás közötti viszonyainak alakulását. A hipotézis szerint, az egymás mellett elforgatott kockák, azaz az egymásra befolyást gyakorló projekt tulajdonságok kapcsolati rendszerét matematikailag is le tudjuk írni, így azok (pl. Nash-féle) egyensúlyi pontja, tehát játékelméleti modellekkel (véges játék, zérusösszegű játékok, oligopol játékok, egyenlő kompromisszumok módszere) determinánsan meghatározhatók.

1.4. Rubik kocka (3x3x3) kirakási módszerek és matematikai megközelítések

A Rubik kockát 1974-ben fedezte fel egy magyar professzor, név szerint Rubik Ernő tervező mérnök. A kezdetben bűvös kockaként ismert magyar játék először a közép-európai országokban volt népszerű, majd fokozatosan meghódította az egész világot, az USA-tól Kínáig. A kocka tervezésének elsődleges célja az volt, hogy egy olyan építészeti demonstrációs eszköz készüljön a hallgatók számára, mely révén a különböző térbeli dimenziók jól értelmezhetőkké válnak az építészhallgatók számára. A 3x3x3-as kocka révén néhány matematikai és logikai összefüggés is jól magyarázható, melyek különösen hasznosak a térbeli gondolkodás elsajátításához. A nemzetközi érdeklődés 1980-tól indult meg a kocka iránt, mikor világossá vált, hogy a Rubik Kocka nemcsak egy szimpla játék, hanem a kocka maga egy különleges rendszer is. A hatszínű kocka minden oldala 3 sorból és 3 oszlopból áll, melyek különleges jelentéssel bírnak. A fő oldalak közötti kapcsolatot a kiskockák reprezentálják és viszik tovább a rendszerben. A kocka alrendszerein keresztül, bizonyos tulajdonságokat behelyettesítve a kockákba, az egyes tényezők kapcsolati rendszere világosan követhető a kocka forgatása során. Mind a helyes, mind a helytelen illesztések, kapcsolati pályák azonosíthatók (Rubik's Revenge, 2011).

Mikor Rubik Ernő megalkotta a 3x3x3-as kockát, egy hónap intenzív „munkára" volt szüksége ahhoz, hogy megtalálja a megfelelő megoldást a kocka kirakásához. A hosszú idő alatt több olyan gondolat is megfogalmazódott benne, mely a későbbiekben a kockát kiemelte a mindennapi játékok sorából, és a Világ egyik legnépszerűbb logikai szerkezetévé tette. Rubik szerint a „kocka" jó példája a szabályos és a természetes szépség egységének. A kocka színezése egyedüli és átgondolt tervezési folyamat eredménye, mely révén, ha bárki megpillantja, első gondolatként a tárgy szépsége ragadja magával (Rubik, 1981). A kocka ugyanakkor tökéletes rendszere az emberi elme tesztelésének, a tudományos megfelelés és fegyelem megfeleltetésének. Rubik Ernő szerint a kocka jelentheti a valóságnak és a szépségnek az egységét is, mely két fogalom ugyanazt is takarhatja egy időben (SZTNH, 2013).

A hármas szám misztikus jelentése végigkísérte a kocka sikertörténetét. Sokak szerint a kocka olyan kapcsolatokat szimbolizál, amelyek az embert a természettel, a természetes léttel kötik össze. Jelentős 3-as szimbólumoknak nevezte Rubik Ernő az „anya-gyerek-apa" kapcsolatrendszert, az „ég-föld-pokol" viszonylatát, az alkotás-megóvás-pusztítás, vagy a „születés-élet-halál" folyamatait (Rubik et al., 1987).

A kocka maga az élet imitációja, vagy még inkább az élet javítására való törekvés. A kocka kirakásának problémája nagyon közel áll az élet problémáinak megoldásához, talán azt mondhatjuk, hogy az egész életünk egy kirakós játék. Ha éhes vagy, találnod kell valamit enni, viszont a mindennapi problémák ettől kicsit bonyolultabbak, nem ennyire egyértelműek. Míg nem ettél, addig csak egy problémád van, miután jóllaktál, utána kezdődik a többi. A kocka kirakásának problémája alapvetően tőled függ. Ki tudod rakni, meg tudod oldani önállóan. Viszont megtalálni a boldogságot az életben már nem ilyen könnyű, ez már nem megy egyedül. Ez a legnagyobb különbség a Kocka és az Élet között (Goudey, 2003). A fenti gondolatmenet azonos a dolgozat első részében megfogalmazott *fenntartható gazdasági érték (FGÉ)* definiálta fenntarthatósági koncepcióval, mely képes – a lokális információkat is felhasználva – integráltan bemutatni a természeti tőke mellett a társadalmi és a technikai tőke-elemek változását is, melyet a klasszikus gazdasági értékmérők csak igen korlátozottan valósítanak meg.

A világ legnagyobb fejtörését jelentő logikai játék esetében már az 1980-as évektől foglalkoztatja a tudományos kutatókat, hogy az összesen 43 252 003 274 489 856 000-féle kezdő pozícióból hogyan lehet megtalálni az "Isteni számot", azaz legfeljebb hány lépés kell a kocka kirakásához. Az eredményeket közzétevő kutató team a Google rendszer számítási teljesítményét és jópár matematikai csavart felhasználva végigellenőrizte az összes lehetséges pozíciót, ami 43 kvintrillió lehetséges állapotot vagy pozíciót jelenthet.

1.4.1. Rubik kocka megoldási módszerek és algoritmusok értelmezése

A tudományos kutatók alapvetően úgy vélték, hogy maximum 18 lépés szükséges a kocka kirakásához, azonban Michael Reid matematikus felírt egy olyan matematikai formulát, mely révén egyértelműnek látszott, hogy 20 lépésnél kisebb számú forgatással nem lehet kirakni a kockát bármely állapotából (Korf, 1997). Ez azt jelenti, hogy elméleti számítások szerint csak húsz vagy annál több forgatással rakható ki a kocka, tehát az Isteni szám is legalább húsz kell hogy legyen. 2008-ban az egyik amerikai kutató, Tomas Rokicki egy csoportelméleti módszerrel kalkulálva bizonyította, hogy 22 forgatásból a kocka bármely elrendezéséből kirakható, tehát gyakorlati példával is bizonyítható a tény, az „Isteni szám" 20 körül van. Azonban már ekkor is tudta mindenki, hogy a 22 -es szám nem lehet a legkisebb szám, a kalkulációkból ez is egyértelmű volt mindenki számára (Rokicki, 2008; Gregory, 2007).

A Rokickiék a csoportelméletből származtatott technikával először felosztották az összes lehetséges kezdő konfigurációt. Ez összesen 2,2 milliárd csoportot jelentett, melyek mindegyike 19,5 milliárd elrendezést foglalt magába. A csoportosítás attól függött, hogyan reagálnak a konfigurációk a kocka tekergetésének 10 lehetséges mozdulatára. A projekten dolgozó matematikusoknak a kocka különböző szimmetriáit kihasználva, sikerült a csoportok számát 56 millióra csökkenteniük. A csoportosítás vagy a lehetőségek számának csökkentése nagyon egyszerű metodika alapján történt, ugyanis ha egy összekevert kockát egyszerűen az oldalára, vagy fejjel lefelé fordítunk, azzal nem lesz nehezebb a kirakása, tehát ezeket az egyenértékű pozíciókat máris el lehetett vetni. Ezektől az egyszerűsítésektől eltekintve még rengeteg induló konfiguráció maradt, ezért a folyamat felgyorsítására egy algoritmus kidolgozására is szükség volt (Rokicki, 2010).

Az új algoritmusnak kiemelt jelentősége volt, mivel a korábbi módszerekkel másodpercenként csak 4000 kockát tudtak végig próbálni. A korábbi algoritmus megvizsgált egy induló mozdulatsort, majd meghatározta, hogy az eredményként kapott pozíció közelebb van-e a megoldáshoz. Ha ez nem sikerült, akkor az algoritmus elvetette ezeket a lépéseket és újra indult.

„Ez a metódus olyan, mintha egy barátunkat meglátogatnánk egy számunkra ismeretlen városban, megkapjuk tőle az útirányt, hogy mikor forduljunk jobbra vagy balra, viszont azt nem árulta el, hogy mi is valójában a kiindulási pontunk. Ha egy véletlenszerű pontról kiindulva követjük az iránymutatást, igen csekély esélyünk lesz eljutni a célállomáshoz, ha azonban sikerül összeilleszteni a megfelelő kiinduló ponttal, akkor biztosan odaérünk!" - mondta erről Joyner (1996a).

Rocicki (2010) felismerte, hogy ezek a zsákutcába torkolló lépések valójában más kiindulási pozíciók megoldásai, ami elvezette egy bizonyos algoritmus kialakításához, mellyel egy másodperc alatt egymilliárd kockát tudott kipróbálni. Az új algoritmus tehát rendkívüli sebességgel párosítja a mozdulatokat a megfelelő kiinduló ponttal, így egy 19,5 milliárdos sorozatot 20 másodperc alatt meg tud oldani, ami rendkívül nagy sebességnek tűnik, de még így is 35 évig tartott volna egy hagyományos számítógép számára a teljes feladat megoldása. Annak érdekében, hogy az időt rövidítsék, egy különlegesen hatékony megoldást kerestek. Nagy szerencse volt a problémamegoldás során, hogy John Dethridge, a Google egyik mérnöke is figyelemmel kísérte a team munkáját, és felajánlotta informatikai rendszerének szabad kapacitásait a kutatáshoz. A számítógépes birodalom szabad számítási kapacitásának felhasználásával, néhány hét alatt megoldotta a problémát.

A 15 éves kitartó kutatás eredménye tehát megerősítette azt a matematikusok által már régóta vélelmezett feltevést, mely szerint a 3x3x3-as Rubik kocka bármely állapotból történő kirakásához nem szükséges 20 –nál több fogatást alkalmazni. Sok kutató egyetért abban, az ilyen kutatások példázzák, hogyan használható a tiszta matematika a nagy számítási kapacitást igénylő problémák leegyszerűsítésére (Rokicki, 2010).

Forgatási matematika

A 3×3×3-as kockára vonatkozó megoldási gondolatmenet a matematikai leírás megkönnyítésére dolgozták ki. Első lépésként észre kell vennünk, hogy ha elméletben rögzítjük a középső, összetartó elemet, akkor a forgatások révén csak a sarkok és az oldalélek változtatják a pozíciójukat, a középső elem nem mozdul el, csak forog önmaga körül. Ez után tegyük föl, hogy a kocka minden állapotát egyenként megszámozzuk 1 től 43 252 003 274 489 856 000-ig. A számokból halmazt alkotunk, amely a kocka lehetséges pozícióinak halmazát jelöli majd (Davis, 2006).

Ha a Singmasteri (1981) alapelveket követve a forgatásokat annak az oldalnak a kezdőbetűjével jelöljük, amelyik oldalt elforgatjuk, a lenti listát kaphatjuk. Forgatási feltételként adhatjuk meg, hogy a forgatás csak a kocka középpontjából kifelé, az óramutató járásának megfelelően történhet. Ezt balkéz szabálynak is nevezzük, mely állapotra jellemző, hogy mindössze hat különböző forgatás lehetséges.

a – az alsó lapot forgatjuk el

f – a felső lapot forgatjuk el

e – a elülső lapot forgatjuk el

h – a hátulsó lapot forgatjuk el

b – a bal oldali lapot forgatjuk el

j – a jobb oldali lapot forgatjuk el

A kocka minden forgatását (nem csak ez a hat) *transzformációnak nevezzük*. Az egyes forgatásokat függvényekként képzelhetjük el, amelyek a kockák állapothalmazán vannak értelmezve. Halmazokat képeznek, a hozzárendelés szabálya pedig az, hogy az adott függvény értéke a megfelelő forgatás végrehajtásával kapott új kockapozíció sorszáma. Egyszerűen felírva: *f*(64 523) = 578 526 687 (Joyner, 1996b).

Ha a forgatások egymásutánját a megfelelő betűk egymásutánjaként jelölhetjük, és a szorzás művelet analógiájára használjuk, akkor könnyen leírhatjuk az egyes forgatási transzformációkat. Az a forgatási kombinációt tehát, amikor a kockán kétszer bal oldali lapot forgatjuk el, ezt követően a felső lapot, majd a hátulsó lapot, a következőképpen írható le: bbfh vagyis b^2fh. Azt a forgatási sort, mikor a kockát csak körbeforgatjuk, ezzel az eredeti helyére állítjuk vissza, 1 –gyel jelölhetjük. Leírva: $bbbb = b{\cdot}b{\cdot}b{\cdot}b = b^4 = 1$, Természetesen az 1-gyel jelölt körbeforgatás is egy transzformáció, de az a forgatás minden pozíciót helybenhagy. Ebben az összefüggésben 1(1)=1, 1(2)=2, … , 1(43 252 003 274 489 856 000) = 43 252 003 274 489 856 000. Ezek alapján már értelmezhetjük az óramutatóval szemben történő forgatást is, amely megfelel három darab óramutató járásával megegyező forgatás egymásutánjának. Tehát például $fff = f^3$ egy ilyen forgatás, amit az előzőek értelmében 1/f-nak vagy f^{-1}-nak is jelölhetünk, hiszen az $1/f \cdot f = 1$ képlet azt írja le, hogy a kocka felső lapjának a középpontból kifelé mutató tengely körüli 90°-os óramutató járásával ellentétes irányú két forgatás, majd azzal megegyezően történő visszaforgatása a kocka elrendezését nem változtatja meg. Egy tranzakció leírása során azt is mondhatjuk, hogy $a^0 = 1$, $f^0 = 1$, $e^0 = 1$, $h^0 = 1$, $b^0 = 1$, $j^0 = 1$ vagyis 0-szor elvégezve a valamelyik forgatási műveletet nem változik meg a kocka (Singmaster, 1981).

A kocka kirakásának halmazelméleti leírása is rettentően izgalmas kutatási terület. Ha az összes lehetséges forgatásból egy halmazt képezünk, amit A-val jelölünk és ezen a halmazon a szorzással jelölt (·) egymás után elvégzés műveletét is értelmezzük (ami másként a forgatási függvények kompozíciója), akkor egy konkrét csoportot kapunk. Ennek jele lehet pl: (A, ·).

Megállapítható ugyanakkor, hogy ezen halmaz véges elemszámú (azaz véges sok különböző forgatás képzelhető el), tehát a csoport véges elemszámú, ugyanis végtelen sok különböző forgatás végtelen sokféleképpen tudná a kockát elrendezni. Az viszont korlátozó feltétel, hogy a 9x6 lapocska mindegyike legfeljebb 6 színt vehet fel, tehát a kocka biztosan kevesebb állapottal rendelkezik, mint 36^6, vagyis ez alapján beláthatjuk, hogy csak véges sok különböző forgatás képzelhető el (Davis, 2006).

Az (A, ·) csoport neutrális eleme az 1 lesz Ugyanakkor azt is észrevehetjük, hogy minden elem az a, az *f*, az *e*, *a h*, *a b* és/vagy a *j* valamilyen egymás utáni végrehajtásából áll. Ilyenkor esetben azt mondjuk, hogy az *{a, f, e, h, b, j}* halmaz generálja az (A, ·) csoportot. A kocka rendezése, azaz minden szín helyére forgatása matematikai alakban a következőképpen néz ki, ha a forgatások x sorozatával összekeverjük. Például: x(1) = 456 358 966 568, ahol $x = bfj^3f^2\ldots aef^2hj$. A megoldáshoz x ismeretének hiányában olyan y forgatást kell találni, amely rendezi a kockát, azaz y(456 358 966 568) = 1. Ebből látható, hogy $x \cdot y = 1$, vagyis $y = x^{-1}$, tehát a cél egy x-et invertáló transzformáció megtalálása. Minden forgatásra igaz, hogy a forgatás inverze megkapható a forgatás fordított irányú végrehajtásával. Például: f inverze $f^{-1} = f^3$, mivel $f \cdot f^{-1} = f \cdot f^3 = f^4 = 1$. Bonyolultabb forgatásoknál az egyes elemeket invertálnunk kell, és fordított sorrendbe hajtjuk végre az átírást. Például: $(afj^3f^2\ldots bef^2hj)^{-1} = j^{-1}h^{-1}f^{-2}e^{-1}b^{-1}\ldots f^{-2}j^{-3}f^{-1}a^{-1}$, ahol $f^{-2} = f^2$ és $j^{-3} = j$, mint az egyszerű szorzással ellenőrizhető (Frey, 1982).

Az alapszabály az, hogy kocka rendezése során alapvetően arra törekszünk, hogy átmozgassunk bizonyos kiskockákat máshová, vagy maradjon a helyén, de kerüljön más pozícióba, például egy sarokelem forduljon el 120°-kal, vagy egy él-elem forduljon át a színével, miközben minden más változatlan maradjon. Természetesen mindenki számára világos, hogy nem minden elképzelt mozgatás valósítható meg egyértelműen. Ha valamelyik sarokelemet szeretnénk elforgatni, akkor tudnunk kell, hogy azt csak egy másikkal együtt tudjuk mozgatni, önállóan nem. Ugyanez vonatkozik a forgásirányra is, mert nem mindig azonos körüljárási irányban forognak a sarokelemek. Vannak olyan forgatások, amelyek csak éleket mozgatnak, és olyanok is, amelyek csak sarkokat. Ez annak a következménye, hogy él-elem nem kerülhet csúcselem helyére illetve fordítva is igaz. Érdekes tény, hogy a 3×3×3-as kocka csúcselemeinek mozgásai megegyeznek egy 2×2×2-es kocka csúcselemeinek mozgásaival, márpedig az utóbbinál nincsenek él-elemek. Ezek leírását is függvényalakú törvényszerűségeken keresztül rögzíthetjük a különböző kirakási módszerekhez (Snap, 2012; Slocum et al.,1981).

1.4.2. 3x3x3-as Rubik kocka kirakási módszerek elemzése

A 3x3x3-as kocka kirakásához sok különböző kockakirakási módszert fedeztek fel egymástól függetlenül az elmúlt évtizedekben, ezek közül a legnépszerűbb módszer David Singmaster által kifejlesztett layer by layer módszer, amelyet 1981-ben a Notes on Rubik's „Magic Cube" című könyvben publikált.

„Layer by layer" módszer:

A módszer lényege, hogy a kocka kirakása sorról sorra történik, először a kocka teteje a felső sorral kerül kirakásra, majd a kocka középső sora, végül az alsó szemközti oldal és harmadik sor. Hatékony gyakorlással ez a módszer 1 perc alatti kockakirakást eredményezhet. Szinte mindenki ezt a módszert tanulja meg először. Fontos a kirakási módszer esetében, hogy itt nincsenek fix algoritmusok, tehát lehet, hogy két ember, akik mindketten Layer by layer módszerrel rakják ki a kockát, mindketten teljesen más algoritmusokat használnak (Hardwick, 2014)!

„Corner First - sarkok először":

Egy másik általános megoldással, melynek neve a sarkokat először „corner first" módszer, révén a kirakás gyorsasága jóval egy perc alá csökkenhet. A kirakás gyorsaságát természetesen a szükséges forgatások száma határozza meg. A „sarkok először” metódus az alapja az egyik leggyorsabb Gilles Roux's metódusnak is. Lényege az, hogy első lépésként az összes sarkot a helyére teszi és beállítja helyes irányba. Majd ezek után az összes középső sort szabadon lehet mozgatni úgy, hogy a sarkokat nem rontjuk el. Ezzel a módszerrel sokkal nagyobb szabadságunk van a kockán, mint a layer by layer metódusná (Doig, 2000)l. A közepek forgatásával pillanatok alatt be lehet állítani az éleket. A kirakóversenyeken nagyon népszerű, mivel a legkevesebb forgatást jelenti általában. Hasonlóan hatékony az *„Edges first method” - „Éleket először módszer”*. Ez az előző módszer fordítottja, itt először az éleket, majd a sarkokat állítjuk be a helyükre. Ezt a módszert használja szinte mindenki a világon a „vakon”, azaz a bekötött szemmel történő kirakáshoz. A módszer esetében különlegesen előnyös, hogy elég mindössze egy algoritmust ismerni, ha azt a kirakó tökéletesen megtanulja, akkor ezzel a módszerrel néhány perc alatt ki is tudja rakni a kockát (Ortega, 2013). A módszer nem nehéz, azonban nem alkalmas a projektfejlesztés szimulációjára, mert a kirakási logika szigorúan automatizált (1. melléklet).

1982-ben Singmaster és Alexander Frey hipotézise szerint, az ideális algoritmussal a kocka 20 forgatással kirakható, viszont ezt az algoritmust nem tudták még leírni (Joyner, 2002a). A minimális forgatási szám bizonyítására először 2007-ben került sor, mikor egy szuper számítógép segítségével Daniel Kunkle és Gene Cooperman (2007) bizonyította, hogy a minimális forgatás 26 db vagy annál kevesebb. 2010-ben Tomas Rokicki és társai bizonyították, hogy az Isteni szám ”God’s number” azaz a minimális forgatási szám 20 forgatás. Ez azonban egy optimális állapottól függ, ahonnan a 20 forgatás eredményre vezethet. Általánosan leírva az állapotot $n \times n \times n$, n=3 Rubik kocka kirakható optimálisan $\Theta(n^2 / \log(n))$ mozdulattal (Rokicki et al., 2010).

Fridrich módszer:

Nagyon általánosan használt kirakási módszer a „kockások” körében. A módszert Jessica Fridrick fejlesztette ki, nagyon hasonló a layer by layer módszerhez, de nagyszámú algoritmust használ a megoldáshoz. A módszerrel, sok gyakorlás mellett átlag 17 másodperc alatt lehet kirakni a kockát, ezért a világ legtöbb „speedcubere” ezt a rendszert használja. Lényege, hogy egy szimpla első kétoldalt, F2L-t (first two layers) csinál, majd egy utolsó oldali orinetációt, OLL-t (orienting the last layer) és végül egy végső permutációt, PLL-t (permuting the last layer) használ. A Fridrich’s megoldás 120 algoritmus megtanulását igényli, a Kocka kirakása 55 mozdulattal végezhető el (Joyner, 2002b).

A kockarajongók esetében rendkívüli jelentőséggel bír a kockaforgatások jelölése. A jelölést Singmaster fejlesztette ki a 3x3x3-as kockára a nyolcvanas évek elején, ezért ezeket Singmaster jelölésnek nevezzük. A jelölési módszer relatív jellege lehetővé teszi, hogy le tudjuk írni kirakásokat úgy, hogy nem kell figyelembe vennünk a színeket vagy a meghatároznunk a felső és alsó oldalt (Kirakások, 2013; Singmaster, 1981):

- *F* (Front): a kirakóval szembenéző „front” oldal
- *B* (Back): a „front” oldallal szembenéző oldal, a kocka hátulja
- *U* (Up): a „front” oldal fölött elhelyezkedő oldal, a „top” oldal
- *D* (Down): a „top” vagy felső oldallal szembenéző oldal, a kocka alja
- *L* (Left): a „front” vagy szemközti oldal bal oldalára eső oldal
- *R* (Right): a „front” vagy szemközti oldal jobb oldalára eső oldal
- *f* (Front two layers): a kirakóval szembenéző „front” oldal középső sora
- *b* (Back two layers): a „front” oldallal szembenéző oldal, a kocka hátuljának középső sora
- *u* (Up two layers): a „front” oldal fölött elhelyezkedő oldal, a „top” oldal középső sora
- *d* (Down two layers): a „top” vagy felső oldallal szembenéző oldal, a kocka aljának középső sora
- *l* (Left two layers): a „front” vagy szemközti oldal bal oldalára eső oldal középső sora
- *r* (Right two layers): a „front” vagy szemközti oldal jobb oldalára eső oldal középső sora

- *x* (rotate): az egész kockát R-be forgatni
- *y* (rotate): az egész kockát U-ba forgatni
- *z* (rotate): az egész kockát F-be forgatni

Mikor a betűket követi a leírásban a (’) szimbólum, akkor a szembe lévő oldalt az óramutató járásával ellentétes irányba kell forgatnunk, ha az alapjel nincs a betű mellett, akkor az óramutatóval megegyező irányba forgatjuk a kockát. Ha a betűt a kettes jel követi, akkor két forgatás következik, azaz 180-fokkal fordítjuk az oldalt. Az R jelenti az óramutató járásával megegyező jobbra fordítást, az R’ pedig a jobb oldal visszaforgatását, vagy az óramutatóval ellentétes irányú forgatást. Az x,y és z betűket akkor használjuk, ha a teljes kockát el kell fordítani az egyik sarok körül. Ennek megfelelően ezt R,U és F irányba tehetjük meg. Mikor az x,y és z betűk alapjelek, akkor a kockát el kell fordítani a szemben lévő irányba. Mikor ezek a jelek be vannak keretezve, akkor a forgatást 180 – fokban kell elvégezni. A legáltalánosabb eltérés a Singmaster jelöléstől ami szintén hivatalos standard, a *„w”* úgymint wide (széles), így például az *Rw*-t használják az *„r”* vagy „front” szemközti oldal jobb oldalára eső oldal középső sora helyett (Frey, 1982; Rask, 2013).

A „Corner First”, vagyis a sarkokat először módszer esetében használatos még továbbá a ”MES” kiterjesztés, ahol a *M, E* és *S* jelek a középső sor forgatását jelölik (Wiki Rubik Kocka, 2013; Demain et al., 2011):

- *M* (Middle): L és R közötti sor, forgatási irány L (fentről lefelé)
- *E* (Equator): U és D közötti sor, forgatási irány D (balról jobbra)
- *S* (Standing): F és B közötti sor, forgatási irány F

A Thistlethwaite algoritmus

A 3x3x3-as kockakirakás optimalizálása, a legkisebb forgatási szám elérése számítógépes megoldással a csoport elméleti alapok feltárásával kezdődött, és ennek alapjait Morwen Thistlethwaite rakta le 1981-ben. A Thistlethwaite koncepció lényege az volt, hogy problémát felbontotta alproblémákra, azaz a kockán belül alcsoportok kialakítása révén kereste a megoldást. Elsősorban a következő főbb csoportok, vagy besorolásokat használta (Heise, 2002):

$$\boldsymbol{G_0} = < L, R, F, B, U, D >$$

$$\boldsymbol{G_1} = < L, R, F, B, \boldsymbol{U^2}, \boldsymbol{D^2} >$$

$$\boldsymbol{G_2} = < L, R, \boldsymbol{F^2}, \boldsymbol{B^2}, \boldsymbol{U^2}, \boldsymbol{D^2} >$$

$$\boldsymbol{G_3} = < \boldsymbol{L^2}, \boldsymbol{R^2}, \boldsymbol{F^2}, \boldsymbol{B^2}, \boldsymbol{U^2}, \boldsymbol{D^2} >$$

$$\boldsymbol{G_4} = \{\boldsymbol{1}\}$$

A fenti csoportosítások egy térbeli koncepciót jelentenek, melynek összefoglalója a következő:

$$\boldsymbol{G_{i+1}} \setminus \boldsymbol{G_i}$$

Minden elemhez, csoporthoz talált egy forgatási sort, amelyek a következő csoportba vezettek. Ezek után a következő alapösszefüggéseket dolgozta ki:

A véletlenszerű kocka általános kocka csoportjai:

$$G_0$$

A megfelelő oldalra forgatott csoport helyzete:

$$G_1 \setminus G_0$$

A folytatásban, azaz a helyes kirakás irányába a következő csoport tartozik:

$$G_1$$

A kocka kirakásának folyamatában nevezte meg aztán a többi csoportot:

$$G_2, G_3, G_4 \dots.$$

Bár a G_0 csoportok száma nagyon magas (~4,3 x 10^{19}), ahogyan egyre több alcsoportot képzünk, a csoportok száma folyamatosan csökkenni fog. Kezdetben úgy ítélték meg, hogy a legfeljebb 85 forgatásból meg lehet oldani a kockát. Aztán a forgatások száma folyamatosan csökkent a csoportosító módszer segítségével, először 63-ra majd 52-re, legvégül 45 forgatás lett a végső szám (Heise, 2002).

További fejlesztések

2007-ben kerül sor arra, hogy Daniel Kunkle és Gene Cooperman egy supercomputer segítségével a forgatások számát 26 forgatásban határozták meg, tehát maximum ennyi forgatással minden helyzetből kirakható az összekevert kocka. 2008-ban a már korábban említett Tomas Rokicki és csapata bebizonyította, hogy 26 helyett 23 forgatás is elég. Aztán 2010-ben további kutatások bizonyították, hogy 20 forgatásnál kevesebb nem lehet az „Isteni szám", tehát ennyi mindig szükséges, ha ki akarjuk kirakni a 3x3x3-as kockát (Richard, 2008). A számítógépes kirakóprogramok is ezekkel a feltételekkel szembesítették a tudományos közéletet.

1.4.3. A Layer by layer módszer egyszerűsített matematikai algoritmusai

A kockakirakási folyamatokat matematikai képletekkel is szükséges leírnunk, ha a Rubik kocka alapú optimalizációt számítógépes alkalmazáson keresztül is vizsgálni akarjuk, vagy esetlegesen egy szoftveralkalmazás keretében, több felhasználó számára is elérhetővé kívánjuk tenni.

Annak érdekében, hogy a matematikai leírást elvégezhessük, ismernünk kell azt a klasszikus nevezéktant, amelyet Singmaster után a következőképpen határozhatunk meg (Singmaster, 1981):

- *F* (Front): a kirakóval szembenéző „front" oldal
- *B* (Back): a „front" oldallal szembenéző oldal, a kocka hátulja
- *U* (Up): a „front" oldal fölött elhelyezkedő oldal, a „top" oldal
- *D* (Down): a „top" vagy felső oldallal szembenéző oldal, a kocka alja
- *L* (Left): a „front" vagy szemközti oldal bal oldalára eső oldal
- *R* (Right): a „front" vagy szemközti oldal jobb oldalára eső oldal

A folyamat matematikai leírásának legismertebb módja, mikor Rubik kocka kirakásának értelmezése az alcsoportok képzésének módszerével történik. Az egyik legfontosabb cél a Rubik kocka kirakásának számítógépes megoldása esetében felépíteni egy természetes csoportosítást, azaz alcsoportokat képezni (Joyner, 2002a):

$$G_n = \{1\} < G_{n-1} < \ldots < G_1 < G_0 = G$$

ahol $G =< R, L, F, B, U, D >$ a Rubik kocka csoport, amelyik megengedi a következő stratégiák alkalmazását:

– *jelöljük a Rubik kocka adott pozícióját egy elemmel* $g_0 \in G$,
– *meghatározva a csoport összes elemeit* $G_{k+1} \,/\, G_k$*:*

$$G_{k+1} \,/\, G_k = \cup_{i=1}^{r_k} g_{k+1,i} G_{k+1}, \;\; ahol \;\; r_k > 1, \forall 0 \leq k < n$$

(megjegyzés: $m_{-1} = 1, g_{n,1} = 1$),

– *(első lépés) ha* $g_0 \in g_{1,i} G_1$ $(ahol \;\; i \in \{1, \ldots, n_1\})$ $akkor\ megengedjük, hogy\ g_1 = g_{1,i}$ *és* $g_1' = g_1^{-1} g_0 (megj.: \;\; g_1' \in G_1,$
– *(következő lépés) ha* $g_k' \in G_k$ $kerül\ meghatározásra, és\ ha\ g_k' \in g_{k+1,j} G_k$ $(ahol\ j \in$ *1,…,n1 akkor megengedjük gk+1=gk+1,j és gk+1'=gk+1−1gk' megj.: gk+1'∈Gk+1,*
– *összegzeve az eddigieket, azt kapjuk hogy* $1 = g_n^{-1} g_{n-1}^{-1} g_{n-2}^{-1} \ldots . g_1^{-1} g_0$, *így*

$$g_0 = g_1 g_2 \ldots . g_{n-1} g_n.$$

Reményeink szerint ezek alapján meg tudjuk határozni az alcsoportok sorrendjét G_i -ben azon az úton, amely lehetőleg a legrövidebb. Relatív egyszerű kockaforgatással találhatjuk meg a megoldási lépéseket, ami ebben a formában nem túl hosszú $g_0 = g_1 g_2 \ldots . g_{n-1} g_n$.

A 3x3x3-as Rubik kocka esetében a megadott csoport $G =< R, L, F, B, U, D >$ szerint kalkulálhatjuk a Rubik kocka lehetséges megoldásainak számát. A csoportok száma alapján a következő permutációs számot kapjuk: 43252003274489856000 ≅ 4,3 x 10^{19}.

A kocka megoldási stratégiája a következő lehet a csoportelméleti koncepció szerint. Ha megengedjük, hogy $x^y = y^{-1} * x * y$ jelölje a kapcsolatot és $[x, y] = x * y * x^{-1} * y^{-1}$ jelöli a kommutátort az x, y csoport elemeinél.

Legyen M_R-rel jelölve az óramutató járásával megegyező negyed fordulat a középső sorral paralel jobb oldalon. A Layer by layer módszer megoldása három alapállapotból rakható össze, melyek a következők (Joyner, 2002b alapján):

1. állapot: Megoldjuk a felső top oldalt és a felső éleket,
2. állapot: Megoldjuk a középső éleket (és az alsó éleket úgy, ahogy azt a legrövidebben lehet),
3. állapot: Megoldjuk az alsó sarkokat (és az alsó éleket, ha szükséges).

A 7. táblázat tartalmazza azokat az algoritmusokat, amelyeket Joyner összesített a *"Mathematics of the Rubik's Cube"* című munkájában, 1996-ban. A táblázat rövidítései Singmaster által megadott kódok alapján jelölik a különböző oldalakat. A táblázatban található M_R jelölés a fent említett forgatást vezeti be a képletbe.

7. táblázat: Módosított Joyner féle matematikai algoritmusok és forgatási rend a 3x3x3-as kocka kirakásához

SORSZÁM	ALGORITMUS	FORGATÁSI REND
1.	$M_R^2 * U^{-1} * M_R^{-1} * U^2 * M_R * U^{-1} * M_R^2$	éleken 3 forgatással (UF, UL, UR)
2.	$(M_R * U)^3 * U * (M_R^{-1} * U)^3 * U$	a top élek felfordítása UF, UB
3.	$(R^2 * U^2)^3$	permutációk (UF, UB) (FR, BR)
4.	$(M_R * U)^4$	felforgatás UB, UL és felforgatás DF, DB
5.	$(r^{-1} * D^2 * R * B^{-1} * U^2 * B)^2$	UFR+, BLD++
6.	$[R, U]^3$	permutációk (UFR,DFR)(UBR,UBL)
7.	$F^2 * L^2 * U^2 * (F^2 * L^2)^3 * U^2 * L^2 * F^2$	permutációk (UF, UB)(ÚR,UL)
8.	$(D^2 * R^2 * D^2 * (F^2 * R^2)^2 * U)^2$	permutációk (UFL,UBR)(DFR,DBL)
9.	$(M_R^2 * U * M_R^2 * U^2)^2$	permutációk (UFL,UBR)(UFR,UBR)
10.	$[R * D * R^{-1}, U]$	sarok 3 forgatással (BRD, URB, ULB)

Forrás: Saját szerkesztés Joyner, 2002b alapján

2. FELHASZNÁLT ANYAGOK ÉS ALKALMAZOTT MÓDSZEREK

2.1. Anyag

Kutatási munkám során mind az interneten elérhető mind a nyomtatott irodalmi forrásokat, ezalatt értendő a hazai, és a nemzetközi szakirodalom egyaránt, alaposan áttekintettem, rendszereztem és kritikai elemezés alá vontam. A szakirodalmi áttekintést a fenntarthatóság különböző értelmezése, a gazdasági stratégiaalkotás során alkalmazott fontosabb játékelméleti megoldások, valamint a Rubik kirakási módszerek elemzése területeken végeztem. A témakör elméleti megalapozását a konkrét szakértői adatbázis felépítése követte a kocka főágens belső tulajdonságainak szoftveres elemzéséhez. A SMART (Simple Multi Attribute Ranking Technic) 3D vizsgálatokhoz felhasznált primer adatokat a Cleantech Incubation Europe (CIE) program gyakorlati vizsgálati eredményeinek szintetizálásával generáltam.

2.2. Módszer

A Rubik kocka kirakó szoftverek elemzéséhez SWOT elemzést végeztem a kutatás során, a Rubik kocka kirakási algoritmusok folyamatainak értékeléséhez pedig Teoretikus folyamatértékelést alkalmaztam. A low-carbon fejlesztési folyamatok fenntarthatósági értelmezéséhez Tartalomelemzést végeztem, melyhez az EU Low-carbon 2050 (http://www.roadmap2050.eu/) stratégiai útmutatót használtam fel. A tartalomelemzés lényege az volt, hogy az adatgyűjtés során az „A practical guide to a prosperous, low carbon Europe" európai társadalmi produktumot, valamint egyéb társadalmi kontrollon végigfutott szakmai dokumentációt "kérdeztem" az empirikus adatok megszerzése érdekében. A keresett játékelméleti algoritmusok esetében toleranciát, azaz a megengedhető eltéréseket vizsgáltam a kockatulajdonságok és a játékelméleti függvények paraméterezése vonatkozásában.

A low-carbon projektfejlesztési modell koncepció kritérium rendszerének meghatározására a Churchman-Ackhoff féle eljárást alkalmaztam. Annak érdekében, hogy a projektet befolyásoló legmeghatározóbb, leghasznosabb tulajdonságok kerüljenek optimalizálásra a meghatározó feltételrendszerek (tényezőcsoportok) közül, a tényezők várható hasznossággát vizsgáltam meg. Az egyes tényezők hasznosságának vizsgálatához hasznosságfüggvényeket írtam fel, amelyek jól reprezentálják a tényezőcsoportok egyenrangúságát vagy sorrendjét.

2.2.1. SWOT elemzés

MÓDSZERTANI LEÍRÁS

A SWOT elemzés olyan stratégiai tervezőeszköz, ami segít értékelni az erősségeket (strenghts), a gyengeségeket (weaknesses), a lehetőségeket (opportunities) és a veszélyeket (threats), amelyek egy termék, projekt, egy üzleti vállalkozás kapcsán, illetve bármely meghatározott cél elérése érdekében meghozandó szervezeti vagy egyéni döntés során felmerülhetnek. A SWOT elemzés tartalmazza a rendszer, az egyén vagy a szervezet belső és külső környezetének felmérését, ezen keresztül támogatva döntéshozót, hogy a legfontosabb témakörökre összpontosítson (Start Up guide, 2012).

Amire a vizsgálatokkal választ kell találni:

Erősségek:

- ✓ Milyen előnyei vannak a vizsgált rendszernek a low-carbon innovációs gyakorlatban, belső tényezők elemzése?
- ✓ Mit csinál jobban, mint másik rendszer?
- ✓ Mit mondanak a rendszerről, melyek az erősségei?

Gyengeségek:
- ✓ Mely részeken lehetne javítani?
- ✓ Mit kellene esetlegesen elkerülni?
- ✓ Mások mit tartanak a rendszer gyengeségének?

Lehetőségek:
- ✓ Milyen lehetőségei vannak a jövőben?
- ✓ Milyen érdekes trendeket, piaci irányzatokat érhetők el?

Veszélyek:
- ✓ Milyen akadályok merülhetnek fel a működése során?
- ✓ Mit csinálnak a versenytársak?
- ✓ Látszanak-e kedvezőtlen változások a működési környezetben?

A fent megfogalmazott kérdésekre az alábbi értékelő táblázatban, rövid válaszokat adva történi a kiértékelés.

	SEGÍTŐ TÉNYEZŐK	**GÁTLÓ TÉNYEZŐK**
Belső tényezők	erősség	gyengeség
Külső tényezők	lehetőség	veszély

A megoldáskereső szoftverek SWOT elemzésének célja:

A vizsgált megoldáskereső szoftver alkalmazások esetén a SWOT elemzése a célja, hogy a szoftver funkciói megfeleltethetők-e a low-carbon projektértékelő modell input és output rendszertulajdonságainak, felhasználói elvárásainak. A rendelkezésre álló adatok alapján azt kell megvizsgálni, hogy az „About low-carbon economy" (LCE Ltd, 2011) és a Launonen (2011) „Hubconcepts - Global best practice for innovation ecosystems" szakmai leírás alapján megfogalmazott low-carbon innovációs és inkubációs célrendszert hogyan és milyen mértékben elégíti ki a választott szoftvermegoldás.

A módszer választásának oka:

A SWOT analízis módszere jó lehetőséget kínál olyan áttekintő összehasonlítás elvégzésére, amelyben nincsenek egzakt, egymással jól összehasonlítható dimenziókban megfogalmazott tulajdonságok. A SWOT elemzés önmagában értelmetlen, de ha egy komplex vizsgálati folyamat részét képezi, akkor különösképpen hatékonnyá teszi a folyamatban gondolkodást.

2.2.2. Teoretikus folyamatértékelés

A 3x3x3 Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatában. A kirakási folyamatok és az azzal parallel beruház tervezési szinteket teoretikusan, a kocka egyes kirakási szintjei, állomásai szerint folyamatértékeltem. Az egyes színt-vizsgálatokat követően „Low-carbon interpretáció" összefoglalókat készítettem.

A folyamatértékelés szerkezeti felépítése a következő

- szektor vagy szint lehatárolás,
- teoretikus evaluáció,
- folyamat és eredmények evaluációja (interpretációk),
- folyamat és eredmények evaluációjának összegzése.

A kocka egyes állapotainak és kirakási szintjeinek ábrázolásához, valamint a low-carbon értelmezések magyarázatához az Online Ruwix Cube Solver programot használtam fel.

A módszer választásának oka:

A kocka kirakásának folyamatát le lehet írni különböző algoritmusok segítségével, de ezek csak a kirakás technikai folyamatát jellemzik számunkra. Mivel vizsgálataim során nem a leggyorsabb forgatási algoritmus elemzése, hanem az egyik leglogikusabb kirakási módszer valamint fenntarthatóság tervezési folyamatának összevetése volt a célom, ezért a teoretikus folyamatértékelésnél hatékonyabb vizsgálati módszert választani nem tudtam.

2.2.3. Adatgyűjtés több dimenziós „low-carbon" fejlesztési folyamatokra tartalom-elemzéssel

A tartalomelemzés az empirikus kutatási technikák ún. beavatkozás-mentes típusába sorolható. A beavatkozás-mentes vizsgálatok nagy előnye, hogy a kutatást végzők a vizsgált probléma realizációjától kellő távolságra, a folyamatokba való beavatkozás lehetőségét kizárva végezheti az adatgyűjtését. Ebben az esetben nem fordulhat elő, hogy adatgyűjtési eljárásunk befolyásolja a válaszadót. A tartalomelemzés az adatgyűjtésnek olyan technikája, amellyel információgyűjtést és elemzést végzünk az arra kijelölt dokumentumokból. A tartalomelemzést olyan társadalomkutatási módszer, amely az emberi közlések tanulmányozására alkalmas (Kérdő, 2008). A kutatási részprogram keretén belül azokat az Uniós társadalmi ellenőrzési mechanizmusokon, vitákon keresztül vezetett programdokumentációkat elemeztem, amelyek a low-carbon fejlesztési koncepciókat illetve a fennárthatóság témakörét egyaránt érintették (Tóth-Fogarassy, 2012). A tartalomelemzés lényege az volt, hogy az adatgyűjtés során az „A practical guide to a prosperous, low carbon Europe" európai társadalmi produktumot, a „Nemzeti Energiastratégia 2030", valamint „Magyarország megújuló energia hasznosítási cselekvési terve 2010-2020", röviden: „NCST 210-2020" társadalmi kontrollon végigfutott szakmai dokumentumokat "kérdeztem" az empirikus adatok megszerzése érdekében

Az elvégzett Tartalomelemzés során azt tartottam lényeges szempontnak, hogy az adatgyűjtés során az előbbiekben jelzett társadalmi produktumokban, a legtöbbször különböző célmeghatározásokat, dokumentumokat úgy "kérdeztem" empirikus adataim forrásaként, hogy az ott fellelhető ellentmondásokat elkerüljem, a közös vezérlőelvek mellé pedig a zöldenergia, vagy klímabarát beruházások preferencia szempontjait világossá tegyem.

A módszer kiválasztásának oka:

Az adatgyűjtés olyan primer formája, mikor a szakterületi dokumentációkban fellelhető ellentmondások feloldására keresünk választ, vagy az ellentmondások újraértelmezésével lépünk tovább az elemzésben.

2.2.4. Játékelméleti algoritmusok tolerancia és alkalmazhatósági vizsgálata

A tolerancia vizsgálat célja a műszaki életben a megadott méretektől, mennyiségtől, vagy minőségtől való megengedett legnagyobb eltérést megállapítását célozza. A játékelméleti algoritmusok esetében azt vizsgáltam, hogy melyik módszer egyezik meg tulajdonságaiban a Rubik kocka kirakási folyamat modell tulajdonságaival, azoktól milyen megengedhető mértékben tér el a reprezentativitás megtartása mellett. A keresett játékelméleti algoritmusok esetében toleranciát, azaz a megengedhető eltéréseket vizsgáltam a kockatulajdonságok és a játékelméleti függvények paraméterezésének vonatkozásában (Ligeti, 2006).

A játékelméleti algoritmusokat sorra vettem, és a modellalkotás folyamatában az egyes forgatási algoritmusokhoz (interpretációkhoz) kapcsolható modelleket. A többcélú optimalizálási feladatok során a következő a modellalkotás folyamata, (Forgó et al., 2005):

1. Kritériumrendszer megadása (főbb tulajdonsághalmazok meghatározása)
2. A tulajdonsághalmazok függetlenségi vizsgálata (tulajdonság átfedések elkerülése)

3. Döntési változók, paraméterek megadása a tulajdonsághalmazon belül (determinisztikus vagy sztochasztikus, azaz valamilyen valószínűségi szinttel teljesülő tulajdonságok megjelölése)
4. Korlátozó feltételek megadása a halmazra vonatkozóan (halmazok felállítása)
5. Lehetséges kritériumok kijelölése a feltételrendszerhez a halmazon belüli célfüggvények megadása (célfüggvények száma véges)
6. A célfüggvények maximumának keresése

A módszer választásának oka:
A lineáris programozásban alkalmazott modellalkotási folyamat világos képet ad a többcélú optimalizálási feladatok kritériumrendszeréről, ezért én is ezt a módszert választottam a háromszintű stratégiai célrendszer-alkotás megvalósítására.

2.2.5. Kritériumok és kockatulajdonságok meghatározása Churchman – Ackhoff féle eljárással

A fejlesztési folyamatokat befolyásoló kritériumok súlyozása szükséges ahhoz, hogy az ismert tényezőcsoportból kiválasszuk azokat a legfontosabb tulajdonságokat, amelyek a projekt fejlesztésének, a beruházások megvalósításának meghatározó feltételeit adják. A Churchman és Ackoff által kidolgozott módszerben a szempontok számának megfelelően két alig különböző eljárás szerepel. Az első 1-7 szempontra, míg a második ennél több szempontra lett kidolgozva.

A módszer az egymást követő összehasonlításokon alapszik, és kisszámú szempont, jelen esetben négy, kiválasztására is alkalmas. (A közismert Guilford-féle módszer csak öt feletti szempont fölött ajánlott.) Első lépésben szempontokat egyelőre csak „érzésre", szakértői becsléssel kell a vélt fontosságuk alapján rendezni. Első közelítésben a legfontosabb szempontnak 1 értéket adva, a többi szempont súlyát ehhez képest kell felvenni. A pontosítás érdekében a legfontosabb szempontot és annak súlyát a többi szempontból készített csoportokhoz (illetve azok súlyainak összegéhez) kell viszonyítani. Ha a kiemelt szempont fontosabb, de a súlyokra felírt reláció ezt nem tükrözi, akkor a súlyokat megfelelően korrigálni kell. A korrigálás után a kiemelt szempontot egy elemmel kisebb csoporthoz kell ugyanígy viszonyítani (Simongáti, 2009a, Russell, 2003). Az eljárás addig folytatandó, amíg a kiemelt szempont és a többi szempontból alkotott (egyre kisebb) csoport azonos fontosságú. Ezt a súlyokra is felírva, tovább lehet lépni a második legfontosabb szempontra, ahol ugyanezt végig kell futtatni. Ha minden szempont súlya megvan, akkor úgy kell normalizálni a dominancia meghatározást, hogy a végső súlyok összege 1 legyen (Churchman-Ackoff, 1957).

A nagyobb, több szempontot is értékelő feladatokra kidolgozott eljárás a fent leírtaktól annyiban eltérő, hogy ott egy tetszőlegesen kiválasztott szempontot kell összehasonlítani olyan csoportokkal, amelyekben a szempontok száma nem több 5-nél. A Churchman és Ackoff módszerek előnye, hogy pontos eredményt lehet elérni, de mindezt kissé időigényesen (Simongáti, 2009b).

A főágensek kijelölésére (Churchman és Ackoff módszer alapjait felhasználva) a következő metodikát dolgoztam ki:

- **0. lépés:** Preferencia sorrend kialakítása előzetes becsléssel (F_1, $F_2 \ldots F_n$)

- **1. lépés:** Fontosság szerint hasznossági értékek hozzárendelése

  - ❑ Az első (F_1) súlyát 1-nek véve meg kell adni a többi szempont relatív súlyát az elsőhöz képest.

  - ❑ Pl. A F_1 szempont fontosabb, ugyanolyan fontos, vagy kevésbé fontos, mint az összes többi együtt?

  - ❑ Képletszerűen: W_1>(=,<) $w_2+w_3+\ldots+w_n$?

- Ezek összevetése, és a fontosság korrigálása:
    - ha F_1 szempont fontosabb, de a súlyokkal felírt egyenlőtlenség nem ezt mutatja, akkor w_1-et úgy kell módosítani, hogy az egyenlőtlenség tükrözze a relációt →2. lépés
    - Ha F_1 nem olyan fontos, akkor annak megfelelően csökkentem a w1-et.
    - Majd hasonlítsuk össze a F_1 szempontot a $\{F_2, F_3, \ldots F_{n-1}\}$ szempontok csoportjával, és ismételjük addig, amíg $\{F_2, F_3\}$ csoporthoz jutunk.
- **2. lépés:** hasonlítsuk össze F_2-t a $\{F_3, F_4 \ldots F_n\}$ csoportokkal a 1. lépés szerint.
- **3. lépés**, folytassuk a sort, amíg a F_{n-2} $\{F_{n-1}, F_n\}$ összehasonlításhoz jutunk.
- **4. lépés:** standardizálás: osszuk el minden szempont súlyát Σw_i-vel

Standardizálás előnye: megbízhatóbb eredményt ad, mint a közvetlen becslés.

Standardizálás hátrány: nem alkalmazható alapvetően csak max. 7 szempontra.

Churchman és Ackoff módszer gyakorlati alkalmazása népszerű az értékbecslési folyamatok, a dominancia sorrend kiválasztása során, ezért szoftveralkalmazásként is jól ismert (Ose L. S., 2008). A rövidített szoftver applikációt használtam fel a vizsgálatokhoz.

A módszer választásának oka:

A súlyozási eljárások kiválasztásához több olyan lehetőség kínálkozik, melyeket a gyakorlat jelenleg preferál (pl. Guilford-féle vagy a közvetlen becslés), de mivel én szeretném a legpontosabb matematikai eszközöket bevezetni a folyamatok leírására, így a Churchman-Ackoff féle eljárást választottam.

2.2.6. Hasznossági függvények alkalmazhatósági vizsgálat a több dimenziós értékelésben (SMART)

Annak érdekében, hogy a legmeghatározóbb, leghasznosabb tulajdonságok kerüljenek optimalizálásra a meghatározó a Rubik kockás low-carbon fejlesztés befolyásoló feltételrendszerek (tényezőcsoportok) közül, a tényezők várható hasznossággát kell megvizsgálnunk. Az egyes tényezők hasznosságának vizsgálatához hasznosságfüggvényeket írhatunk fel, amelyek jól reprezentálják a tényezőcsoportok egyenrangúságát vagy sorrendjét. A hasznosságfüggvény az egyes állapotok kívánatosságának kifejezésére, minden állapothoz egyetlen számot rendel. A hasznosságokat a cselekedetek következményeinek a valószínűségével kombinálva kapjuk az egyes cselekedetekhez tartozó várható hasznosságot (Russel-Norvin, 2003).

Egy S állapotnak a döntést meghozó tulajdonságok szempontja szerinti hasznosságára az $U(S)$ jelölést használhatjuk. Az állapotokat az adott körülmények pillanatfelvételeinek tekinthetjük, így egy nem determinisztikus A cselekvésnek az $Eredmény_1$(A) állapotok a lehetséges következményei, ahol az i index a különböző következményeken fut végig. Russel-Norvin szerint, az A végrehajtása előtt az ágens egy $P(Eredmény_1(A)|Tesz(A), E)$ valószínűséget rendel minden egyes következményhez, ahol az E az ágens által a világról elérhető tényeket jelöli, és a $Tesz(A)$ egy állítás, hogy az A cselekvés végrehajtódik a jelenlegi állapotban. Ekkor a következő formulával kiszámíthatjuk a cselekvés $EU(A|E)$ várható hasznosságát adott tények esetén:

$$EU(A/E) = \sum_i P\,(Eredény_1(A)/Teszt(A), E)\; U\,(Eredmény_i(A))$$

A maximális várható hasznosság (MVH) elve azt mondja ki, hogy egy racionális ágensnek azt a cselekvést kell választania, ami maximalizálja az ágens várható hasznosságát. Ha cselekvések egy legjobb sorozatát szeretnénk kiválasztani ennek az egyenletnek a felhasználásával, akkor az összes lehetséges cselekvéssorozatot számba kellene venni, és a legjobbat kiválasztani (David, 2002; Russel- Norvin, 2003).

A kutatási programban végzett hasznosság vizsgálatok, illetve a kiértékelő függvények kerültek be a SMART szoftveralkalmazásba. A SMART (Simple Multi Attribute Ranking Technic) a programban a hasznosság vizsgálatok eredményeit és az egyes tényezők kapcsolatát 3D-ben is tudtam vizualizálni, itt a vizsgálatok elvégzéséhez primer adatokat használtam fel a CIE kutatási program adatbázisából.

A SMART (Simple Multi Attribute Ranking Technic) egy olyan értékelő technika, amely sorrendbe tudja rakni számunkra azokat a tulajdonságokat, amelyek a döntés szempontjából meghatározóak lehetnek. A módszer az egymás közötti kapcsolatokat és kereszt-összefüggéseket is mutatja számunkra. Fejlesztését a Harvarde University, az MIT és a University of Southern California közösen végezte az elmúlt években. Az alkalmazás az alternatívák egyes szempontok szerinti alapértékeit hasznossági függvények segítségével hasznossági értékké transzformálják. A szoftveralkalmazás legértékesebb tulajdonsága, hogy a jellemző kapcsolatrendszereket 2D és 3D formában is képes követni (Huhn, 2013). A hasznossági függvények hasznossági értékké történő alakítása Simongáti (2009b) által meghatározott lépések szerint történt.

A módszer választásának oka:

A hasznossági függvények és az értékelési rendszer adatait háromdimenziós kiterjesztésben tudja ábrázolni a SMART program.

3. EREDMÉNYEK

3.1. Rubik kocka megoldó szoftverek és alapösszefüggéseinek összehasonlítása SWOT áttekintéssel

A szoftverfejlesztés folyamatát a csoport elmélet eszközei leegyszerűsíthetik a számításokat több száz vagy millió elrendezés alcsoportjainak meghatározásával, melyek közös matematikai tulajdonságokkal rendelkeznek. 1992-ben Herbert Kociemba, német matematikus egy ravasz módszerrel állt elő a kocka 43 milliárd lehetőségének lecsökkentésére (Ajay, 2011). A kalkuláció matematikai alapja (csoportelmélet alapján) az, hogy hogyan számoljuk ki a variációs lehetőséget, azaz hányféle minta lehet a kockán:

- 8 sarok = 8! féle pozíció / mindegyik 3-féle orientációval = 3^8
- 12 él = 12! féle pozíció / mindegyik 2-féle orientációval = 2^{12}
- Lehetetlen helyzetek:
  - nem lehet elemcsere (2),
  - nem lehet élorientáció (2),
  - nem lehet sarokorientáció (3).
- Azaz 2x2x3 = 12-vel osztás, ami összesen = (8! x 3^8 x 12! x 2^{12}) / 12 ~= 4.3 x 10^{19}

Kociemba máshogyan közelítette a kocka matematikai összefüggéseit, ahelyett hogy a megszokott módon, adott elrendezésekre alapozta volna, felvázolt egy alcsoportot, ami a kocka 18 lehetséges mozgatásából 10 mozgatás sorozatán alapult. Ennek a 10 mozdulatnak a kombinációjával rájött, hogy elérhet 20 milliárd különböző konfigurációt egy kirakott kockából. Ez azért számít fontos lépcsőfoknak, mert az így kapott alcsoport már elég kicsi ahhoz, hogy beférjen egy hétköznapi asztali számítógép memóriájába. A célra Kociemba megalkotott egy programot is, a Cube Explorert, amit egy amerikai matematikus, Michael Reid 1995-ben továbbfejlesztett, és alkalmazásával 30-ra becsülte a minimálisan szükséges forgatások számát. Az elméleti tudósok már 1982-ben a 20-at tartották Isten számának, vagyis a minimálisan szükséges forgatások számát, de a bizonyításhoz szuperszámítógépekre lett volna szükség. Az Isteni szám (azaz a 20 forgatás) bizonyítása végül csak 2010 júliusában történt meg, amikor Tomas Rokicki, Herbert Kociemba, Morley Davidson és John Dethridge (Rokicki et al., 2010), büszkén jelentették be a Világnak, hogy bizonyítva van:„God's Number for the Cube is exactly 20”.

A Kociemba által fejlesztett Cube Explorer Rubik kocka kirakó program volt tehát az első olyan program, amely 30 körüli forgatási számmal tudott miden állapotból kirakott kockát létrehozni. Az első szoftver után, illetve ez alapján pedig a Világ minden pontján elindultak a különböző kirakó programhoz kapcsolódó egyedi fejlesztések. Annak érdekében, hogy röviden áttekintsük a Rubik kockás szoftverfejlesztés, valamint a Rubik kocka kirakási algoritmusára épülő low-carbon projektfejlesztési módszertan kapcsolati hálóját, három fontos fejlesztési irány SWOT elemzését végzem el. A low-carbon fejlesztés során cél, hogy a vizsgált fejlesztés, beruházás folyamata akár szoftverek segítségével is gyorsabbá, egyszerűbbé tehető legyen. A szoftverek szerepe akkor lehet jelentős, ha kocka egyes oldalainak tulajdonságokhoz rendelését követően, a projekt kiindulási állapotát a Rubik kocka rendezetlenségi állapotával is jellemezni tudjuk. Ha a kocka állapottal írtuk le a rendezetlenség fokát, akkor a megoldáskereső szoftver már könnyen megmondja a felhasználó számára, hogy milyen útvonalon lehet a rendezettség különböző szintjeire eljutni. A szoftverekkel végzett megoldáskeresés mindössze azt a kérdést veti fel, hogy az alkalmazott útvonal megfelelő-e vagy sem, be tudja-e tartani a megoldáskeresés folyamata azokat a szakmai szempontokat (Global best practice for innovation ecosystems), amelyek a sikeres projektfejlesztés alapjait jelentik.

A módszertani részben már részletes vázolt SWOT elemzés célja az, hogy világos legyen számomra, hogy a szoftverek funkciói megfeleltethetők-e a low-carbon projektértékelő modell input és output elvárásainak. Az elemzést a klasszikus SWOT elemzés szabályai szerint végeztem, ennek részleteit nem, csak eredménytábláit mutatom be a fejezetben. Az érthetőség kedvéért azért az egyes szoftveralkalmazásokról rövid tájékoztatót adok.

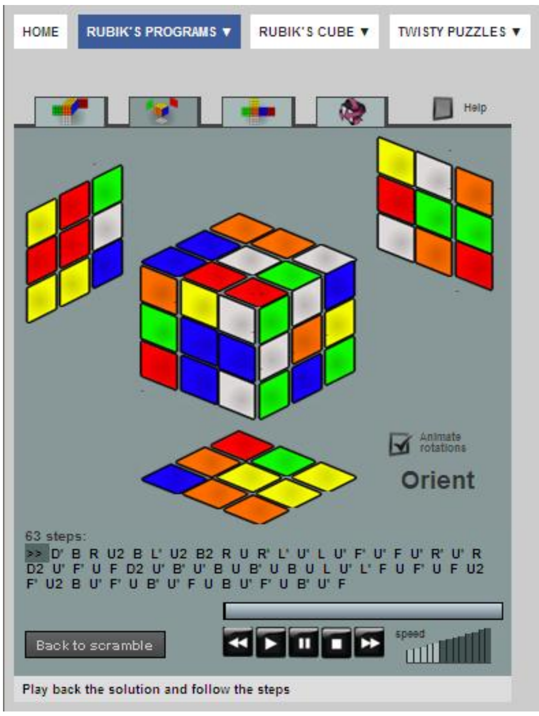
SWOT elemzéssel értékelt szoftverek:

- ✓ RUWIX PROGRAM (KOCIEMBA CUBE EXPLORER FEJLESZTÉS)
- ✓ MEGOLDÁSKERESŐ LBL SZOFTVER (NAGY GÁBOR)
- ✓ RUBIKSOLVE PROGRAM (ERIC DIEC)

3.1.1. Ruwix program (Kociemba Cube Explorer fejlesztés) SWOT elemzése

Ruwix program Kociemba által, 2005-ben fejlesztett solver program alapján készült, a magyar származású, Dénes Ferenc által fejlesztett komplex kirakó és demonstrációs program. A szoftver a legrövidebb kirakási utat választja bármely kevert állapotból. Átlagos forgatási szám 50-60 forgatás, mely nem a *Layer by layer* algoritmusokat preferálja. Ebben az esetben lényegesen nagyobb számú algoritmus töltöttek fel a fejlesztők az optimális megoldáskereső programba, így az sokkal több jó megoldást talál az optimalizáció során. Az online megoldó szoftver nagyon látványos és minden fontos információt megoszt a használókkal (3. ábra).

A Ruwix online weben futó Java alkalmazás, mely nem használ semmilyen egyéb támogató felületet. A Rubik kocka tanulásához, az egyes módszerek tanulmányozásához nélkülözhetetlen funkciókkal látta el a fejlesztő. Alkalmas többek között arra, hogy a kirakás folyamatát lépésről lépésre animálva mutassa meg az érdeklődőnek. A megoldáskereső bármely kocka állapotból képes animáltan végigvezetni a megoldási, forgatási lépéseket, mely lehetőséggel a Rubik kocka kirakó versenyre készülő játékosok élnek nagyon nagy előszeretettel. A Ruwix programban lehet online játszani a különböző Rubik termékekkel (2x2x2, 3x3x3, 4x4x4, 5x5x5 stb.), melyek 3D formában nyújtanak kiemelkedően kellemes játékélvezetet.



3.ábra: A Ruwix program grafikája és legrövidebb megoldási képlete
Forrás: Dénes, T. (2005) Ruwix.com

A Ruwix program, megoldáskereső SWOT értékelése (8. táblázat) a low-carbon projektértékelő modell input és output elvárásainak függvényében:

8. táblázat: Ruwix program SWOT táblája

	SEGÍTŐ TÉNYEZŐK	GÁTLÓ TÉNYEZŐK
Belső tényezők	**ERŐSSÉG** Kiváló grafikával és vizuális megjelenítéssel működik, sokan a Világ legjobb kirakó szoftvereként emlegetik, Rubik játékokra specializálódott, a 3x3x3-as kockán kívül sok-sok logikai megoldását kínálja.	**GYENGESÉG** Low-carbon megoldások szempontjából kedvező layer-by layer módszer mellett, más, gyorsabb algoritmusokat használ, jelen formában nem alkalmas az értékelésre
Külső tényezők	**LEHETŐSÉG** A kiváló megjelenítési és használhatósági jellemzők miatt, low-carbon módszertani specifikációt is célszerű lenne ráfejleszteni.	**VESZÉLY** A program online verzióban fut, programhoz speciális adatok hozzárendelése nem lehetséges ebben a formában ingyenes verzió összekapcsolása a fizetős SMART segédprogrammal bonyolult használatot jelenthet egy esetleges low-carbo specifikáció esetében is.

Forrás: saját szerkesztés

3.1.2. Rubik Kocka Megoldáskereső szoftver értékelése és SWOT elemzése

A Rubik Kocka megoldó/kirakó szoftver bemutatására egy hazai szoftverfejlesztést, Nagy Gábor (Debreceni Egyetem) informatikus mérnök által fejlesztett, „Megoldáskereső módszerek” című leírását, illetve módszertani útmutatását használtam fel elsősorban, melyben egyedi módon – állapottér reprezentációval - került kidolgozásra a többszintű megoldáskeresés problematikája. A szoftver kiválasztásának másik fontos szempontja az volt, hogy a megoldáskeresés a layer by layer módszert preferálja, kizárólag ezzel az algoritmussal tudomásom szerint egyetlen megoldáskereső alkalmazás sem fut jelenleg, mivel ezt a módszer „túl lassú”-nak tekintik. (Viszont bármely megoldáskereső tudná futtatni, ha ezzel programozzuk.)

A fejlesztő által írt program 2008-ban, Java nyelven íródott, NetBeans IDE 6.1 fejlesztői környezetben. Annak érdekében, hogy világos legyen a program felépítése, a két alapcsomag – az állapottér és a kocka csomag – struktúráját kell, hogy megértsük.

Az „Alapotter” elnevezésű csomag két absztrakt osztályt és egy interfészt tartalmaz, melyek pontosan rögzítik az állapotterek általános elemeit, jellemzőit. Az alapprobléma implementálásakor ezeket az elemeket konkretizálja a program az adott probléma állapottér-reprezentációjához. A program az eltérő állapotok esetében folyamatosan ellenőrzi, hogy az célállapot-e vagy sem. A tervező leírása szerint, a kialakult Heurisztikus állapotot a „HeurisztikusAllapot” interfész biztosítja, melyet szintén implementálni a kell program kezdetén. A megoldáskereső szoftver esetében „KockaAllapot” osztályt, vagy kocka csomagot neveztek meg kiindulásként, mely osztály elemei az állapottér egy-egy elemét írják le. Ez az osztály tartalmazza a Rubik Kocka állapotát nyilván tartó 54 elemű byte-tömbön kívül eső konstruktorokat és az állapotokra meghívható összes módszert. A célállapot ellenőrző függvény végigjárja a kocka állapotát leíró háromdimenziós tömböket, és ha valamelyik oldalon egy oda nem illő szint talál, akkor „hamis” üzenettel tér vissza, ha sikerül végigjárnia tömböt, és nincs „hamis" üzenet akkor a kocka ki van rakva, minden szín a helyére került (Nagy, 2008a).

A program leírásában az szerepel, hogy a kocka állapotát 54 szám határozza meg, ezek a [0,5] intervallumból kerülhetnek ki, s a számok egy-egy színt szimbolizálnak (Nagy, 2008 alapján):

$$H = \{(0,0, \ldots..,0), (1,0, \ldots..,0), \ldots.., (5,5, \ldots.,5)\}$$

A ≠ H, mert H nem lehet minden eleme valódi állapot.

$$A = \{a | a \in H_1 \times \ldots. \times H_n\}$$

A kocka csomag leírása

Az „Állapotter csomag" osztályait és interfészét felhasználva készített osztályok a „kocka csomagban" foglalnak helyet, és ezek az osztályok már szorosan a Rubik-kockához, illetve annak struktúrájához kapcsolódnak. A „KockaAllapot" osztály példányai az állapottér egy-egy állapotát írják le, de az osztály tartalmazza a Rubik-kocka állapotát nyilvántartó 54 elemű byte-tömbön kívül az osztály példányosításához használt konstruktorokat és módszereket is, melyek a következők (Nagy, 2008b):

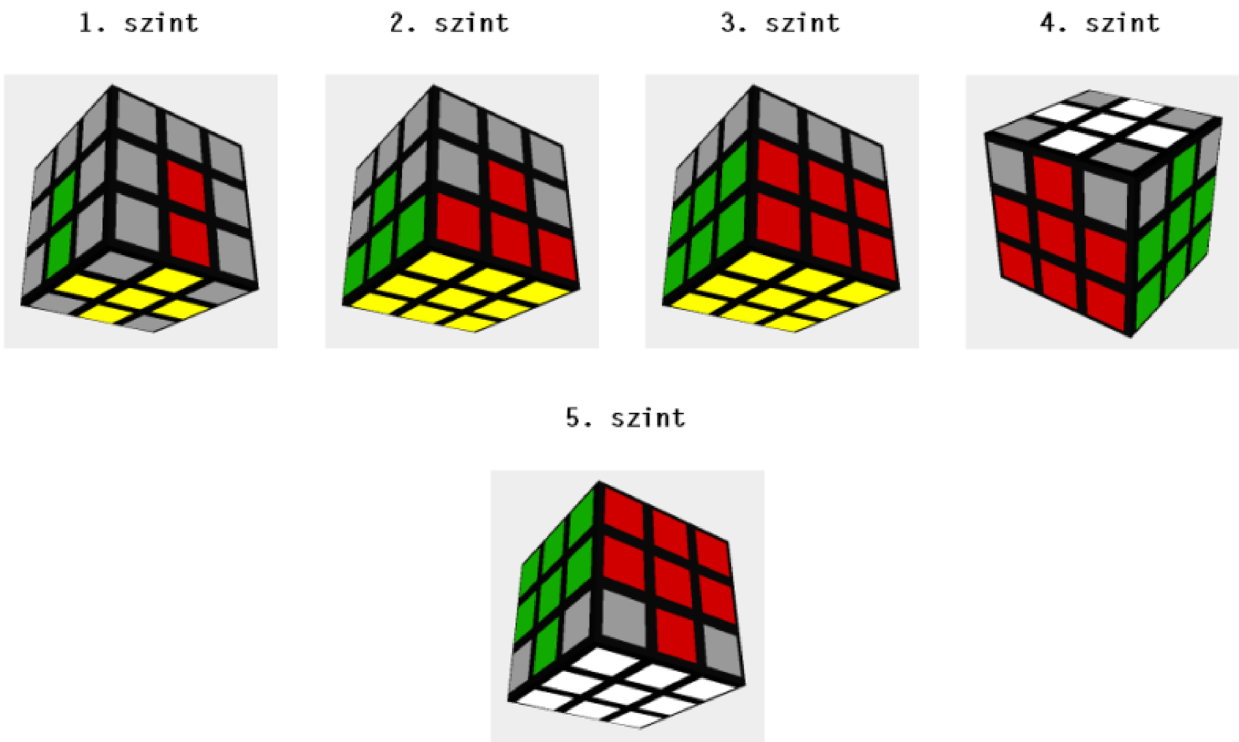
- ✓ „Célállapot ellenőrző függvény", melynek visszatérési értéke igaz vagy hamis lehet. Három egymásba ágyazott fór-ciklus segítségével végigjárja a kocka állapotát leíró háromdimenziós tömböt, és ha egy oldalon oda nem illő színt talál, akkor azonnal „hamissal" tér vissza. Ha sikerült végigjárnia a tömböt, akkor minden szín a helyén van és visszatérési értéke „igaz" lesz.
- ✓ „Operátor" - alkalmazási előfeltételt ellenőrző függvény, mely megvizsgálja, hogy az adott állapotra alkalmazható-e a paraméterként kapott operátor feltétel. Visszatérési értéke ennek is logikai érték, ami a Rubik - kocka esetében minden esetben „igaz".
- ✓ Az „Alkalmaz függvény", mely a paraméterként kapott operatort alkalmazza az adott állapotra, visszatérési értéke a keletkezett állapot függvénye. Ehhez másolatot készít a kocka állapotáról, majd a másolat tömbjén elvégzi az operátorok megfelelő összes értékmásolást és visszatér a másolattal.
- ✓ Az adott állapotot egy paraméterként kapott állapottal összehasonlító függvény. Visszatérési értéke logikai mely „igaz", abban az esetben, ha a vizsgált állapotok tömbjének minden eleme megegyezik. Egyébként hamis értéket hordoz..
- ✓ Egy kiértékelő függvény, amely vizsgálatunk szempontjából különösen fontos.
- ✓ Az állapotot nyilvántartó „adattag" lekérdező módszer.
- ✓ Kiíratással kapcsolatos módszerek, metódusok.

A Layer by layer módszer és a kiértékelő függvény

A fejlesztő választása okán, a program egy *MOHÓ* keresőt (greedy seach) használ a kocka megoldásához, és ennek megfelelően a kiértékelési függvény egyedül a heurisztikus függvényből áll, amit a program esetében, a kocka csomagoknál már említett, "KockaÁllapot" osztály „heurisztika metódusa" implementál. A metódus az adott állapotokat a korábban mar említett soronként haladó, vagy más néven „Layer by layer" módszer szerint pontozza le. A heurisztikus csomag hatására tehát a program a „Layer by layer" módszer, azaz a sorról-sorra történő kirakás alapján keresi a megoldást, bár a megoldáskeresésében köztudottan nem ez a leghatékonyabb, legrövidebb úton elérhető módszer. A program nem elemzi a kiindulási állapotot, mert a kezdő oldal optimalizálásához egy komplikált kiértékelő függvényre volna szükség, melyet a fejlesztő szükségtelennek ítélt a programcsomagban, így a kiértékelő függvény használata helyett a program mindig a sárga oldallal kezd. A módszerrel kapcsolatban könnyen ellenőrizhető rétegekről (layerekről), vagy más szóval szintekről beszélünk, ezért a „heurisztikus függvény” is a szint ellenőrzésével kezdődik, hogy elkerülje az aktuálisnál is alacsonyabb szinteken fontos ellenőrzéseket (Nagy, 2008c alapján).

Ezek a szintek a következők, 4. ábra:

0. szint: a kocka még az 1. szint követelményeinek sem felel meg.
1. szint: a sárgát is tartalmazó élek a helyükön vannak és jó irányban állnak, azaz kész a „sárga kereszt”.
2. szint: a sárgát is tartalmazó sarkok a helyükön vannak és jó irányban állnak, azaz kész a legfelső sáv.
3. szint: kész van a középső sáv.
4. szint: a fehéret is tartalmazó élek a helyükön vannak és- jó irányban állnak, azaz kész a „fehér kereszt”.
5. szint: a fehéret is tartalmazó sarkok a helyükön vannak és jó irányban állnak, azaz a kocka a célállapotban van.

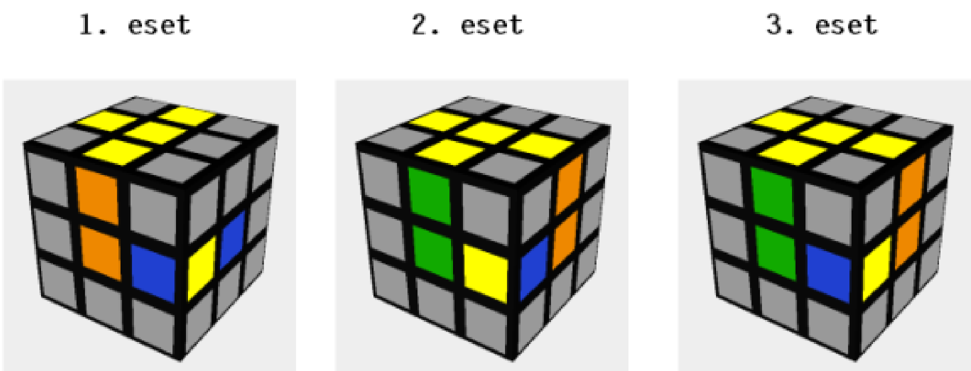


4. ábra: A Layer by layer módszer szintjei a programban
Forrás: saját szerkesztés (Nagy, 2008 alapján)

A fejlesztő leírása alapján előfordulhat olyan eset, amikor a heurisztika vagy akár a szint lerontása nélkül nem tudunk továbblépni a programmal. Ebben az esetben segítségünkre lehetnek az úgynevezett megoldó algoritmusok, melyek olyan lépéssorozatok, melyeket a megfelelő állapotra alkalmazva, először lerontják ugyan a heurisztikát, de a lépéssorozat végére közelebb kerülünk a célhoz, mint az alkalmazása előtt voltunk. Az első szintet (első sor kirakása) alapvetően elérhetjük algoritmusok nélkül is, mégis ez a legnehezebben megvalósítható része a heurisztikus függvénynek. Nagy szerint ennek oka, hogy a magasabb szintekkel ellentétben, ahol egy-két forgatást leszámítva már csak algoritmusokat alkalmazunk, míg ezeknél az egyszerű, de sok különböző forgatási alternatívát jelentő lépéseknél sokkal nehezebb az emberi tudást átadni a gépnek. A felsőbb szinteken nem okoz gondot a heurisztikus algoritmusok alkalmazása, minden állapotra tudunk illeszteni néhány fix algoritmust, csak azt kell eldöntenünk, hogy melyiket alkalmazzuk előbb.

Egy állapot heurisztikája alatt az állapotra meghívott heurisztikus függvény visszatérési értékét, azaz az állapot „jóságát" érti a programozó. Az elgondolása az volt, hogy minél, alacsonyabb szinten vagyunk, az állapot heurisztikája annál magasabb értékről indul, majd a szintnek megfelelő vizsgálatok során minél távolabbinak tűnik az eggyel magasabb szint, ez az értek annál jobban növekszik. A növekedés mértékét tehát a szint teljesítéséhez szükséges élek, vagy sarkok helyzete és / vagy iránya határozzák meg. Minden ilyen él vagy sarok többé-kevésbé megnöveli a heurisztikát. Mértéke attól függ, hogy mennyire messze vagy közel esik a helyétől, vagy egy olyan helytől ahonnan egy algoritmus segítségével helyére tehető. A fejlesztő szerint egy szinten belül sohasem nő meg annyival a heurisztika értéke, hogy egy alacsonyabb szintű állapotnak a heurisztikája kisebb legyen ettől. Ez a feltétel elengedhetetlen ahhoz, hogy a kereső a módszerhez mérten, a lehető legrövidebb megoldást találja meg. Ennek egyik következménye, ha elértünk egy bizonyos szintet a programmal, akkor elég csak ahhoz a szinthez kapcsolódó vizsgálatokat elvégeznünk, hiszen a többi már úgyis vagy még úgysem teljesül. Ezeknek megfelelően a pontozás a programban a következő módon zajlik (Nagy, 2008 alapján):

- A ***szint meghatározás*** a pontozás, a kiértékelés első lépese. Minél magasabb szinten vagyunk, ez a szám annál kisebb lesz. A heurisztika kezdő értéke ötös szint esetén, a kiértékelő függvényben ”0”**.**
- ***0. szinten:*** Egy helyén lévő és jó irányban álló él alig növeli meg a heurisztikát, míg a többi él minél messzebb esik a helyétől, annál jobban növeli azt. Ha már legalább két él áll jó helyen és irányban, akkor megengedhetjük az algoritmusok használatát, de ezzel egy időben az élek kisebb növekedést is okoznak a heurisztikában, ha közel vannak ahhoz, hogy egy algoritmus segítségével helyre rakjuk őket. Ezek az algoritmusok csak 3-5 lépésből állnak, viszont különböző egyéb, járulékos hatással is rendelkeznek. A programban három ilyen algoritmus alkalmazására kell figyelnünk oldalanként. Ennek oka, hogy a szoftver rögzített szemszögből értelmezi az operátorokat, mindig a sárga oldal van fent és a kék oldal szemben. E miatt ugyanaz a mozdulatsor más operátorokból épülhet fel a különböző oldalakra nézve, de tudnunk kell a helyeset kiválasztani. Jó példa erre, ha megnézzük a kék-sárga él három olyan helyzetét, melyből csak algoritmus segítségével tehető helyére (5. ábra).



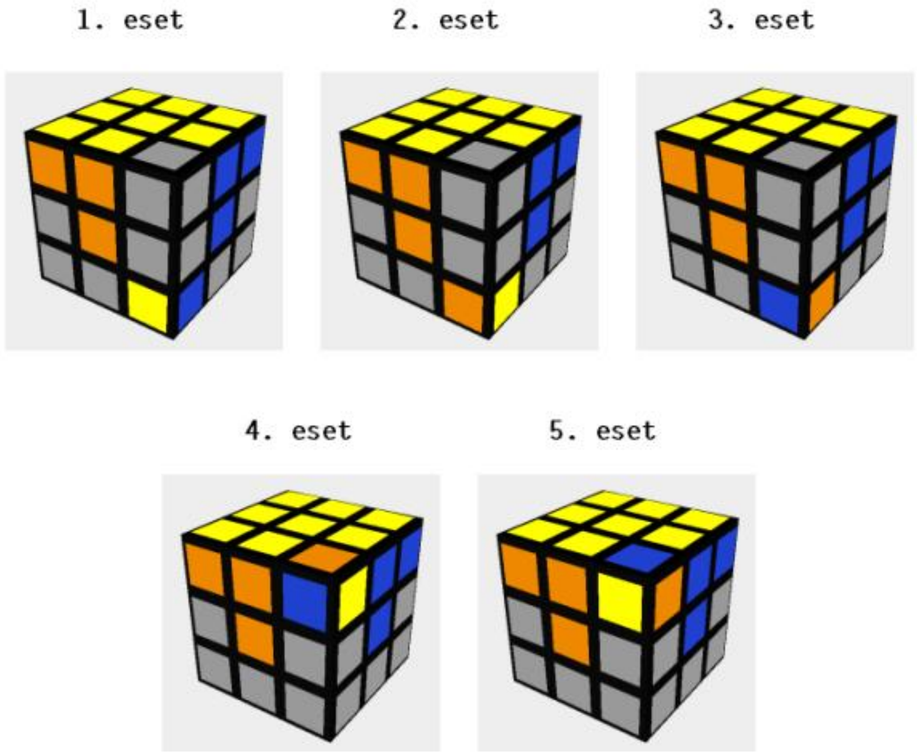
5. ábra: Kizárólag algoritmus alapján helyre rakható élek
Forrás: saját szerkesztés (Nagy, 2008 alapján)

1. algoritmus: UR, LB, UL.
2. algoritmus: UR, LF, UL. 3. algoritmus: UR, UR, RR, UL, UL.

A *rövidítések* az angol szavak kezdőbetűiből állnak össze:

F (Front): a kirakóval szembenéző „front" oldal,
B (Back): a „front" oldallal szembenéző oldal, a kocka hátulja,
U (Up): a „front" oldal fölött elhelyezkedő oldal, a „top" oldal,
D (Down): a „top" vagy felső oldallal szembenéző oldal, a kocka alja,
L (Left): a „front" vagy szemközti oldal bal oldalára eső oldal,
R (Right): a „front" vagy szemközti oldal jobb oldalára eső oldal.

- ***1. szinten:*** Ezen a szinten már csak algoritmussal lehet helyrerakni egy sarkot. A heurisztika a szintnek megfelelő kezdőértéken túl, a sarkok „algoritmus lehetőségtől" való távolságától függően nőhet. Ezen a szinten már 4-5 különböző algoritmusra kell figyelnünk. A 6. ábrán látható példákon keresztül nézzük végig a kék – sárga - narancs sarokra vonatkozó öt különböző algoritmust:



6. ábra: Sarkok algoritmussal leírható helyei
Forrás: saját szerkesztés (Nagy, 2008 alapján)

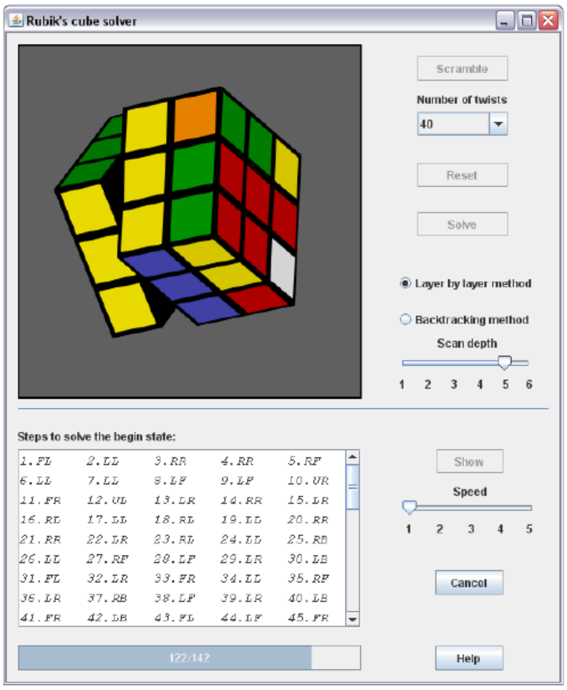
1. algoritmus: LF, LL, LB. 2. algoritmus: FL, LR, FR. 3. algoritmus: LF, LR, LB.
4. algoritmus: FL, LL, FR. 5. algoritmus: LF, LL, LB.

A megoldó program tehát a fent meghatározott 7 szint *MOHÓ* keresési programja révén oldja meg a kocka kirakását (9. táblázat). A fenti módszertani leírás értékelése során világos számunkra, hogy a program alkalmas a Layer by layer módszer alapján, bármilyen kiindulási állapotból eljutni a célállapotba, azaz a kirakott kockaállapotba. A forgatások száma a kiindulási állapottól függ, de rendszerint több mint 70 forgatás. Egyszerűbb kiindulási állapotból azonban itt is lecsökkenhet 40-45 forgatásra (7. ábra).

9. táblázat: A 3x3x3 –as Rubik kocka program alapú, Layer by layer kirakásának algoritmusai a MOHÓ keresővel (2., 3., 4., 5. szinteken)

Szint száma	Fázis	Algoritmusok
2.	A második sáv éleinek algoritmussal leírható pozíciói	1. algoritmus: FL, LL, FR, LB, FR, LF, FL. 2. algoritmus: LF, LR, LB, FR. LB. FL LF. 3. algoritmus: LF, LL, LB, LR.
3.	Élcsere érdemes állapot, élcsere a záró oldalon	1. algoritmus: LF, LL, LL, LB, LR, LF, LR, LB, LR.
	Élforgatás, záró oldal színhelyes állapotba forgatása	1. algoritmus: LB,RB, FL, LF, RF, LR, LB,RB,FL,LF,RF, LR, LB,RB,FL,LF,RF,LR
4.	Sarokcsere	1. algoritmus: LB, LL, RB, LR, LF, LL. RF, LR. 2. algoritmus: FR, LR, RR LL, FL, LR EL, LL
5.	Sarkok színre forgatása, rossz sarkok helyre rakása	1. algoritmus: RB, LL, RF, LL, RB, LR, LR, RF, LB, LR, LF, LR, LB, LR, LR, LF. 2. algoritmus: LB, LL, LL, LF, LL, LB, LL, LF, RB, LL, LL, RF, LR, RB, LR, RF.

Forrás: Saját szerkesztés Nagy, 2008 alapján



7. ábra: Rubik kocka megoldáskereső kiértékelő képernyője
Forrás: Rubik Kocka megoldáskereső program

A Rubik Kocka megoldáskereső program, szoftveralkalmazást azért tartottam lényegesnek ilyen részletességgel bemutatni, mert a megoldáskeresés folyamatában, szinte lépésről lépésre követi a kézzel történő forgatás mechanizmusát, a layer by layer módszer minden algoritmusát használja a megoldáskeresés közben, viszont más módszert nem alkalmaz.

A Rubik Kocka megoldáskereső program, megoldó szoftver SWOT értékelése (10. táblázat) a low-carbon projektértékelő modell input és output elvárásainak függvényében:

10. táblázat: Rubik Kocka megoldáskereső SWOT értékelése

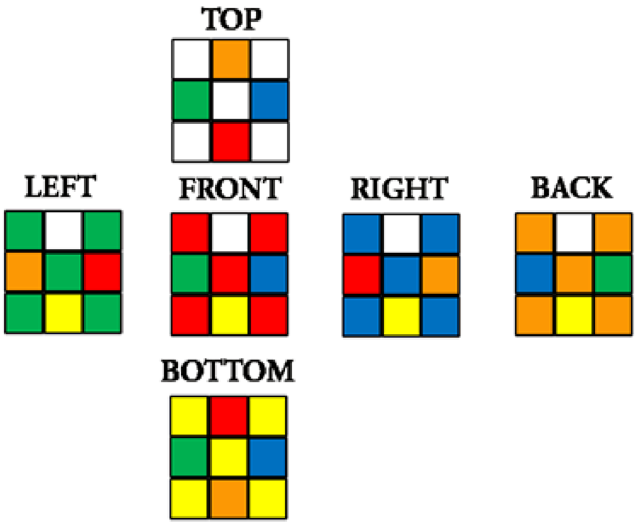
	SEGÍTŐ TÉNYEZŐK	GÁTLÓ TÉNYEZŐK
Belső tényezők	**ERŐSSÉG** Elméleti és gyakorlati kirakás lépései megegyeznek, Layer by layer megoldást végigvezeti a megoldó programban Világos előrehaladási és korrekciós lépésekkel dolgozik , könnyen fejleszthető, egyszerű programozási megoldás, minden algoritmus értelmezhető a low-carbon projektértékelő modellben is.	**GYENGESÉG** Vizuális felület korszerűtlen, viszonylag lassú feldolgozási képesség, online formában nem elérhető. Jelenleg csak a 3x3x3 –as kocka megoldásarása képes.
Külső tényezők	**LEHETŐSÉG** Vizuális felület kialakítása, a program könnyű összehangolása a SMAT kiértékelő szoftveralkalmazással, low-carbon értelmezési tartomány megadása nem igényel fejlesztést a szoftveren, olcsó piacra lépést jelenthet az egyszerű programozási megoldás miatt.	**VESZÉLY** Korszerűtlen, régi fejlesztés, a program lassú lehet, ha nem gyorsítható megfelelő mértékben az Állapottér konfigurációk miatt, könnyen „másolható” szoftver.

Forrás: saját szerkesztés

3.1.3. Rubiksolve program SWOT elemzése

A legismertebb online megoldó program az interneten. A fejlesztő Eric Dietz, gyerekkora óta foglakozik a Rubik kocka matematikájával, programozási lehetőségeivel. 2002-ben publikált, illetve osztott meg a Rubik „fun” közösséggel először kockamegoldó programsort. 2005-ben Kociemba 3x3x3 módszerét használta online programjának népszerűsítéséhez. 2007-ben készített egy olyan Solver programot, amelyet folyamatosan fejlesztett és csökkentette a megoldáshoz szükséges forgatások számát újabb és újabb algoritmusok beillesztésével. A jelenleg futó algoritmus 2010-ben véglegesítette, Kociemba féle algoritmust használ a Solver, így bármely állapotból történő készre forgatás 25 forgatásnál kevesebbet igényel. Eric Dietz a megoldáshoz mindig a Kociemba algoritmusokat használta fel, melynek egyik kiindulási megoldási algoritmusa követhető a 8. és a 9. ábrákon írta a fejlesztő a programhoz kapcsolódó linken (Dietz, 2010).

A program a csak a 2x2x2, 3x3x3 és 4x4x4 kockák megoldási algoritmusaira kínál megoldást, más Rubik játékokat nem találni a portfólióban. Kétdimenziós formában jelenít meg minden részletet, a felület nem igazán nyújt vizuális élvezetet. A forgatásokhoz kapcsolódó magyarázó ábrák nagyon egyszerűen értelmezhetők, az elmúlt években kockajátékosok tízezrei tanulták meg kirakni a Rubik kockát a Rubiksolve program útmutatásai alapján.

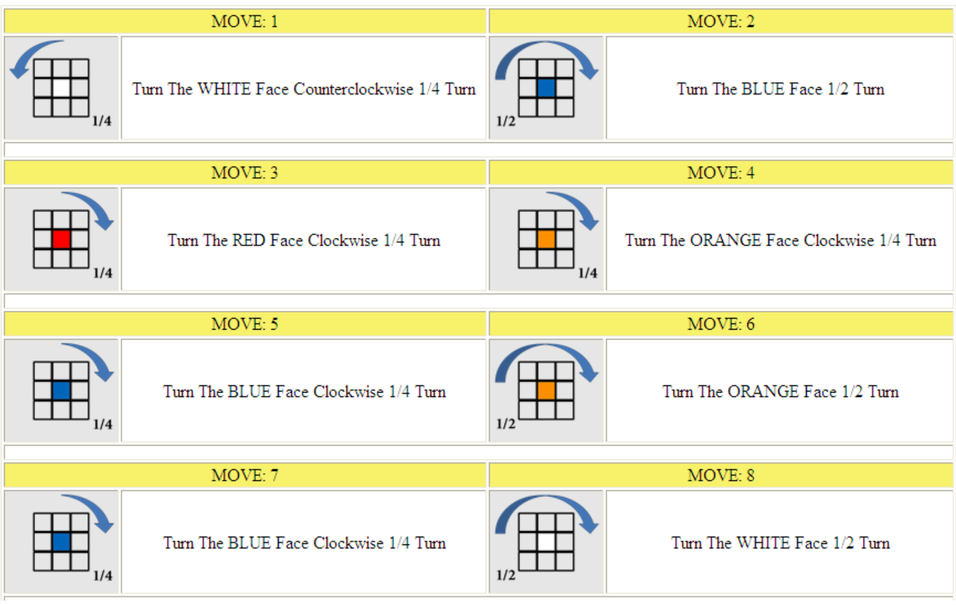


8. ábra: Az oldalak jelölése a program kirakó felületén (flip állapot)
Forrás: Dietz, 2010 alapján

A Rubiksolve programban a csökkentett algoritmusok száma miatt, nem ugyanazokkal a szintekkel találkozunk, mint az előbbiekben bemutatott Rubik megoldáskereső esetében. A Layer by layer módszert nem alkalmazza kirakó megoldásként a program, de természetesen egyes algoritmusok a módszerek között megegyeznek, így a különböző megodáskereső programokban is azonosak.

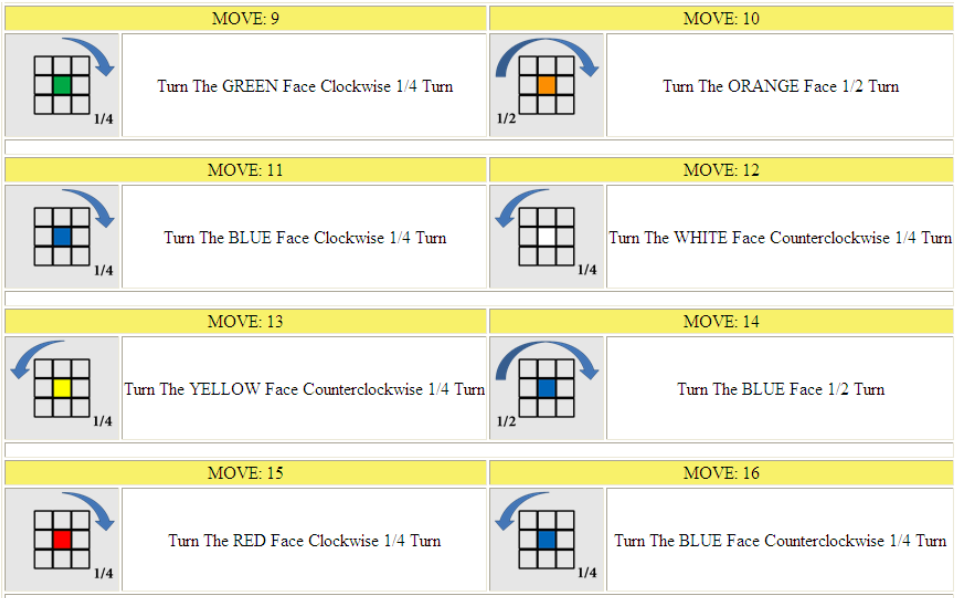
A program igen gyorsan dolgozik, a beírt kombinációk megáldási képletét néhány másodperc alatt írja fel a képernyőre. Ezzel ellentétben, a Ruwixnak és a Rubik megoldáskeresőnek is akár több tíz másodperc vagy akár percek is kellenek a megoldó képlet (8. ábra) kiírásához.

1. szint

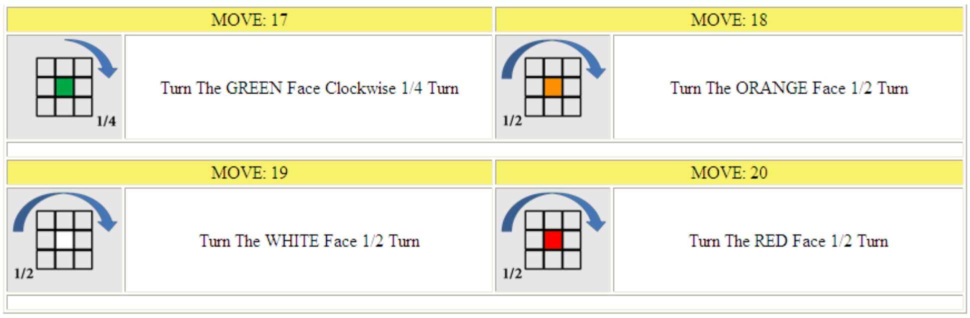


MOVE: 1		MOVE: 2	
1/4	Turn The WHITE Face Counterclockwise 1/4 Turn	1/2	Turn The BLUE Face 1/2 Turn
MOVE: 3		MOVE: 4	
1/4	Turn The RED Face Clockwise 1/4 Turn	1/4	Turn The ORANGE Face Clockwise 1/4 Turn
MOVE: 5		MOVE: 6	
1/4	Turn The BLUE Face Clockwise 1/4 Turn	1/2	Turn The ORANGE Face 1/2 Turn
MOVE: 7		MOVE: 8	
1/4	Turn The BLUE Face Clockwise 1/4 Turn	1/2	Turn The WHITE Face 1/2 Turn

2. szint



3. szint



9. ábra: A Rubiksolve megoldási képlete 20 lépésben levezetve
Forrás: http://mk2.rubiksolve.com/

A Rubiksolve megoldáskereső program, megoldó szoftver SWOT értékelése (11. táblázat) a low-carbon projektértékelő modell input és output elvárásainak függvényében:

11. táblázat: Rubiksolve megoldáskereső szoftver SWOT értékelése

	SEGÍTŐ TÉNYEZŐK	GÁTLÓ TÉNYEZŐK
Belső tényezők	**ERŐSSÉG** Gyors működés, folyamatos fejlesztés alatt áll, layer by layer módszert is tud használni.	**GYENGESÉG** Kétdimenziós megjelenítés, a Layer by layer elvek nem értelmezhetők az adatbevitelnél egyébfelhasználói funkciók hiányoznak .
Külső tényezők	**LEHETŐSÉG** A funkciók egyszerű bővítése jó lehetőséget kínál a low-carbon felhasználás kialakításához.	**VESZÉLY** Elsősorban a gyors megoldásra összpontosít, nem értelmezhető a működés minden részlete egyszerűbb felhasználók számára.

Forrás: saját szerkesztés

A bemutatott Ruwix solver és Rubiksolve megoldó program egyaránt annak a Kociemba *Cube Explorer* fejlesztésnek a továbbgondolt változatai, amely már 2005 óta alapvetően meghatározza a Rubik kocka rajongók fejlesztési munkáját, fejlesztési elképzeléseit. A megoldó programok áttekintését követően megállapíthatjuk, hogy alapvetően bármilyen algoritmus bevezetésére lehetőség nyílik a megoldó programok esetében, de természetesen minden fejlesztő arra törekedett az elmúlt évek során, hogy minél gyorsabb, kevesebb forgatással működő megoldó programot adjon a versenyzők kezébe. A Rubiksolve program esetében ez minden esetben 25 lépés alatt van.

A Rubik Megoldáskereső program a meghatározott 7 kirakási szint *MOHÓ* keresési programja révén oldja meg a kocka kirakását. A módszertani leírás értékelése során világos számunkra, hogy a program alkalmas a Layer by layer módszer alapján, bármilyen kiindulási állapotból eljutni a célállapotba, azaz a teljesen színre rakott kockaállapotba. Ugyanakkor bármelyik szinten megállítható a folyamat. A forgatások száma a kiindulási állapottól függ, de rendszerint több mint 70 forgatás. Egyszerűbb kiindulási állapotból azonban itt is lecsökkenhet 40-45 forgatásra.

A SWOT értékeléseket elemezve továbbá megállapítható, hogy gyors GYELV (gyengeségek/erősségek/lehetőségek/veszélyek) áttekintés eredménytáblája egyértelműen a Layer by layer algoritmusokra specializált, hazai fejlesztésű Rubik Megoldáskeresőt preferálja. A low-carbon projektértékelő modell input és output elvárásainak ez a Java nyelven íródott alkalmazás felel meg leginkább funkcionalitásában, mutatja ezt az a szerkezeti sajátosság is, hogy a kézzel történő kirakás algoritmusai is közel ugyanazokat a kirakási szinteket jelölik meg, mint a szoftver Állapottér elnevezésű csomagja. (A többi vizsgált kirakó szoftver esetében ezek a szintek teljesen eltérőek lehetnek.)

3.2. Layer by layer kirakási módszer elve és fenntarthatósági összefüggései

A 3x3x3 Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatában. A kirakási folyamatok és az azzal parallel beruház tervezési szinteket teoretikusan, a kocka egyes kirakási szintjei, állomásai szerint folyamatértékeltem. Az egyes színt-vizsgálatokat követően „Low-carbon interpretációkat", a kirakási lépésekhez illeszkedő projekttervezési összefoglalókat készítettem. A kocka egyes állapotainak és kirakási szintjeinek ábrázolásához, valamint a low-carbon értelmezések magyarázatához az Online Ruwix Cube Solver programot használtam fel.

Kocka lélek

Rubik Ernő azt írta 1980-ban, hogy a kocka olyan, mintha élő lenne, mintha forgatás közben életre kelne a kezedben. A Rubik kockának három sora és három oszlopa van, ezek szimbolikus vagy misztikus jelentéssel is bírnak. Ha megnézzük az egyes blokkok, a 3x3x3-as oldalak tulajdonságait, rögtön szembe tűnik, hogy olyan rendszerelemekről, vagy különböző sajátosságokkal bíró kiskockákról van szó egy-egy oldal esetében („középkockák", „élkockák", „sarokkockák"), melyek magukban rejtenek egy bizonyos jelentéstartalmat, és ezt a sajátos jelentéstartalmat viszik magukkal, bárhová is forgatjuk őket a rendszeren belül. Rubik Ernő szerint a 3-as szám különleges jelentésén keresztül, a kocka alkalmas modellezni az életet magát. Képes láttatni az ember és a természet viszonyát, az alkotás, megóvás és a rombolás folyamatát, és így a különböző erőforrás rendszereink egymással való együttműködésének viszonylatait (Rubik, 1981). Az gondoljuk, hogy a „bűvös kockajáték" kirakásának problematikája jól tükrözheti a jelen kor egyik legnagyobb kihívásának, a helyes és hatékony energiafelhasználásnak a legfontosabb kérdését is. Ma az egész energiafelhasználási rendszer egy óriási puzzle játéknak tűnik, mely esetében sehogy sem sikerül megtalálni a megfelelő darabokat. Azonban úgy ítéljük meg, hogy a 3x3x3-as Rubik kocka kirakási módszere talán segít megtalálni az egyes tényezők közötti valós kapcsolatot, a két vagy három dimenzióban is értelmezhető rendszertulajdonságok releváns számbavételét, és adhat jó megoldásokat az energiafogyasztás keresleti és kínálati oldalának fenntartható módon történő értelmezéséhez (Fogarassy, 2013).

„Layer by layer method" (Sorról-sorra módszer) a legismertebb, és az egyik legegyszerűbb módszer a Rubik kocka kirakásának, viszont tudni kell, hogy ez a módszer a legtöbb fejlett metódus (Fridrich, Corner first stb.) alapja is. Lényege, hogy sorról sorra rakjuk ki a kockát a megoldás során. Tehát először az első soron egy keresztet csinálunk, majd a sarkokat berakjuk, ezek után jön a középső sor-, végül az alsó sor él-, majd a utolsó sor sarokkockái (Fogarassy et al., 2012).

Az amatőrök között legtöbben a „Layer by layer method"-ot használják, mivel ezt a legkönyebb megtanulni, és ennek van professzionálisan kidolgozott algoritmusa. A többi fejlett kirakási módszer is mind ezen a módszeren fejlődött ki. A kirakás menetét „www.rubikkocka.hu" hivatalos oldalán is megtalálható leírás alapján ismertetem. A hivatalos kirakási módszer-leírást azonban kiegészítettem a fenntarthatósági alapelvekhez szorosan kötődő, UNFCCC által bevezetett fejlődési alapelvekkel, nevezetesen a LEDS - úgymint „Low-Emission and low-carbon Development Strategies" (OECD, 2010), azaz alacsony emissziós és szénfelhasználást segítő stratégiai gondolkodás alaplépéseivel is. Azt a feltételezést tettük, hogy mivel a Rubik kocka hármas számában rejlő mitikus rendezőelve sok, az életben megoldatlan kérdésre is tud indirekt válaszokat adni, feltételezhető, hogy a kocka kirakását ismerő egyén is tud „Rubik módon" gondolkodni a stratégiaalkotás vagy a gazdasági egyensúlykeresés kérdéseiben. A következő leírásban megtalálhatók azok a kockakirakáshoz kötődő módszertani lépések, amelyek egy stratégiai fejlesztés során (pl. fosszilis-megújuló energiaellátó rendszer cseréjének esetében), a kocka kirakása szerint, rendezőelvként vehetők figyelembe.

3.2.1. A 3x3x3-as Rubik kocka Layer by layer kirakásának folyamatértékelése

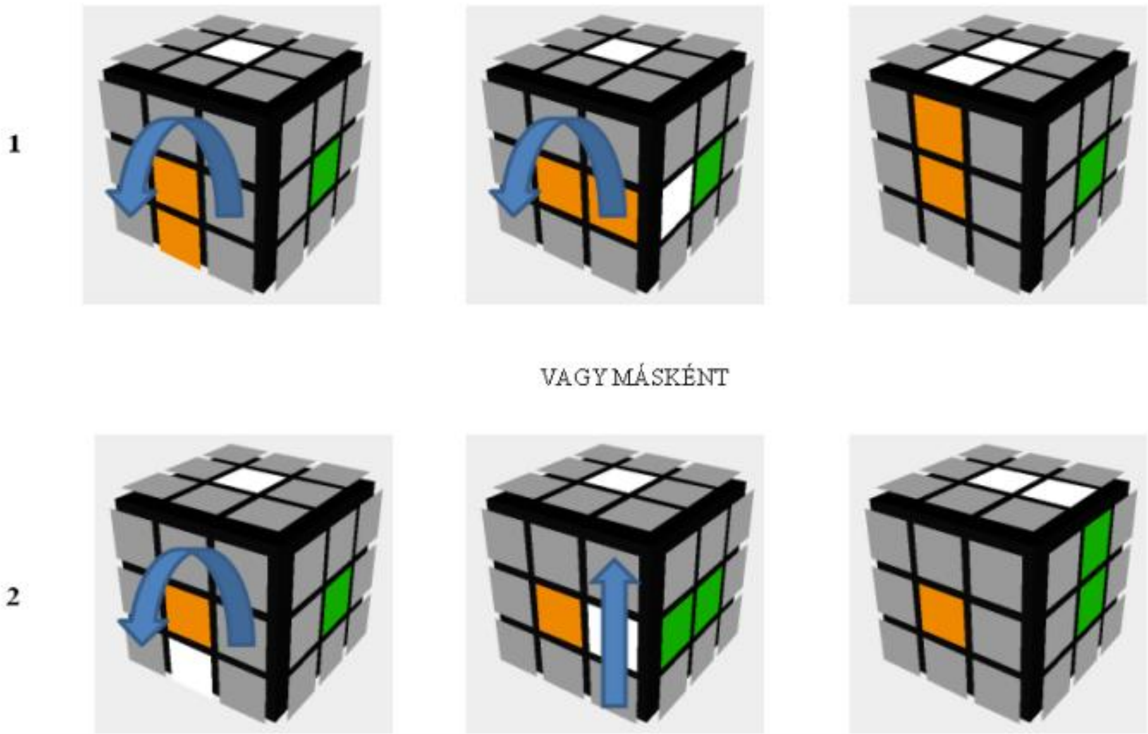
A Layer by layer módszer tulajdonképpen egy olyan strukturált rendezésnek tekinthető, amely mérföldköveket, állapotokat rendel a kirakás folyamatához (fehér kereszt, második sor kirakása, sárga kereszt stb.), ezek az állapotok többféle úton is elérhetők a kirakásokkal, sőt mindenki másként csinálja a saját kényelmi szempontjainak megfelelően, azonban az egyes állapotok, fázisok betartása nélkül a következő állapotba lépni nem lehet. A fenntarthatósági alapelvek és low-carbon fejlesztési koncepciók esetében azért van kiemelt jelentősége a fázisonként történő fejlesztési lépések betartásának, mert a körülmények, adottságok ugyan jelölhetnek ki eltérő utakat az egyensúlykeresés folyamatában, azonban a rendezőelvnek ugyanannak kell lennie, akárhol keressük is az egyensúlyi pontokat, legyen az Kínában vagy Magyarországon. A sorról sorra történő kirakás fázisainak leírása során a „www.rubikkocka.hu” hivatalos oldal módszertani leírásaira, valamint Singmaster 1980-ban közzétett kirakási rajzaira támaszkodtam. A folyamat azonban ezektől a leírásoktól jelentősen eltérő megjelenítésben kerül interpretálásra a a low-carbon módszertani átkötések miatt. A kocka egyes állapotainak és kirakási szintjeinek ábrázolásához az Online Ruwix Cube Solver programot használtam fel.

3.2.1.1. Fehér kereszt, a kiindulásai feltételek több szintű szinkronizációja

A „Layer by layer módszer” sajátossága, hogy mindig a fehér oldalt tekinti kiindulási színnek, illetve a fehér középkockát (az a kockaelem, amelyiknek csak egy színe van) a kiindulási pontnak. Természetesen bármelyik szín lehet a kirakási folyamat kezdő színe, ezért a forgatási logikát bármely színről indulva ugyanúgy lehet alkalmazni. Tehát miután a fehér középkockánk megvan, az első lépés során megkeressük egymás után azt a négy élkockát (élkocka az, aminek két színe van), amiben van fehér szín. Ezeket egymás után a fehér középkocka mellé forgatjuk. A többi kockát egyelőre bárhová forgathatjuk, azok nem kerültek végleges helyükre, tekintsük őket szürkének! Ha már minden fehér fent van, akkor a fehér oldal elforgatásával úgy állítjuk be ezeket, hogy közülük minimum kettő egyezzen az alatta levő egyszínű középkockával! Általános elvárás tehát, hogy kettő vagy mind a négy elem pontosan helyezkedjen el lefelé is, a 10. ábra szerinti módon. Ez a kocka kirakásának első lépése tehát az un. „Fehér kereszt” kirakásának folyamata.

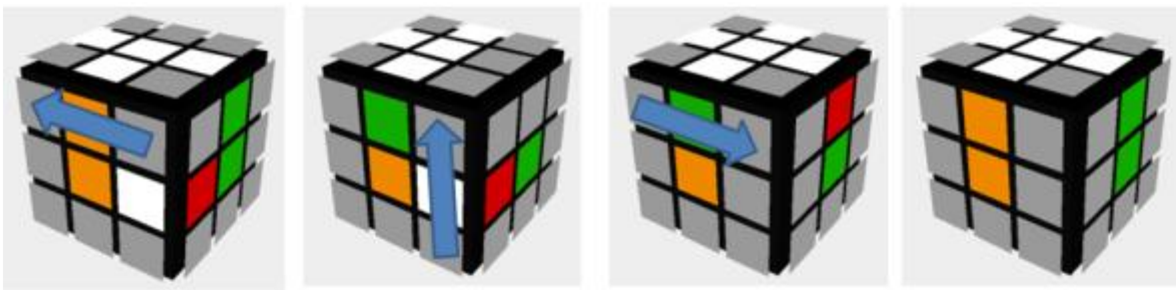
10. ábra: Fehér kereszt a helyükre forgatott oldalsó élkockákkal
Forrás: saját szerkesztés

Nagyon fontos, hogy a fehér kereszt úgy álljon az induló oldalon, hogy a középkockák mind a négy oldalra egyezzenek az átlókban. Ha nem ezt a pozíciót veszik fel a fehér élkockák, akkor a kirakással a módszer szerint nem lehet tovább haladni. A fehér élkockák felhozása az kiindulási oldalra többféle pozícióból és eltérő módon, de ugyanazt a logikai sort követve történhet. Leggyakrabban az alsó sorból kell felhoznunk az élkockákat a kiindulási oldalra. Alulról, a lenti sorból felfelé történő forgatás folyamata látható a 11. ábrán. A két különböző eset egy-egy speciális kockaállapotot jelenít meg. A 11. ábra felső sorában (1) alulról, egy 180º fokos fordítással visszük fel a helyére a kockát. A második sorban (2) egy 90º fokos felforgatást egy arra merőleges, másik 90º, az óramutató járásával megegyező elfordítás követ. A zöld-fehér élkocka így kerül a helyére.



11. ábra: Élkocka helyre forgatása az alsó sorból
Forrás: saját szerkesztés

Ha a fehér két elkészült él közé van beékelődve, akkor a 12. ábrán látható forgatást alkalmazzuk. Ez a forgatás első látásra rossz pozícióba viszi fel az élkockát, de aztán ezt könnyen a végleges helyére tudjuk fordítani.



12. ábra: Élkocka helyre forgatása a középső sorból
Forrás: saját szerkesztés

Ha a kiforgatott keresztből csak kettő stimmel a középkockával, akkor a másik két oldalt úgy cserélhetjük meg, hogy megkeressük hol van az az elem, amit meg akarunk cserélni a másikkal, majd azt az oldalt elfordítjuk kettővel, úgy hogy a fehér alulra kerüljön. Ezt követően az elemet odafordítjuk a saját színéhez, majd ezen az oldalon fordítunk kettőt. Most ebből a helyzetből, ami rosszul állt, az került le alulra. Ezek után ezt az elemet állítjuk a saját színéhez, és végül ezen az oldalon fordítunk kettőt, azaz 180º-ot (11. ábra 1. sor)! Ez a módszer működik akkor is, ha egymás mellett található két kockát kell kicserélni, illetve működik akkor is, ha kettőt átelleneset kell megcserélni. Ha mind a négy szín jó helyen áll (fehér + 4szín élkockák stimmelnek a négyszínű középkockákkal, amint az a 10. ábrán látható), akkor a következő lépésre, a fehér sarkok kirakására térünk át, de előtte nézzük meg, hogy a fenntarthatóság keresésének folyamatában mit is jelent ez a fázis.

„LOW-CARBON INTERPRETÁCIÓ" NO1:

Az Input oldal, azaz a kiindulási állapot peremfeltételeinek meghatározása, a teljes vagy részleges rendszer átalakítás (fosszilis energiaellátó rendszer teljes vagy részleges cseréje) határozza meg a célrendszerünket. Itt definiáljuk magát a fejlesztési programot, a keretfeltételeket, a projekt vagy feladat határait. Megfogalmazzuk azt, hogy milyen kapcsolati rendszerek befolyásolják az adott folyamat, projekt, koncepció létrehozását. Ez lesz a fehér középkockánk, ami a megváltozhatatlan célrendszert, azaz a kiindulási folyamat fix pontját jelenti. Jelen esetben szakértői álláspontok szerint az Energiaracionalizálást jelölhetjük meg fix pontként. Továbbá négy olyan viszonyítási pont kialakítása szükséges, melyek meghatározó hatást gyakorolnak a projektkörnyezetre. Ezek lehetnek a stratégiai alapkapcsolat, technológiai alapkövetelmény, finanszírozási elvárások valamint a piaci alapillesztés kétdimenziós értelmezései. Ezek az egyes élkockákkal és a meghatározó tényezőkhöz tartozó fix tulajdonságokkal (narancs, kék, piros, zöld középkockákkal) korreláló tulajdonságok adják a fejlesztés 2D-s kiinduló tulajdonságait.

Példa: Ha az energiaellátás rendszerét 100 %-ban, azonnal lecserélem (1. stratégia) az új tiszta technológiára, vagy megvárom, míg a régi rendszer életciklusa döntő mértékben lejár (2. stratégia), az két különböző stratégiai célt jelent. Az (1) verzióban azonnali és végleges beavatkozást indukálok jelentős költségekkel, a (2) verzióban a fosszilis-megújuló rendszerek cseréje fokozatos lesz, hosszabb ideig tart, a beruházás költségeit időben elnyújtva valósít meg teljes rendszer átalakítást. Ennek a folyamatnak az okszerűségét kell vizsgálni. Ha nem végezzük el a „régi-elavult" és „új-tiszta technológiai" rendszer működési feltételeinek szinkronizációját, akkor a kocka kirakása, vagy a projekt fenntartható tovább tervezése nem folytatható. Ekkor a projekt következő lépése nem teljesíthető, vagy ha mégis folytatódik, akkor rossz irányú fejlesztés valósulhat meg. A kiindulási alapokat (fehér szín kirakása a kezdő oldalon) tehát nem elegendő csak azokkal az egyértelmű szempontokkal megfogalmaznunk, amelyek első látásra alapvetően meghatározzák az indulási feltételeket, hanem a következő tervezési szint fix pontjaihoz is illesztenünk kell azokat. A gyakorlatban ez úgy értelmezhetjük, hogy a fehér oldal (vagy a projekt alapjai) úgy is kirakhatók, hogy azok nincsenek összhangban az első sorral vagy a második tervezési szint fix pontjaival, azaz a (narancs, kék, piros, zöld) középkockákkal. Ez a projekt és kockaállapot látható a 13. ábrán. A projekt ebből az állapotból nem lesz fenntartható, eleve kudarcra van ítélve a fejlesztési program.

13. ábra: Fehér oldal rosszul kirakva, avagy a projekt kiinduló állapota rosszul megtervezve
Forrás: saját szerkesztés

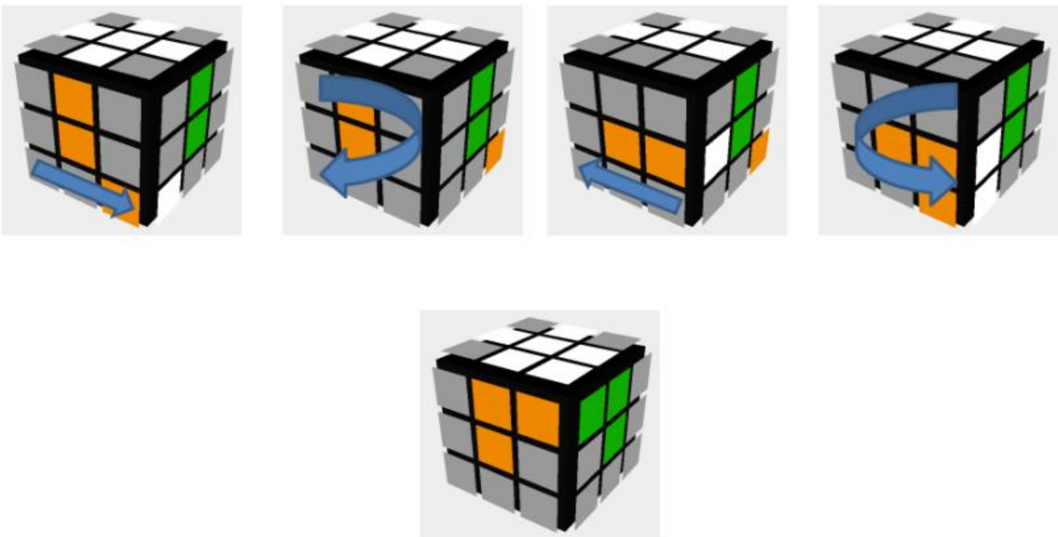
3.2.1.2. A Fehér sarkok kirakásának algoritmusa, egyensúlykeresés a kiindulási állapothoz

A fehér kereszt kirakása után a sarkok helyre rendezése a következő lépés (14. ábra). A kocka oldalának sarkaiba színhelyesen a sarokkockák illeszkednek. Sarokkockának azokat a kockákat nevezzük, amelyeknek három színe van (pl. fehér, narancs, zöld). A kockában összesen nyolc db sarokkocka található, a feladatunk tehát az, hogy a négy fehér színt tartalmazó sarokkockát kell átforgatnunk a fehér szemközti oldalra, a fehér kereszt sarkaihoz.

14. ábra: A fehér sarkok helye és az első sor kirakási képe
Forrás: saját szerkesztés

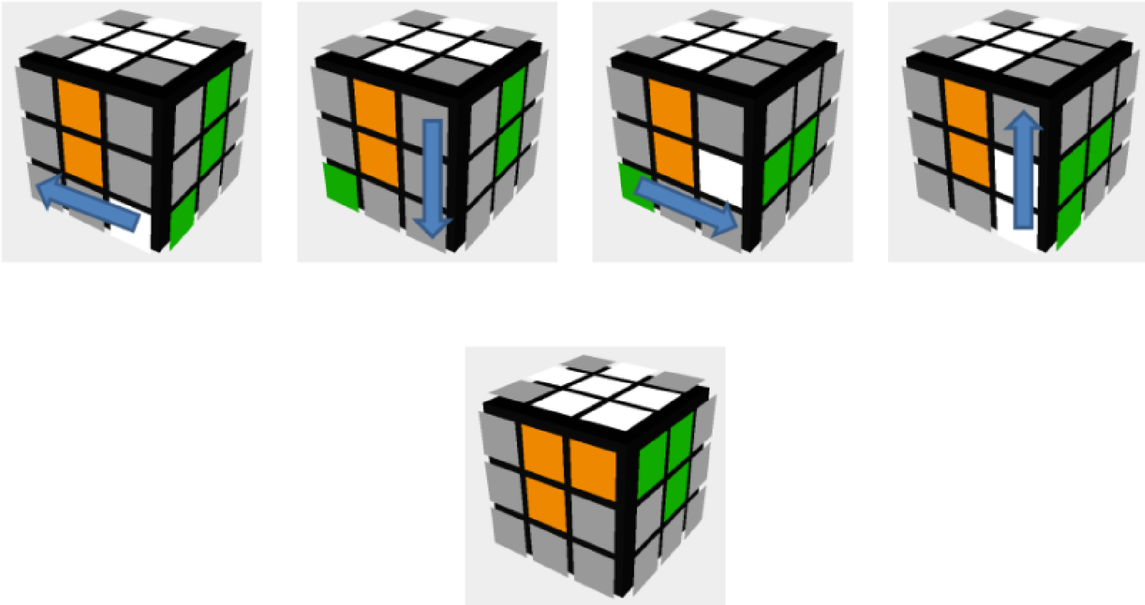
Először keressük meg a négy fehér sarokkockát, majd rakjuk fel a helyükre az alábbi (a) és (b) algoritmusok (forgatáskombinációk) segítségével. Mindkét forgatáskombináció (a) és (b) esetében célszerű a fehér keresztet úgy beállítanunk, hogy az felfelé nézzen. A legegyszerűbb kirakás következik akkor, ha a legalsó sorban találunk fehér sarokkockát. Először nézzük meg, hogy a fehér mellett milyen másik szint találunk. Ezt a színt állítsuk az alsó sor elforgatásával a lehető legközelebb a saját színű középkockához. Ekkor most ez a sarokkocka vagy jobbra, vagy balra esik a középtől. Amelyik irányba áll, abba az irányba visszük el az alsó sort, majd ugyanoda levisszük a szemközti oldalt. A forgatás zárásaként visszafordítjuk az alsó sort és végül visszafordítjuk a szemközti oldalt is. A két forgatáskombináció a 15. és 16. ábrákon látható.

(a) A sarokkockán a fehér szín jobbra áll, onnan kerül fel a fehér frontoldalra



15. ábra: Jobbos sarokkocka helyre forgatása az alsó sorból
Forrás: saját szerkesztés

(b) Megoldás, ha a sarokkockán a fehér szín balra áll, és onnan kerül fel a fehér oldalra



16. ábra: Balos sarokkocka helyreforgatása az alsó sorból
Forrás: saját szerkesztés

A (b) megoldás helyre forgatása rendkívül egyszerű, jól látható a 16. ábrán, hogy a sarokkockát csak el kell forgatnunk az „útból”, majd azt a kockarészt, ahová kerülni fog a sarokkocka, beforgatjuk arra a helyre, ahonnan a sarokkockát elvittük. A sarokkockát visszafordítva, majd a már egymás mellett található fehér sarok- és élkockákat felforgatjuk a frontoldalra. Ekkor a sarok a végleges helyére kerül.,m,m

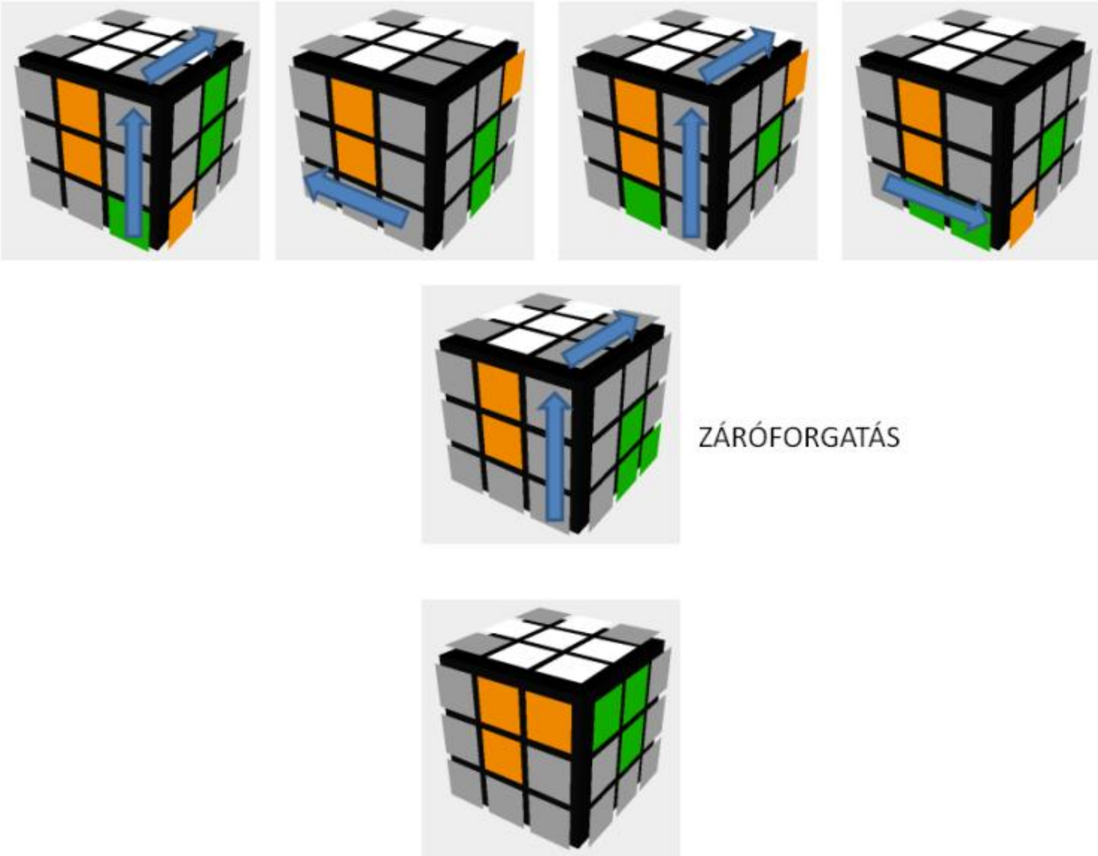
(c) Megoldás, ha a sarokkockán a fehér szín lefelé áll

Első pillanatra az a legnehezebb állás, ha a sarokkockánk fehér színnel lefelé néz. Ebben az esetben azonban egy 180 fokos forgatással a szín felhozható az induló oldalra, majd arra könnyen rá tudjuk rendezni a megfelelő élkockát (17. ábra). Abban az esetben, ha sarokkocka a felső sorba rosszul áll, akkor le kell vinni az alsó sorba és valamelyik korábbi forgatást alkalmaznunk. Többféle kombinációját alkalmazhatjuk az előzőekben leírt forgatásoknak, melyek közül a választás egyrészt egyéni térlátásmód, másrészt kézügyesség (bal kezes, jobb kezes) függvénye.

Ha a fehér színű kockák elfogytak az alsó sorokból, akkor a kezdő fehér oldal, az induló sorunk elkészült. De vigyázni kell, mert a kocka fehér oldala teljes lehet úgy is, ha hogy a sarokkockák első látásra a helyükön vannak, de oldalra nem egyeznek a színek. A sarokkocka fent lehet az első sorban úgy is, hogy a fehér oldal kifelé áll. Egyik állásból sem folytatható a kirakás a második sor építésével, mert később ideálisan nem forgathatók helyükre a rosszul álló kockarészek.

(d) Ha a sarokkocka fent van a felső soron, de rosszul áll, akkor több verziót használunk

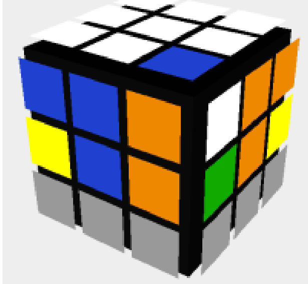
A gyakorlatban fordítsuk úgy a kockát, hogy az adott sarokkocka velünk szemben a jobb oldalon legyen, majd fordítsuk el a kocka jobb oldalát magunk felé. Ekkor a sarokkockánk teljesen lekerült az alsó sorra. Az alsó sort fordítsuk el az óramutató járásával ellentétesen, azaz visszafelé, majd a jobb oldalt fordítsuk magunktól kifelé. Ezzel a mozdulatsorral kihoztuk az (a), (b) vagy a (c) eset valamelyikét, amelyekkel helyre tudjuk rakni a sarkot!



17. ábra: Lefelé néző sarokkocka helyreforgatása
Forrás: saját szerkesztés

„LOW-CARBON INTERPRETÁCIÓ" NO2:

Projekt fenntartható fejlesztési irányának megadása, az induló állapot fix pontokhoz kötésének véglegesítése a cél. A meghatározó feltételek összehangolása és egymás kapcsolati rendszerének meghatározása a három tulajdonságot egyben meghatározó sarokkocka segítségével lehetséges elsősorban. Az összes szempont egymástól független, de mégis egymásra épülő szinkronizálásának folyamata a legrövidebb úton, de mégis a leghatékonyabb módon valósulhat meg. Fontos megjegyezni, hogy a sarokkocka fent lehet az első sorban úgy is, hogy a fehér oldal kifelé áll. Ez az állapot látható a 18. ábrán.



18. ábra: A felső sorban található sarokkocka helyén van, de kifelé néz
Forrás: saját szerkesztés

Ebből az állásból sem folytatható a kirakás a második sor építésével, mert később ideálisan nem forgathatók helyükre a rosszul álló kockarészek. Az eset jól mutatja, hogy a projektfejlesztés folyamatában is találkozhatunk olyan projekt tényezővel, amely első látásra megfelelő helyen van, de mégsem az egyensúlyi állapotban. Nem tudjuk tovább építeni programunkat, vagy ha mégis tovább haladunk vele, akkor a fejlesztés rossz irányú lesz. Ennek a kiindulási egyensúlypontnak a keresése történik a kirakás és projektfejlesztés jelen ciklusában. A keresett egyensúlyi állapotot Nash-egyensúlynak nevezzük. A függvény felírása az első layer tervezése során, pl. a szabályozáspolitika és a finanszírozás politika leírása esetében alkalmazható a projekttervezés folyamatában. A kooperatív játékok esetén az egyensúlyi állapot még akkor is stabil lehet, ha egy stratégia-kombináció nem Nash-egyensúly, amennyiben a játékosok egyezségre jutnak, hogy ezt a stratégia-kombinációt választják.

A Nash-egysúlypontra vonatkozó *Definíció* szerint:

Egy $J = (n, S, (\varphi_i)_{i=1}^{n})$ n személyes játék egyensúlyi pontján vagy a stratégiáján olyan $(x_1^*, \ldots\ldots, x_n^*) \in S$ pontot (stratégiai n-est) értünk, melyre a kifizetőfüggvény

$$\varphi_i(x_1^*, \ldots, x_{i-1,}^* x_i^*, x_{i+1,}^*) \geq \varphi_i(x_1^*, \ldots, x_{i-1,}^* x_i, x_{i+1,}^*)$$

teljesül minden $i = 1, \ldots\ldots, n$ játékosra. Az egyensúlyi pontot tehát *Nash-féle equilibrumnak* nevezzük. Az első layer kirakását követően csak a Nash-féle egyensúlyi ponttal rendelkező kapcsolat építhető tovább, azaz a kocka kiforgatását csak ebből az állapotból építhetjük tovább. Az első layer mindig korrelál a második layer középkockájához, tehát csak színazonos lehet. Az első layer és a középkocka valódi egyensúlypontját nevezhetjük Nash-egyensúlypontnak.

Példa: A célrendszerhez kapcsolódó technológiai fejlesztés és pénzügyi hatékonysági keretfeltételeinek, alapösszefüggéseinek összeillesztése direkt módon, illetve a szabályozórendszernek (szabványok, normák) való megfeleltetés indirekt módon, történhet meg a három szempontra kiterjedő beforgatás segítségével. Jó példa lehet az amerikai szabványok nem megfelelősége az európai felhasználói környezetben, a lokális termékbeszerzés preferenciája a globális beszerzéssel szemben elv érvényesítésére itt helyénvaló és fenntartható egyensúlypontot jelent.

3.2.1.3. A második sor kirakása az élkockák helyreforgatásával (3 algoritmus felhasználásával)

A 19. ábrából egyértelmű, hogy az első sor elkészítése után, a középkockák is a helyükön lesznek, így alapvetően az illeszkedő élkockák megfelelő helyre forgatása a következő forgatások feladata. Az első sor kirakási technikájával összevetve a most következő algoritmusokat, azt kell mondanom, hogy hosszabb forgatássort kell alkalmazni a megoldás során, amely alapesetben 7 db forgatást feltételez egy-egy kocka helyreforgatásánál. Érdekes, hogy a második sor kirakása sokkal inkább automatizálható (pl. szoftverrel), mint az első sor helyreforgatása. Itt nem okoz gondot a heurisztikus algoritmusok alkalmazása, minden állapotra tudunk illeszteni néhány fix algoritmust, csak azt kell eldöntenünk, hogy melyiket alkalmazzuk előbb.

19. ábra: Két teljes sor kirakva az élkockák helyreforgatásával
Forrás: saját szerkesztés

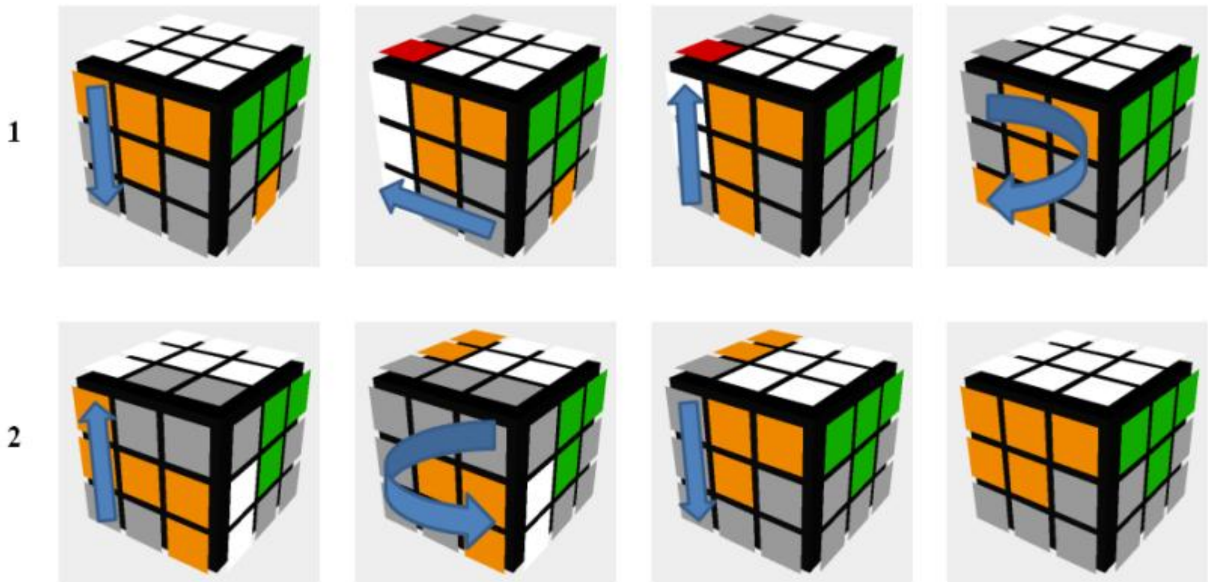
Az élkockák beállításával tehát elkészül a teljes második sor. Az élkockák elhelyezkedésére három lehetőség (*a), b), c),*) áll fent, melyeknek a következő megoldásai léteznek:

a) és *b)* megoldások esetében, a kocka alsó oldalán keresnünk kell egy olyan élkockát a sárga közép mellet, amelyekben nincs sárga szín. Ennek oka, hogy azokban az élkockákban, amelyekben nincs sárga, az mindegyik a középső sorba tartozik. Ha megtaláltuk a középső sorba tartozó élkockákat, akkor egyenként állítsuk a saját színéhez, ami azt jelenti, hogy húzzuk a középkockája alá közvetlenül. Ha ezt az oldat magunkkal szemben tartjuk, akkor nézzük meg, hogy az élkockának melyik a másik színe. A párosító szín így nekünk jobbra (20. ábra) vagy balra (21. ábra) fog állni.

A középkocka és az alsó kocka színe azonos lesz, a következő lépésben pedig megnézzük, honnan hiányzik ez a bizonyos élkockánk. (Az a szín nekünk vagy jobbra, vagy balra kell, hogy álljon!) Attól a színű középtől, ami az alsó színünk bizonyos élkockája, attól elfordítjuk az alsó oldalt az ellentétes irányba! Miután realizáltuk, hogy hová kell beforgatnunk az élkockát, akkor ezt az oldalt lefordítjuk magunk elé, illetve az élkockánkat visszafordítjuk az eredeti állásba. Ekkor marad a szemben lévő oldalon két fehér szín, ezeket felrakjuk a többi fehérhez!

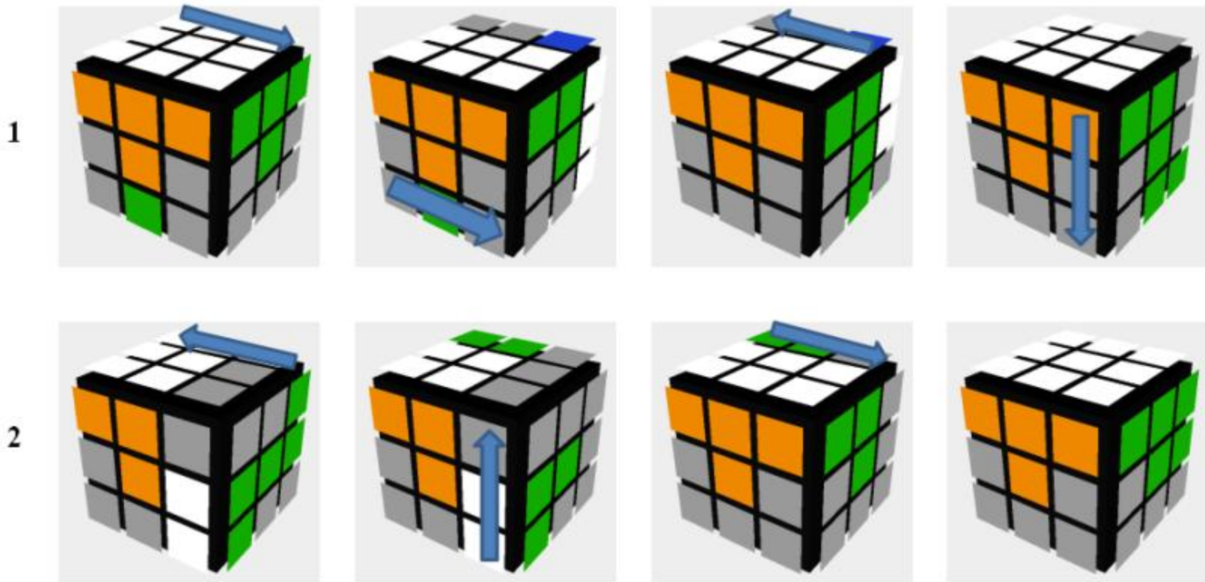
Ha most megnézzük a kockát, azt láthatjuk, hogy a szemközti oldalon lent lévő sarokkocka (amiben van fehér) „összepárosodott" a hozzá tartozó élkockával (azzal, amit eredetileg kinéztünk). Ebből a helyzetből már egyszerű dolgunk van, a sarokkockát berakjuk a saját helyére ugyanúgy, mint ahogy az az előző részben (Fehér sarkok berakása) le volt írva.

a) Jobbról történő beforgatás folyamata (20. ábra)



20. ábra: Élkocka helyre forgatása, ha a hiányzó kocka jobboldalra áll
Forrás: saját szerkesztés

b) Balról történő beforgatás folyamata (21. ábra)



21. ábra: Élkocka helyre forgatása, ha a hiányzó kocka jobboldalra áll
Forrás: saját szerkesztés

c) az élkocka fent van a második sorban, de rosszul, vagy rossz helyen (22. ábra)

A *c)* megoldás alkalmazására tehát azért kerülhet sor, mert az élkocka jó helyen van ugyan, de például színben elfordulva helyezkedik el. Ebben az esetben csináljuk végig az *a)* vagy a *b)* forgatáskombinációt, amivel azt érjük el, hogy az élkocka, ami eddig a második sorban rosszul volt beállítva, az most lekerül az alsó sorra, ahonnan be tudjuk tenni majd a helyére az *a)* vagy a *b)* algoritmus segítségével.

„LOW-CARBON INTERPRETÁCIÓ" NO3:

A fenntartható projekttervezés folyamatában a középső sor élkockáinak helyreforgatásával az a célunk, hogy az egyes befolyásoló tulajdonsághalmazok egymás közötti kapcsolatrendszerét tovább rendezzük, találjuk meg azokat az egyensúlypontokat, amelyek mentén az egymásra közvetlenül ható tényezők, azaz az élkocka egyik és másik színe által hordozott tulajdonságok, valamint a középkocka másik oldalán elhelyezkedő szín azonos, de más színnel párosodó élkocka tulajdonságai jelenthetnek. Az indirekt módon egymásra ható tényezők, illetve az általuk reprezentált tulajdonságok összehangolása nélkül az egyensúlyi helyzet nem optimális (mert több egyensúlyi helyzet vagy egyensúlypont is létezik). Ezt az állapotot le tudjuk írni a korábban már ismertetett többváltozós folytonos függvényekkel.

A φ_i két célt kifizető függvényeknek, az $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, vektorokat pedig stratégia vektoroknak tekintve egy folytonos kétszemélyes játékot definiálhatunk, legalább két egyensúlyponttal.

$$\varphi_i(\boldsymbol{u}) = \varphi_i(\boldsymbol{u}_1, \boldsymbol{u}_2,)$$

Több egyensúlypont kialakulásának a legfőbb oka, hogy az egymásra ható tényezők több ponton is optimalizálódhatnak (az élkockát, vagy az azt reprezentáló tulajdonságot tudjuk balra és jobbra is optimalizálni, külön-külön, de csak akkor megfelelő az egyensúlyi állapot, ha ezekből a pontokból tovább is tudunk lépni, azaz a kocka kirakása folytatható). A rossz egyensúlypont kialakulásának kockaképe látható a 22. ábrán.

22. ábra: Élkocka helyre forgatása, ha a hiányzó kocka jobboldalra áll
Forrás: saját szerkesztés

Példa: a leggazdaságosabb technológiai megoldásokat közvetlenül össze tudjuk hangolni a magas minőséggel és innováció-tartalommal, de piaci változások hatása a finanszírozási rendszerre (kamatváltozás), deviza kockázati tényezők és globális hatások elemzése (indirekt hatásként) nem került figyelembe vételre, akkor a projekt nem, vagy csak jelentős újrarendezéssel, jelentős változtatások (innováció nélkül vagy alacsonyabb minőségben) árán valósítható meg.

3.2.1.4. Sárga kereszt kirakásának algoritmusa és az output oldal hangolása

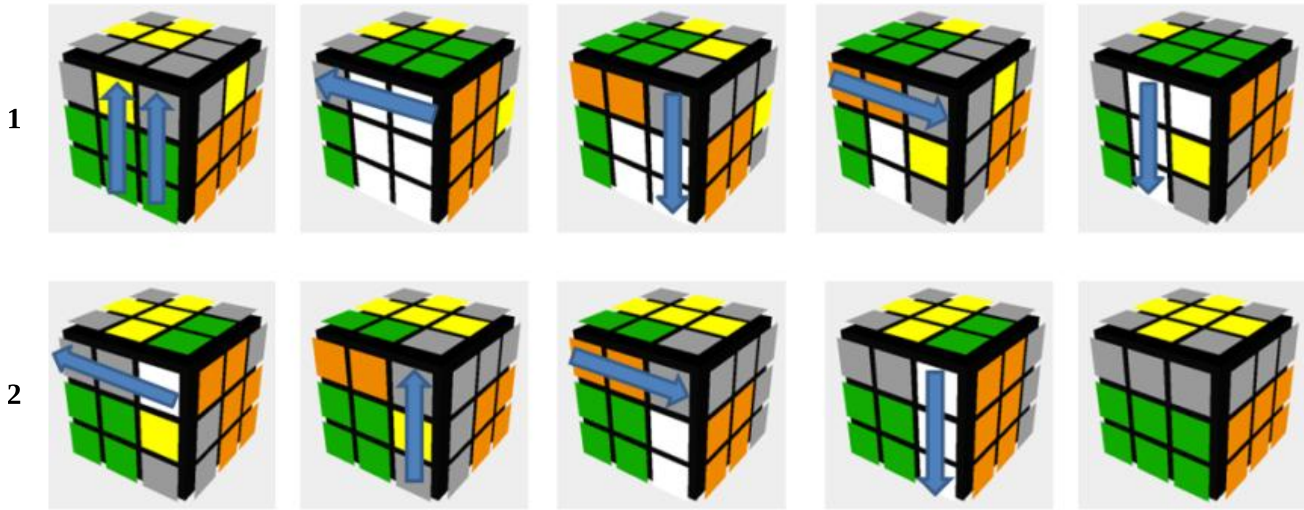
A „sárga kereszt" kiforgatása az egyik legfontosabb fázis a kocka kirakásának befejezése előtt. Ezzel a forgatással kezdjük össze hangolni a fehér oldalt és a sárga oldalt. A forgatás végén a sárga színű élkockák színnel kifelé kerülnek frontoldalra. A „sárga kereszt" kirakása esetén nem szempont, hogy a sárga élkockák színhelyesek legyenek, azaz oldalra nem egyeznek meg a középkocka színeivel (23. ábra).

23. ábra: A „sárga kereszt" kockaképe
Forrás: saját szerkesztés

Helyreforgatás során a két nem színhelyes kockarészt szemben és jobbra tartjuk magunktól (24. ábra) úgy, hogy a sárga közép felfelé nézzen! Alulról felhozunk egy hatos blokkot a szemközti oldalra úgy, hogy a velünk szemben lévő oldalon csak a bal sáv ne legyen fehér! Ebből a hatos blokkból csinálunk egy fordított L betűt (24. ábra 1. sor utolsó kocka). Ennek menete, hogy elfordítjuk a sor tetejét az óramutató járásával megegyezően, majd a jobb oldalról a két fehér színt letesszük alulra és végül visszafordítjuk a tetejét.

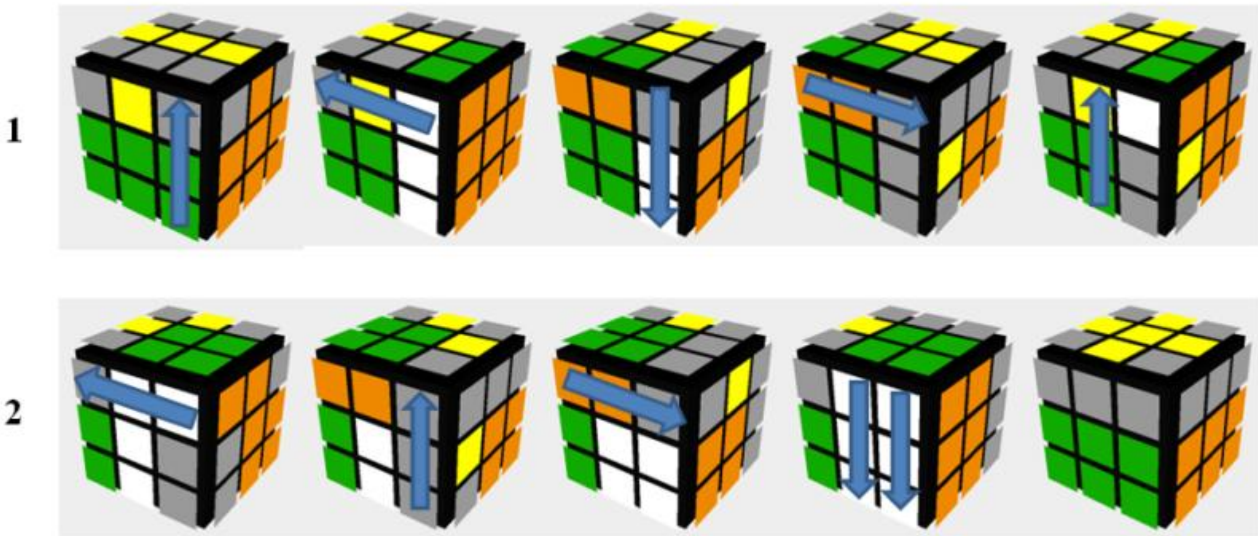
Ahogy megvan a fordított L-betűnk, levisszük az L-betű szárát (középső sort) alulra, majd visszafordítjuk a kockát úgy, hogy a fehér szín felfelé nézzen, végül balról behúzzuk a hiányzó sarkot és lefordítjuk a kirakott oszlopot.

a) ha két egymás melletti élkocka nincs megfelelő helyen, a forgatási rend a következő



24. ábra: Élkockák helyre forgatása a sárga oldalon
Forrás: saját szerkesztés

b) ha két egymással átellenes oldalon találjuk az élkockákat, a forgatási rend a következő



25. ábra: Élkockák helyre forgatása a sárga oldalon, ha a kockák átellenesen vannak
Forrás: saját szerkesztés

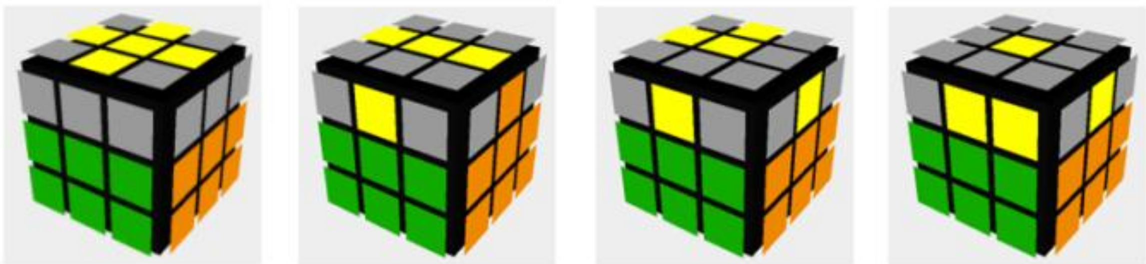
A kirakás menete az, hogy a két rossz helyen lévő kockát magunkkal szemben és a túloldalt tartjuk, ez látható a (b) változatban (25. ábra). Jobb oldalon felhozunk egy hármas fehér sort, elfordítjuk a tetejét (óramutató járása szerint), majd levisszük a jobb oldalról az ott maradt két fehéret. Visszafordítjuk a felső sort és felhozunk a középső sor hátrafordításával a három fehér kockát. Ebben az esetben egy fordított L betűt kapunk. Ezt az L betűt ki kell egészítenünk egy hatos blokká. Ez úgy valósítható meg, ha elfordítjuk a felső oldalt (az óramutató járása szerint balra), felhozunk a jobboldal hátrafordításával a két fehéret a szemközti oldalra, majd visszafordítjuk a felső oldalt. Az elkészült hatos blokkot helyre kell forgatnunk lefelé a többi fehérhez.

c) nincs sárga élkocka a frontoldalon

Előfordulhat, hogy nem találunk sárga színt tartalmazó élkockát a frontoldalon. Ebben az esetben végigforgatjuk az *a)* vagy a *b)* algoritmust, aminek következtében egy vagy két élkocka felkerül a frontoldalra. Ezután vagy az *a)* vagy a *b)* forgatás algoritmusait használjuk az élek helyrerakásához.

„LOW-CARBON INTERPRETÁCIÓ" NO4:

A sárga kereszt kirakása önmagában az output elvárások (sárga oldal) összehangolása az input oldal (fehér oldal) és fejlesztési célrendszer minden részletével Itt elsősorban az input és output trendvonalak indirekt módon történő összehangolása a cél. Az indirekt összehangolás azért fontos, mert ebben a fázisban még lehetőség nyílik bizonyos korrekciókra, vagy rugalmas szempontok kisebb mértékű módosítására az egyensúlypontok rendeződésének függvényében. Az indirekt rendezést az teszi lehetővé, hogy a sárga kereszt kirakása során, a felső sort élkockáit nem hangoljuk össze a középkockákkal, azok nem színhelyesen találkoznak a forgatási fázis végén. A második sor kirakását követően, a sárga élkockák különböző helyzetben lehetnek a legfelső sorban. Ha a sárga közép kivételével nem találunk sárga színt a frontoldalon (26. ábra „D" állapot), akkor a helyreforgatás hosszabb időt vesz igénybe, mivel egy algoritmus úgy kell végigforgatnunk, hogy alapvetően nem haladtunk előre a kirakással, csak átrendezést hajtottunk végre. Az átrendezés követően kezdhetjük a kijelölt algoritmus alkalmazását. A fenti körülmény világosan jelzi számunkra, hogy találhatunk olyan helyzetet, mikor a kocka záró oldala a vártnál rendezetlenebb állapotban van, mert egyik élkocka sincs a helyén. A projektfejlesztés során is előfordulhat az a helyzet, hogy a projekt kimeneti elvárásait át kell rendeznünk a korábban tervezett kimeneti elvárásokhoz képest. Ez a helyzet könnyen előfordul, mivel a gyakorlati megvalósítás során sokszor találkozhatunk olyan helyzetekkel, hogy egy adott beruházás, projektterv megvalósítása elhúzódik, több hónapot vagy akár éveket csúszik, így a gazdasági környezet (piac, szabályozás) alapvető változásokat generál az elvárások terén is. A 2010-es évek gazdasági újrarendeződési folyamatának egyik nagyon jellemző beruházási momentuma volt ez a jelenség, mely nemcsak hazánkban, de a Világ több pontján is okozhat, okozott kudarcba fulladt „giga" fejlesztéseket (kínai szellemvárosok, sikertelen európai etanol és biodízel üzemek stb.). A „sárga kereszt" kirakásának algoritmusa ezért méltán számíthat kiemelt figyelemre a gyakorlati hasznosíthatóság területén.



A B C D

26. ábra: Az élkockák lehetséges állapota a középső sor rendezését követően
Forrás: saját szerkesztés

Példa: A flexibilis technológia feltételek esetleges megváltoztatása a tervezett rendszerhez képest, ebben a fázisban még lehetséges úgy, hogy az output kritériumok és egyensúlypontok ne változzanak. Ugyanilyen tényező lehet az adózási és egyéb pénzügyi feltételekhez történő, még rugalmasan kezelhető változások figyelembe vétele is. Alapvetően azt feltételezem, hogy egy jól tervezett és hosszútávon kiszámítható gazdasági környezetben olyan output feltételek alakulásával számolhatunk, amelyek közel állnak az eredeti üzleti tervben megfogalmazott elvárásokhoz, azok új egyensúlyi állapotba rendezésére nincs szükség. A kocka logikát követve, amennyiben a „sárga kereszt" a második sor kirakását követően azonnal a frontoldalon van, akkor könnyen befejezhető a kocka kirakása, mivel csak a sarokkockák helyreforgatása maradt hátra. A projektfejlesztés folyamatában is ez az állapot feltételezhető, ha a projekt output elvárásai a „sárga keresztet" formázzák, akkor változtatások nélkül befejezhető a projekt vagy beruházás (26. ábra „C" állapot). Ha projektfejlesztés folyamatában a 26. ábrán látható „B" vagy „C" állapot jellemzi a befejező fázist, akkor a projektet új egyensúlyi állapotba kell hozni, egy közepes szintű beavatkozás, újratervezés szükségszerű. Abban az esetben azonban, mikor a projektfejlesztés állapotára a 26. „D" kockaállapot írható fel, azaz tulajdonképpen egy kimeneti tényező sem a korábbi elvárások szerint alakul, nagyon komoly újratervezés, egyensúlyátrendezés szükségszerű, amely esetek nagy részében hosszabb időt vesz igénybe (plusz egy vagy két algoritmus alkalmazása), és a projekt befejezését késleltetheti.

3.2.1.5. Sárga sarkok kirakása és fenntarthatósági kritériumok rendezése kimeneti állapotra

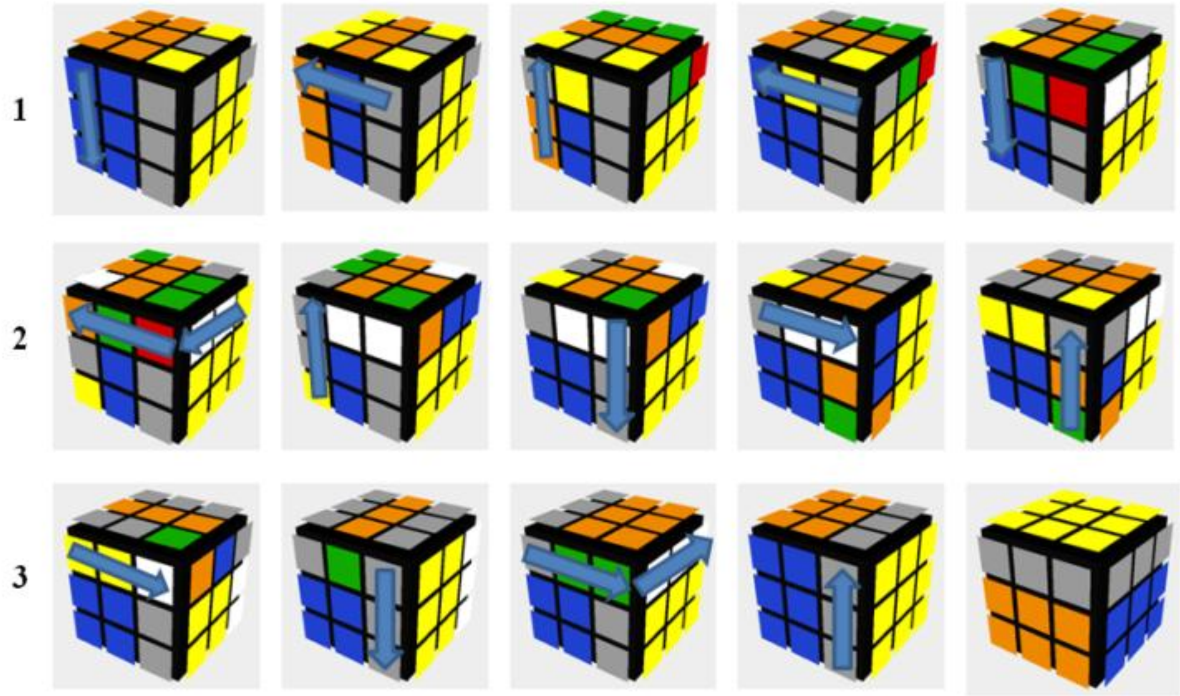
Ebben a forgatási kombinációban az összes sárga sarokkockát (négy db) megforgatjuk egy helyben, mégpedig úgy, hogy a felső sárga oldal nem lesz színhelyes az alatta lévő sorral.

27. ábra: A sárga oldal független kirakása
Forrás: saját szerkesztés

Erre a forgatásra nagyon sok variáció/algoritmus készült a elmúlt években, így ezek felsorolása jelen esetben szükségtelenül és nagyon hosszan történhetne csak meg. Annak érdekében, hogy a sarkok helyreforgatását el tudjuk végezni, elegendő egy egyszerűbb forgatási kombinációt meghatároznunk, melynek többszöri ismétlésével bármelyik kockaállásból eljuthatunk a kívánt sárga oldal kirakásához.

A 28. ábrán, illetve ábrasoron azt az esetet láthatjuk, amikor csak két sarokkocka áll rossz helyen, és azok is úgy állnak, hogy a két sárga szín egy oldalon van. A kockát mindig úgy kell tartani, hogy a két megfordítandó sarokkocka jobb oldalra kerüljön. Arra is figyeljünk, hogy az az oldal nézzen felfelé, ahol a két megfordítandó sarokkockán a sárga szín kifelé van. Kezdő lépésként fordítsuk magunk felé a bal oldali oszlopot, majd fordítsuk el a tetejét (óramutató járása szerint). Ezek után a baloldalt hátrafelé, és újra a tetejét, majd a bal oldalt fordítsuk lefelé, és a tetején fordítsunk kettőt. Befejezésként fordítsuk a bal oldalt ismét fel. Ezt kell most megismételnünk a jobb oldallal is. Abban az esetben, ha a két egymás melletti sarokkockán a sárga szín nem egy oldalon van, mint a fenti kiindulásnál, hanem egymással átellenesen, akkor is ugyanezt az algoritmust alkalmazzuk, de a kockát itt úgy kell tartanunk, hogy a sárga oldal legyen fent és a megfordítandó sarkok jobbra álljanak. Minden más esetben két lépésben tudjuk megoldani a sárga sarkok megforgatását.

Abban az esetben, ha három sarok helyezkedik el rosszul, azaz kifelé néz a frontoldalról, akkor is ezt a forgatáskombinációt alkalmazzuk. Azzal a „rossz" sarokkal kezdjük el a forgatáskombinációt, amelyik a jó helyen álló sarokhoz a legközelebb van. Ennek a forgatásnak az eredményeként a következő sarokkocka is felkerül, vagy befordul sárga színnel a frontoldalra. Így a 28. ábrán látható álláshoz hasonlót, vagy olyan állapotot kapunk, amelyben a két rosszul álló sarokkocka egymás mellet, egy oldalon helyezkedik el. A 28. ábrán látható forgatáskombinációt alkalmazva ebből az állásból már könnyedén meg tudjuk oldani a helyreforgatást.



28. ábra: Sárga sarkok helyre illesztése, a fenntarthatóság feltételeinek biztosítása
Forrás: saját szerkesztés

„LOW-CARBON INTERPRETÁCIÓ" NO5:

Az alsó (sárga) oldal sarkainak kiforgatása során az output elvárások rendezésére kerülhet sor. A sárga kereszt kirakásával egy olyan egyensúlyi állapotba került a rendszer, amely világos feltételeket jelent a „fogyasztói" oldal, vagy megrendelő, esetleg politikai döntéshozó számára is. Az output oldal tulajdonságainak véglegesítése a sarkok helyrerakásával valósul meg. Azt feltételezem, hogy a fenntartható üzleti stratégiák egyik kulcsa, ha projekt vagy beruházás úgy rendeződik a piaci feltételekhez, hogy azok legalább négy stratégiai célrendszer szerint vannak összerendezve. Ez könnyen megtehető a négy sárga sarokkocka segítségével. A négy sárga sarokkocka összesen 12 tulajdonságot hordoz, amely a teljes kocka vonatkozásában jelentős részhalmazt képez. A kocka összes oldalával 54 különböző tulajdonságot tudunk leírni, melyek közül a sarokkockák a különböző színekkel egyenként 3-3 tulajdonságot hordoznak a rendszerben. Ez azt jelenti, hogy ez az egy forgatási algoritmus 22%-ban határozza meg a tulajdonságok rendezettségi szintjét. A Rubik kockához kapcsolódó többdimenziós problémakezelés elmélete ugyan a következő fejezetben kerül bemutatásra, de már ebből az egyszerű összefüggésből is látszik, hogy vannak olyan rendszerelemek (kockák/tulajdonságok), amelyek helyzete döntő mértékben befolyásolja az egész állapottér egyensúlyi rendszerét. A szakirodalmi elemzésben bemutatott, játékelméleti megoldásokkal történő egyensúlypont keresés ebben az esetben is szükségszerű lehet, amennyiben a sarokkockák nem a megfelelő pozíciót foglalják el. A projektfejlesztéssel párhuzamba hozható egyensúlypont keresés a gyakorlati megvalósítás során úgy képzelhető el, hogy a sarokkockához nevesített tulajdonságok (3db) egyensúlyi helyzetét keressük az állapottérben. Ennek függvényállapotát a következőképpen írhatjuk le.

A φ_i három célállapotot optimalizáló kifizető függvényeknek, az $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, $\boldsymbol{u}_3$ vektorokat pedig stratégia vektoroknak tekintve, egy folytonos háromszemélyes játékot definiálhatunk, legalább három egyensúlyponttal, ahol a megfelelő stratégiavektorok, $\boldsymbol{u} = (\boldsymbol{u}_i)_{i=1}^{3}$.

$$\varphi_i(\boldsymbol{u}) = \varphi_i(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3,)$$

Példa: A *No4* példában említett „flexibilis technológia feltételek esetleges megváltoztatása a tervezett rendszerhez képest ebben a fázisban még lehetséges úgy, hogy az „output kritériumok és egyensúlypontok ne változzanak", kiegészítésre kerülhet azzal, hogy a direkt kapcsolatban lévő szomszéd tulajdonságok (sarokkocka három oldalán) együttműködési, kooperációs stratégiái véglegesítődnek. Ilyen lehet a technológiai változás és azt követő javított finanszírozási konstrukció rendszerbe illesztése. Ezek a tulajdonságok a projekt életképességét, a megváltozott gazdasági környezetben való fenntarthatóságát határozzák meg. Nagyon fontos tudnunk, hogy a gazdasági egyensúlypontok, tehát az üzleti életképességet befolyásoló tényezők folyamatosan és gyorsan változnak. A beruházás tervezés vagy üzleti tervek készítése során erre nagyon nehéz felkészülni, így a kötelező fenntarthatósági feltételekhez kötött beruházások (környezetvédelmi, megújuló energiatermelést célzó, klímabarát stb.) sok esetben lehetetlen célállapotba kerülnek. Többek között erre kínál megoldást a Rubik kocka alapú projekttervezés fenntarthatósági algoritmusának alkalmazása.

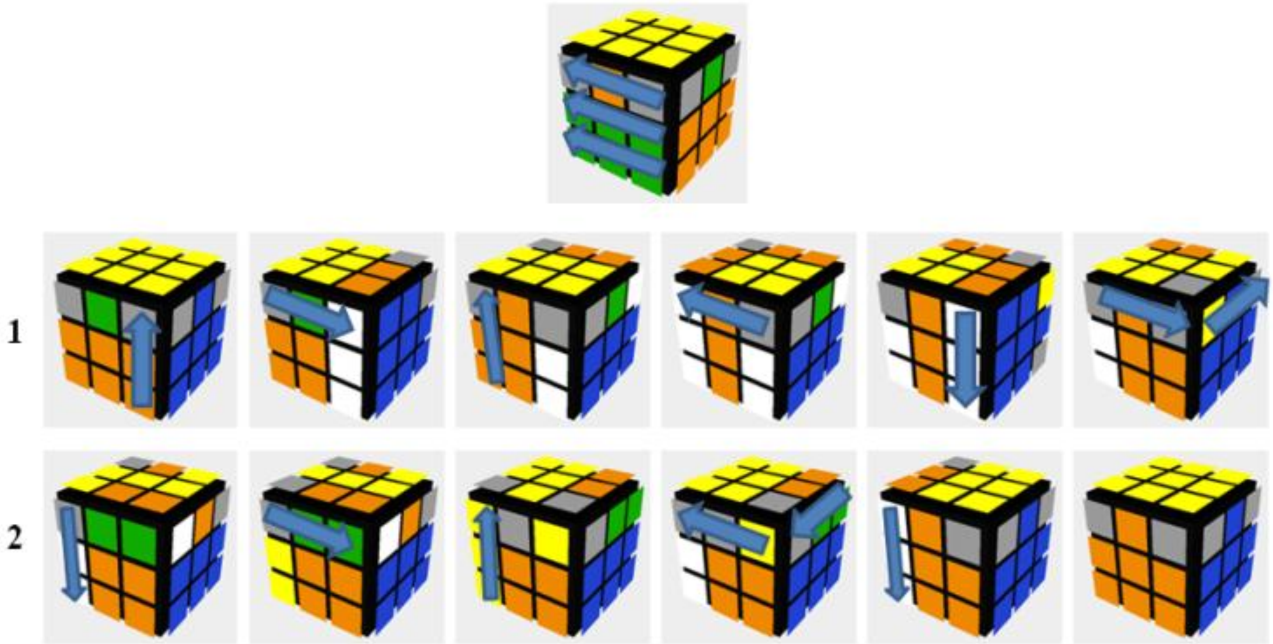
A forgatás során kevés kockakapcsolat változik, amely azt jelzi, hogy az egymás közötti kapcsolatok optimalizációja rövid időt és kevés munkaráfordítást vesz igénybe, de az említett intenzív rendező hatás miatt, végrehajtása rendkívül jelentős.

3.2.1.6. Alsó és felső sor összekötése élcserével, input/output tényezők szigorú összehangolása

Ebben a forgatási sorozatban az összes sárga élkockát a saját helyére kell állítanunk. Ez az a kockaállapot, mikor mindenki úgy érezheti, hogy harmóniában van a kockája, és már csak egy nagyon kis lépés választja el a teljes sikertől. A 29. ábrán látható a sárga és fehér oldalak harmonikus összerendezésének első fázisa.

29. ábra: A sárga és fehér oldal összerendezése az állapottérben főtulajdonságok alapján
Forrás: saját szerkesztés

Hasonlóan a fehér kereszt esetében elmondottaknál, a kirakás során a sárga oldal elforgatásával vagy két élkockát tudunk beállítani, vagy mind a négyet egyszerre. Ha két élkockáról van szó, akkor ez a két élkocka lehet egymás mellet, és lehet egymással szemben elhelyezkedő. Mindkét esetre ugyanazt az algoritmust használjuk, de ha a kicserélendő kockák szemben vannak egymással, akkor kétszer kell az algoritmust végigforgatnunk.



30. ábra: Záró oldal sárga élkockáinak helyreforgatása
Forrás: saját szerkesztés

A élkockák helyreforgatása során tartsuk a két kicserélni való élkockát szemben és magunktól balra. Aztán fordítsuk a jobb oldalt egyszer fel és a tetejét pedig jobbra, majd a bal oldalt újra fel és a tetejét balra. A forgatás után a jobb oldalon három fehér kocka lesz szemben, azokat tegyük le az alsó sorra (30. ábra 1. sor). Ezután fordítsunk a kocka tetején, azaz a felső soron kettőt, majd a bal oldalt fordítsuk le. Ezek után a tetejét fordítsuk jobbra, a bal oldalt fordítsuk hátrafelé, ezzel visszajött az élkocka a szemközti oldalra, a bal oldali oszlopban két fehér kocka lesz (30. ábra 2. sor), amihez a legfelső sor dupla elfordításával állítsuk oda a harmadik fehéret. Utolsó lépésként ezt a kész hármas fehér oszlopot fordítsuk le a többi fehérhez alulra.

„LOW-CARBON INTERPRETÁCIÓ" NO6:

Az input (fehér) és az output (sárga) oldal összekötése a forgatás célja. A legfontosabb bemeneti és kimeneti feltételek szigorú összehangolásáról van szó a forgatás vagy egyensúlykeresés folyamatában. A sárga oldal élkockáinak helykerülése révén a stratégiai fix pontok (a négy meghatározó középkocka), valamint az input oldal bemeneti tényezői direkt, nem változtatható kapcsolatba kerülnek az output tulajdonságokkal, feltételekkel. A teljes folyamat, tervezés, stratégiaalkotás, fejlesztés alapvetően végleges állapotba kerül az élcserével.

Példa: az élcsere jól modellezi, hogyan véglegesítődik a projekt tervezése, kivitelezése szempontjából fontos input és output tulajdonságok összessége. Ilyen eset lehet, mikor az input oldalon feltüntetett politikai feltételrendszer az outputhoz, a program megvalósulásához kapcsolódóan is véglegesítődik. Hasonlóan kezelhető a folyamatot érintő „korrupciós faktorok" vagy a globális tényezők változása és fixálása a projekt fejlődése során.

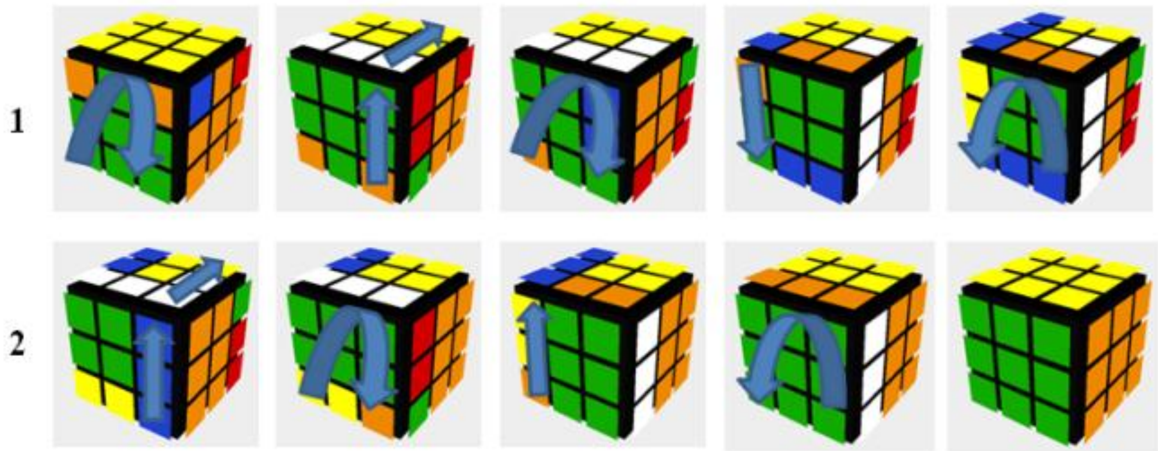
3.2.1.7. Sarokcsere és a rendszertulajdonságok végső egyensúlyának meghatározása

A sarokcsere a kocka kirakásának utolsó fázisa, a rendszertulajdonságok végső egyensúlyi pontjának meghatározása (31. ábra).

31. ábra: Egyensúlyba rendezett kockaállapot
Forrás: saját szerkesztés

A kocka állapot jól ismert ebben a fázisban, vagy három sarokkocka áll rossz helyen, vagy mind a négy. A három sarok kirakása elvezet a negyedik sarok helyreforgatásához is, így ezt külön nem kell megtanulni. Ha nem akarunk több, gyorsabb kirakási algoritmust is megtanulni, akkor elegendő egy algoritmus ismernünk ennek az állapotnak a rendezéséhez, mivel ezt a forgatás többször végrehajtva, a sarokkockák a helyükre fognak kerülni.

Ha van olyan sarkkockánk, amelyik jó helyen áll, akkor az balra tartva a kezünkben, a jobb oldali soron kezdjük meg a sarkok helyre forgatását. Az első sárga soron fordítsunk kettőt (ekkor feljön egy teljes fehér sor), majd a második egyenesen is fordítsunk kettőt, így kialakítathatunk egy szabályos L betűt (vagy tükrözött L betűt). Ez után fordítsuk úgy a kockát, hogy az L betű rendes irányba, betűként álljon (32. ábra 1. sor). Ekkor az L betű szárát magunk felé fordítjuk, az átellenes oldalt pedig abba az irányba (a 32. ábra 1. sor 4. kocka), ahol eddig az L betű szára volt, majd az L betűt a szára visszafordításával visszaállítjuk. Az L betűből ezután egy I betűt fordíthatunk ki, amit úgy érünk el, hogy a L betű talpát lefordítjuk az aljára, azaz kettőt fordítunk rajta. Ezután fordítsuk az I betűt magunk felé, és a túloldalt az I betű helyétől fordítsuk el. Utolsó lépésként fordítsunk egyet úgy, hogy kész legyen a kocka (sárgát sárgához, fehéret fehérhez). Ezt a fogatás sorozatot addig kell ismételjük, amíg mindegyik sarokkocka helyére nem kerül. Ha két sarok nincs a helyén, akkor kétszer egymás után, ha három sarok nem volt a helyén, akkor háromszor egymás után. Ismerünk több olyan algoritmust is, amelyek különböző „sarokállás” esetén, egy rövidebb forgatás kombinációval juttatják el a sarkkockákat a végállapotba. Ezek ismerete természetesen rövidíti a kirakás időtartamát.



32. ábra: Sarkok cseréje helyreforgatással
Forrás: saját szerkesztés

„LOW-CARBON” INTERPRETÁCIÓ NO7:

Fenntarthatósági kritériumok meghatározása, a végső egyensúlyi állapot beállítása a cél a fogatással. A sarokcsere során a forgatás olyan karakterisztikájú, hogy abban az input oldal és az érintett kockaoldal minden tulajdonsága összehasonlításra és utolsó átvizsgálásra kerül. Az élcsere minimum három sarkon végigmegy, de sokszor mind a négy sarok helyreforgatása megtörténik az élcserékkel. A projekttervezés vagy projektfejlesztés mozzanatait modellezve kijelenthető, hogy az élcserével egy végleges keretet kap a vizsgálati rendszer. A három tulajdonságot hordozó sarokkockák révén, 4x3, azaz 12 releváns projekttulajdonság kerül végső egyensúlyi pontba, amely a kirakás folyamatában az egyik legnagyobb horderejű forgatás. A Rubik kocka alapú projekttervezés során ezt a végső egyensúlykeresési folyamatot a fenntarthatósági kritériumok betartásának nevezhetjük. Mint azt az előbbi forgatásokból láthattuk, a sárga vagy output oldal egyensúlypontja többször is kialakítható a fejlesztés folyamatában (sárga kereszt, sárga sarkok kirakása), viszont egyik esetben sem jelenti a háromdimenziós illesztés a fenntarthatósági kritériumok felvételét, csak a sarkok cseréjével.

A záró sarokkockák egyensúlypontjának/fenntarthatósági optimumának keresése: három tulajdonság összehangolására épülő forgatási kombináció az egyik legfontosabb értékét, a végső harmónia megteremtését adja meg a fejlesztési programnak, stratégiának. Ennek a beállításnak a hiányában nincs meg a végső összhang az input és az output oldalak között, a rendszer rugalmassága nagymértékben csökken, mivel nem adaptálta azokat a feltételeket, amelyek az életképességet, a rendszertulajdonságok esetleges változásához kapcsolódó adaptációs képességet jelentenék.

A fentiek értelmében három különböző stratégia programot fogalmazhatunk meg a low-carbon stratégiaalkotás folyamatában:

A. A technológiailag elfogadható tervezési opció (túltervezés, elavulás elkerülése).
B. Pénzügyi fenntarthatóság és likviditás optimalizálása (biztonságos önfenntartás és jövedelemtermelés).
C. Káros projekthatások elkerülése a kapcsolódó termékpályákon (funkciójában önmagának megfelelő rendszer).

A fenti célok matematikai megfogalmazása nem egyszerű feladat, illetve a matematikai értelmezést követő játékelméleti kifizető-függvények felírása is speciális feltételrendszerek beállítása után valósulhat csak meg.

Feladatunk matematikailag többek között egy háromszemélyes játékkal írható le, ahol $\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3$, a megfelelő stratégiavektorok, $\boldsymbol{u} = (\boldsymbol{u}_i)_{i=1}^{3}$ a szimultán stratégia vektor.

$$\varphi_i(\boldsymbol{u}) = \varphi_i(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3,) = c^T_{i1}\,\boldsymbol{u}_1 + \boldsymbol{c}^T_{i2} + \boldsymbol{u}_2 + \boldsymbol{c}^T_{i3} + \boldsymbol{u}_3 = \boldsymbol{c}^T_i + \boldsymbol{u}$$

a célfüggvények a stratégiavektorok.

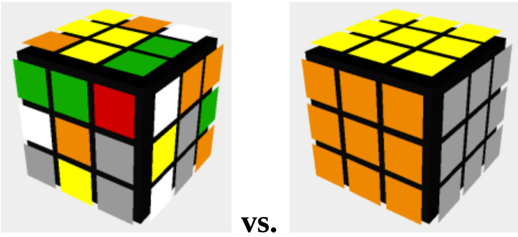
$$\boldsymbol{A}_1\,\boldsymbol{u}_1 + \boldsymbol{A}_2\,\boldsymbol{u}_2 + \boldsymbol{A}_3\,\boldsymbol{u}_3 \geq \boldsymbol{b}$$

alakú feltételeknek tesznek eleget. Itt az együtthatók az előző modellegyütthatókból keletkező vektorok és mátrixok lesznek (függvény levezetése a 25. oldalon található).

Példa: Jó példa erre a gyakorlatból a pénzügyileg és technológiailag is elfogadható végső tervezési opció megtalálása (reális megtérülést biztosító technológiai megoldással), mivel ha ez nem valósul meg, akkor a fejlesztés akár káros fejlesztésként jelenhet meg a társadalom számára. A fenntarthatósági feltételek teljesülése esetén nem fordulhatna elő az Európai Unióban mégis oly gyakori eset, nevezetesen ha a támogatási források megszűnnek egy-egy fejlesztési környezetben, akkor jobb esetben csak a kapcsolódó tevékenységek (pl. hulladékgyűjtő rendszerek, hulladékkezelés) vállnak működésképtelenné, rosszabb esetben az egész termékpálya omlik össze (pl. vállalkozói inkubációs programok vagy K+F programok).

3.2.2. Folyamatelemzés összefoglaló értékelése

A 3x3x3-as Rubik kocka sorról-sorra kirakási módszere alapján történő projekttervezési és fejlesztési folyamatok jól demonstrálják a körülöttünk lévő erőforrások fenntartható felhasználásának, kapcsolati rendszerének összefüggéseit, így a jövőben javasolt ezen elvek mentén kialakítani fejlesztési és stratégiai elképzeléseinket. A Rubik kocka kirakási folyamatára alapozott folyamatszabályozás garantáltan gyors, hatékony, kis költséggel járó tervezési protokoll, illetve a bemutatott folyamatanalízis egyértelművé tette számunkra, hogy figyelembe vétele esetén, biztonságosan teljesülnek a hosszú távú (fenntartható) működtetés feltételei, nagy valószínűséggel feltételezhető, hogy a fejlesztési folyamat eredménye a társadalom számára nem lesz káros fejlesztés.



33. ábra: Kocka entrópikus és egyensúlyi állapotban
Forrás: saját szerkesztés

A kocka kirakásával a projektfejlesztés folyamatát imitáltuk, tehát a rendezetlenségi állapotból a teljes rendezettség állapotába való eljutás útvonalát. A Rubik kocka egyensúlyi kockaállapota a teljesen kirakott Rubik kocka. Nem véletlen, ha valaki meglát egy összekevert Rubik kockát, azonnal szeretné megoldani, kirakni, mivel a kívánt vagy vágyott állapot, a színre kirakott kocka állapot (33. ábra). A Rubik kocka színösszeállításában is hordozza a harmóniát, mint az már korábban említettem, a kocka színeinek kiválasztása a feltaláló részéről tudatosan történt, a színek egymásmellettisége szintén nem a véletlen műve. A kocka misztifikálása nélkül kijelenthető, hogy a kocka már látványában is magában hordozza azt a színgazdag harmóniát, mely révén tökéletes egyensúlyt és hibátlan logikát feltételezünk a konstrukcióban.

A teoretikus folyamatértékelés során, a forgatások bemutatásának célja annak szemléltetése, hogy egyes állapotokba való eljutás milyen kocka interakciókat feltételez, tehát mely kockák/tulajdonságok egymásra hatását kell vizsgálnunk a forgatási folyamat alapján. Ezek pontos helyét és interakcióit jelen kutatás során nem határoztam meg, de a folyamat fázisokra történő felosztása megtörtént, illetve a kirakási szakaszok és projektfejlesztés mechanizmusainak összevetését elvégeztem. A párhuzamok egyértelműen igazolták, hogy a két logikai művelet egymást erősem támogathatja. A folyamatértékelés alapján bebizonyosodott, hogy a 3x3x3 Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatának.

A 12. táblázatban összesítettem azokat értelmezési szinteket, amelyek a projektfejlesztés folyamatában is értelmezhető tartományokat jelentenek, illetve ahol lehetségesnek ítéltem felírtam azokat a játékelméleti egyensúlykeresési összefüggéseket, melyekkel az egyes színekhez vagy fázisokhoz rendelt projekttulajdonságok - a fenntarthatóság érdekében - egyensúlyi állapotba hozhatók.

12. táblázat: A modellalkotási folyamat és az eredmények evaluációja

KOCKA INTERPRETÁCIÓK (forgatási algoritmus száma)	MODELL-ALKOTÁS SZINTJE	BEFOLYÁSOLT /low-carbon/ PROJEKTTULAJDONSÁG	JÁTÉKELMÉLETI ÖSSZEFÜGGÉS
NO1	INPUT	„Fehér kereszt" – kiindulási peremfeltételek megadása	n- személyes Zérusösszegű folytonos játékkal leírható állapot
NO2	INPUT	Fehér sarok kirakása – fenntartható fejlesztési irányok kijelölése, Egyensúlykeresés, nem kooperatív optimum	Nash-egysúlypontra vonatkozó függvények alapján, nem kooperatív stratégia, véges játékkal leírható
NO3	KOCKA-KÖZÉP	Második sor kirakása - kapcsolati pontok rögzítése, egyensúly megteremtése, Kétdimenziós tényezők összerendezése, fixpont igazítás	Élkockák helyre forgatása konfliktus feloldási módszerrel lehetséges A fixpontok igazítása függetlenségi vizsgálat és Zérusösszegű játékkal célszerű
NO4	KOCKA-KÖZÉP	„Sárga kereszt" – input/output oldalak indirekt összehangolása	Véges oligopol játékkal, vagy egyenlő kompromisszumok módszerével leírható
NO5	OUTPUT	Sárga sarok kirakása – fenntarthatósági kritériumok értelmezése az outputok rendezése során	Folytonos háromszemélyes játékkal leírható, Nash-egyensúly keresés szükséges
NO6	OUTPUT	Sárga oldal élcsere – input/output oldalak szigorú összehangolása	Zérusösszegű játékkal leírható folyamat, konfliktus feloldási módszer kooperatív stratégia
NO7	OUTPUT	Sarokcsere – a végső egyensúlyi állapot beállításának fázisa, egyensúlykeresés, fenntarthatósági kritériumok véglegesítése	Oligopol játékok, kooperatív egyensúlyi stratégiára alkalmazva, vagy Nash-egyensúlypontra vonatkozó függvények alapján, kooperatív stratégia.

Forrás: saját szerkesztés

3.3. Rubik kocka alapú „low-carbon” optimalizáció elmélete

Létezik néhány jól működő szoftverfejlesztési eljárás, amelyeket főként az ipari fejlesztések területén alkalmaznak, de mindegyiknek megvan a maga előnye, hátránya, korlátoltsága. A modellek általában kötődnek valamilyen vállalathoz, szervezethez is, akik továbbfejlesztik, támogatják, promótálják ezeket a módszereket. Egy specifikus fejlesztési modell azonban soha nem lehet minden projektre alkalmas fejlesztési eljárás, a technológiák, a források korlátozottsága, a piacra jutás ideje, a gyorsan változó fogyasztói igények azok a speciális faktorok, amelyeket a programfejlesztőknek figyelembe kell venniük, ha adoptálnak egy-egy módszert a saját projekt ciklusaik esetében.

A kutatási témához kapcsolódóan meg kell említeni egy már ismert Rubik kockás szoftver fejlesztési módszert (Rubik’s Cube software development methodology - RCM), egy speciális célrendszerű módszertan, amely rendkívül hasznos megoldást kínál a szoftverek életciklusainak modellezésére. Az RCM modell koncepció különösen hasznos régi szoftverek felújításban, a kifutó szoftveralkalmazások új fogyasztói igényeknek történő megfeleltetésében. Ez már egyfajta recycling vagy újrahasznosítás, amellyel eleve sok energiát és anyagot takaríthatunk meg (ez a fejlesztési vonal megfelel az EU low-carbon koncepciójának is). Az RCM alapeleme a Rubik kocka kirakásának legismertebb módszere a Layer by layer kirakás, mely módszer alapvetően a sorról sorra történő kirakást követi, és könnyen értelmezhető a szoftverek újraprogramozásának folyamatában. A szoftverfejlesztés és a Rubik kocka kirakás analógiájára 2011-ben jöttek rá az indiai szoftverfejlesztők, melyet részben bevezettek a gyakorlatba és rendkívül sok programozási munkától kíméli meg a szoftverfejlesztőket. A szoftverek új fogyasztási igényeknek történő megfeleltetése rendkívül időigényes és költsége folyamat, amely a RCM módszerrel jelentősen csökkenthető, így az elgondolás bizonyítottan egy új és költséghatékony megoldást kínál a szoftverfejlesztésben (Fogarassy, 2012).

Az általam megfogalmazott, Rubik kocka alapú low-carbon optimalizációs elmélet egyik legfontosabb tulajdonsága, hogy egy-egy projekt többszintű (bemeneti, kimeneti, kapcsolati) vizsgálatát az egyes projekttényezők valós interakciói alapján vizsgálja, így a szükségtelen elemzésének elkerülésével időt és rengeteg munkát takaríthatunk meg. A kocka egyes oldalaihoz rendelt rendszerkapcsolatok (élkocka tulajdonságok, sarokkocka tulajdonságok) szükségtelenné teszik, hogy bizonyos tulajdonságokat direkt módon vizsgáljunk, tehát minden egyes tényezőkapcsolat nem kell „beszéljen” egymással. A szükséges rendszerelemek közötti „kommunikációk” tehát elérhetők egyszerű határfelületi kapcsolatok célirányos vizsgálatával, vagy áttételes rendszerkapcsolatok leírásán keresztül.

3.3.1. A „low-carbon” fejlesztések fenntarthatósági összefüggései

Fontos rendszertulajdonság, mikor a technikai megfelelőséget vizsgáljuk egy projekt fejlesztés során, akkor nem kell az outputhoz kötődő piaci lehetőségeket is direkt módon figyelembe vennünk az elemzésnél, de a két tulajdonság közötti kapcsolat mégis megvan és figyelembe vételre kerül az interakciókon keresztül. Ugyanilyen példa lehet, ha egy-egy pénzügyi megfelelőség vizsgálata esetén a likviditási kérdések tárgyalása, amely nem függ közvetlenül a piaci kereslettől, de mégis befolyással vannak egymásra, mely kapcsolatot vizsgálatok nélkül is közvetíti - a megfelelőséget biztosítva - a Rubik kockás módszertan. A fent említett szempontokra épül a korábban már említett UN LEDS programja (LED-low-emission development strategies), amelyet 1992-óta szeretnének a gyakorlati megvalósítás reális szintjére terelni, de a program közgazdasági értelmezése nem tudott megfogalmazódott az elmúlt évtizedekben.

A « low-carbon economy » főbb prioritásainak hazai célrendszere (Fogarassy, 2013 alapján):

- Minden erőforrás, és főként az energiahordozók esetében a hatékonyság javítására kell törekednünk. Sokkal hatékonyabb rendszerben kell az energia transzformációs rendszereinket működtetni, fokozott figyelmet kell fordítani a villamos energiatermelés hőenergia kapacitásainak lokális és maximális kihasználására.

- A tudatos fogyasztást nagyon magas szinten kell megvalósítani – legyen az akár környezetvédelmi, vagy társadalmi felelősségvállalásra vonatkozó kezdeményezés - jelenjen meg a termelés, a kereskedelem, vagy akár az egyén szintjén.
- Lokális termelés és fogyasztás preferenciája elsődleges! Akármilyen keresleti igény merül fel – legyen az energia, anyag vagy szolgáltatás – azokat feltétlenül a lokális rendszereink erőforrásaival kell kielégítenünk. Az energiahordozókat az alacsony emisszió kibocsátási rendszerekben kell előállítani, megújuló és alternatív energiaforrásokat kell használni és a CO_2 kibocsátást a lehető legnagyobb mértékben megkötni.
- Minden hulladékot minimalizálni kell – újrahasznosítani, visszaforgatni, csökkenteni, mivel ezzel rendkívül sok nyersanyag és energia takarítható meg.

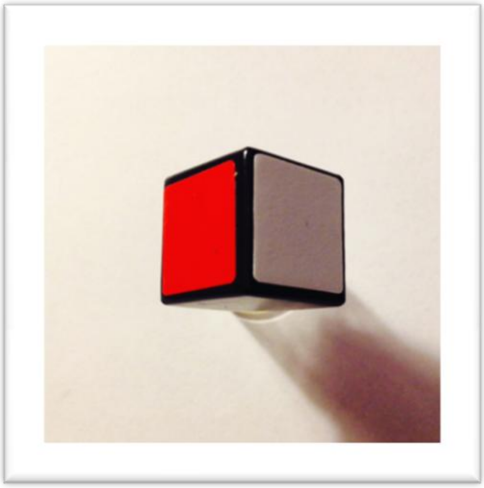
A „low-carbon-economy" esetében nagyon nehéz lesz megvalósítani ezt a komplex követelményrendszert, vagy az alapelveket tudatosan beépíteni azokba a beruházási folyamatokba, melyek esetén a BAU követelmények (business as usual) alapvetően mellőzik a fenntarthatósági szempontrendszert. Azonban ezeknek a szempontoknak, ezeknek a prioritásoknak a figyelembe vételéhez ajánlható a Rubik kocka kirakásának módszere, mely módszer (a több dimenzióban is értelmezhető térbeli struktúrájával) jól modellezi a fenntarthatóság „low-carbon" célrendszerét az egyes fejlesztési és beruházási programokban.

3.3.2. Fenntarthatósági kritériumok optimalizálása és a háromdimenziós problémakezelés elmélete

A komplex kockázat menedzsmentet segíti az egy, kettő és három szempontú, Rubik kocka alapú egyidejű problémakezelés. A kocka váza, nem mozgatható kockarészek. Az 1 dimenziós problémakezelés alapjai a 6 db középkocka (34. ábra). A projektfejlesztés vagy beruházás tervezés szempontjából olyan tulajdonságokat rögzítünk a középkockákhoz, amelyek a folyamatban a fejlesztés vázát jelentik, tehát megváltoztathatatlan, fix pontokat jelentenek a meghatározó területeken (pl. technológia, szabályozás, finanszírozás, piac).

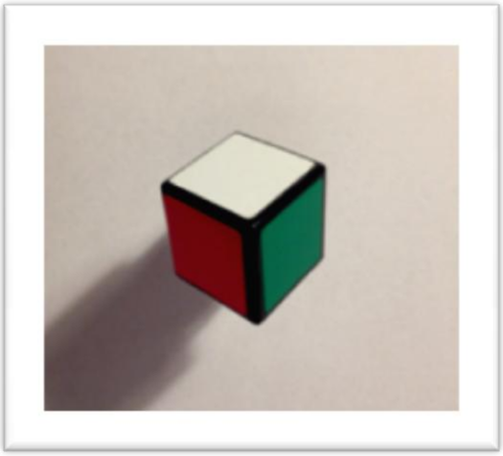
34. ábra: Rubik kocka váza a forgatható, de helyükről nem elmozdítható fix állású középkockákkal
Forrás: saját szerkesztés

Az élkockák száma 12. A kétdimenziós problémakezelés alapja az együtt mozgatható, vagy optimalizálható két tulajdonság (pl. technológiai szabályozás és finanszírozás). Ez hagyományos értelemben (x,y) tengelyek mentén történő vizsgálatot jelent (35. ábra).



35. ábra: A Rubik kocka élkockája, mely kétirányú illesztést igényel a helyrerakáskor
Forrás: saját szerkesztés

A sarokkockák száma 8 db. A speciális háromdimenziós problémakezelés alapja az együtt mozgatható, vagy optimalizálható három tulajdonság. Ez hagyományos értelemben (x,y,z) tengelyek mentén történő vizsgálatot jelent (36. ábra).



36. ábra: A sarokkocka, háromoldalú illesztést jelent a kirakás során
Forrás: saját szerkesztés

Feltételezzük, hogy hatoldalú, 3x3x3-as Rubik kocka minden oldala és minden kiskockája harmonizál a projektfejlesztés egy-egy elemével:

A. Középkocka - stabil vagy fix összetevője, tulajdonsága a kocka minden oldalának és a projektfejlesztés fázisának is. A 3x3x3-as Rubik kocka esetében 6 középkockát különböztetünk meg, mely stabil kockák és egyben fix tulajdonságok alapvetően körvonalazzák és meghatározzák a fenntartható projekttervezés folyamatát is.

B. Élkocka - közvetlen kapcsolatot jelent két szín és két tulajdonság között. Az élkockák száma a 3x3x3-as Rubik kocka esetében 12. Az élkockák kétdimenziós tulajdonsághordozást jelentenek, mely révén az összetartozó rendszerjellemzők a projekt fejlődése során is együtt határozzák meg az egyes tulajdonságok illesztését és a szükséges rendszerfeltételeket.

C. Sarokkocka – A projektfejlesztés vagy tervezés koordinálásának legfejlettebb eleme, mivel 3 tulajdonságot egyszerre optimalizál a fejlesztési folyamat során. Nagyon komplex és direkt kapcsolatot jelent a három szín, azaz három tényezőcsoport végső illesztési elemei között. A fehér (input) oldali sarokkockák a stabil, minden szempontot és tényezőt mérlegelni tudó elemeit jelentik a projekt struktúrának, a sárga (output) oldali kockák pedig a fenntarthatóságok, a harmonikus erőforrás felhasználást, a káros fejlesztések kiküszöbölésének szelekcióját valósítják meg.

A low-carbon projektfejlesztési vagy tervezési eljárás hipotézisem szerint egy paralel protokollnak nevezhető a Rubik kocka layer-by-layer kirakási módszerével. Az oldalak és az egyes színek projekttulajdonságokhoz rendelésével elérhetjük, hogy egy speciális, a fenntartható projektfejlesztési folyamatot valósíthassunk meg, melynél a fejlesztés és eljárás specifikumát az adja, hogy az egyes tényezők vagy tényezőcsoportok, mint egy-, két- vagy háromdimenziós rendszerelemek vehetők figyelembe, kezelhetők a fejlesztési programok során. A low-carbon fejlesztések és Rubik kocka kirakás célrendszere ugyanazt a vezérlőelvet követi, logikus és alacsony energia bevitel révén igyekszik eljutni az „egyensúlyi állapotba".

3.3.3. Rubik kocka alapú egy, kettő és háromdimenziós problémakezelés módszere

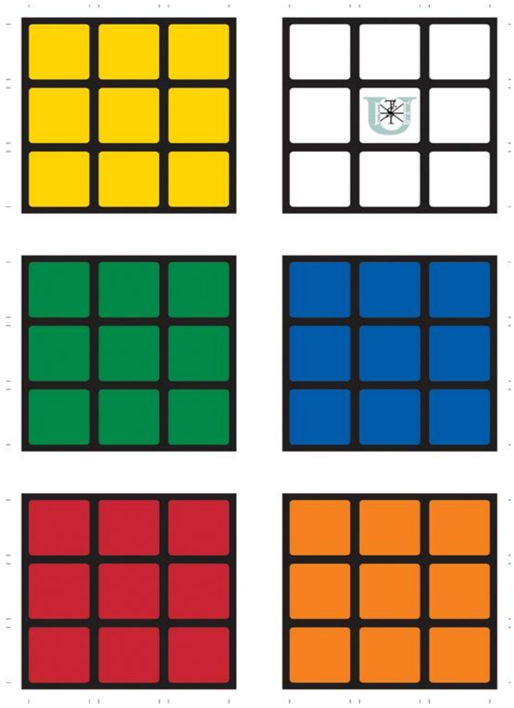
A projekttervezés és fejlesztés alapvetően egy olyan folyamat optimalizáció, amely különböző tényezők együttes kezelésén alapul, mégpedig úgy, hogy a vizsgált szegmensek egymáshoz képest a legharmonikusabb konstellációba kerüljenek. Egy feltételezett „low-carbon optimalizációs protokoll" esetében négy különböző determinációs terület (tényezőcsoport) kijelölése szükséges, és ezeket a területeket a 3x3x3-as kocka egymás melletti színes oldalaihoz tudjuk rendelni. Két szemközti oldal (fehér, sárga) lesz a projektünk input és output oldala. Az optimalizációt meghatározó tényezőcsoportok a következők lehetnek egy demonstrációs projektben : stratégiai célrendszer optimalizációja (piros oldal), piaci lehetőségek elemzése (zöld oldal), megvalósítás, technológiai feltételrendszerek területe (kék oldal), pénzügyi hatások illesztése (narancs oldal), illetve az input oldali célokat összesítő (fehér front oldal) és az output oldali eredményösszesítő (sárga oldal) feltételrendszer.

A Rubik kocka alapú low-carbon optimalizációs koncepció egyik legfontosabb tulajdonsága (az indiai szoftverfejlesztési tapasztalatokra alapozva) az, hogy egy-egy projekt többszintű vizsgálatát az egyes komponensek valós interakciói alapján vizsgálja, így a szükségtelen interakciók elemzésének elkerülésével időt és rengeteg munkát takaríthatunk meg (low-carbon megoldás). A kocka egyes oldalaihoz rendelt rendszerkapcsolatok (élkocka tulajdonságok, sarokkocka tulajdonságok) szükségtelenné teszik, hogy bizonyos tulajdonságokat direkt módon vizsgáljunk, tehát minden egyes tényezőkapcsolat nem kell „beszéljen" egymással. Ezek a rendszerelemek közötti „kommunikációk" tehát elérhetők egyszerű határfelületi kapcsolatokkal, vagy áttételes rendszerkapcsolatokon keresztül.

Fontos rendszertulajdonság, mikor a technikai paramétereket vagy megfelelőséget vizsgálunk egy projekt fejlesztés során, akkor nem kell az outputhoz kötődő piaci lehetőségeket is direkt módon figyelembe vennünk az elemzésnél, de a két tulajdonság közötti kapcsolat mégis megvan, és figyelembe vételre kerül az interakciókon keresztül. Ugyanilyen példa lehet, ha egy-egy pénzügyi megfelelőség vizsgálata esetén a likviditási kérdések tárgyalása, amely nem függ közvetlenül a piaci kereslettől, de mégis befolyással vannak egymásra, mely kapcsolatot vizsgálatok nélkül is közvetíti - a megfelelőséget biztosítva - a Rubik kockás módszertan.

Nagyon fontos megjegyezni, hogy egyes tulajdonságok sokkal több interakciót kívánnak meg egymás között, mint mások. Ez azt jelenti, hogy egy-egy tulajdonság (ami a kocka színhelyeivel paralel) megfelelő helyre kerülése, vagy végleges karakterisztikájának biztosítása csak a többi tulajdonság együttes optimalizációjával biztosítható. Ebből az aspektusból is világos, miért olyan fontos egy három dimenzióban is értelmeződő projektfejlesztési modell alkalmazása.

Alapvetően a logikai tervezés és a modellezés kétdimenziós stratégiai modellekben folyik (pl. logframe mátrix – LFA/Logical Framework Approach), melyhez kapcsolódóan leképeztük azt az esetet, hogy mit jelentene a Rubik kocka szerkezete kétdimenziós értelmezésben. Az 37. ábra jól szemlélteti, hogy a kétdimenziós (x,y) kocka struktúra ugyan több tulajdonságot is kezelhetne együtt, de az azok közötti kapcsolatot, amelyet csak a térbeli értelmezés adhat, azt nem tudja megfogalmazni.



37. ábra: A kétdimenziós Rubik kocka szerkezetének értelmezése
Forrás: saját szerkesztés

Alapvetően kijelenthetjük, hogy a projektfejlesztést befolyásoló tényezők helyes összeillesztésében, a Rubik kocka háromdimenziós értelmezési tartománya jelenti az újszerűséget és a gyakorlati előnyt. Rubik Ernőnek, a Rubik kocka feltalálójának is az volt az alapvető célja a kocka tervezésével, hogy az építészhallgatók a részek közötti kapcsolatot, a térbeli egységek elmozdulását, kapcsolatuk változását három dimenzióban is tudják értelmezni. Az újszerű, háromdimenziós protokoll lehetőséget teremt arra, hogy gyorsabban, olcsóbban és a helytelen, hibás fejlesztéseket kiküszöbölve valósuljanak meg a jövőben fejlesztési programjaink. Az egyes oldalakhoz rendelt jelentéstartalom és a multi dimenzionálás értelmezés látható az 13-17. táblázatokban. A háromdimenziós feladatkezelés az input és az output oldali feltételek úgymond alaposabb elemzését teszi lehetővé, mely háromdimenziós illesztés természetesen végigvonult a projekttervezés teljes szakaszán a kocka folytonos újrarendezésének köszönhetően. (A kocka „újrarendezése” azt jelenti, hogy miden egyes újabb él vagy sor színhelyes beillesztése úgy történik meg, hogy a kész kockaelemeket az algoritmus eredeti helyükre rendezi vissza.)

A következő táblázatban (13. táblázat) egy konkrét projekt input és output oldalának karakterisztikáit foglaltam össze. A tipizált projektfejlesztési cél jelen esetben a fosszilis energiahordozó cseréje megújuló energiaforrásokra, vagy azok kombinált rendszereire vonatkozik. A 14-17. táblázatokban szakértői értékeléssel, súlyozás alapján került kijelölésre a négy főágens, amelyeket a színes oldalakhoz rendeltem. A színek közül a piros oldal reprezentálja a „Stratégiai programillesztés/jogi és szabályozási feltételeket”, a zöld szín „Piaci lehetőségek vizsgálatát”, a kék szín a „Technológiai feltételrendszert”, a narancssárga pedig a „Pénzügyi hatások összesítését”. Az egyes oldalakon (főágenseken) belül pedig a kockák kapcsolati rendszerén keresztül határoztam meg az egyszerre két vagy három tulajdonságot is magában hordozó kockajellemzőket. Azok a tulajdonságok, amelyek úgymond függetlenek a projekt vagy beruházás fejlesztése, megvalósítása szempontjából, a középkockákhoz kapcsolódnak, ezekből oldalanként egy darab van, egy-egy meghatározó tulajdonsággal (természetesen az input és output oldalakon is van fixkocka, de ezek nem főágenshez kapcsoltak). A két vagy három tulajdonságot is összekapcsoló jellemzőket az élkockákhoz és sarokkockákhoz rendeltem. A tulajdonságok meghatározása és az egyes kockákhoz rendelése hasznosságuk alapján történt. Az egy irányba, vagy pontszerűen értelmezhető tulajdonságokat „1D", azaz egy dimenziós (x) tulajdonságnak, a két irányba ható (x,y) tulajdonságokat „2D", vagy két dimenzióban értelmezhető tulajdonságoknak, a három irányba ható tulajdonságokat (x,y,z), háromdimenziós vagy „3D" tulajdonságnak neveztem el.

13. táblázat: Rubik kocka bemeneti (fehér/F) és kimeneti (sárga/S) oldalainak jelentése

OLDALSZÍNEK	**A SZÍNEK JELENTÉSE** 1D – egy dimenzióban értelmezhető tulajdonság (x) 2D – kettő dimenzióval értelmezhető tulajdonság (x,y) 3D – három dimenzióval értelmezhető tulajdonság (x,y,z)
FEHÉR (F)	**INTPUT**: Az input követelmények megfogalmazása, a piac és az állami szabályozás összeillesztése mellett megfogalmazott termék vagy szolgáltatás alaprendszere. **FEHÉR OLDALI INPUT LEKÉPEZÉS:** FEHÉRKÖZÉP (1D) ✓ energiaracionalizálás (F) ÉLFEHÉR (2D) ✓ stratégiai alapkapcsolat (FP), ✓ technológiai alapkövetelmény (FK), ✓ finanszírozási elvárás (FN), ✓ piaci alapillesztés (FZ), SAROKFEHÉR (3D), ✓ piaci igényű megtérülési alapfeltétel (NZF), ✓ támogatási eszközök és technológiai feltételeknek történő megfeleltetés (NKF), ✓ technológia kockázatok és innovációs prioritások összehangolása az alapcéllal (KPF), ✓ stratégiákkal összehangolt piaci szegmens kijelölése az alapfeltételeknél (PZF).
SÁRGA (S)	**OUTPUT:** Az erőforrás-felhasználási lehetőségek maximumának figyelembe vételével körvonalazott Pareto optimális termék vagy szolgáltatási rendszer. **SÁRGA OLDALI OUTPUT LEKÉPEZÉS:** SÁRGAKÖZÉP (1D) ✓ erőforrás-hatékony energiafelhasználás/ jövedelmező termelés (S) ÉLSÁRGA (2D) ✓ az energia- és CO2 mérleg stratégiai megfeleltetése (PS), ✓ technológiai kockázatcsökkentés minimalizálása (KS), ✓ adózási és kedvezményezési feltételek az energiatermelő rendszerben (NS), ✓ mesterséges és valós piacra lépés pénzügyi feltételei (ZS), SAROKSÁRGA (3D) ✓ stratégia célrendszerekhez illeszkedő struktúra, amelynél a piaci életképesség is igazolt (PSZ), ✓ pénzügyileg és technológiailag is elfogadható tervezési opció (reális megtérülést biztosító technológiai megoldás) (NSK), ✓ piaci feltételek között életképes termelési szolgáltatási feltételek megléte (ZSN), ✓ a választott technológiai megoldás a stratégiai célokat maximálisan alátámasztó, hosszú távú valós opció (KSP), (fenntarthatósági szempontok érvényesítése a fejlesztésben). (Jelentések: D=dimenzió, F=fehér, S=sárga, Z=zöld, P=Piros, K=kék, N=Narancssárga)

Forrás: saját szerkesztés

14. táblázat: Rubik kocka „Piros/P tulajdonság oldalainak" jelentése

OLDALSZÍNEK	**A SZÍNEK JELENTÉSE** 1D – egy dimenzióban értelmezhető tulajdonság (x), 2D – kettő dimenzióval értelmezhető tulajdonság (x,y), 3D – három dimenzióval értelmezhető tulajdonság (x,y,z)
PIROS (P)	**STRATÉGIAI PROGRAMILLESZTÉS/JOGI ÉS SZABÁLYOZÁSI FELTÉTELEK** : A tervezett profilhoz kapcsolódó meghatározó információk, szinergiák, kooperációk biztosítása vállalati, lokális, szektor, regionális vagy uniós gazdaságpolitikai szinten. PIROSKÖZÉP (1D) ✓ lokális/vállalati stratégia megvalósítása (P) ÉLPIROS (2D) ✓ piaci stratégia követése és összehangolása a gazdaságpolitikai prioritásokkal (PZ), ✓ technológiai rendszerek megtérülő változatokhoz történő megfogalmazása, támogatási prioritások műszaki paramétereinek illesztése a projekthez (PK), ✓ az energia- és CO2 mérleg stratégiai megfeleltetése (PS), ✓ stratégiai alapkapcsolat (PF) SAROKPIROS (3D) ✓ technológia kockázatok és innovációs prioritások összehangolása az alapcéllal (KPF), ✓ stratégiákkal összehangolt piaci szegmens kijelölése az alapfeltételeknél (PZF), ✓ stratégia célrendszerekhez illeszkedő struktúra, amelynél a piaci életképesség is igazolt (PSZ), ✓ a választott technológiai megoldás a stratégiai célokat maximálisan alátámasztó, hosszú távú valós opció (KSP) (Jelentések: D=dimenzió, F=fehér, S=sárga, Z=zöld, P=Piros, K=kék, N=Narancssárga)

Forrás: saját szerkesztés

15. táblázat: Rubik kocka „Zöld/Z tulajdonság oldalainak" jelentése

OLDALSZÍNEK	**A SZÍNEK JELENTÉSE** 1D – egy dimenzióban értelmezhető tulajdonság (x) 2D – kettő dimenzióval értelmezhető tulajdonság (x,y) 3D – három dimenzióval értelmezhető tulajdonság (x,y,z)
ZÖLD (Z)	**PIACI LEHETŐSÉGEK VIZSGÁLATA**: Piaci lehetőségek, piaci pozíciók értékelése a mesterséges és valós piac szegmensekben. ZÖLDKÖZÉP (1D) ✓ kereslet és kínálati egyensúlyban tervezhető ár ÉLZÖLD (2D) ✓ piaci stratégia követése és összehangolása a gazdaságpolitikai prioritásokkal (PZ), ✓ piaci változások hatása a finanszírozási rendszerre, deviza kockázati tényezők és globális hatások elemzése (NZ), ✓ mesterséges és valós piacra lépés pénzügyi feltételei (ZS), ✓ piaci alapillesztés (FZ). SAROKZÖLD (3D) ✓ stratégia célrendszerekhez illeszkedő struktúra, amelynél a piaci életképesség is igazolt (PSZ), ✓ piaci feltételek között életképes termelési szolgáltatási feltételek megléte (ZSN), ✓ piaci igényű megtérülési alapfeltétel (NZF), ✓ stratégiákkal összehangolt piaci szegmens kijelölése az alapfeltételeknél (PZF). (Jelentések: D=dimenzió, F=fehér, S=sárga, Z=zöld, P=Piros, K=kék, N=Narancssárga)

Forrás: saját szerkesztés

16. táblázat: Rubik kocka „Kék/K tulajdonság oldalainak" jelentése

OLDALSZÍNEK	**A SZÍNEK JELENTÉSE** 1D – egy dimenzióban értelmezhető tulajdonság (x) 2D – kettő dimenzióval értelmezhető tulajdonság (x,y) 3D – három dimenzióval értelmezhető tulajdonság (x,y,z)
KÉK (K)	**TECHNOLÓGIAI FELTÉTELRENDSZER**: A piaci lehetőségek és a technológiai megoldások összehangolása. Fontos a technológiai kockázatok és lehetőségek feltérképezése. KÉKKÖZÉP (1D) ✓ BAT technológiai követelményeknek megfelelő technológiai alkalmazás ÉLKÉK (2D) ✓ technológiai rendszerek megtérülő változatokhoz történő megfogalmazása, támogatási prioritások műszaki paramétereinek illesztése a projekthez (PK), ✓ technológiai alapkövetelmény (FK), ✓ technológiai kockázatcsökkentés minimalizálása (KS), ✓ leggazdaságosabb technológiai megoldás, magas minőséggel, innováció-tartalommal (KN). SAROKKÉK (3D) ✓ támogatási eszközök és technológiai feltételeknek történő megfeleltetés (NKF), ✓ pénzügyileg és technológiailag is elfogadható tervezési opció (reális megtérülést biztosító technológiai megoldás) (NSK), ✓ technológia kockázatok és innovációs prioritások összehangolása az alapcéllal (KPF), ✓ a választott technológiai megoldás a stratégiai célokat maximálisan alátámasztó, hosszú távú valós opció (KSP). (Jelentések: D=dimenzió, F=fehér, S=sárga, Z=zöld, P=Piros, K=kék, N=Narancssárga)

Forrás: saját szerkesztés

17. táblázat: Rubik kocka „Narancs/N tulajdonság oldalainak" jelentése

OLDALSZÍNEK	**A SZÍNEK JELENTÉSE** 1D – egy dimenzióban értelmezhető tulajdonság (x) 2D – kettő dimenzióval értelmezhető tulajdonság (x,y) 3D – három dimenzióval értelmezhető tulajdonság (x,y,z)
NARANCS (N)	**PÉNZÜGYI HATÁSOK ÖSSZESÍTÉSE**: Finanszírozási típus, kormányzati eszközök jelenléte, adózás, deviza kockázat, likviditási kérdések NARANCSKÖZÉP (1D) ✓ megtérülési idő, cégérték (N) ÉLNARANCS (2D) ✓ adózási és kedvezményezési feltételek az energiatermelő rendszerben (NS), ✓ finanszírozási elvárás (FN), ✓ leggazdaságosabb technológiai megoldás, magas minőséggel, innováció-tartalommal (KN), ✓ piaci változások hatása a finanszírozási rendszerre, deviza kockázati tényezők és globális hatások elemzése (NZ) SAROKNARANCS (3D) ✓ pénzügyileg és technológiailag is elfogadható tervezési opció (reális megtérülést biztosító technológiai megoldás) (NSK), ✓ piaci igényű megtérülési alapfeltétel (NZF), ✓ támogatási eszközök és technológiai feltételeknek történő megfeleltetés (NKF), ✓ piaci feltételek között életképes termelési szolgáltatási feltételek megléte (ZSN). (Jelentések: D=dimenzió, F=fehér, S=sárga, Z=zöld, P=Piros, K=kék, N=Narancssárga)

Forrás: saját szerkesztés

A Rubik kocka tulajdonság oldalainak és az oldalakon belüli kockák jelentésének kiválasztása (14., 15., 16., 17. táblázat) ebben az esetben véletlenszerű, de mivel a tényezők fontosságát, egymáshoz viszonyított preferenciát meg tudjuk határozni, célszerű a hasznosságukat is vizsgálnunk valamilyen függvényszerű kapcsolattal. Kritériumok súlyozására, a dominancia meghatározására vezetünk be módszert a modell jóságának fokozása érdekében.

3.4. „Rubik kockás projektfejlesztés" folyamata játékelméleti értelmezésekkel

A Low-carbon Rubik kockás projekttervezés és projektfejlesztés egy speciálisan felépített tervezési koncepció, amely jelenleg egyedüliként képes a folyamatokat befolyásoló tényezők háromdimenziós értelmezésére. Az input és output oldalak gazdasági vagy erőforrás-felhasználási egyensúlypontjának „beállítására", illetve a köztük lévő kapcsolat leírására olyan játékelméleti megoldásokat alkalmaztam, melyeket eddig még nem használtak fel hasonló célokra a tudományos kutatások során.

A modellezést megelőző tolerancia vizsgálat célja a műszaki életben a megadott méretektől, mennyiségtől, vagy minőségtől való megengedett legnagyobb eltérést megállapítását célozza. A játékelméleti algoritmusok esetében azt vizsgáltam, hogy melyik módszer egyezik meg tulajdonságaiban a Rubik kocka kirakási folyamat modell tulajdonságaival, azoktól milyen megengedhető mértékben tér el a reprezentativitás megtartása mellett. A keresett játékelméleti algoritmusok esetében toleranciát, azaz a megengedhető eltéréseket vizsgáltam a kockatulajdonságok és a játékelméleti függvények paraméterezésére.

A komplex modellkészítés során a játékelméleti modelleket sorra vettem, és a modellalkotás folyamatában az egyes forgatási algoritmusokhoz (interpretációkhoz) kapcsolható modelleket rendeltem. A kocka tulajdonság halmazait három fő csoportba osztottam, melyek az INPUT oldali tulajdonságok, a KOCKAKÖZÉP oldalainak tulajdonságai és OUTPUT oldali tulajdonságok. A három tulajdonsághalmaz egyensúlypontjainak beállítására játékelméleti módszereket alkalmaztam. Ennek lényege az volt, hogy azokon a tulajdonság elemeken, amelyeken a SMART (Simple Multi Attribute Ranking Technic) vizsgálat nem megengedhető eltéréseket jelzett, ott az egyensúlyi pontok (Nash-egyensúly) felé vezető stratégiák modellezésére írtam fel paramétereket. A vizsgálatok és a modellalkotás három szinten (input, kockaközép, output) történt, így a teljes folyamat játékelméleti modellezése is 3 szinten, azaz 3 különböző típusú játékelméleti modell (különböző kifizető-függvényeinek) összekapcsolásával végeztem.

Az egyes modellezési szintekhez kapcsolódó játékelméleti kifizető-függvények felírása a modellben úgy történt, hogy a megvizsgáltam az input oldal, a kockaközép valamint az output oldal azon tulajdonságait, amelyek a SMART analízis alapján, a csoportba tartozó tulajdonságok közül a „nem megengedhető eltéréseket" mutatják, majd ezeket az üzleti környezetben értelmezve, fenntarthatósági stratégiaként optimalizáltam.

3.4.1. Input oldali leképezések algoritmusai

A projektindítás folyamata történik ebben a fázisban. Választ kapunk arra, hogy mit kell figyelembe venni a projekt elindítása során! Az első 'layer' vagy kockasor helytelen forgatása rossz irányú, sikertelen folytatást is eredményez, tehát nem tudunk a következő, középső 'layer'-re jutni.

Energiacserés példával ezt könnyen magyarázhatjuk. Ha az eredeti energiaellátó rendszerünket úgy cseréljük le, hogy az a korábbi rendszer vagy annak fő elemei még életciklusának 60-80 %-os elavulási fázisába jár, akkor jelentős pénzügyi veszteséget okozhatunk a beavatkozással. Ennek elkerülésére számolhatunk a pl. Nash-egyensúly beállításával optimális beavatkozási időpontot.

A környezetvédelmi célú fejlesztések ellentétes irányúak a gazdaságfejlesztési prioritásrendszerrel (pl. az üvegházgáz csökkentést, fosszilis energia felhasználást célzó program az energiafelhasználás minimalizálását célozza, a másik a szennyező energiahordozó növelését. Első layer tervezése során, a szabályozáspolitika és a finanszírozás politika esetében alkalmazható a projekttervezés folyamatában (38. ábra). Ugyanez ez a helyzet a vízbázis védelem és a támogatott vízigényes energianövények termelési feltételeinek vonatkozásában is. Egy-egy projekt esetében, a világos üzleti és egyéb szabályozási feltételek, valamint a fenntartható üzleti stratégiák megvalósítása érdekében, a nem kooperatív gazdasági szereplők által diktált feltételeket is figyelembe kell vennünk. Ebben a helyzetben a Nash-egyensúly megtalálása különösen nehéz feladat, de feltétlenül szükséges, mert az ellentmondásos helyzetből a projekt nem építhető tovább.

Definíció:

A Nash-egysúlypontra vonatkozó *Definíció* szerint:

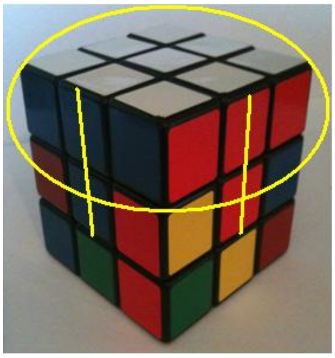
Egy $J = (n, S, (\varphi_i)_{i=1}^{n})$ n személyes játék egyensúlyi pontján vagy a stratégiáján olyan $(x_1^*, \ldots\ldots, x_n^*) \in S$ pontot (stratégiai n-est) értünk, melyre

$$\varphi_i(x_1^*, \ldots, x_{i-1}^*, x_i^*, x_{i+1}^*) \geq \varphi_i(x_1^*, \ldots, x_{i-1}^*, x_i, x_{i+1}^*)$$

teljesül minden $i = 1, \ldots\ldots, n$ játékosra. Az egyensúlyi pontot tehát Nash-féle equilibrumnak nevezzük.

Tétel:

Az első 'layer' kirakását követően csak a Nash-féle egyensúlyi ponttal rendelkező kapcsolat építhető tovább, azaz a kocka kiforgatását csak ebből az állapotból forgathatjuk tovább. Az első 'layer' mindig korrelál a második 'layer' középkockájához, tehát csak szín azonos lehet.



38. ábra: Első sor vagy 'layer' Nash-féle egyensúlyi pontja (bekarikázva), a középkocka mindig szín azonos (a függőleges vonalakkal jelölve)
Forrás: saját szerkesztés

Bizonyítás:

Legyen $x^* = (x_1^* \ldots\ldots, x_n^*)$ a játék valamely egyensúlypontja. Ekkor tetszőleges $y = (y_1 \ldots\ldots y_n) \in S$ esetén:

$$\varphi_k(x_1^*, \ldots\ldots, x_k^*, \ldots\ldots, x_n^*) \geq \varphi_k(x_1^* \ldots\ldots, y_k, \ldots\ldots x_n^*) \;\; (k = 1,2,3 \ldots, n),$$

és innen összeadással közvetlenül látható, hogy $\varphi(x^*, x^*) \geq \varphi(x^*, y)$. A tétel alapján hatékony algoritmust lehet ajánlani a tervezést befolyásoló tényezők egyensúlypontjának meghatározására, a halmaz leképezések fix-problémáinak megoldására.

Példa:

A biomassza alapú megújuló energiatermelés tervezése során kritikus pont, hogy a rendkívül nagy mennyiségű vízfelhasználás korlátozhatja a projekt jövedelemtermelő képességét, illetve a leghatékonyabb technológia lehetőségek alkalmazásának feltétele lehet. A kérdést, illetve a feltételt ezért alapvetően technológiai jellegűnek tekintjük, ezért a háromdimenziós tulajdonságokat hordozó sarokkocka (színe: piros-zöld-fehér), amely színkombináció az input oldal (fehér), a stabil szabályozási feltételek (piros), valamint a technológiai megoldások (zöld) együttes kezelését próbáljuk játékelméleti optimummal és stratégiával párosítani (39. ábra).

Szerencsére a vízelosztási problémák megoldása központi szerepet játszik a játékelméleti megoldások között is, rendszerint sok-sok rendkívül bonyolult függvénykapcsolat leírásán keresztül juthatunk csak olyan egyensúlypontokhoz, amelyek egy megnyugtató rendszerhasználat feltételeit körvonalazzák, és a matematikai összefüggések felírása különleges kihívást jelent. A többcélú vízfelhasználás, illetve a vízfelhasználáshoz kapcsolódó szereplők érdekei és kifizető-függvényei eltérő helyeken kínálnak optimumokat, melyek rendszerint egy többszereplős, nem lineáris, de mégis valamilyen Nash-egyensúlyon nyugvó nem-kooperatív játékot feltételeznek.

A probléma megfogalmazásához - a Rubik kocka alapú low-carbon fejlesztésekhez kapcsolódóan - egy háromszereplős vízfelhasználáshoz kapcsolódó optimalizálást fordítottam le a Rubik kocka alapú stratégiai alkotás folyamatára, melyet Szidarovszky-Molnár (2013) leírása alapján végeztem.

Tehát a többcélú vízelosztás, mint döntéselméleti feladat évtizedekre visszamenőleg jelent kihívást a kutatás számára, sokféle megoldási opcióval. Jelen esetben nem-kooperatív Nash-egyensúlyt keresünk három játékos (mezőgazdasági vízfelhasználó /öntözésre/, ipari vízfelhasználó /hűtésre/, háztartási felhasználó /funkcionális/) részvételével. A low-carbon stratégiai probléma központi eleme, hogyan döntheti a mezőgazdasági (biomassza termelő) vízfelhasználási projekt, hogy számára elegendő víz jut a rendelkezésre álló erőforráskészletből.

SMART ÉRTÉK	KOCKATÍPUS ÉS DOMINANCIA	FŐÁGENS BELSŐ TULAJDONSÁGOK
73	sarokkocka, átlagon felül domináló tulajdonsággal	stratégiákkal összehangolt technológia kijelölése az alapfeltételeknél (FZP)
57	**élkocka**, átlagon alul domináló tulajdonsággal	stratégia, szabályozási alapkapcsolat (FZ)
87	sarokkocka, átlagon felül domináló tulajdonsággal	pénzügyi eszközöknek és szabályozási feltételeknek való megfeleltetés (FNZ)

39. ábra: „Fehér-zöld-piros” sarokkocka 3D tulajdonsága (FZP) a reális megtérülést biztosító technológiai megoldás (háromszereplős vízfelhasználáshoz kapcsolódó optimalizálás)
Forrás: saját szerkesztés

A probléma háromdimenziós, a Rubik megoldás input oldali eldöntendő kérdése. A rendelkezésre álló vízfogyasztás bázisa lehet *felszíni víz*, *felszín alatti víz* és *tisztított szennyvíz.* Legyen $k = 1,2,3$ a három játékos, akik a döntési folyamatban a következő döntési változatokat követhetik:

A stratégia minden játékos számára egy ötelemű vektorral írható fel:

$$\boldsymbol{x_k} = (\boldsymbol{f_k}, \boldsymbol{t_k}, \boldsymbol{k_k}, \boldsymbol{f_k^*}, \boldsymbol{t_k^*}).$$

ahol

f_k = felszíni víz lokális	*t_k =felszín alatti víz lokális*	*k_k = kezelt, tisztított víz*
f_k^* =felszíni víz import	*t_k^* =felszín alatti víz import*	

A kifizető-függvény pedig a teljes felhasznált vízmennyiségre, minden játékos esetében:

$$\phi_k = \boldsymbol{f_k} + \boldsymbol{t_k} + \boldsymbol{k_k} + \boldsymbol{f_k^*} + \boldsymbol{t_k^*}$$

A játékosok mindegyikének van két közös korlátozó feltétele, mely szerint a felhasznált vízmennyiség nem lehet kevesebb, mint a minimálisan szükséges mennyiség D_k^{min}, illetve nem lehet nagyobb, mint a maximális technológiai igény D_k. (A feltétel a vízpazarlás elkerülésére szolgáló, fenntarthatósági kritérium.)

$$f_k + t_k + k_k + f_k^* + t_k^* \geq D_k^{min}$$

$$f_k + t_k + k_k + f_k^* + t_k^* \leq D_k$$

A mezőgazdasági szereplőnek *(k=1)* a fentieken kívül be kell vezetnie még másik két korlátozó feltétel a vízhasználatra vonatkozóan, a következő változókkal:

G = azon növények csoportja, amelyek esetében csak felszín alatti víz használható
a_i = növényarány (i) a teljes mezőgazdasági terület arányában
w_i = a növények vízigénye (i) hektáronként
T= az a növénycsoport, amelyet szennyvízzel is lehet öntözni
$W = \sum_i a_i w_i = teljes\ vízigény\ minden\ növény\ esetében\ hektáronként$

tudjuk, hogy a felszín alatti víz nyújtja a legjobb öntözővíz minőséget, a tisztított szennyvíz pedig a legrosszabbat, így meg kell határoznunk azoknak a növényeknek (szenzitív élelmiszernövény) a volumenét a mezőgazdasági portfólióban, melyek nem öntözhetők tisztított vízzel. Az elérhető felszín alatti vízbázisra támaszkodó vízszükséglet nem lehet alacsonyabb, mint azoknak a növényeknek a vízigénye, amelyek csak tiszta, jó minőségű felszín alatti vízzel öntözhetők:

$$\frac{t_1 + t_1^*}{f_1 + t_1 + k_1 + f_1^* + t_1^*} \geq \frac{\sum_{i \in G} a_i w_i}{W}$$

ha az egyenletet átírjuk lineáris formába, akkor

$$a_1 f_1 + (\alpha_1 - 1)t_1 + a_1 k_1 + \alpha_1 f_1^*(\alpha_1 - 1)t_1^* \leq 0$$

$$\text{ahol } \alpha_1 = \frac{\sum_{i \in G} a_i w_i}{W}$$

Ehhez hasonlóan a tisztított víz felhasználásának aránya és elérhetősége is felírható. A tisztított vízszükséglet nagysága sem lehet nagyobb, mint a rendelkezésre álló vízmennyiség. Ez az összefüggés adja azoknak a növényeknek a volumenét, amelyek tisztított szennyvízzel is vagy csak ezzel öntözhetők (pl. energetikai célra termelt növények).

$$\frac{t_1}{f_1 + t_1 + k_1 + f_1^* + t_1^*} \geq \frac{\sum_{i \in T} a_i w_i}{W}$$

ha az egyenletet átírjuk lineáris formába, akkor

$$-\beta_1 f_1 - \beta_1 t_1 + (1 - \beta_1) k_1 - \beta_1 f_1^* - \beta_1 t_1^* \leq 0$$

$$\text{ahol } \beta_1 = \frac{\sum_{i \in T} a_i w_i}{W}$$

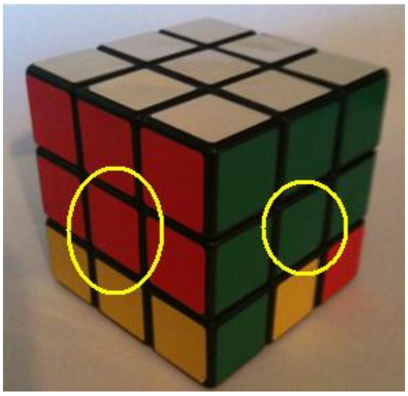
A többi játékos (ipari, háztartási) szintén fel kell írnunk a megadott korlátozások alapján a megfelelő függvénykapcsolatot, melynek összefüggésrendszere megtalálható a hivatkozott publikációban, csak eztán rendelhetünk numerikus adatokat a feladatmegoldáshoz.

A fentiek alapján egyértelműen megfogalmazható, ha mezőgazdasági rendszereinket az első szinten úgy tervezzünk meg energetikai biomassza felhasználás céljára, hogy a korlátozó erőforrás felhasználását (vízigény) már a kiindulási állapotban játékelméleti megoldással egyensúlyba tereljük, akkor a tervezési folyamat megfeleltethető a fenntarthatósági kritériumrendszereknek is. A vizsgálat konkrét eredménye az lehet, hogy a projekt tervezése során nem kalkuláljuk túl aránytalanul a vízfelhasználást, illetve ha már ebben a pontban világos, hogy rendelkezésre álló vízmennyiség nem elegendő a Pareto optimális termelés megvalósításához, azaz vélhetően vízdeficitet okoz a vizsgált rendszerben, akkor a projektet elvetjük, mert nem teljesíti a fenntarthatósági kritériumokat.

3.4.2. Input és output kapcsolatok leírása játékelméleti összefüggésekkel

KÖZÉPKOCKA KAPCSOLAT JÁTÉKELMÉLETI MODELLEZÉSE (2. SZINT)

A középkocka helyben tartása, a ‘layer’ vagy sor kirakása a zérusösszegű játékokat imitálja, mivel a középkocka helye megváltozhatatlan, így ahhoz igazodni kell a többi kockával. A középkockák helye és állandó (helyükről, kapcsolati rendszerűkből nem kiforgathatók) és az általuk meghatározott értékösszetevők konstansnak tekinthetők (40. ábra).



40. ábra: Zérusösszegű játékok a mindig fix középkockára (sárgával bekarikázva), az élkockák (kétszínű) erre a feltételre optimalizálódnak
Forrás: saját szerkesztés

Definíció:

Egy $n-$ személyes J játékot konstans összegűnek nevezzük, ha a játékos által elért nyeremények és veszteségek végösszege egy állandó c szám, az összes lehetséges stratégia mellett.

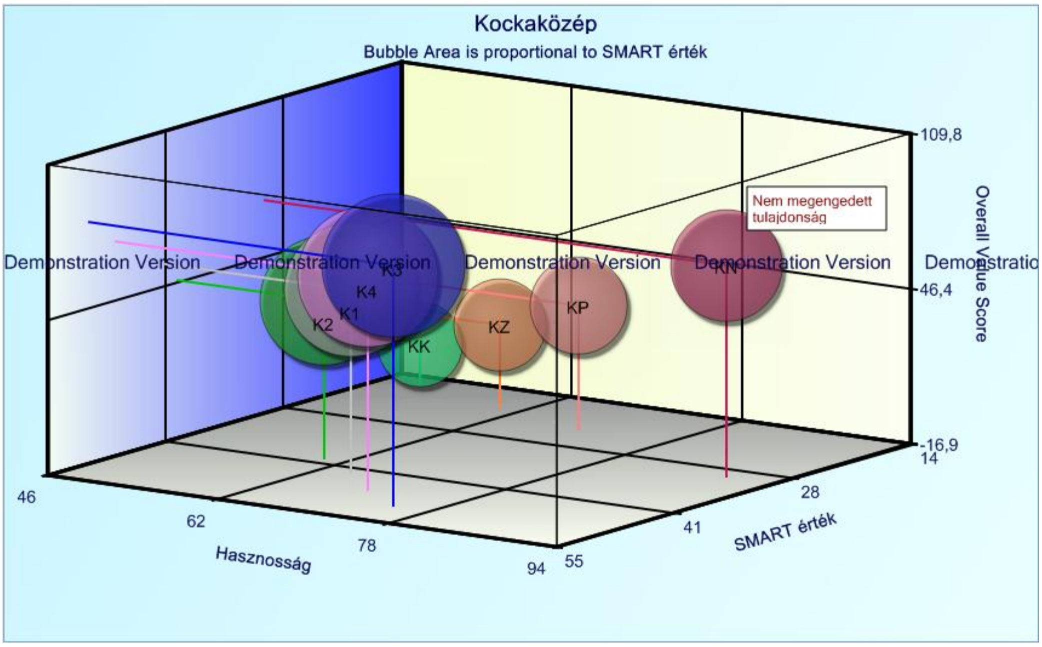
Képletszerűen:

$$\sum_{i=0}^{n} \varphi_i(x) = c \ (x \in S).$$

Ha $c = 0$, a játékot *zérusösszegűnek* nevezzük.

Tétel:

A zérusösszegű játékkal konstans összegű optimalizálást végzünk, mivel fixpont tulajdonság miatt, az erőforrás véges összegű, így az adott erőforrás harmonikus elosztása a cél, a tulajdonságcsoport fix egyensúlyi pontját keressük vele (41. ábra). Az előzetes SMART analízis alapján megállapítottuk, hogy a narancs-közép, azaz a Rubik kocka narancsszínű oldalának középkockája (KN), nem megengedett tulajdonság eltérést mutat. A narancs oldal a jelenlegi főtulajdonság halmazban a „a projekt pénzügyi értéket, a megtérülési időt”, mint projekt tulajdonságot hordozza magával. Ennek a tulajdonságnak a vizsgálata játékelméleti optimalizálással megoldást ad arra, hogy a low-carbon karakterisztikájú projekt fix erőforrásai hogyan optimalizálódnak Nash-egyensúlyi helyzetbe.



41. ábra: SMART elemzés „nem megengedett” tulajdonsága (megtérülési idő, projekt érték nincs egyensúlyban a többi erőforrással)
Forrás: saját szerkesztés

A 41. ábrán látható egyensúlytalanság oka az lehet, hogy a projekt, beruházás megtérülését befolyásoló külső tényezők stabilitása nem megfelelő. Ennek megállapítására az új belépő piacra lépésének körülményeit szükséges elemeznünk.

A feladat megoldása nem egyszerű, ha a befolyásoló tényezők között vannak olyanok, melyek nem piaci elemek (externáliák), de mégis hatással vannak az megtérülés hosszára (pl. adó- és szabályozáspolitika, szennyezés kontroll, devizapolitika stb.).

Bizonyítás:

A Rubik kocka középkockáinak (4 különböző fix tulajdonság) Nash-egyensúlyi pontját úgy határozom meg, hogy megállapítom, mely tényező nincs a Pareto optimális helyzetben.

Az $n -$ személyes konstans összegű játékok alkalmasak arra, hogy szemléltetni tudjuk a négy tényező egyensúlyi pontját.

Alapul véve egy $(x_1^*, x_2^*, x_3^*, x_4^*) \epsilon\ S$ egyensúlyi pontot, ami alapján felírhatjuk

$\varphi_1(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_1(x_1, x_2^* x_3^*, x_4^*)$ minden $x_1 \in S_1$ mellett

és

$\varphi_2(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_2\left(x_1^*, x_{2,} x_3^*, x_4^*\right)$ minden $x_2 \in S_2$ esetén.

és

$\varphi_3(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_3(x_1^*, x_2^*, x_3, x_4^*)$ minden $x_3 \in S_3$ esetén.

és

$\varphi_4(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_4(x_1^*, x_2^*, x_3^*, x_4)$ minden $x_4 \in S_3$ esetén.

A játék zérusösszegű, így

$$\varphi_1(x_1, x_2, x_3, x_4) + \varphi_2(x_1, x_2, x_3, x_4) + \varphi_3(x_1, x_2, x_3, x_4) + \ \varphi_4(x_1, x_2, x_3, x_4) = 0\ ,$$

második egyenlőség alakulása

$$\varphi_1(x_1^*, x_2^*, x_3^*, x_4^*) + \varphi_2(x_1^*, x_2^*, x_3^*, x_4^*) + \varphi_3(x_1^*, x_2^*, x_3^*, x_4^*) + \ \varphi_4(x_1^*, x_2^*, x_3^*, x_4^*) = 0$$

Bármelyik tulajdonság nem megengedett eltérése, legyen $\varphi_1(x_1, x_2^* x_3^*, x_4^*)$

$$\varphi_1(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_1(x_1, x_2^* x_3^*, x_4^*)$$

Konstans összegű játék esetében a játék minden stratégiájára érvényes:

$$\begin{aligned}&\varphi_1(x_1^*, x_2^*, x_3^*, x_4^*) + \varphi_2(x_1^*, x_2^*, x_3^*, x_4^*) + \varphi_3(x_1^*, x_2^*, x_3^*, x_4^*) + \ \varphi_4(x_1^*, x_2^*, x_3^*, x_4^*)\\ &\quad \geq \varphi_1(x_1, x_2^* x_3^*, x_4^*) + \varphi_2(x_1^*, x_2^*, x_3^*, x_4^*) + \varphi_3(x_1^*, x_2^*, x_3^*, x_4^*) + \ \varphi_4(x_1^*, x_2^*, x_3^*, x_4^*)\end{aligned}$$

Az 4 tényezős konstans összegű játék egyensúlyi pontja megszűnik, ha bármely tényező esetében stratégiaváltás következik be:

$$(x_1^*, x_2^*, x_3^*, x_4^*) \longrightarrow (x_1, x_2^*, x_3^*, x_4^*)$$

a stratégia bármely elemének megváltozatása egyenlőtlenséghez vezet,

$$\varphi_1(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_1(x_1, x_2^*, x_3^*, x_4^*)$$

Ez az egyenlőtlenség-rendszer azt mondja, hogy ha az $(x_1^*, x_2^*, x_3^*, x_4^*)$ egyensúlyi pontból az első játékos egyoldalúan kilép valamilyen x_i^*-tól különböző stratégiát választva, a kifizető függvény csak kisebb vagy egyenlő lehet. Ha a negyedik tényező tér el nem megengedett módon, és a többiek nem változtatnak stratégiájukon akkor az ő kifizető függvénye is kisebb vagy egyenlő lesz, mint a többiek $\varphi_{1,2,3}(x_1^*, x_2^*, x_3^*, x_4^*)$ esetében.

$$\varphi_{1,2,3}(x_1^*, x_2^*, x_3^*, x_4^*) \geq \varphi_4(x_1^*, x_{2,}x_3^*, x_4^*)$$

a $\varphi_{1,2,3}(x_1^*, x_2^*, x_3^*, x_4^*)$ tényezők kifizető függvénye nagyobb vagy egyenlő lesz – mivel a játék zérusösszegű, ez azt jelenti, hogy az összkifizetés nem növekedhet és nem csökkenhet.

3.4.3. Output oldali leképezések algoritmusai

A konfliktus-feloldási módszerek a kooperatív játékelméleti megoldások egyik népszerű családját jelentik. Ezek közül kiemelhetjük a Nash axiomatikus megoldásrendszerét, amely axióma halmazok megadásával biztosította, hogy a megoldás mindig a *Pareto-vonalon* legyen. A Kálai-Smorodinsky megoldás pedig a konfliktushelyzet legrosszabb kimeneteli pontjának meghatározásával megadja a minimálisan elérhető, vagy a konfliktus megoldásaként megadható utolsó lehetséges pontot, vagyis a még elfogadható legrosszabb kimenetelét

OUTPUT OLDAL JÁTÉKELMÉLETI MODELLEZÉSE (3. SZINT)

Az kimenti oldalon végrehajtott sarokcsere a végső egyensúlyi állapot beállításának fázisa, egyensúlykeresés, fenntarthatósági kritériumok véglegesítése legtöbb esetben csak kooperatív stratégiával végezhető.

Definíció:
A *kooperatív játék*ot a következő fogalmakkal határozzunk meg. $N = \{1, \ldots\ldots, n\}$ a *játékosok halmaza*, amelynek egy tetszőleges S részhalmazát közismert szóval *koalíció*nak nevezzük: $S \subseteq N$. Legyen S a részhalmazok, azaz a lehetséges koalíciók halmaza. Az N alaphalmazt *teljes koalíció*nak nevezzük.

Tétel:
A low-carbon beruházási koncepciónkban zöldáramot termel a projekt, de a megtermelt áram csak úgy juthat el a fogyasztóhoz, ha mind a zöldáram termelő (Beruházó/B), mind a hálózat (Hálózat/H) tulajdonosa megállapodnak abban, hogy a termék a hálózaton keresztül eljut a fogyasztóhoz. Az együttműködés feltétele, hogy a beruházás használati/szállítási díjat fizet a hálózat tulajdonosának, a hálózat tulajdonosa pedig belenyugszik abba, hogy a korábbi értékesítési mennyiség helyett, kisebb mennyiségben szállít saját terméket a hálózaton. A hálózat kompenzációként megkapja a beruházó kifizetését. Ez a kompromisszum tulajdonképpen azt jelenti, hogy piaci feltételek között életképes termelési és szolgáltatási feltételekben kell megállapodni (SKP). A leírt Rubik kockás projektfejlesztés „zöld-sárga-narancs" tulajdonságkockáját próbáljuk a modellel egyensúlyba hozni, megfelelő stratégiát hozzárendelni az együttműködéshez.

Bizonyítás:
A konfliktus-feloldási módszert egy kétszemélyes játékkal vezethetjük be. A példában jelöljük S_1 és S_2 a játékosok stratégia halmazát φ_1 *és* φ_2 a két kifizetőfüggvényt. A lehetséges kifizetések halmaza így 2-dimenziós lesz és következőképpen írható fel:

$$H = \{\varphi_1(x,y), \varphi_2(x,y) \mid (x,y) \in S_1 \times S_2\}$$

Ebben az esetben is, mint mindig, mindkét játékos a kifizetésének maximalizálására törekszik, de természetesen kifizetése függ a másik játékos stratégiájától és általános szabály, hogy az egyik játékos kifizetésének növelésével a másik csökkeni fog. A feladat tehát az, hogy olyan megoldást találjunk, amely mind a Beruházó mind a Hálózat tulajdonos, tehát mindkét játékos számára elfogadható megoldást jelent. Minden megoldás előtt fel kell tennünk azt, ha nem jön létre megállapodás, akkor mindkét játékos alacsonyabb kifizetést, vagy büntetést kap.

Általános jelölések:

$$\boldsymbol{f}_* = (f_{1*}, f_{2*})$$

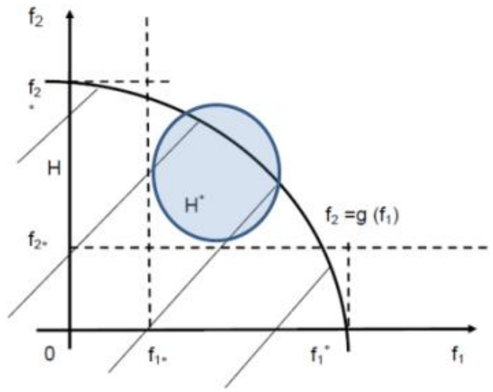
ez lesz a kifizetési vektorunk, amelynél feltételezzük, hogy létezik olyan $(f_1, f_2) \in H$, amely esetén $f_1 > f_{1*}$ és $f_2 > f_{2*}$. A konfliktus matematikailag $(H, \boldsymbol{f}_*)$ párral definiált. A 42. ábrán ezt a párt definiáltuk. Feltesszük továbbá, hogy a H halmaz zárt, konvex és korlátos, azaz

$$(f_1, f_2) \in H \text{ és } \bar{f_1} \leq f_1, \bar{f_2} \leq f_2$$

esetén szükséges $(\bar{f_1}, \bar{f_2}) \in H$, valamint mindkét koordinátájában korlátos, azaz

$$sup\,\{f_i | (f_1, f_2) \in H\} < \infty$$

$$i = 1{,}2 \quad \text{esetén.}$$



42. ábra: A konfliktushelyzet ábrázolása a kifizető-függvény helyével
Forrás: saját szerkesztés (3. ábra módosítása)

Feltesszük továbbá, hogy a H határvonala egy $f_2 = g(f_x)$ függvény gráfja, amely szigorúan csökken f_1-ben és konkáv. A $g-$ függvény gráfját Pareto-vonalnak szokták nevezni, tehát a fenntarthatósághoz kapcsolódó optimum kritériumok kielégítésének feltétel itt teljesülhet. A játék és megoldásfeltételek között figyelembe kell venni, hogy racionális játékos nem fogad el olyan megállapodást, amely rosszabb kifizetést jelentene, mint a megállapodás nélküli kifizetés.

A lehetséges kifizetési halmazt így leszűkíthetjük a következőképpen:

$$H^* = \{f_1, f_2 | f_1 \geq f_{1*}, f_2 \geq f_{2*}, (f_1, f_2 \in H\}$$

Következtetés:

A projektfejlesztési folyamat (biomassza alapú energiatermelés) low-carbon tervezéséhez a különböző (kocka)szinteken komplex, unortodox játékelméleti optimumkeresést hajtottuk végre. A játékelméleti optimumkeresés során elméleti modell struktúrát határoztam meg, mely alapvetően három különböző típusú játékelméleti megoldás egymás után illesztését jelenti annak függvényében, hogy az adott gazdasági feltételrendszert, milyen játékelméleti módszerrel a leghatékonyabb jellemeznünk:

1. kockaszint : nem kooperatív háromszemélyes játék (input oldal nem megengedett eltéréseinek korrigálására),
2. kockaszint: nem kooperatív zérusösszegű játékkal írjuk le (középkocka kapcsolat nem megengedett eltéréseinek korrigálására),
3. kockaszint: konfliktus-feloldási módszert kétszemélyes játékkal (output oldal nem megengedett eltéréseinek korrigálására).

A három játékelméleti modell együtt tudja leírni a projektfejlesztés során szükséges Nash-egyensúly állapotokat, melyek a fenntarthatóságot biztosítják a projekt megvalósítása során. A megfelelő Nash-egyensúly kiválasztását a kockák kapcsolati rendszere alapján történő SMART érték meghatározás biztosítja. Ennek részletezésére a későbbiekben kerül sor. Ugyanakkor hangsúlyoznunk kell, hogy az általam kiválasztott játékelméleti sor (háromszemélyes kooperatív játék – nem kooperatív zérusösszegű játék – konfliktus-megoldási módszer), elsősorban a tipizált energetikai fejlesztés és meghatározott gazdaságfejlesztési környezetben (Magyarország/Közép-és Kelet-Európa) alkalmazható. Kijelenthetjük tehát, hogy más gazdasági körülmény vagy más fejlesztési cél ezek alapján, eltérő játékelméleti megoldássorral is jellemezhető.

3.5. Kritériumok és kockatulajdonságok súlyozása és Churchman-Ackoff féle dominancia meghatározás

Amint azt már a többdimenziós problémakezelés esetében említettük, a két szemközti kocka oldal (fehér, sárga) lesz a projektünk input és output oldala, a „low-carbon optimalizációt" meghatározó tényezőcsoportok/főágensek pedig a következők lehetnek demonstrációs projektünkben:

- ✓ stratégiai célrendszer optimalizációja (piros oldal),
- ✓ piaci lehetőségek elemzése (zöld oldal),
- ✓ megvalósítás, technológiai feltételrendszerek területe (kék oldal),
- ✓ pénzügyi hatások illesztése (narancs oldal).

Felvetődik a kérdés, hogy miért pont a felsorolt főágensek lettek kijelölve meghatározó tényezőcsoportként! A tényezőcsoportok vagy főágensek kiválasztását természetesen egy szisztematikus elõválogatás előzte meg, melynek folyamatát és módszertani elemei a következőképpen foglalhatjuk össze.

A kiválasztás során tehát alapvetően a fejlesztési folyamatokat befolyásoló kritériumok súlyozása szükséges ahhoz, hogy az ismert tényezőcsoportból kiválasszuk azokat a legfontosabb tulajdonságokat, amelyek a projekt fejlesztésének, a beruházások megvalósításának meghatározó feltételeit adják. A 3x3x3-as Rubik kocka hat különböző oldala párosult tulajdonság csoportokkal, de mint az korábban már meghatároztuk, a kocka egymással szemben található oldalai (fehér és sárga) adják a projekt folyamatok input és output oldalait. Ebben az összefüggésben a főtulajdonságok meghatározása a négy domináns tulajdonságra korlátozódik, melyek így a narancs, kék, piros, zöld oldalakkal párosulnak. A Módszertani fejezetben leírt módon és okból a kiválasztás menete speciálisan kialakított lépésekből állt.

Az értékelési folyamatban, illetve az egyes főágens tulajdonságainak megjelenítésére a SMART (Simple Multi Attribute Ranking Technic) szoftver használtam, ezért dominancia meghatározás menetét és adattartalmát is a szoftver adatbeviteli feltételeinek feltettem meg. A felhasznált módszer három fő összetevőre épít az értékbecslés során, nevezetesen:

1.) Az eredmények összegyűjtése és megfeleltetése a vizsgálatnak,
2.) A szakértői csoportok közötti lekérdezés,
3.) Szakértői csoportok direkt beavatkozása online korrekcióval.

A vezérelt folyamat operatív lépései pedig a következők:

Lépés1: A layerek vagy szempontok sorrendbe rendezése (szakértői becsléssel) 1-100 között.

Lépés2: Layerek által meghatározott csoportok alcsoportokba rendezése, a kontroll tulajdonság kiválasztása, a kontroll tulajdonságok az 50 –es értékkel jellemezhető tulajdonságok lesznek. A kontroll tulajdonságot minden alcsoporthoz hozzá rendezzük. Szakértői becsléssel súlyozzuk a tulajdonságokat.

Lépés3: Fontosság megítélése, szakértői becsléssel. A szakértői becsléssel a kontroll tulajdonsághoz viszonyítva, három kategóriát jelölünk meg az alcsoportban:

- fontosabb, mint a kontroll (+)
- kevésbé fontos, mint a kontroll (-)
- ugyanolyan fontos, mint a kontroll (0)

A folyamat végén a megadott értékek és válaszok alapján generál a módszer algoritmusa végső eredményt a tulajdonságoknak. Az elért eredmények alapján állít sorrendet, majd beállítja az érzékenységet, vagyis azt a pontot, amely estében egy-egy tulajdonság nem megengedett átlagtól való eltérőségét adjuk meg.

Az általam vizsgált fosszilis-megújuló energiacserés rendszerek esetében a Lépés1 után a SLépés2 –t csak úgy tudjuk választani, hogy nem képezünk alcsoportokat, viszont kontroll tulajdonságokat mind a négy oldalhoz rendelünk. Lépés3 lépéshez szakértői team bevonása célszerű az alapkritériumoknak megfelelően. Az elvégzett dominancia vizsgálatban öt tényezőcsoportot adtam meg, melyből négy került kiválasztásra. Ezeket a kocka oldalaihoz rendeltem és meghatároztam az egyes főtulajdonságok dominanciáját.

Feltételezett protokoll a kritériumok és kockatulajdonságok súlyozása fosszilis-megújuló energiacserés rendszerek esetében:

Step1:

- ✓ Pénzügyi hatások összesítése
- ✓ Technológiai feltételrendszer
- ✓ Stratégiai Programillesztés
- ✓ Piaci lehetőségek vizsgálata
- ✓ Jogi, szabályozási adaptálás

Step2:

Group1: a tulajdonságok súlyozása

1. Pénzügyi hatások összesítése: 90 (N)
2. Jogi, szabályozási adaptálás: 70 (Z)
3. Piaci lehetőségek vizsgálata: 60 (K)
4. Technológiai feltételrendszer: 50 (P)
5. Stratégiai Programillesztés: 30 (kieső tulajdonság)

Kontroll tulajdonság: Technológiai feltételrendszer (50 %)

Step3:

Pénzügyi hatások összesítése:	+
Jogi, szabályozási adaptálás:	+
Piaci lehetőségek vizsgálata:	+
Technológiai feltételrendszer:	0
Stratégiai Programillesztés:	-

A Churchman és Ackoff dominancia vizsgálat eredménye:

A főoldal dominancia vizsgálat eredményét összesítve megállapítható, hogy a projektet befolyásoló 4 fő tulajdonság csoport preferenciájának meghatározása Churchman és Ackoff módszerének átalakításával, jól megvalósítható. A főtulajdonság és kocka oldal/szín megfelelő biztonsággal párosítható, az alternatíva jósága (x_j) a következő függvénnyel írható fel (18. ábra):

$$x_j = \sum_{i=0}^{n} w_i a_i \qquad j = 1,2,3,4$$

$$w_i \ = i - dik\ szempont\ súlya\ w_i > 0; \quad a_i = i - dik\ szempont\ szerint\ érték\ a_i > 0$$

18. táblázat: Súlyozott tulajdonságok, fontosság és magyarázatok a dominancia vizsgálathoz

Tulajdonságcsoport	Súlyozott tulajdonságok (w_i)	Fontosság vizsgálat	Megjegyzés
1. Pénzügyi hatások összesítése	90	+	Narancs oldal (N) tulajdonsága, elsőségi preferencia, a kezdő oldalon
2. Jogi, szabályozási adaptálás	70	+	Zöld oldal (Z) tulajdonsága, a kocka rendszerint az óramutató járásának megfelelően, balra rendeződik először, így az erősebb tulajdonságokat az ellenkező irányba jelöljük ki.
3. Piaci lehetőségek vizsgálata	60	0	Kék oldal (K) tulajdonsága, legkisebb súllyal szerepel (ebben az esetben a feltétel nem igaz).
4. Technológiai feltételrendszer	50	0	Piros oldal (P) tulajdonsága, a kontroll tulajdonság helye, a kocka hátsó oldala, egyensúlypont.
5. Stratégiai Programillesztés	30	-	Kiesett tulajdonság. Vizsgálni kell a tulajdonság párosítását egyéb domináns tulajdonsághoz. Szakértői döntés értelmében a 2-es tulajdonsághoz integráljuk. Annak dominanciájában 2.7/9 arányban szerepeltetjük.

Forrás: saját szerkesztés

A 18. táblázatban látható rendezési szempont az, hogy a legdominánsabb tulajdonság kerüljön a top oldalra (jelen estben Narancs - N), illetve a leggyengébb tulajdonság a vele szemközti oldalra (jelen esetben a Piros - P). Ennek oka, hogy kapcsolati profiljuk leírása (ellentmondásokkal, hibákkal együtt) úgy kerülhet legjobban leírásra, ha két másik tulajdonságon (jobbról, balról) keresztülvezetve történik meg. A dominancia listában középen, gyengülésüknek sorrendjében elhelyezkedő tulajdonságok „gyengülésüknek" sorrendjében, az óramutató járásával ellentétes irányba kerüljenek kijelölésre. A sorrendezés lényege, hogy az erősebb dominanciát mutató tulajdonsághalmaz feltételezhetően rendezettebb állapotban van, míg a kisebb „dominanciát" mutató tulajdonságcsoport esetlegesen távolabb esik az egyensúlyi ponttól. A legerősebb relevanciát mutató tulajdonságcsoport lesz a kocka top oldalán, a legkisebb relevanciájú tulajdonságcsoport lesz a legdominánsabb tulajdonsággal szemben, az átellenes oldalon. A kocka helyre forgatása leggyakrabban az óramutató járásával megegyező irányba történik az egyes algoritmusokban, így azok az egyes lépések után a legrövidebb úton optimalizálódnak az egyensúlyi pont felé. Ezért az óramutató járásával ellentétes irányba (dominanciájuk sorrendjében) adjuk meg a másik két halmazt, így az erősebb kocka tulajdonságcsoport az induló narancs oldaltól jobbra eső zöld oldalra, a másik, illetve a sorban harmadik legdominánsabb tulajdonság pedig a kék oldalra kerül.

3.6. Hasznossági függvény alkalmazása a főoldalak tulajdonságainak összekapcsolására

A Rubik kockás projektfejlesztés egyik legfontosabb lépése az, hogy hogyan választjuk ki a vizsgálati szempontrendszer résztulajdonságait abból a tényezőcsoportból (főágensen belül), amelyet előzetesen a dominancia vizsgálattal meghatároztunk. A tényezőcsoportok, azaz kocka oldalakra vetített főágensek belső résztulajdonságait, a Rubik kocka egyszínű középkockája, élkockái és sarokkockái adják. „Kiskockák” kiválasztását meghatározza a Rubik kocka szerkezeti alaptulajdonsága, mely szerint az input oldalként kezelt fehér oldal, valamint az output oldalként kezelt sárga kockaoldalon kívül négy olyan, egymással szemben, illetve közvetlenül egymás mellett elhelyezkedő oldalt (tényezőcsoportot) reprezentáló felületet vehetünk figyelembe, amelyek a projektünket körülvevő feltételek optimalizálási folyamatában az optimum rendezés részét képezik. A kocka oldalakra vetített főágensek belső résztulajdonságainak, azaz „kiskockák” tulajdonságainak meghatározására, többváltozós hasznossági függvényeket használtam.

Többváltozós hasznossági függvények alkalmazása

A döntéshozóknak mérlegelniük kell a környezeti problémák megelőzését és azokat a közgazdasági nehézségeket, amelyek bizonyos termékek és folyamatok megszüntetése miatt jelentkezhetnek. A következmények hasznainak és költségeinek számbavétele nélkül dönteni nem lehet. Azokat problémákat, ahol a döntés kimenetele két vagy több attribútum jellemez, a több-attribútumú hasznosságelmélet kezeli. Általában minden attribútumról feltételezzük, hogy diszkrét vagy folytonos értékekkel rendelkezik. Az egyszerűség kedvéért feltételezzük, hogy az attribútumokat úgy határoztuk meg, hogy a nagyobb attribútum értékek nagyobb hasznosságértékekhez tartoznak, ha minden más változatlan.

Jelölhetjük az $\boldsymbol{x} = x_1,\ x_2, \dots\ x_{nb}$ az attribútumokat,

az attribútum értékek vektorát pedig $\mathbf{x} = \langle x_1, x_2, \dots, x_n \rangle$ összefüggéssel,

$U(x_1, \dots, x_n)$ hasznosságfüggvény megadásához.

A preferenciák rendszere és a több attribútumos hasznossági függvények értelmezése

A több attribútumos hasznosságelmélet azzal a feltételezéssel él, hogy a hasznosságfüggvények jól körülhatárolható strukturáltsággal rendelkeznek. Az elfogadott elméleti megközelítés az, hogy a viselkedés preferenciáiban várható szabályszerűségeket azonosítunk, és az úgynevezett reprezentációs tételeket felhasználva megmutatjuk, hogy az adott preferenciarendszerrel rendelkező tulajdonság leírható a következő hasznosságfüggvénnyel:

$$U(x_1, \dots, x_n) = f\,[f_1(x_1), \dots, f_n(x_n)]$$

ahol f remélhetőleg olyan egyszerű függvény, mint egy összeadás. Látható, hogy ez az összefüggés ahhoz hasonló, ahogy a valószínűségi hálókat használtuk az együttes valószínűségeloszlás-függvény felbontására. Ennek azért van kiemelt jelentősége, mert a Rubik kocka esetében is hálószerű kapcsolatban jelenítjük meg az egyes tényezőcsoportok, majd tényezők valószínűség-eloszlását.

Preferenciák bizonytalanság nélkül

A determinisztikus preferenciastruktúrában megtalálható alapvető szabály-szerűséget *preferenciafüggetlenségnek* nevezzük. Két attribútum, X_1 és X_2 preferenciálisan független X_3-tól, ha a preferencia $\langle x_1, x_2, x_3 \rangle$ és $\langle x_1', x_2', x_3' \rangle$ között nem függ X_3 konkrét értékétől.

Ennek a definíciónak – *preferenciafüggetlenség* - azért van kiemelkedő jelentősége a Rubik kocka módszertan dominancia meghatározás szempontjából, mivel ennek a „kiválasztásnak” az alapja az a gyors, és csak a meghatározó szempontok alapján történő optimalizálás, ami miatt többek között az indiai szoftverfejlesztő informatikusok is ezt a módszert (RCM) választották a régi szoftveralkalmazások felújítására.

Ha az X_1, ..., X_n tulajdonságok kölcsönösen preferenciálisan függetlenek, akkor az ágens viselkedési preferenciája leírható a következő függvény maximalizálásával:

$$V(x_1,\ldots x_n) = \sum_i V_i\,(x_i)$$

ahol minden egyes V_i egy értékfüggvény, amely csak az X_i attribútumnak a függvénye.

Az ilyen típusú értékfüggvényt *additív értékfüggvénynek* nevezzük. Az additív értékfüggvények egy teljesen illeszkedő módszert adnak a meghatározó tulajdonságok értékfüggvényének a leírására, és számos „valósvilág”-beli helyzetben érvényesek. Még ha a költségpreferencia függvény nem is teljesül teljes mértékben, ahogy ez az attribútumok szélső értékei esetén megtörténhet, egy additív értékfüggvény még mindig jó közelítést adhat a főtulajdonságok preferenciáinak. Ez különösen igaz, ha a költségpreferencia függvény csupán azon attribútum tartományokban sérül, amelyek a gyakorlatban csak kis valószínűséggel fordulnak elő.

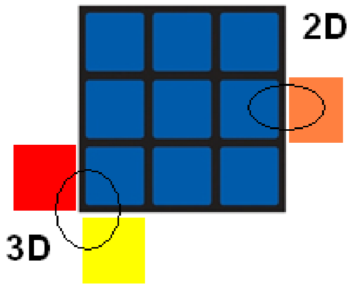
A fenti leírásból jól érzékelhető, hogy egy beruházási döntés meghozatala a klasszikus döntéshozatali mechanizmusok szerint támaszkodhat alapvetően a meghatározó ágensekre, főtulajdonságokra, és például olyan főtulajdonságok figyelembe vétele nélkül, mint

- a piaci és a szabályozási körülmények kockázatát,
- a környezetre, klímaváltozásra gyakorolt hatások stb.

hozhatjuk meg a támogató beruházási döntésünket. Rubik kocka kirakásán alapuló módszertan esetében kikötjük, hogy a kocka oldalak (főágens) belső tulajdonságai esetében nem hozhatunk döntést preferenciafüggetlenség alapján, viszont a főágensek egymás közötti kapcsolatának összehasonlítása során, ez alapvetően kötelező teendő.

A Rubik kocka alapján történő fenntarthatósági értékelés egyik legértékesebb jellemzője az, hogy a főágenshez (kockaoldal), vagy meghatározó tulajdonsághoz kapcsolódó hasznosságokat és kapcsolati rendszert együtt tudja kezelni az összes többi főágenssel (a kocka többi oldalához rendelt tulajdonsággal). A kétdimenziós (x,y) mellett, a háromdimenziós (x,y,z) kapcsolatokat is le tudja írni, illetve a kiforgatással az egyes tényezőket befolyásoló tulajdonságokat azonosítani képes. Ennek oka, hogy a kétszínű kockákhoz 2D tulajdonság értelmezést, a háromszínű kockákhoz 3D értelmezést társíthatunk, azaz olyan tulajdonságok kezelésére is alkalmas, amely egyszerre kettő vagy három főágenshez is tartozik.

A kockák által hordozott tulajdonságokat különbözőképpen értelmezhetjük. (Mivel a fehér és a sárga oldalak a vizsgálat input és output oldalait jelentik, funkcionális értelmezésük a kirakás során eltér a másik négy oldalszín értelmezésétől). Annak érdekében, hogy világos legyen a kapcsolat, a következő ábrán magyarázom az összefüggést (42. ábra). A 2D megjelölés azokra a kockákra és tulajdonságokra vonatkoznak, amelyek kétszínűek (pl. a 42. ábrán kék-narancssárga színű élkocka), a 3D megjelölés pedig azokra a kockákra, vagy kapcsolati tulajdonság jellemzőre vonatkozik, amelyek háromszínűek. A 43. ábrán a kék-piros-citromsárga színű sarokkocka nevezhető tipizáltan 3D tulajdonságnak.



43. ábra: A kockaoldal, vagy főágens 2D és 3D értelmezése
Forrás: saját szerkesztés

A 2D kockatulajdonság azt jelenti, hogy a kék élkocka olyan tulajdonságot hordoz, amelyet a másik főágens (narancssárga) is befolyásol. A 3D kockatulajdonság azt jelenti, hogy jelen esetben a kék sarokkocka olyan tulajdonságokat hordoz, amelyet másik két főágens (piros és sárga) is befolyásol, és ez oda vissza érvényes. A főágensek, vagy kockaoldalak alkotórészeinek, vagy kiskockáinak tulajdonságokkal való megjelölése tehát ennek az adott oldalon belüli elhelyezkedésnek is a függvénye kell hogy legyen. Így a kiskockák tulajdonsággal történő megjelölése, a főágensen belüli hasznosságuk, dominanciájuk meghatározása ennek figyelembe vételével megtörténhet. A hasznosságvizsgálatot a SMART (Simple Multi Attribute Ranking Technic) attribútum értékelő szoftverrel végeztem, melynek használatához a 12. ábrán bemutatott folyamatértékelést és játékelméleti optimalizálási szinteket vettem alapul. A 19. ábra jól szemlélteti, hogy a projekttervezési modell szintjei hogyan hangolódtak össze a SMART szoftveralkalmazással végzett attribútum vizsgálatokhoz.

19. táblázat: SMART vizsgálati szintek összehangolása a modellalkotási szintekkel

SMART VIZSGÁLATI SZINT	MODELL-ALKOTÁS SZINTJE	BEFOLYÁSOLT PROJEKTTULAJDONSÁG (1D, 2D, 3D = egy-, kettő-, három dimenziót hordoz)
1. SZINT	INPUT	„Fehér kereszt” – kiindulási peremfeltételek megadása (1D, 2D)
	INPUT	Fehér sarok kirakása – fenntartható fejlesztési irányok kijelölése, Egyensúlykeresés, nem kooperatív optimum (3D)
2. SZINT	KOCKA-KÖZÉP	Második sor kirakása - kapcsolati pontok rögzítése, egyensúly megteremtése, Kétdimenziós tényezők összerendezése, fixpont igazítás (1D, 2D)
	KOCKA-KÖZÉP	„Sárga kereszt” – input/output oldalak indirekt összehangolása (1D, 2D)
3. SZINT	OUTPUT	Sárga sarok kirakása – fenntarthatósági kritériumok értelmezése az outputok rendezése során (3D)
	OUTPUT	Sárga oldal élcsere – input/output oldalak szigorú összehangolása (2D)
	OUTPUT	Sarokcsere – a végső egyensúlyi állapot beállításának fázisa, egyensúlykeresés, fenntarthatósági kritériumok véglegesítése (3D)

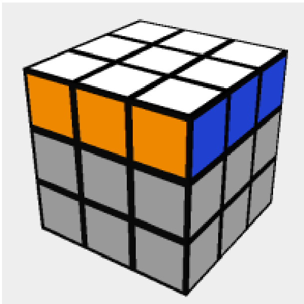
Forrás: saját szerkesztés

3.6.1. Háromszintű logikai vizsgálat értelmezése

A játékelméleti optimumkeresés hármas tagolása (input-középkocka-output) már jól mutatta számunkra, hogy a Rubik kocka kirakásra épülő projektfejlesztés tervezési szintjei ugyan követik a kirakási logikát, de célszerű nagyobb egységekre bontanunk az egyensúlykeresés folyamatát is, vagyis egyes fázisokat (NO_{1-7}) összevonnunk. A háromszintű vizsgálatok előkészítése során az első két kirakási fázis (NO_1, NO_2) az input szintre került. A következő két kirakási fázist (NO_3, NO_4) a középkockához rendeltem, majd az utolsó három kirakási fázist (NO_5, NO_6, NO_7) az output oldalhoz rendeltem. Így tehát a 3x3x3 Rubik kocka Layer by layer, azaz sorról sorra történő kirakásához kapcsolódó 7 fázist összevontam, és három elemzési szintet adtam meg, melyek a következők:

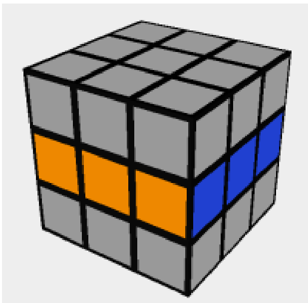
1. kockaszinten : Input oldal nem megengedett eltéréseinek korrigálására (43. ábra),
2. kockaszinten: Középkocka kapcsolatok nem megengedett eltéréseinek korrigálására (44. ábra),
3. kockaszinten: output oldal nem megengedett eltéréseinek korrigálására (45. ábra).
   (Megjegyzés: a „nem megengedett eltérést”, mint nem fenntartható tulajdonságot definiáltam a tulajdonsághalmazban.)

Az első kockaszinten vagy 'layer'-en a 9 darab kocka definiálásával 21 belső tulajdonság jelölhető meg. Ezek a szomszédos kockákkal együtt értelmezve, azaz élkocka (4x2) és sarokkocka (4x3) és egy középkocka (1x1) a többi főágens (sárga-, kék-, zöld-, piros színű oldalak) dimenzióban is értelmeződnek (44. ábra).



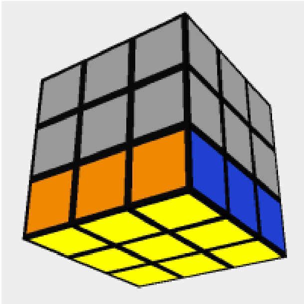
44. ábra: Első kockaszint vagy input oldal értelmezése
Forrás: saját szerkesztés

A következő szint a kocka középső értelmezési tartományának (1x1x3) definiálásával adhatók meg! Ebben az esetben 12 belső tulajdonságot viszünk be az értelmezési dimenzióba. Ez 4 db élkockát (4x2) és négy darab középkockát jelent (4x1). Ebben az esetben 8 db kockával kell kalkulálnunk, mint rendszerelemmel (45. ábra).



45. ábra: Második kockaszint vagy középkocka kapcsolatok értelmezése
Forrás: saját szerkesztés

Az értelmezési dimenziók (szintek) megadásának utolsó lépésében, az első lépéshez hasonlóan, a szomszédos kockákkal együtt értelmezve, azaz élkocka (4x2) és sarokkocka (4x3) és egy középkocka (1x1) a többi főágens dimenzióban adjuk meg. Így ugyancsak 21 belső tulajdonság adható meg az 4 főágenshez kapcsolódóan (46. ábra).



46. ábra: Harmadik kockaszint vagy output oldal értelmezése
Forrás: saját szerkesztés

A SMART értékeléshez tehát úgy képeztünk tulajdonsághalmazokat, hogy az első kockasor, a középső kockasor, valamint a harmadik kockasor közép-, él-, és sarokkockáit kirakási szintenként csoportosítottuk. Összefoglalva elmondható, hogy az értékkel jellemezhető kocka tulajdonságok esetében, észre kell vennünk, hogy 9+8+9= 26 db komplex tulajdonsággal le tudjuk írni a vizsgált rendszerünket, amely egyéb 54 különálló tulajdonságot hordoz a főágenseken keresztül (6x9). Ki kell emelni továbbá azt is, hogy komplex tulajdonságkezelés miatt (1D, 2D, 3D), az elemzés az egyes tulajdonságok kapcsolatrendszerét is világosan leírja.

3.7. SMART (Simple Multi Attribute Ranking Technic) elemzés

A kockák tulajdonságainak értelmezéséhez, a kiskockákhoz rendelhető tulajdonságok meghatározásához a SMART módszert (Simple Multi Attribute Ranking Technic) választottam, amely képes a 2D és 3D tulajdonságok együttes kezelésére és megjelenítésére. A módszertani részben leírtak szerint választottam ki a vizsgálati technikát, melyet a különböző attribútumok vizuális megjelenítésének vonatkozásában, egyedülálló szoftveralkalmazásnak számít.

A SMART elemzés menete a következő volt:

1. Főágensek dominancia vizsgálati adatainak értékelése, adatbevitel,
2. Vizsgálati szintek „kiskocka tulajdonságainak”, feltételezett hasznossági értékeinek meghatározása,
3. SMART táblázatok kialakítása és 3D –ben történő megjelenítés.

A Churchman és Ackoff dominancia vizsgálat eredményeit alapul véve, a SMART értékelés kiinduló adatsora a következő, melyhez a dominancia vizsgálat eredményei alapján határoztam meg az egyes színekhez rendelhető tulajdonságokat:

Group1: a tulajdonságok súlyozása
Pénzügyi hatások összesítése: 90 --------► narancs
Jogi, szabályozási adaptálás: 70 -----------► zöld
Piaci lehetőségek vizsgálata: 60 --------------► kék
Technológiai feltételrendszer: 50 -----------------► piros
Stratégiai Programillesztés: 30 -----------------► kieső tulajdonság

Az előző fejezetben választ kaptunk arra, hogy a Rubik kocka alapú projekttervezés folyamatában hogyan párosítjuk az egyes kirakási algoritmusokat a különböző szintekhez, a Churchman-Ackoff féle eljárással megkapjuk azt a 4 legfontosabb tulajdonságot a befolyásoló tényezők listájából, amelyeket a kocka oldalaihoz is párosíthatunk. Ha a fehér (F) az input oldal, akkor ebben az esetben a legdominánsabb tulajdonságcsoportot hordozza a narancs oldal (N), így meglesz az FN kiindulási oldalpárunk, majd tovább megyünk az óramutató járásával megegyező irányba a fehér oldal körül, és FK (fehér-kék) majd FB (fehér-piros) és FZ (fehér-zöld) oldalak jelölhetik a releváns tulajdonságokat (ágenseket) a tervezésben.

A rendezési szempont az, hogy a legdominánsabb tulajdonság kerüljön a top oldalra (jelen estben Narancs - N), illetve a leggyengébb tulajdonság a vele szemközti oldalra (jelen esetben a Piros - P). Ennek oka, hogy kapcsolati profiljuk leírása (ellentmondásokkal, hibákkal együtt) úgy kerülhet legjobban leírásra, ha két másik tulajdonságon (jobbról, balról) keresztülvezetve történik meg. A dominancia listában középen, gyengülésüknek sorrendjében elhelyezkedő tulajdonságok „gyengülésüknek" sorrendjében, az óramutató járásával ellentétes irányba kerüljenek kijelölésre. A sorrendezés lényege, hogy az erősebb dominanciát mutató tulajdonsághalmaz feltételezhetően rendezett állapotban van, míg a kisebb „dominanciát" mutató tulajdonságcsoport esetlegesen távolabb van az egyensúlyi ponttól. A legerősebb relevanciát mutató tulajdonságcsoport lesz a top oldalon, a legkisebb relevanciájú tulajdonságcsoport lesz a legdominánsabb tulajdonsággal szemben, a szemközti oldalon, majd az óramutató járásával ellentétes irányba (dominanciájuk sorrendjében) adjuk meg a másik két halmazt. A kocka helyre forgatása leggyakrabban az óramutató járásával megegyező irányba történik az egyes algoritmusokban, így azok az egyes lépések után a legrövidebb úton optimalizálódnak az egyensúlyi pont felé, ezért kerül az erősebb kocka tulajdonságcsoport az induló narancs oldaltól jobbra eső zöld oldalra, a másik, illetve a sorban harmadik legdominánsabb tulajdonság pedig a kék oldalra.

Hasznossági függvények megadása

A SMART szoftver az általunk leírt vizsgálati módszerhez képest általános függvény-definíciós módszert kínál. A következő struktúrában vázolt adatok és kritériumok egyszerűen bevezethetők a program adatbázisába és a beépített algoritmusok segítségével kiértékelhetjük azt.

KISKOCKA SZÁMA	**HASZNOSSÁG (1-100) (feltételezett)**	**KAPCSOLAT, DIMENZIÓ ÉRTÉK**	**SMART ÉRTÉK (V)**	**KOCKATÍPUS ÉS DOMINANCIA**	**FŐÁGENS BELSŐ TULAJDONSÁGOK**

A táblázatok feltöltése során a hasznossági függvények hasznossági értékké történő alakítása a következő lépéseken keresztül történik:

1. a maximális hasznosságot jelentő érték kap 100 pontot,
2. a minimális, vagyis 0 hasznosságot jelentő érték kap 0 pontot,
3. a két szélsőérték alapján meghatározandó az az érték, amely a maximálishoz képest fele hasznosságot jelent, ez 50 pontot ér,
4. a középső és a maximális hasznosság alapján meghatározandó az az érték, amely hasznosság szempontjából a kettő között helyezkedik el, ez 75 pont,
5. minimális és közepes hasznosság alapján megkeresendő az az érték, amely a hasznosság szempontjából a kettő között helyezkedik el, ez 25 pontot kap.

Hasonló módon további belső függvényértékek nyerhetők, az így kapott pontokhoz pedig matematikai módszerekkel, egy közelítő függvényt tudunk rendelni. Az alternatívák jóságát a hasznossági érték súlyozott számtani középértéke adja. A kapcsolatdimenzió értéket a 2-3. mellékletben is jól követhető módon, háromszintű kapcsolat (sarokkocka) esetében 3/3, kétszintű kapcsolatok (élkocka) esetében 2/3, a fix kocka esetében pedig 1/3 értékkel jelöltem. Ennek értelmében a felírható hasznossági függvény a főgráfra vagy főtulajdonságra a következő:

$$V(x_1, \ldots . x_n) = \sum_i V_{i=\frac{1}{3};\frac{2}{3};\frac{3}{3}} \left(x_i = \frac{\sum_{i=0}^{n} w_i a_i}{\sum_{i=0}^{n} w_i} \right) \qquad i = 1, \ldots . . n;\ n = 9$$

$$w_i = i - dik\ szempont\ súlya,\ w_i > 0;$$

$$a_i = i - dik\ szempont\ szerint\ érték,\ a_i > 0$$

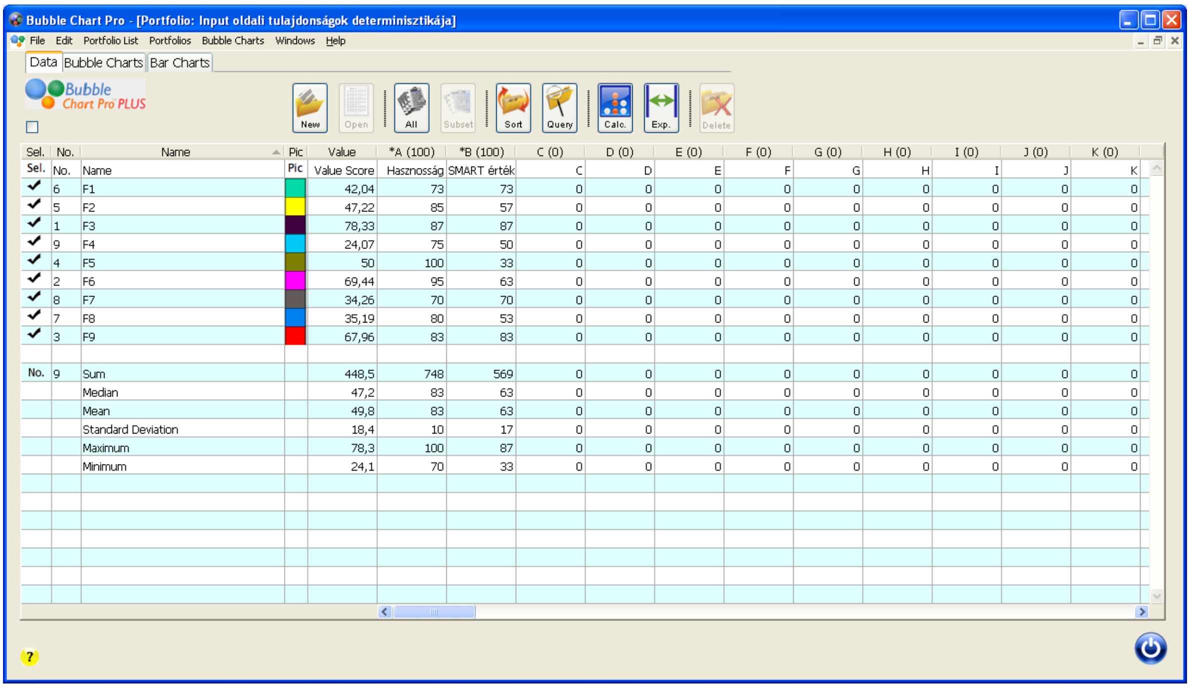
A fenti képlet alapján az egyes szintek SMART értékeinek megjelenítésével kapunk képet a főágensen belüli hasznosság különböző dimenziókat érintő jellemzőiről. A főágensek (N,P,Z,K) belső tulajdonságainak egymáshoz való viszonyát és az input oldal hasznossági jellemzőit láthatjuk a 20. táblázatban.

20. táblázat: SMART input értékek generálása az adatbevitelhez ('►X' = hasznosság)

FEHÉRKÖZÉP (1D) – energiaracionalizálás	Dominancia:100	(100) maximális érték
ÉLFEHÉR (2D) Dominancia: Narancs/ 90; Zöld/70; Kék/60; Piros/50		
stratégia, szabályozási alapkapcsolat (FZ),		100/70 ► 85,0
technológiai alapkövetelmény (FP),		100/50 ► 75,0
finanszírozási elvárás (FN),		100/90 ► 95,0
piaci alapillesztés (FK)		100/60 ► 80,0
SAROKFEHÉR (3D) –		
piaci igényű megtérülési alapfeltétel (FNK),		100/90/60 ► 83,3
pénzügyi eszközöknek és szabályozási feltételeknek való megfeleltetés (FNZ),		100/90/70 ► 86,6
techn. kockázatok és a piaci prioritások összehangolása az alapcéllal (FKP),		100/60/50 ► 70,0
stratégiákkal összehangolt technológia kijelölése az alapfeltételeknél (FZP),		100/70/50 ► 73,3

Forrás: saját kutatás

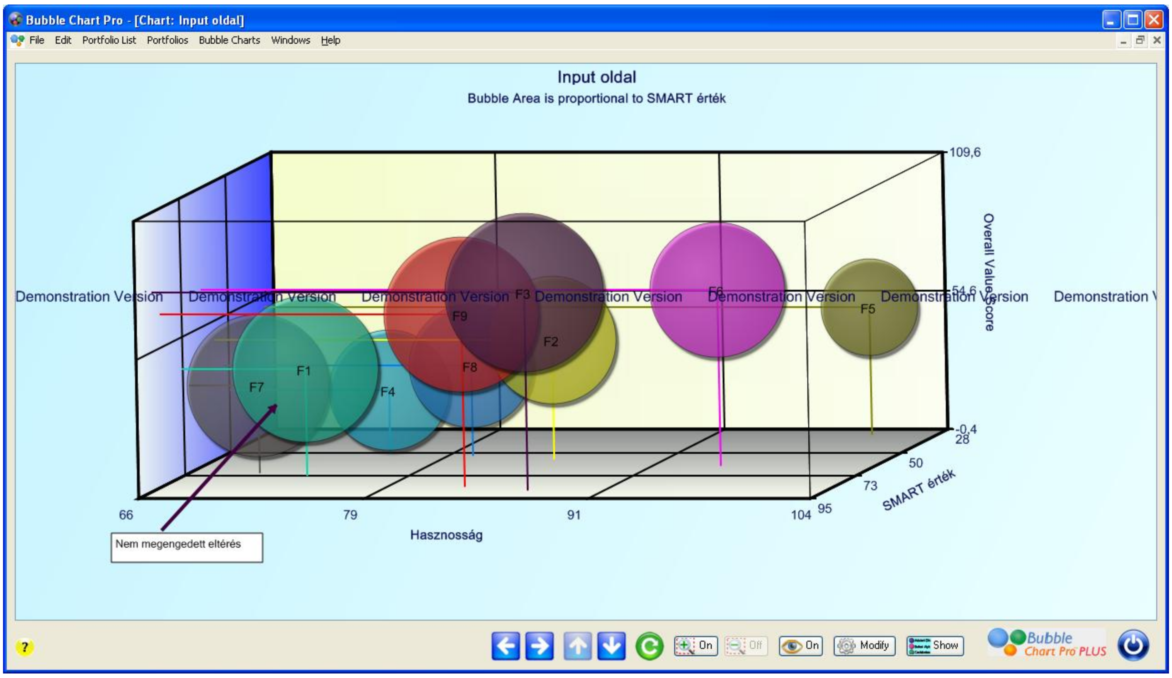
A 20. táblázatban generált adatok bevitelét követően a SMART Bubble Chart Pro demo verzióban lefuttatott "Value Score" értékpontozás alapján készíti a program a 3D tulajdonságmegjelenítést, amely az alkalmas arra, az összevetett tulajdonságok vizuális kiválasztással is megkülönböztethetők legyenek jóságuk, vagy stratégiai fontosságuk szerint. A SMART program input adattábláját a 47. ábrán láthatjuk.



Sel.	No.	Name	Pic	Value	*A (100)	*B (100)	C (0)	D (0)	E (0)	F (0)	G (0)	H (0)	I (0)	J (0)	K (0)
Sel.	No.	Name	Pic	Value Score	Hasznosság	SMART érték	C	D	E	F	G	H	I	J	K
✓	6	F1		42,04	73	73	0	0	0	0	0	0	0	0	0
✓	5	F2		47,22	85	57	0	0	0	0	0	0	0	0	0
✓	1	F3		78,33	87	87	0	0	0	0	0	0	0	0	0
✓	9	F4		24,07	75	50	0	0	0	0	0	0	0	0	0
✓	4	F5		50	100	33	0	0	0	0	0	0	0	0	0
✓	2	F6		69,44	95	63	0	0	0	0	0	0	0	0	0
✓	8	F7		34,26	70	70	0	0	0	0	0	0	0	0	0
✓	7	F8		35,19	80	53	0	0	0	0	0	0	0	0	0
✓	3	F9		67,96	83	83	0	0	0	0	0	0	0	0	0
No.	9	Sum		448,5	748	569	0	0	0	0	0	0	0	0	0
		Median		47,2	83	63	0	0	0	0	0	0	0	0	0
		Mean		49,8	83	63	0	0	0	0	0	0	0	0	0
		Standard Deviation		18,4	10	17	0	0	0	0	0	0	0	0	0
		Maximum		78,3	100	87	0	0	0	0	0	0	0	0	0
		Minimum		24,1	70	33	0	0	0	0	0	0	0	0	0

47. ábra: SMART program eredménytáblájának megjelenítése 2D-ben
Forrás: saját szerkesztés SMART program alapján

Az input oldal egyensúlyi helyzete instabil, ez az F1 tulajdonságon keresztül azonosítható, amely stratégiákkal összehangolt technológia kijelölése az alapfeltételeknél (FZP). A tulajdonságot és helyzetét a tulajdonság-halmazban a 48. ábrán láthatjuk. A gömbre kattintva megkapjuk az adott gömb esetében jellemző (x,y,z) koordinátákat is, melyek speciális hasznossági függvényeket jelentenek. A három dimenziós megjelenítés miatt, a 48. ábrán jól érzékelhető a tényezőkapcsolat, illetve a kapcsolatok mélysége.



48. ábra: SMART program ’output’ eredményeinek és ’nem egyensúlyos’ tulajdonságának megjelenítése
Forrás: saját szerkesztés SMART program alapján

A SMART program segítségével elvégeztem az input oldal (ezek követhetők 47. és 48. ábrán), a kockaközép oldal-tulajdonságok és az output oldal értékelését. Az értékeléshez kapcsolódó táblázatok és összesítő diagramok a 2., 3., 4. mellékletekben találhatók.

A vizsgálat összefoglalásaként elmondhatjuk, hogy a Rubik kocka oldalszíneihez, a tulajdonságcsoportok dominanciájának függvényében, sikeresen hozzárendeltem a projektfejlesztési szempontokat. A dominancia vizsgálat a lefuttatott SMART pilot programmal a következőképpen alakult, az eredmények összegzése a következő:

- ✓ A projekttulajdonságok input és output oldala, a fehér és sárga kocka-oldalra történő leképezésével megtörtént,
- ✓ A főtulajdonságok esetében a narancsszín dominált a legjobban, a piros szín pedig a legkevésbé, így a narancs a legmagasabb, míg a pirosa legalacsonyabb dominancia értékkel bír. Ha a narancsszín a frontoldalon van, akkor vele átellenesen találjuk a piros oldalt, jobbra pedig zöld, majd a narancstól balra a kék színű oldalt,
- ✓ A dominancia meghatároz alapján, a dominancia erősség függvényében felállított sorrendben a legerősebb tulajdonság a narancsszínhez, a második tulajdonság a zöld színhez, a harmadiktulajdonság a kék színhez, a leggyengébb tulajdonság a piros színhez rendelődik a módszertanban leírtak szerint,
- ✓ A játékelméleti módszerek alkalmazása, illetve a SMART program alkalmazhatósága érdekében a vizsgálati módszert három különböző tulajdonságcsoportot bevonva, szintekre bontottam: 1) Input ► 2) Kocka közép ►3) Output.
- ✓ A SMART program által az Input oldalon kijelölt nem megengedett eltérés alapján a „technológia kijelölése az alapfeltételeknél” tulajdonság esetében egyensúlytalanságot mutatott, így erre a feltételre állíthatunk be játékelméleti optimalizálást.
- ✓ A SMART program által az Kockaközép tulajdonság-oldalon kijelölt nem megengedett eltérés alapján „a projekt pénzügyi értéket, a megtérülési idő” tulajdonság esetében

egyensúlytalanságot mutatott, így erre a feltételre állíthatunk be játékelméleti optimalizálást.

- ✓ A SMART program által az Output oldalon kijelölt nem megengedett eltérés alapján a „piaci feltételrendszer, piaci instrumentumok és szolgáltatások egyensúlyba terelése" tulajdonság esetében egyensúlytalanságot mutatott, így erre a feltételre állíthatunk be játékelméleti optimalizálást.
- ✓ A fosszilis/megújuló energiacsere projekttervezési folyamatra elvégzett projektértékelési vizsgálatok eredménye tehát az, hogy a tervezés bemeneti (input) oldalához kapcsolódóan a „technológiai kérdések tisztázása, alapfeltételek egyensúlyba állítása" befolyásolja leginkább a fenntartható egyensúlyhoz közeli állapot elérését. A középső, vagyis összekötő tulajdonságcsoportban a „pénzügyi érték, megtérülés" kap túl nagy hangsúlyt, ezért ennek fenntarthatóbb értelmezése szükségszerű, ami közelíthet az említett fenntarthatósági gazdasági érték (FGE) fogalmához. A kimeneti oldalon (output) a piaci feltételrendszer, piaci felületek hiánya, egyensúlytalansága okozhat egyensúlyvesztést és sikertelen projekt megvalósítást, ennek egyensúlyi tervezése lehet jó stratégia.

3.8. Új és újszerű tudományos eredmények

Az irodalomelemzés és az elvégzett módszertani vizsgálatok eredményeit két csoportba sorolva mutatom be. Új tudományos megállapításként (M_n) közlöm azokat a tudományos értékű eredményeket, amelyek további kutatásokkal, vagy a vizsgálati adatok megfelelő szintű kiterjesztése révén, a későbbiekben tudományos tézisként is közölhetők.

Új tudományos eredmények címmel közlöm azokat a tudományos eredményeket (T_n), melyek jelen kutatás keretében új tudományos tézisként megfogalmazhatók.

3.8.1. Új tudományos megállapítások

M_1: A környezetvédelmi célú vagy a klímaváltozást előnyösen befolyásoló beruházások életképességét, fenntarthatóságát meghatározó projekt tulajdonságok kapcsolati rendszerét a Rubik kocka kirakás „*Layer by layer"* módszer, illetve a módszer egyes fázisaihoz rendelt játékelméleti modellekkel le lehet írni.

M_2: Gazdasági események fenntartható modellezésére vonatkozó egyes játékelméleti megoldások gyakorlati alkalmazását, illetve a helyes egyensúlyi állapotok kijelölését sok esetben a modellek sok tényező miatt fellépő bonyolultsága gátolja. Játékelméleti módszerekkel történő fenntarthatósági egyensúlykeresés során, az összehasonlításra kerülő tényezők egymás közötti kapcsolatának egyszerű függvényszerű leírását, illetve ezeknek a kapcsolatoknak tervezési fázisokra épülő, szintenként történő kezelésével célszerűen lehet megvalósítani.

M_3: A többváltozós próbafüggvényeket alkalmaztam azoknak a tulajdonságcsoportoknak a kiválasztására, amelyek egy-egy tervezési folyamatban olyan tulajdonságokkal jellemezhetők, melyek a fenntartható projekttervezés, kivitelezés folyamatát elterelik az egyensúlyi irányból. A projektfejlesztés különböző állapotaiban - dominancia analízissel és hasznossági függvények alkalmazásával - a projekt sikeres megvalósítását pozitív irányba tudjuk terelni.

M_4: A low-carbon projektfejlesztési vagy tervezési eljárás hipotézisem szerint egy paralel protokollnak nevezhető a Rubik kocka layer-by-layer kirakási módszerével. Az oldalak és az egyes színek projekttulajdonságokhoz rendelésével elérhetjük, hogy egy speciális, a fenntartható projektfejlesztési folyamatot valósíthassunk meg, melynél a fejlesztés és eljárás specifikumát az adja, hogy az egyes tényezők vagy tényezőcsoportok, mint egy-, két- vagy háromdimenziós rendszerelemek vehetők figyelembe a fejlesztési programok során. A low-carbon fejlesztések és

Rubik kocka kirakás célrendszere ugyanazt a vezérlőelvet követi, logikus és alacsony energia bevitel révén igyekszik eljutni az „egyensúlyi állapotba”.

3.8.2. Új tudományos eredmények

T_1: Összehasonlító vizsgálataim igazolták, hogy a 3x3x3-as Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatának. A „*Layer by layer*” kirakási módszer „7” forgatási algoritmusa közül az első kettő a beruházás tervezés *Input* tényezőivel korrelál, a harmadik és negyedik algoritmus a kezdő és befejező fázis kapcsolatrendszerét írja le, az ötödik, hatodik és hetedik fázis a kimeneti oldal tényezőivel állítható párhuzamba.

T_2: Vizsgálataim bizonyították, hogy a fenntarthatóságot is biztosító projektfejlesztési folyamat *Input* oldalának egyensúlyi állapota egyszerűen egy *konstans összegű játékkal* vagy *nem-kooperatív véges játékkal* is leírható. Az *Input* és *Output* oldalak kapcsolati rendszere, illetve az erre vonatkozó egyensúlyi állapot *konfliktus feloldási módszerrel, zérusösszegű játékkal vagy véges oligopol játékkal* is felírható. A fejlesztési folyamat *Output* oldalának egyensúlya pedig *kooperatív oligopol* játékkal vagy *Nash-egyensúlypontra* vonatkozó kooperatív stratégiákkal írható le.

T_3: Számításokkal igazoltam, hogy a Rubik kocka (3x3x3) záró sarokkockái az egyensúlypont vagy fenntarthatósági optimumkeresési folyamatban kulcsszerepet játszanak. A záró kocka három tulajdonság összehangolására épülő forgatási kombinációja az ideális Nash-egyensúly állapotát biztosíthatja a beruházási programok során. Ennek a beállításnak a hiányában nincs meg a végső összhang az *Input* és az *Output* oldalak között, a rendszer rugalmassága nagymértékben csökken, mivel nem adaptálta azokat a feltételeket, amelyek az életképességet, a rendszertulajdonságok esetleges változásához kapcsolódó adaptációs képességet jelentenék.

T_4: A Rubik kocka (3x3x3) alapján történő fenntarthatósági értékelés egyik legértékesebb jellemzője az, hogy az egyes kockaoldalakhoz, vagy az értékelést meghatározó tulajdonsághoz kapcsolódó hasznosságokat és kapcsolati rendszert együtt tudja kezelni az összes többi meghatározó tulajdonsággal (a kocka többi oldalához rendelt tulajdonsággal). A *kétdimenziós (x,y)* mellett, a *háromdimenziós (x,y,z)* kapcsolatokat is le tudja írni, illetve a forgatás renddel az egyes tényezőket befolyásoló tulajdonságokat azonosítani képes. A kétszínű élkockákhoz 2D tulajdonság értelmezést, a háromszínű sarokkockákhoz 3D értelmezést társíthatunk, azaz olyan tulajdonságok kezelésére is alkalmas, amely egyszerre kettő vagy három főtulajdonsághoz is tartozik.

T_5: Megmutattam, hogy a fenntartható üzleti tervezés folyamata jól leírható játékelméleti modellekkel, melyek a Rubik kocka layer by layer módszerének algoritmusára fenntartható projektfejlesztési folyamatot modelleznek. Vizsgálataim igazolták, hogy akkor tekinthető egy projektfejlesztési folyamat fenntarthatónak, ha a következő feltételek teljesülése biztosított:

A. A technológiailag elfogadható tervezési opció van jelen (túltervezés, elavulás elkerülése).
B. Pénzügyi fenntarthatóság és likviditás optimalizálása biztosított (biztonságos önfenntartás és jövedelemtermelés min. 10 éven keresztül).
C. Káros projekthatások elkerülése a kapcsolódó termékpályákon (funkciójában önmagának megfelelő rendszer).

4. KÖVETKEZTETÉSEK ÉS JAVASLATOK

Ahogy az a szakirodalmi feldolgozásból világosan kiderült, a fenntarthatóság értelmezésére a közgazdaságtanban rendkívül eltérő megközelítésekkel találkozhatunk. Alapvető dilemma tehát a fenntarthatósági célrendszerek/stratégiák megfogalmazásakor, hogy a *gyenge fenntarthatóság*, avagy az *erős fenntarthatóság* elméleti összefüggéseit alkalmazzuk az egyes megközelítések esetében. A természeti és mesterséges tőke viszonya határozza meg alapvetően a gyenge és erős fenntarthatóság közötti különbséget. A gyenge fenntarthatóság elmélete kimondja, hogy a természeti és mesterséges tőke egymással helyettesíthető viszonyban áll. Egy viszonylag új mutatószám, a teljes gazdasági érték (TGÉ) rendszere, mely alapvetően a környezeti értékek számbavételét is egy korszerű módon kezelő közgazdaságtani adaptáció, mégsem tudja megfelelően kezelni a pénz időértékéhez kapcsolódó erőforrás transzformációk kérdését. Erre a kérdésre adott jó választ a fenntartható gazdasági érték (rövidítve: FGÉ) mutatószám, mely egy olyan értékelési mód, amely képes – a lokális információk figyelembe vételével, azokat célirányosan felhasználva – integráltan bemutatni a természeti tőke mellett, a társadalmi és a technikai tőke-elemek változását, melyet a teljes gazdasági érték csak igen korlátozottan valósít meg. A Teljes Gazdasági Érték (TGÉ), mint értékmérő igen erősen támogatja a közgazdasági irányok egyik legújabb változatát, az un. „low-carbon economy” célrendszerét.

A „low-carbon economy”, azaz az alacsony szén-intenzitású gazdasági koncepció lényege, hogy a lokális gazdaság vagy piac szintjén, elsősorban az alacsony energia- és anyaginputtal működő struktúrákat preferálja, mely révén az erőforrások megfelelő kezelésének hosszú távú feltételeit is biztosítva látja. A fenntarthatósághoz kapcsolódó kritériumok értelmezése során ezért vehetjük alapul jogosan a „low-carbon economy” prioritásrendszerét, mely a fenntarthatósági kritériumok jelenleg legmodernebb szemléletben megfogalmazott, egyensúlykereső koncepciója. A fenntarthatósági értelmezések értelemszerűen az erőforrások egyensúlyt teremtő felhasználására keresnek megoldásokat, gyakorlati modelleket, megvalósítható értelmezéseket. Azt gondolhatjuk tehát, hogy a fenntarthatóság modellezésére a jövőben azok a tudományos megközelítések alkalmasak, melyek jól definiáltan és félreérthetetlenül képesek megoldásokat nyújtani. Dolgozatomban azt kívántam bizonyítani, hogy a játékelméleti stratégia vagy modellalkotás egy alkalmas eszköz a fenntartható egyensúlykeresés folyamatára, és módszereivel megfelelő elméleti alapokat adhat gyakorlat számára.

Szakirodalom áttekintés főbb következtetései:

- A fenntarthatóság gazdasági értelmezése még mindig sok olyan nehézséget hordoz magában, amely miatt leképezése nehezen jut el az operatív megvalósítás szintjére. A fenntarthatósági kritériumok és gazdasági feltételek ma leghatározottabban a „low-carbon economy” szakterületi leírásaiban érhetők tetten, melyet ezért mind a politikai döntéshozók, mind a gazdasági szereplők követhető, elfogadható környezetvédelmi és/vagy klímastratégiának tartanak.
- Az erőforrások kezelésének, fenntartható használatának kérdése a gazdasági egyensúlykeresés területén nyitott új lehetőségeket az elmúlt években, melyek közül egyik kiemelkedően érdekes és izgalmas megoldást a játékelméleti módszerek alkalmazása jelentett. Annak ellenére, hogy a gazdasági döntések vonatkozásában sok csalódás és bizalomvesztés is érte a játékelméleti stratégiai modellek alkalmazását, az új elméleti megközelítések, a megfelelő Nash-egyensúly keresés új módszerei, az egyszerűsített modellek alkalmazása, biztató jelenségnek tekinthetők a matematikai modellalkotás területéről.
- A szakirodalom feldolgozás során említést tettem arról az új „low-carbon” szoftverfejlesztési megközelítésről, amely a régi, elavult szoftverek esetében jelent egy kedvező újrahasznosítási irányt. A megoldással lényegesen kevesebb anyag és munkaráfordítás révén vagyunk képesek régi szoftveralkalmazásainkat megfeleltetni az új

fogyasztói elvárásoknak, azaz újra piacra vinni elavultnak hitt termékeinket. A szoftver regenerációs megoldás a Rubik kocka „layer by layer”, azaz sorról-sorra kirakás módszerének alkalmazásában rejlik, mely módszer imitálja a 3x3x3-as Rubik kocka kirakását, így rövid interakciókon keresztül tudja az új fogyasztói elvárásoknak is megfelelő állapotba (egyensúlyba) juttatni a fejlesztési folyamatot.

Saját kutatási eredményeim főbb következtetései:

- A fenntarthatósági alapelvek és a Rubik kocka klasszikus „layer by layer” kirakási módszere szinkronizálhatók egymással. A Rubik kocka felépítésének és kirakásának fenntarthatósági értelmezése, a fenntarthatóság és a Rubik kocka kirakási algoritmusok közötti kapcsolat vizsgálatai egyértelműen bizonyítják, hogy a kocka térszerkezeti egyensúlyi állapotához vezető folyamat egyensúlykeresési folyamat, mely esetünkben a környezetvédelmi folyamatok egyensúlykeresésének modellezésére is alkalmas.
- A játékelméleti stratégiai modellek kapcsán végzett elemzéseim jó mutatják, hogy a jelenlegi gyakorlatban több olyan gazdasági stratégiai modellel találkozhattunk, amelyek nem működnek igazán jól. A nem megfelelő működést alapvetően a modellek bonyolultsága, a sok-sok tényező, kritérium figyelembe vétele okozza. Annak érdekében, hogy a modellalkotás folyamata, és a modellek működési mechanizmusa ne essen a „túl komplikál és nem kérjük” kategóriába, olyan egyszerűsített, de mégis működő modell szükséges, ami jól átlátható, megbízhatóan feltölthető adatokkal és könnyen korrigálható. Erre a kihívásra kínál megoldást a játékelméleti modellek alkalmazásához kapcsolódó három szintű „unortodox játékelméleti optimumkeresési” modellem.
- A játékelméleti modellek, stratégiai optimumkereső rendszerek elemzése során megállapítottam, hogy sokkal célravezetőbb több kisebb különálló, és az üzleti környezetben másként reagáló, egyedi játékelméleti megoldásokkal jellemezni az egyensúlykeresés folyamatát, mint komplex soktényezős modellstruktúrákkal egyben függvényesíteni a folyamatot. A megújuló energiatermelés, azaz a fosszilis/megújuló cserére irányuló fejlesztési folyamatok esetében, három szintre bontva a fejlesztési programot, célszerű induláskor *nem–kooperatív játékelméleti módszert*, ezt követően *konstans állandó vagy zéróösszegű* játékot, majd a kimeneti feltételek megfogalmazására során *kooperatív Nash-egyensúlykeresést, többszemélyes oligopol játékkal* alkalmaznunk.
- Az általam felvázolt unortodox játékelméleti módszer a gyakorlati alkalmazás során feltételezi, hogy az egyes szakaszokra (input, kockaközép vagyis összekötő és output) időben rugalmasan módosítható függvénykarakterisztikák alkalmazunk, ezért sikeres megoldás lehet a bonyolultabb folyamatok modellezésére is, ha egymás után több egyszerű játékelméleti modellt rakunk optimumkeresési folyamatba. Az üzleti környezet változásának függvényében ezek módszerek/játékok változtathatók és rugalmasan adaptálhatók egyéb gazdasági feltételrendszerek esetében is.

5. ÖSSZEFOGLALÁS

A dolgozat témája a fenntarthatósághoz kapcsolódó ismérvek újszerű értelmezése játékelméleti modellekkel, mely kutatás célja elsősorban az, hogy a gazdasági és politikai instabilitás tudatában is fenntarthatóan tervezhetők legyenek a különböző környezetvédelmi célú vagy a klímaváltozást előnyösen befolyásoló beruházások.

A játékelméleti megoldásokat manapság nagyon gyakran használják a gazdasági élet stratégiaalkotási feladatainak, vagy egyéb műszaki, erőforrás felhasználási problémáinak megoldására. Kiemelt szerepük van olyan helyzetek megoldásában, amelyekben egyidejűleg több döntéshozó egymással ellentétes érdekeit kell figyelembe venni, és a helyzet kimenete nagyban függ a különböző döntéshozók egyéni döntéseitől, a kialakított döntési stratégiák egymásra gyakorolt hatásától. A játékelméleti megoldások sok esetben találnak megfelelő egyensúlypontot a döntések kialakításához, azonban mégis nagyon gyakran szembesülünk olyan helyzetekkel, ahol a körülmények befolyásoló tényezők sokasága, vagy hatásuk bonyolultsága miatt, a játékelméleti megoldások több egyensúlypontot is mutatnak, megnehezítve a helyes döntés kiválasztását, vagy még rosszabb, ha egyáltalán nem találnak megoldást az ismertetett (matematikai összefüggésekkel csak bonyolultan leírható) feltételek között.

A szakirodalmi feldolgozás során ezért kiemelt hangsúlyt fektettem a gazdasági érték és a fenntarthatóság viszonyának klasszikus és új szemléletű bemutatása, mivel a fenntarthatóság értelmezése még mindig sok olyan nehézséget hordoz magában, amely miatt leképezése nehezen jut el az operatív megvalósítás szintjére. Sokak szerint a fenntarthatóság értelmezésnek kulcsa, ha meg tudjuk fogalmazni a fenntarthatóság kritériumrendszerét és elvárásait függvényszerű kapcsolatokon keresztül is. Fenntarthatósági szempontok matematikai értelmezése, a fenntartható gazdasági egyensúly vagy vállalati stratégiák játékelméleti megközelítésének bemutatása, klasszikus és fenntartható gazdasági egyensúlypontok keresésének értelmezése olyan kihívás melyhez kapcsolódóan számtalan elmélet, tudományos dolgozat, generatív képletek és számtalan tudományos próbálkozás kapcsolódik már évtizedek óta, de nem igazán átütő sikerrel.

A dolgozatban bemutatott, a Rubik kocka funkcionálisan strukturált kirakásán alapuló, Layer by layer metódussal működő projektfejlesztési módszer játékelméleti megoldásokkal keresi a fenntartható projektfejlesztés optimum megoldását, de koncepcionálisan is alapvető különbséget jelent a korábbi megközelítésekkel szemben. A bemutatott Rubik logikai megközelítés lényege, hogy a gazdasági szereplőt, vagy szereplőket befolyásoló körülményeket külön-külön kezeljük, ezekhez tartozó egyensúlypontokat elsősorban lineárisan rendezhető játékelméleti megoldásokkal szintekre rendezzük, majd ezeket az egyensúlyponttal már rendelkező megoldásokat rugalmasan adaptáljuk. A módszer különlegessége abban rejlik, hogy a Rubik kocka megoldási metódusán keresztül újra és úja, elsősorban a fenntarthatóság kritériumait figyelembe vevő állapotba tereljük a fejlesztési folyamatokat.

A fent megfogalmazott „tiszta”, azaz biztos egyensúlypontot adó játékelméleti megoldásokat az a paraméter kiválasztási módszer garantálja, melyet a low-carbon projektfejlesztés során vezetünk be, és a lényegi tulajdonságokat a Rubik kocka megfelelő oldalaihoz, illetve kiskockáihoz rendeli a Churchman-Ackhoff eljárás és a SMART (Simple Multi Attribute Ranking Technic) hasznosságelemzés során.

A Rubik kocka kirakáson alapuló unortodox játékelméleti módszer így képes a jelenlegi gazdasági környezetet terhelő piaci hibák, externális, azaz a paci feltételrendszeren kívül elhelyezkedő, de azt jelentősen befolyásoló feltételrendszer feltérképezésére, mely létfeltétele a fenntartható környezetvédelmi beruházások megvalósításának. Társadalmi elvárás, hogy a fenntartható gazdasági struktúrákat támogató fejlesztések externális hatásainak pontos definiálásával, azok tipizálásával, az extern hatások összesítésével olyan (matematikai függvényekkel is leírható) törvényszerűségek kerüljenek megfogalmazásra, melyek a fenntarthatóság jelenleg érvényes kritériumrendszerét meg tudják fogalmazni a piaci szereplők, politikai döntéshozók számára.

Az előbbi logikai menetet követve, a dolgozatban bemutatott „low-carbon rubik kockás projektfejlesztési módszertan” tehát úgy próbál megoldást találni a fenntartható beruházási, technológiai és egyéb, elsősorban „cleantech” gazdaságfejlesztési folyamatok megvalósítására, hogy azok gyorsan és lehetőség szerint a legolcsóbban, viszont hosszú távon is életképesen megvalósíthatók legyenek. A „low-carbon rubik kockás projektfejlesztési módszertan” fejlesztése során kiemelt vizsgálati területnek tekintettem a megújuló energia beruházásokhoz kapcsolódó erőforrás-környezetet, mely meglehetősen sok ellentmondást hordoz magában, viszont elképesztően gyors fejlődésnek indult az elmúlt évek során. Az Rubik kockás módszer reményeink szerint megteremtheti annak a projektfejlesztési gyakorlatnak az alapjait, mely a jövőben kiküszöbölheti a nem fenntartható beruházások megvalósítását, és elkerülhetővé válnak az olyan fejlesztési folyamatok, amelyek a jóléti mutatókat leginkább csak negatív irányba terelik.

A fenntarthatóság értékrendjének, vagy gazdasági értelmezésének elemzése során világossá vált, hogy a hagyományos értékmérés rendszerei nem képesek olyan kritériumhalmazt megfogalmazni, amelyek valóban a fenntartható gazdasági értékelés rendszereit, a hosszú távon is életképes gazdasági struktúrák kialakítását alapozhatják meg. A játékelméleti alapokra támaszkodó Rubik logikás projektfejlesztés egy új szemléletű megközelítést jelent ebben a folyamatban.

Igazolt hipotézisek:

- H1: A környezetvédelmi célú vagy a klímaváltozást előnyösen befolyásoló beruházások életképességét, fenntarthatóságát befolyásoló projekt tulajdonságok kapcsolati rendszerét modellekkel le lehet írni.
- H2: Játékelméleti módszerekkel történő fenntarthatósági egyensúlykeresés során, az összehasonlításra kerülő tényezők egymás közötti kapcsolatának függvényszerű leírása megvalósítható.
- H3: A többváltozós próbafüggvények alkalmasak azoknak a tulajdonságcsoportoknak a kiválasztására, amelyek a projekt sikeres megvalósítását dominánsan befolyásolják.

Részben igazolt hipotézisek:

- H4: A 3x3x3 Rubik kocka egyes kirakó algoritmusaival a fenntarthatósági elvek szinkronizálhatók, a kocka oldalainak kapcsolatrendszere olyan térszemléletet és tervezési stratégiát ír le, amely új tudományos szemléletet ad a beruházás tervezés folyamatában.

Összefoglalóan elmondható, hogy jelen dolgozatban hitelt érdemlően bizonyításra került az a tudományos felvetés, mely szerint a Rubik kocka kirakási algoritmusa, nevezetesen a *„Layer by layer” módszer,* alkalmas a projektfejlesztés folyamatának modellezésére. Ugyanakkor, megfelelő játékelméleti modellekkel úgy reprezentálható az egyes projekttulajdonságok kapcsolati rendszere, hogy a különböző környezet és klímabarát beruházások mind a humánerőforrás tervezés, mind a környezeti feltételek megőrzése, javítása szempontjából is jól tervezhető, alacsony kockázatú gazdasági környezetben valósuljon meg. A Rubik kocka kirakási metódusa, az egyes karakterkockák (középkocka, élkocka, sarokkocka) kapcsolati rendszerének és helyreforgatásának értelmezése játékelméleti módszerek segítségével biztosítani tudja, hogy a legrövidebb idő alatt, viszonylag kis erőforrás felhasználási feltételek mentén tudjuk modellezni a fenntartható gazdasági érték koncepció által megfogalmazott kritériumrendszert.

6. SUMMARY

The topic of the dissertation is the novel approach of the characteristics related to sustainability by applying game theory models. The primary goal of the research is to support the planning process – despite the economic and political instability – of sustainable investments aiming environment protection or influencing climate change heavily.

Game theory solutions are used widely nowadays to carry out the tasks of strategy-creation, or to solve other technical, resource consumption-related problems in business life. They have important role in solving situations in which the opposing interests of more decision makers must be taken into consideration in the same time, and the circumstances of the situation depend highly on individual decisions of different decision makers, and on the effects of decision strategies on each other.

Game theory solutions may find the sufficient balance point to develop decisions in many cases, however, we usually face situations in which, due to the great number and difficulties of factors influencing the circumstances, game theory solutions show more balance points. This makes the selection of the right decision more difficult, or in a worse case, they cannot find solutions (which can only be written down difficultly by mathematic relations) among the circumstances. Therefore, during processing the relevant literature I emphasized the introduction of the classic and the new approach of the relations of economic value and sustainability, because the interpretation of sustainability still carries many difficulties, which causes the practical implementation to be hardly achieved.

Many think that the key of interpreting sustainability is to be able to conceptualise the criteria-system and requirements of sustainability also through function-like relations. The mathematical interpretation of sustainability factors, the introduction of sustainable economic balance or company strategies through a game theory approach, interpreting the search for classic and sustainable economic balance points are challenges for which many theories, scientific papers, generative formulas and a great number of scientific attempts have been connected for decades now, but none of them has been fully successful.

The project-development method shown in the dissertation, based on the functionally structured solving process of the Rubik’s Cube, which uses a layer by layer method searches for the optimal solution of sustainable project-development by using game theory solutions, uses conceptionally different approach compared to the earlier ones.

The essence of the shown Rubik logical approach is that the factors influence the actors of the economy or the circumstances around the actors are handled individually. The balance points related to them are organised to levels by primarily linearly organisable game theory solutions, then these solutions possessing balance points are flexibly adapted. The specialty of the method lies in the fact that through the method of solving the Rubik’s Cube, the development processes can be carried out in a way in which primarily the criteria of sustainability are considered, again and again.

The above conventionalized „pure” game theory solutions, in other words, the ones which give certain balance points are guaranteed by the selection method, which are introduced during the low-carbon project-development, and the important characteristics are assigned to the corresponding sides of the Rubik’s Cube and to the small tiles during the Churchman-Ackhoff method and the SMART (Simple Multi Attribute Ranking Technic) utility analysis.

This way, the unorthodox game theory method - based on the solving of the Rubik’s Cube – is able to map the external criteria-system of market mistakes (which is outside the market criteria-system, but it influences is greatly) burdening the economic environment, and it is the primary criteria of existence of carrying out sustainable environment-protection investments.

It is a social requirement, that through exactly defining, typing and summarizing external effects of developments supporting sustainable economic structures, patterns (being able to written down by mathematic functions) should be conceptualised, which are able to conceptualise the currently relevant criteria-system of sustainability for market actors and political decision makers.

Following the previous, „the low-carbon Rubik's Cube project-development method" shown in the dissertation tries to find solution for the materialisation of sustainability, technological, and other, primarily „cleantech" economic development processes, that those would happen quickly, the cheapest way possible, but feasibly on the long run.

The resource-environment related to renewable energy-source investments were considered as primary research field during the development of „the low-carbon Rubik's Cube project-development method". These contain much opposition, however, it has improved very quickly throughout the past few years. The Rubik's Cube method can, hopefully, create the basis of that project-development practice, which can eliminate the implementation of non-sustainable investments in the future, and development processes can be avoided which direct the indicators of welfare to a negative trend.

It became obvious during the analysis of the set of values or the economic interpretation of sustainability that the traditional value measuring systems are not able to create such criteria-system which provides a basis of a system for sustainable economic evaluation, and for the design of economic structures feasible on the long run. The project development, using Rubik's logic, and which is based on game-theory gives a new approach in this process.

Proven hypotheses:

- H1: The connection system of project characteristics, which influence the feasibility, sustainability of investments aiming environment protection or which influence climate-change positively, can be written down by models.
- H2: During the search for balance, by using game-theory methods, the function-like description of connection between the factors to be compared can be done.
- H3: The multivariable test functions are capable of selecting the characteristic-groups, which predominantly influence the successful implementation of a project.

Partly proven hypotheses:

- H4: By the individual solving algorithms of the 3x3x3 Rubik's Cube the sustainability principles can be synchronised, the connection-system of the sides of the cube writes down a space-approach and planning strategy, which gives new scientific approach during the planning process of investments.

As a summary, it was proven in this dissertation – in a credible way – that the scientific theory, according to which the solving algorithm of the Rubik's Cube, the *„Layer by layer"* method, respectively, is capable to create models for the processes of project-development. However, the connection-system of certain project characteristics can be represented with the corresponding game-theory models in a way that different environment- and climate-friendly investments can be planned easily, from human resource view, and from the view of preserving, improving environmental factors as well, in a low-risk economic environment.

Understanding the solving method and the connection-system of the Rubik's Cube, its character tiles (middle, corner and edge tiles) can ensure – by the game-theory methods – that the criteria-system conceptualised by sustainable economic value concepts can be modelled through the shortest time, with the consideration of using relatively little amount of resources.

MELLÉKLETEK

M1: IRODALOMJEGYZÉK

1. **AJAY**, J. (2011) Rubik’s Cube Model of Software Engineering for Incremental and legacy projects. Journal of Computing, Volume 3. Issue 2. Februar 2011 pp. 99-101.
2. **AXELROD**, R. (1984) The Evolution of Cooperation. New York, Basic Books, 1984. pp. 3-8. http://www-ee.stanford.edu/~hellman/Breakthrough/book/pdfs/axelrod.pdf
3. **AXELROD**, R. (2000) On six advances in Cooperation Theory/ The evalution of Cooperation – University of Michigan, 2000 pp. 5-9
4. **CIE** (2012-2014) Cleantech Incubation Europe program, Documents for downloads. http://cleantechincubation.eu/downloads
5. **CHURCHMAN,** C.W**.** – ACKOFF, R.L. (1957) Introductions to Operational Research, Wiley, New York, 1957 (IFORM Resource Collection) pp. 45
6. **DAVID,** T. (2002). Adventures in group theory: Rubik's Cube, Merlin's machine, and Other Mathematical Toys. Baltimore: Johns Hopkins University Press. ISBN 0-8018-6947 p. 7
7. **DAVIS,** T**.** (2006) Group Theory via Rubic’s Cube. Geometer Org, http://geometer.org/rubik/group.pdf pp.10-12
8. **DÉNES,** T. (2005) A Ruwix program grafikája és megoldási képlete, programleírása. 2005Forrás: Ruwix.com/Online Ruwix Cube Solver program p.1
9. **DEMAINE,** E. et al. (2011) Algorithms for Solving Rubik”s Cubes, MIT Computer Science, Cambridge, USA, Medford, 2011 pp. 4-7 http://arxiv.org/pdf/1106.5736v1.pdf
10. **DOIG A.** (2000) Community planning and management of energy supplies - international experience. Renewable Energy. 2000; pp.325-331. https://sites.google.com/site/journalofcomputing/ www.journalofcomputing.org, p. 99
11. **DIETZ,** E. (2007-2010) About me. USA, Minnesota, 2007, pp. 4-5 http://www.wrongway.org/?programmer)
12. **FREY,** A. – Singmaster, D. (1982) Handbook of Cubik Math, Enslow, 1982. App.: This book supplements Singmaster's Notes on Rubik's Magic Cube, by taking a closer look at the group theory behind the cube, 1982 pp. 44-45.
13. **FOGARASSY,** C**.** (2012) Karbongazdaság (Low-carbon economy). L’Harmattan Kiadó, Budapest, 2012 pp. 23-24
14. **FOGARASSY,** C. – Herczeg, B. – Szőke, L. Balogh, K. (2012) Low-carbon project development (Rubic’s Cube solutions) - Sustainable energy and material management. Hungarian Agricultural Engineering, 24/2012 Gödöllő, pp. 5-9.
15. **FOGARASSY,** C. (2013) Low-carbon - Rubik kockás projektfejlesztési módszer az energia- és anyaghatékonyságot javító projektekben. Mezőgazdasági Technika, Gödöllő, LIV. évfolyam 2013. tavasz/különszám pp. 3-4
16. **FOGARASSY,** C. -KAPOSZTA, J., -NAGY, H. (2007) Externalitiy aggregation of the field of biomass production. Engineering for Rural Development, 6th Scientific Conference, Latvia-Jelgava 2007 pp. 122
17. **FORGÓ,** F. **–** PINTÉR, M. – SIMONOVICS, A. – SOLYMOSI, T. (2005) Játékelmélet. Elektronikus jegyzet, BME Budapest, OTKA T046196 2005 pp. 17
18. **FORGÓ,** F. - SZÉP, J.- SZIDAROVSZKY, F. (1999) Introduction to the Theory of Games. Kluwer Academic Publisher, Dordrecht, 1999 pp. 22
19. **GOWDY,** J. – Erickson, J. (2005) The approach of ecological economics. Cambridge Journal of Economics 2005, 29/2005 pp. 209-2012 http://steadystate.org/wp-content/uploads/Gowdy_Erickson_EE_Approach.pdf
20. **GOUDEY**, C. (2003) All about the Rubik’s Cube. Cubeland. http://www.cubeland.fr.st/ , 2003 pp.1
21. **GREGORY,** Intermaggio (2007) Solve a Rubik’s Cube like a Pro. pp.1 http://www.instructables.com/files/orig/F8T/N90H/FABRWSIA/F8TN90HFABRWSIA.pdf
22. **HARDIN,** G.: The tragedy of the Commons. Science 162, 1968 pp. 1241–1248.
23. **HARDWICK,** C. (2014) Solving the Rubik's Revenge. pp.1-2

http://www.speedcubing.com/chris/4-solution.html
24. **HARSÁNYI,** J. (1995) A racionális viselkedés alapjai. Magyar Szociológiai Társaság, Tanulmányok, 1995/ 4. szám, Budapest 1995 pp. 1-5.
25. **HEISE**, R. (2002) Human Thistlethwaite Algorithm. pp.1 http://www.ryanheise.com/cube/human_thistlethwaite_algorithm.html
26. **HOBSON,** E. (2012) Can Game Theory be Applied to Family Law Negotiations?: Round Two. Family Law Dispute Resolution, 2012 pp. 3-7 http://www.riverdalemediation.com/wp-content/uploads/2009/07/Hobson-Can-game-theory-be-applied-to-family-law-negotiations-round-two.pdf
27. **HUHN,** G. (2013) How Smart Project Ranking Works. Quick Start Guide. http://www.bubblechartpro.com/files/Bubble_Chart_Quick_Start_Guide.pdf
28. **ICHIISHI,** T. (1983) Game theory for economic analysis; Academic press, New York 1983 pp.10
29. **JOYNER,** D. (2002a). Adventures in group theory: Rubik's Cube, Merlin's machine, and Other Mathematical Toys. Johns Hopkins University Press. ISBN 0-8018-6947-1. http://mike.verdone.ca/media/rubiks.pdf pp.22-30
30. **JOYNER,** D**.** (2002b). Adventures in group theory: Rubik's Cube, Merlin's machine, and Other Mathematical Toys. Johns Hopkins University Press. ISBN 0-8018-6947-1. http://mike.verdone.ca/media/rubiks.pdf pp. 48-50
31. **JOYNER,** D. (1996a) Mathematics of the Rubik's cube. Furman Univesity, Mathematical Quotations Server, Forrás:http://math.furman.edu/˜mwoodard/mquot.html pp. 248.
32. **JOYNER,** D**.** (1996b) Mathematics of the Rubik's cube. Furman Univesity, Mathematical Quotations Server, Forrás:http://math.furman.edu/˜mwoodard/mquot.html pp. 250.
33. **KEREKES,** S. (2005) A fenntarthatóság közgazdaságtani értelmezése. In: Fenntartható fejlődés Magyarországon. Stratégiai Kutatások – Magyarország 2015, MTA/MH Új Mandátum Könyvkiadó, Budapest, 2005 p. 196-198.
34. **KÉRDŐ,** A. (2008) A tartalomelemzés elmélete és gyakorlata. BGF Budapest, 2008 http://elib.kkf.hu/edip/D_13763.pdf p. 7-10
35. **KISS,** KÁROLY szerk. (2004) Környezetvédelmi szempontból káros támogatások. Lélegzet Alapítvány, Budapest, 2004. p. 6-10.
36. **KIRAKÁSOK: 3X3X3** (2013) RUBIK KOCKA HIVATALOS OLDALA (2013) http://www.rubikkocka.hu/3x3-kirakas/
37. **KUNKLE,** D. – Cooperman, G. (2007) Twenty-Six Moves Suffice for Rubic's Cube, ISSAC'07, July 29–August 1, 2007, Waterloo, Ontario, Canada, 2007 pp. 3-8
38. **KORF, R.** E. (1997) Finding optimal solutions to Rubik's Cube. Using Pattern Databases, AAAI-97 Proceedings.1997, AAAI (www.aaai.org). pp. 700-703 http://www.aaai.org/Papers/AAAI/1997/AAAI97-109.pdf
39. **KORTEN,** D. (2013) Sacred Earth a New Economy and the 21st Century University, University of British Columbia, CIRS 2013, pp. 3-10 http://livingeconomiesforum.org/sites/files/KortenSacredEarth.pdf
40. **KORTEN,** D. (2011) Taking ecological economics seriously: Is the Biosphere, stupid. USSEE Society for Ecological Economics June 19, 2011, Lansing, Michigan, 2011 pp. 3-5
41. **LAUNONEN,** M. (2011) Hubconcepts - The Global Best Practice for Managing Innovation Ecosystems and Hubs, Hubconcepts Inc., Hiilikatu 3, 00180 Helsinki, Finland www.hubeconcept.com pp. 5-25
42. **LIGETI,** Gy. (2006) Sztereotípiák és előítéletek. Társadalmi Riport/2006 pp. 382-399 http://www.tarsadalomkutatas.hu/kkk.php?TPUBL-A-717/publikaciok/tpubl_a_717.pdf
43. **LCE** LTD. (2011) About a Low-carbon Economy. http://www.lowcarboneconomy.com/LCE/ pp. 3-5.
44. **MÁLOVICS,** GY. – BAJMÓCY, Z. (2009) A fenntarthatóság közgazdasági értelmezése. Közgazdasági Szemle, LVI. évf., Budapest, 2009. május, pp. 464-483.
45. **MEZŐ,** I. (2011a) Játékelmélet. Unideb.hu, Debreceni Egyetem, Debrecen, 2011, pp. 6-8

46. **MEZŐ,** I. (2011b) Játékelmélet. Unideb.hu, Debreceni Egyetem, Debrecen, 2011, pp. 9
47. **MÉSZÁROS,** J. (2005) Játékelmélet, Gondolat Könyvkiadó Kft., Budapest, 2005 pp.12-15 http://szoc.tarstud.hu/survey/jatekelmelet/jelmossz.pdf
48. **MIEA** (2005) Mesterséges Intelligencia. Modern megközelítésben/Mesterséges Intelligencia Elektronikus Almanach, 2005 pp. 16.2 http://project.mit.bme.hu/mi_almanach/books/aima/ch16s02
49. **MOLNÁR,** F. (2005) A fenntarthatósági gazdasági érték (FGÉ), avagy a gazdaság és a gazdasági érték egy tartalmazó rendszerek felőli megközelítése. BCE PhD dolgozat, Budapest, 2005 pp. 8-10.
50. **MOLNÁR,** S – Szidarovszky, F. – Molnár, M. (2010a) Játékelmélet és döntési módszerek. Szent István Egyetem. Egyetemi jegyzet, Gödöllő, 2010 pp. 25-27
51. **MOLNÁR,** S. – Szidarovszky, F. – Molnár, M. (2010b) Játékelmélet és döntési módszerek. Szent István Egyetem. Egyetemi jegyzet, Gödöllő, 2010 pp. 30-33
52. **MOLNÁR,** S. – Szidarovszky, F. – Molnár, M. (2010c) Játékelmélet és döntési módszerek. Szent István Egyetem. Egyetemi jegyzet, Gödöllő, 2010 pp. 72-77
53. **MOLNÁR,** S. (2010) Környezetinformatikai modellek. Szent István Egyetem. Gödöllő, 2010, pp. 191
54. **MOLNÁR,** S. (1994) On the optimization of INPUT-OUTPUT systems cost functions, Pure Mathematics and Applications, Vol. 5. No. 4, 1994, pp. 404
55. **MOLNÁR,** S. - SZIDAROVSZKY, F. (1994) A diszkrét idejű oligopólium játékok stabilitásáról, SZIGMA, Vol. 25. No. 3, Budapest, 1994 pp. 103-105
56. **MOLNÁR,** S. - SZIDAROVSZKY, F. (1995) A folytonos idejű termelői-fogyasztói modellek stabilitásáról, SZIGMA, Vol. 26. No. 3-4., Budapest, 1995 pp. 95
57. **MOLNÁR,** S. – SZIDAROVSZKY, F. (2011a) Játékelmélet. Többcélú optimalizáció, konfliktuskezelés, differenciáljátékok. ComputerBooks, Budapest, 2011 pp. 16-17
58. **MOLNÁR,** S. – SZIDAROVSZKY, F. (2011b) Játékelmélet. Többcélú optimalizáció, konfliktuskezelés, differenciáljátékok. ComputerBooks, Budapest, 2011 pp. 18-19
59. **MOLNÁR,** S. – SZIDAROVSZKY, F. (2011c) Játékelmélet. Többcélú optimalizáció, konfliktuskezelés, differenciáljátékok. ComputerBooks, Budapest, 2011 pp. 46-47
60. **MOLNÁR,** S. – KELECSÉNYI, S. (2009) Hungarian National Climate Change Strategy Mitigation, adaptation and a low carbon economy. CIRCLE-APM, 2009. pp. 22.
61. **NAGY,** G. (2008a,b,c) Megoldáskereső Módszerek. Informatikai szakdolgozat. Debreceni Egyetem Informatikai Kar, Debrecen, 2008, pp.12-16, pp. 22-25, pp. 40-45
62. **NCST 2010-2020** (2011) „Magyarország megújuló energia hasznosítási cselekvési terve 2010-2020", Nemzeti Fejlesztési Minisztérium, Budapest, 2011, pp. 13-20 http://videkstrategia.kormany.hu/download/2/7e/60000/Meg%C3%BAjul%C3%B3%20Energia%20Hasznos%C3%ADt%C3%A1si%20Cselekv%C3%A9si%20Terv(1).pdf
63. **NEMZETI ENERGIASTRATÉGIA 2030** (2012) Nemzeti Energiastratégia – National Energy Strategy, Nemzeti Fejlesztési Minisztérium, Budapest, http://www.terport.hu/webfm_send/2657 pp. 50-70.
64. **NEUMANN,** J. – Morgenstern, O. (2007) Theory of games and Economic behavier. Sixtieth-Anniversry Edition. Princeton University Press, New Yersey, 2007 pp. 13.
65. **NOWAK, M.** – May, R. (1992) Evolutionary games and spatial chaos. Nature 359, 1992 pp. 826–827.
66. **NUCLEAR POWER** in France (2014) World Nuclear Association in France. WNA, 2014. http://www.world-nuclear.org/info/Country-Profiles/Countries-A-F/France/ pp. 2-4
67. **OECD** LEDS (2010) Low-emission development strategies (LEDS): Technical, Institutional and Policy lessons. OECD/IEA France, Paris, pp. 10 http://www.oecd.org/environment/cc/46553489.pdf
68. **ORTEGA,** V. – JELINEK, J. (2013) A corners-first method for Rubik's Cube kirakási módszer leírása és képletei, http://cube.misto.cz/_MAIL_/cfsm.html p.1

69. **OSE,** L. S. (2008) Applying the Churcman/Ackoff Value Estimation Procedure to Spatial Modeling, MGIS Capstone Presentation, Penn State University – Word Campus, MDA Federal Inc, 2008 pp. 8-9
70. **PEARCE,** D.W.; - Atkinson, G.D. (1993). "Capital theory and the measurement of sustainable development: an indicator of weak sustainability". Ecological economics **8**: pp. 105–108
71. **PEARCE,** D. W. – ATKINSON, G. D. (1995): Measuring sustainable development. In: Bromley, D.(szerk.): The handbook of environmental economics. Blackwell Publishers, Cambridge, 1995 pp.167–168
72. **RASK-** THE BLOGGER (2013) Rubik's Cube Algorithms List. forrás: http://rakstheblogger.hubpages.com/hub/Rubik-Cube-Algorithms p. 1
73. **RICHARD,** B. (2008) A Rubik-kocka legnehezebb rejtvények nyomában. Informatika és Tudomány sg.hu, 2008 http://www.sg.hu/cikkek/76302/20_mozdulattal_kirakhato_barmely_rubik_kocka
74. **ROAD Map 2050** (2013) A practical guide to a prosperous low-carbon Europe. ECF The Hague, 2013, http://www.roadmap2050.eu/
75. **RUBIK, E. et al.** (1987) Rubik's Cubic Compendium, by Ernő Rubik, Tamas Varga, Gerzson Keri, Gyorgy Marx and Tamas Vekerdy, Oxford University Press, 1987. pp. 6-7
76. **RUBIK,** E. (szerk.) (1981) A bűvös kocka. Műszaki Kiadó, Budapest, 1981 pp. 2-22
77. **RUBIK'S REVENGE** (2011) Rubik's Revenge leírás. http://en.wikipedia.org/wiki/Rubik's_Revenge
78. **RUBIK** KOCKA WIKI (2013) http://hu.wikipedia.org/wiki/Rubik-kocka
79. **ROKICKI,** T**.** et al. (2010) God's Number is 20. http://www.cube20.org/ p.1
80. **ROKICKI,** T. (2008) Twenty-Five Moves Suffice for Rubik's Cube. Source: http://tomas.rokicki.com/rubik25.pdf pp. 1-4
81. **RUSSELL,** S. – NORVIN, P. (2003) Artificial intelligence. A modern approach. 2nd Edition. 2003, 1995 Pearson Education, magyar fordítás: Mesterséges intelligencia. Modern megközelítésben/Mesterséges Intelligencia Elektronikus Almanach, 2005 pp. 11.6 http://project.mit.bme.hu/mi_almanach/books/aima/index
82. **SALAZAR,** R. – SZIDAROVSZKY, F. – ROJANO, A. (2010) Water distribution Scenarios int he Mexican Valley. Water Resources Management, 24 (12) pp. 2959-2960
83. **SIMONGÁTI,** GY. (2009a) STPI (Fenntartható közlekedés mutatója) kidolgozása a belvízi hajózás fenntarthatóság elve szerinti értékeléshez, PhD. Disszertáció, BME KK, Budapest, 2009 pp. 56-57.
84. **SIMONGÁTI,** GY (2009b) STPI (Fenntartható közlekedés mutatója) kidolgozása a belvízi hajózás fenntarthatóság elve szerinti értékeléshez, PhD. Disszertáció, BME KK, Budapest, 2009 pp. 60.
85. **SIMONOVITS**, A. (2003) Neumann János és a játékelmélet Természet Világa, 2003. III. különszám, Neumann-emlékszám Különszámaink /Neumann emlékszám, pp. 1. (http://www.termeszetvilaga.hu)
86. **SINGMASTER**, D. (1981) Notes on Rubik's Magic Cube, Pengium Book, Enslow, 1981 pp. 22
87. **SLOCUM,** J. – Singmaster, D.- Wei Hue Vang - Gebhart, D. (2009) The Cube: The Ultimate Guide to the World's Bestselling Puzzle. Secrets, Stories, Solutions. Black Dog & Leventhal Publishers, March 2009. pp. 15-19
88. **SNAP,** R. (2012) Rubik's magic cube (12), University of Vernon, 2012 pp. 3-10 http://www.cems.uvm.edu/~rsnapp/teaching/cs32/lectures/rubik.pdf
89. **SPASH,** C. (1999) The development of environmental thinkig in economics. Environmental Values, vol. 8. pp. 425-427 http://www.clivespash.org/1999_Spash_EV_Development.PDF
90. **SPASH,** C. (2011) Social Ecological Economics: Understanding the past to see the future. American Journal of Economics and Sociology 70 (2) pp. 359-360 http://www.clivespash.org/2011_Spash_AJES_Social_Ecol_Econ.pdf

91. **START UP GUIDE** (2012) Üzleti tanácsok fejlődő kisvállakozók részére, Online publikáció, magyar nyelvű, http://startupguide.hu, pp. 2.2.
92. **SOLYMOSI,** T. (2009) Kooperatív játékok. Magyar Tudomány. A Magyar Tudományos Akadémia Lapja. 2009/05 pp. 547 http://www.matud.iif.hu/2009/09maj/05.htm
93. **SZIDAROVSZKY,** F. (1978a) Nash-féle kooperatív megoldási koncepció általánosításáról. Szigma, Budapest, 1978 pp. 70
94. **SZIDAROVSZKY,** F. (1978b) Játékelmélet. ELTE TTK jegyzet, Budapest, 1978 pp. 7-9.
95. **SZIDAROVSZKY,** F. – MOLNÁR, S. (1986) Játékelmélet műszaki alkalmazásokkal. Műszaki Könyvkiadó, Budapest 1986 pp. 220-225
96. **SZIDAROVSZKY,** F. - MOLNÁR, S.- MOLNÁR, M. – FOGARASSY, CS. (2013) Modeling Conflicing Interests in Water Distribution Problem with Game Theory. University of Arisona. (2013) pp.162-165
97. **SZILÁGYI,** M. (2005): Találkozás a játékelmélettel, Beszélő folyóirat, 2005. június–július, Évfolyam 10, Szám 6 pp. 1
98. **SZTNH** (2013) Magyar feltalálók és találmányok. Ernyey Gyula: "Tárgyvilágunk 1896-1996", Budapest-Pécs, 1998 alapján. pp.1 http://www.sztnh.gov.hu/feltalalok/rubik.html?printable=1
99. **TÓTH,** L. – FOGARASSY, CS. (2012) Low-carbon energiaellátási rendszerek a gyakorlatban. Szaktudás Kiadó Ház Zrt., Budapest, 2012 pp. 115.
100. **TURNER, R.** K. – Perrings, C. – Folke, C. (1996) Ecological Economics: Paradics or Perspectives. Cserge Working Papers GEC 95-17, UK London, 1996 pp. 24-26. http://www.cserge.ac.uk/sites/default/files/gec_1995_17.pdf
101. **TURNER, R.** K. – Pearce, D. – Bateman, I. (1993) Environmental economics. An elementary intorduction. The Johns Hopkins University Press, Baltimore, 1993 pp. 16-30
102. **VINCZE, J.** (2009) A játékelmélet és gazdasági intézmények. Magyar Tudomány. A Magyar Tudományos Akadémia Lapja. 2009/05 pp. 568 http://www.matud.iif.hu/2009/09maj/07.htm
103. **WIKI RUBIK KOCKA** (2013) A Rubik Kocka, mint térbeli logikai játék. (http://articles.portal-tol.net/english-language-hu/Rubik-kocka)

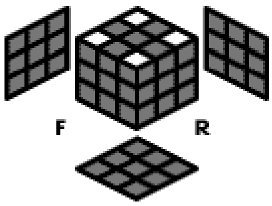
M2: CORNER-FIRST MÓDSZER

A Corners-First solution method for Rubik's cube
kirakási módszer leírása, képletei

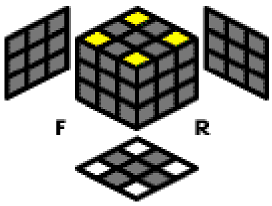
Victor Ortega and Josef Jelinek leírása alapján

Stage I: Solve the corners

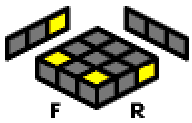
1. Orient U corners



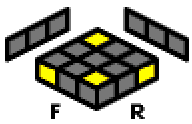
2. Orient D corners



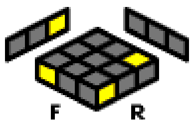
Rotate the whole cube so that D becomes U. Orient the corners depending on which of the seven patterns below you see:



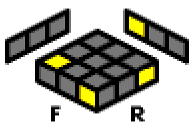
letter T pattern:
R U R' U' F' U' F



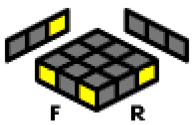
letter L pattern:
F R' F' U' R' U R



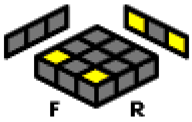
sune pattern #1:
R U2 R' U' R U' R'



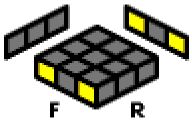
sune pattern #2:
R U R' U R U2 R'
(inverse of #1 in both respects)



letter pi pattern:
R U R2 F' R2 U R'

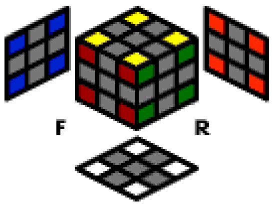


letter U pattern:
R' F' U' F U R

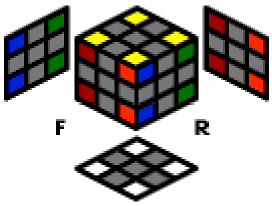


letter H pattern:
R2 U2 R' U2 R2

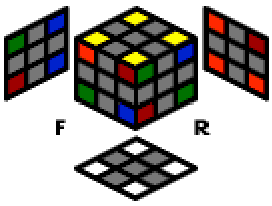
3. Permute all corners, by method of number of solved "edges"



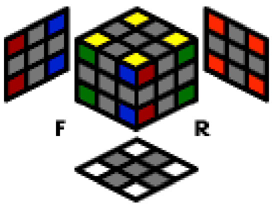
Proceed with one of the following sequences depending on how many solved edges you have:



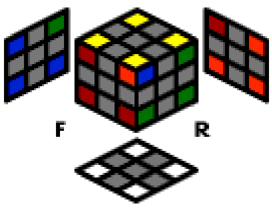
0:
F2 R2 F2



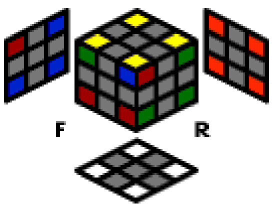
1 (DB solved):
R U' F U2 F' U R'
((UB solved): R' U R' B2 R U' R)



2 (DB and UB solved):
R2 U F2 U2 R2 U R2



4 (D solved):
F2 U' R U' R' U F2 U R U R'

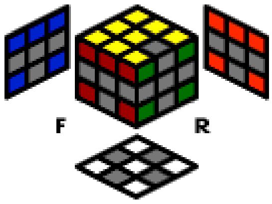


5 (UF not solved):
R U' R F2 R' U R F2 R2

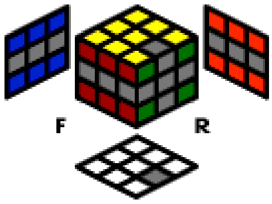
Stage II: Solve the edges

At this point, align corners and position centers. The cube is now fully symmetric except for edges.

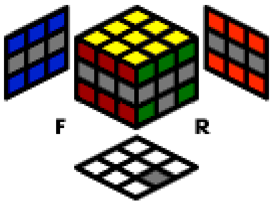
4. Solve three U edges

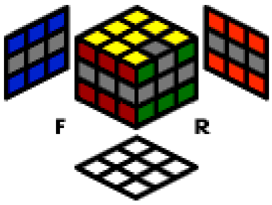


5. Solve three D edges



6. Solve one more U or D edge, depending on which is easier

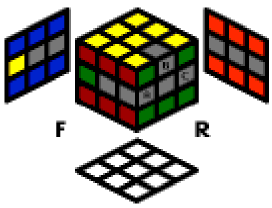


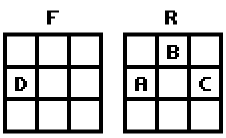


7. Solve last U edge, orient middle layer

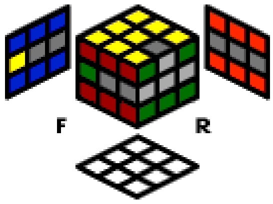
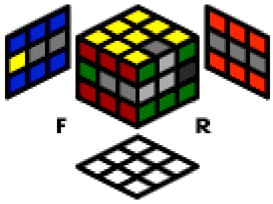
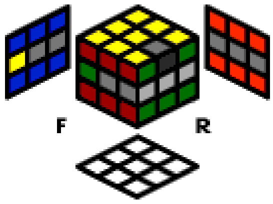
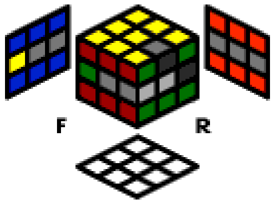
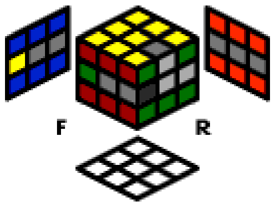
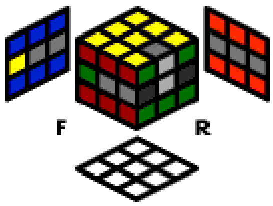
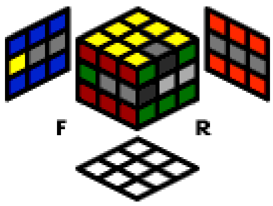
a) U edge in the middle layer

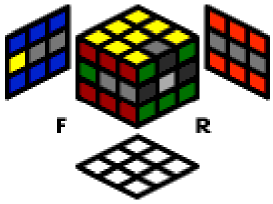
In the diagram below, edges A, B, or C are oriented correctly (C) if that facelet's color matches the adjacent center or the opposite center. Otherwise it's incorrectly oriented (I).





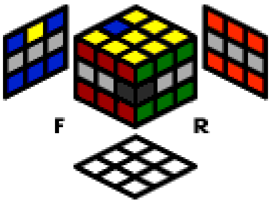
(facelet D belongs in facelet position B)
("adjacent center" between A and C)

Edge A	Edge B	Edge C	Pattern	Sequence
C	C	C		⇨R ⇨R' ⇦R' ⇨R
C	C	I		⇨R' ⇦R' ⇨R' ⇦R'
C	I	C		⇦R ⇨R2 ⇨R
C	I	I		R' ⇦R ⇨R' ⇨R
I	C	C		R2 ⇨R ⇦R ⇨R ⇦R'
I	C	I		R' ⇨R' ⇦R' ⇦R ⇨R ⇦R
I	I	C		R' ⇦R' ⇨R ⇨R

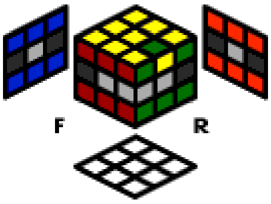
I	I	I		**R' ⇨R' ⇨R' ⇦R2 ⇦R**

b) U edge in position but twisted

There will be 1 or 3 twisted edges in the middle layer:



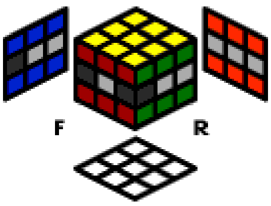
FR twisted:
R U2 R' ⇨⇨R2 ⇦R' U2 R'



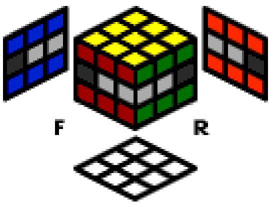
FR not twisted:
R' ⇦R' ⇦R' ⇦R'

c) U edge solved

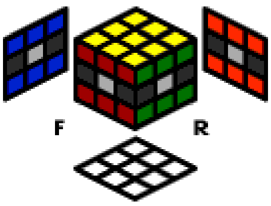
There will be 0, 2, or 4 edges twisted in the middle layer:



2 adjacent (FL and FR):
R2 F ⇩F' R2 F ⇧F'

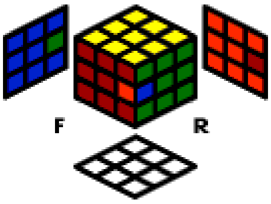


2 opposite (FL and BR):
F ⇩F' R2 F ⇧F' R2

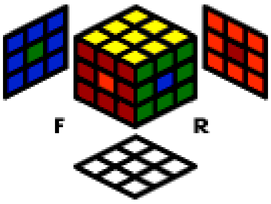


all four:
R F2 R2 ⇨R ⇨⇨R F2 R2 ⇨R
all four, when all edges are positioned correctly:
F' U' F (⇨ R)4 F' U F

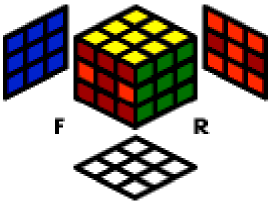
8. Position middle layer



send FR to BL, BL to BR, and BR to FR:
R2 ⇦R2 (⇨)



exchange centers with opposites:
⇩⇩⇦⇩⇩(⇨)



exchage FR with BR, FL with BL:
R2 ⇨⇨R2 (⇨⇨)

►READY

M3: SMART TÁBLÁZAT ÉS KIMENETI ÁBRÁK AZ INPUT OLDALI TULAJDONSÁGOKRA

Input (fehér) oldali tulajdonságok determinisztikája:

SMART input értékek generálása az adatbevitelhez (’►X’ = hasznosság)

FEHÉRKÖZÉP (1D) – energiaracionalizálás Dominancia:100 (100) maximális érték

ÉLFEHÉR (2D) Dominancia: Narancs/ 90; Zöld/70; Kék/60; Piros/50

I. stratégia, szabályozási alapkapcsolat (FZ), 100/70 ► 85,0
II. technológiai alapkövetelmény (FP), 100/50 ► 75,0
III. finanszírozási elvárás (FN), 100/90 ► 95,0
IV. piaci alapillesztés (FK) 100/60 ► 80,0

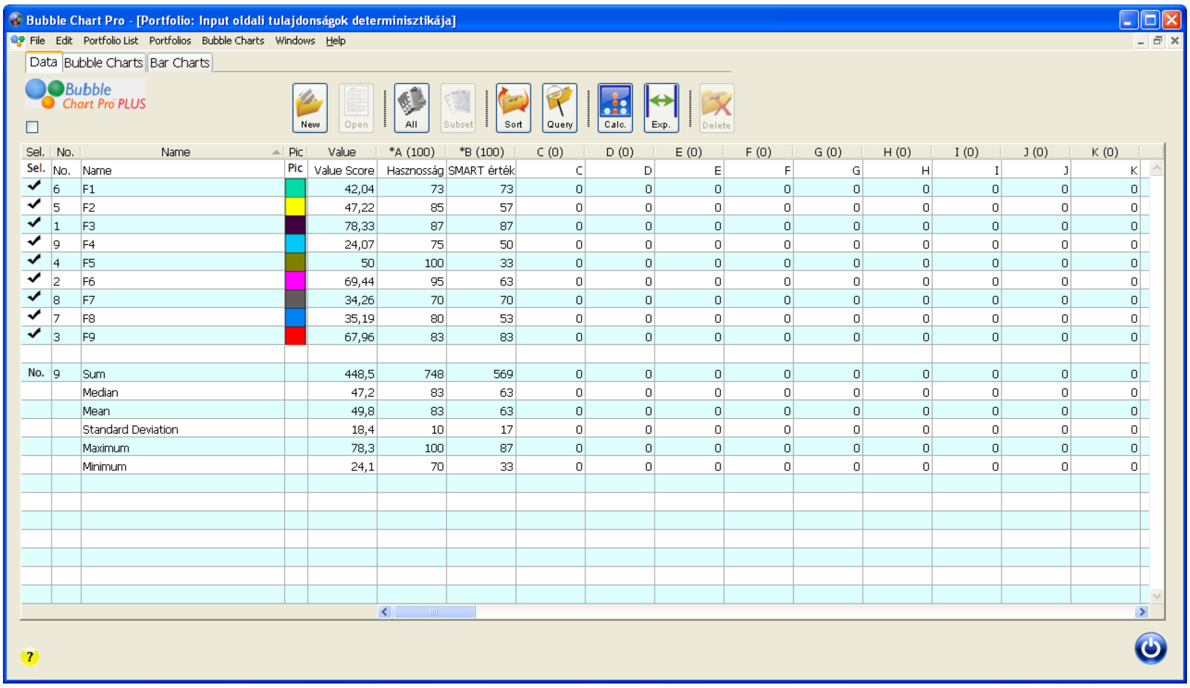
SAROKFEHÉR (3D) –

I. piaci igényű megtérülési alapfeltétel (FNK), 100/90/60 ► 83,3
II. pénzügyi eszközöknek és szabályozási feltételeknek való megfeleltetés (FNZ), 100/90/70 ► 86,6
III. techn. kockázatok és a piaci prioritások összehangolása az alapcéllal (FKP), 100/60/50 ► 70,0
IV. stratégiákkal összehangolt technológia kijelölése az alapfeltételeknél (FZP), 100/70/50 ► 73,3

KISKOCKA SZÁMA	HASZNOSSÁG (1-100) (kerekített)	KAPCSOLAT, DIMENZIÓ ÉRTÉK	SMART ÉRTÉK	KOCKATÍPUS ÉS DOMINANCIA	FŐÁGENS BELSŐ TULAJDONSÁGOK
F1	**73**	3/3	**73**	sarokkocka, átlagon felül domináló tulajdonsággal	stratégiákkal összehangolt technológia kijelölése az alapfeltételeknél (FZP)
F2	**85**	2/3	**57**	**élkocka**, átlagon alul domináló tulajdonsággal	stratégia, szabályozási alapkapcsolat (FZ)
F3	**87**	3/3	**87**	sarokkocka, átlagon felül domináló tulajdonsággal	pénzügyi eszközöknek és szabályozási feltételeknek való megfeleltetés (FNZ)
F4	**75**	2/3	**50**	**élkocka**, átlagon alul domináló tulajdonsággal	technológiai alapkövetelmény megléte (FP)
F5	**100**	1/3	**33**	KÖZÉPKOCKA, tulajdonság dominancia összehasonlítása a többi középkockával lehetséges	energiaracionalizálás (F)
F6	**95**	2/3	**63**	**élkocka,** átlagon felül domináló tulajdonsággal	finanszírozási rendszer, deviza kockázati tényezők és globális hatások elemzése (FN)
F7	**70**	3/3	**70**	sarokkocka, átlagosan domináló tulajdonsággal	techn. kockázatok és a piaci prioritások összehangolása az alapcéllal (FKP)
F8	**80**	2/3	**53**	**élkocka,** átlagon alul domináló tulajdonsággal	piaci alapillesztés (FK)
F9	**83**	3/3	**83**	sarokkocka, átlaggal azonos vagy azon felül domináló tulajdonsággal	pénzügyileg stabil és piaci igényű megtérülés (reális megtérülést biztosító technológiai megoldás) (FNK)

Forrás: saját kutatás

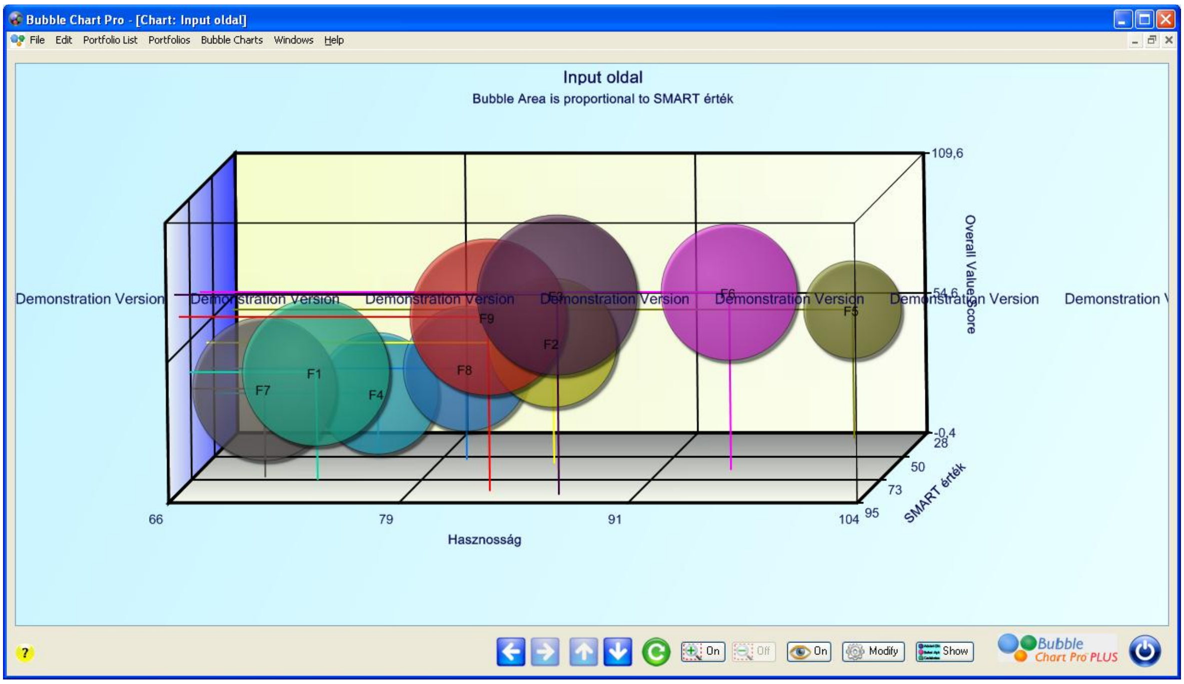
Input oldal karaktertáblája a SMART programban:



Bubble Chart Pro - [Portfolio: Input oldali tulajdonságok determinisztikája]

Sel.	No.	Name	Pic	Value	*A (100)	*B (100)	C (0)	D (0)	E (0)	F (0)	G (0)	H (0)	I (0)	J (0)	K (0)
Sel.	No.	Name	Pic	Value Score	Hasznosság	SMART érték	C	D	E	F	G	H	I	J	K
✓	6	F1		42,04	73	73	0	0	0	0	0	0	0	0	0
✓	5	F2		47,22	85	57	0	0	0	0	0	0	0	0	0
✓	1	F3		78,33	87	87	0	0	0	0	0	0	0	0	0
✓	9	F4		24,07	75	50	0	0	0	0	0	0	0	0	0
✓	4	F5		50	100	33	0	0	0	0	0	0	0	0	0
✓	2	F6		69,44	95	63	0	0	0	0	0	0	0	0	0
✓	8	F7		34,26	70	70	0	0	0	0	0	0	0	0	0
✓	7	F8		35,19	80	53	0	0	0	0	0	0	0	0	0
✓	3	F9		67,96	83	83	0	0	0	0	0	0	0	0	0
No.	9	Sum		448,5	748	569	0	0	0	0	0	0	0	0	0
		Median		47,2	83	63	0	0	0	0	0	0	0	0	0
		Mean		49,8	83	63	0	0	0	0	0	0	0	0	0
		Standard Deviation		18,4	10	17	0	0	0	0	0	0	0	0	0
		Maximum		78,3	100	87	0	0	0	0	0	0	0	0	0
		Minimum		24,1	70	33	0	0	0	0	0	0	0	0	0

Input oldal tulajdonságainak 3D ábrázolása a SMART programban:



A 3D ábrázolás alapján nem megengedett eltérést mutat az egyensúlyi állapottól az F7 illetve F1 tulajdonságokra. A vizsgálatok alapján a technológia feltételrendszer (pl. vízhasználat) egyensúlyba terelése, azaz játékelméleti optimumkeresése javasolt.

M4: SMART TÁBLÁZAT ÉS KIMENETI ÁBRÁK AZ OUTPUT OLDALI TULAJDONSÁGOKRA

Output (sárga) tulajdonságok determinisztikája:

SMART output értékek generálása az adatbevitelhez (’►X’ = hasznosság)

SÁRGAKÖZÉP (1D) – erőforrás-hatékony energiafelhasználás/ jövedelmező termelés (S) (100)

ÉLSÁRGA (2D)

I.	technológiai kockázat minimalizálása (PS)	100/50 ►75
II.	az energia- és CO2 mérleg stratégiai megfeleltetése (ZS)	100/70 ►85
III.	adózási és kedvezményezési feltételek az energiatermelő rendszerben (NS)	100/90 ►95
IV.	mesterséges és valós piacra lépés pénzügyi feltételei (KS)	100/60 ►80

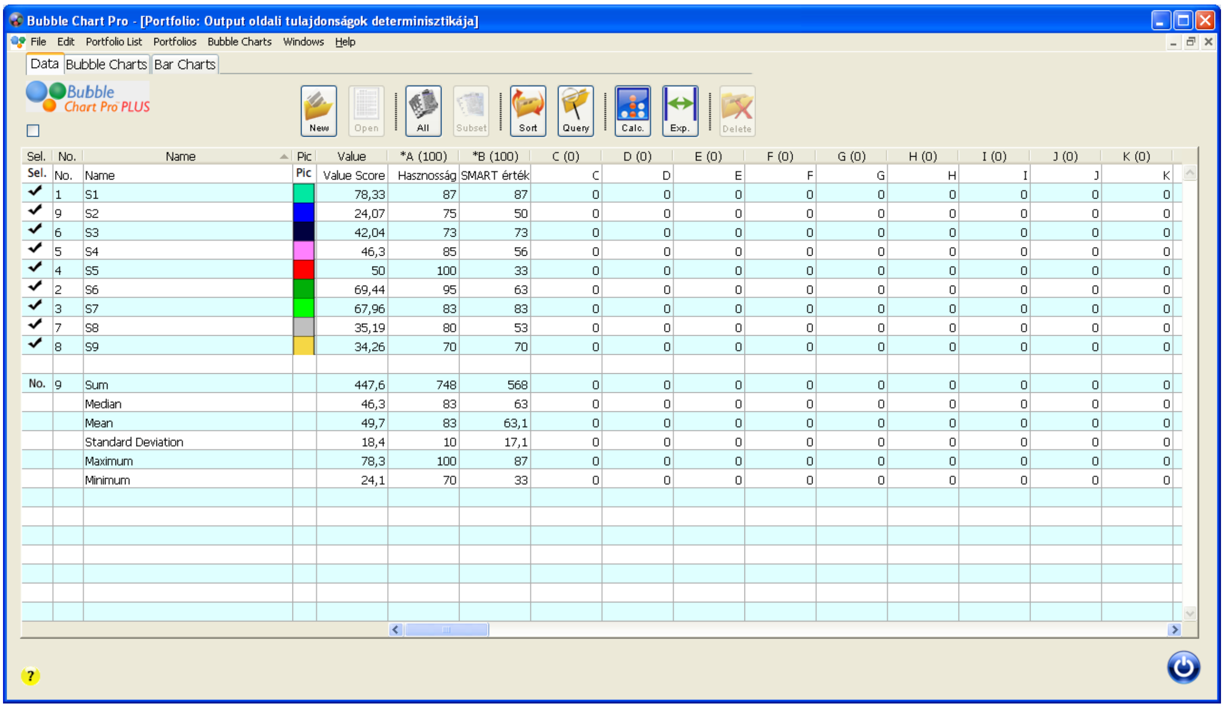
SAROKSÁRGA (3D)

V.	stratégia célrendszerekhez illeszkedő technológia és szabályozás (SZP),	100/70/50 ►73,3
VI.	pénzügyileg és piaci szempontból is elfogadható tervezési opció (reális megtérülést) (SNK),	100/90/60 ►83,3
VII.	piaci feltételek között életképes technológiai feltételek megléte (SKP),	100/60/50 ►70
VIII.	a technológiai megoldás hosszú távú valós pénzügyi opció (SNZ)	100/90/70 ►86,6

KISKOCKA SZÁMA	HASZNOSSÁG (1-100) (kerekített)	KAPCSOLAT, DIMENZIÓ ÉRTÉK	SMART ÉRTÉK	KOCKATÍPUS ÉS DOMINANCIA	FŐÁGENS BELSŐ TULAJDONSÁGOK
S1	**87**	3/3	**87**	sarokkocka, átlagon felül domináló tulajdonsággal	a technológiai megoldás hosszú távú valós pénzügyi opció (SNZ)
S2	**75**	2/3	**50**	**élkocka**, átlagon felül domináló tulajdonsággal	technológiai kockázat minimalizálása (PS)
S3	**73**	3/3	**73**	sarokkocka, átlagosan domináló tulajdonsággal	stratégia célrendszerekhez illeszkedő technológia és szabályozás (SZP)
S4	**85**	2/3	**56**	**élkocka**, átlagosan domináló tulajdonsággal	az energia- és CO2 mérleg stratégiai megfeleltetése (ZS)
S5	**100**	1/3	**33,3**	KÖZÉPKOCKA, tulajdonság dominancia összehasonlítása a többi középkockával lehetséges	erőforrás-hatékony energiafelhasználás (S)
S6	**95**	2/3	**63**	**élkocka,** átlagon felül domináló tulajdonsággal	adózási és kedvezményezési feltételek az energiatermelő rendszerben (NS)
S7	**83**	3/3	**83**	sarokkocka, átlagosan domináló tulajdonsággal	pénzügyileg és technológiailag is elfogadható tervezési opció (SNK)
S8	**80**	2/3	**53**	**élkocka,** átlagosan domináló tulajdonsággal	mesterséges és valós piacra lépés pénzügyi feltételei (KS)
S9	**70**	3/3	**70**	sarokkocka, átlagosan domináló tulajdonsággal	piaci feltételek között életképes termelési szolgáltatási feltételek megléte (SKP)

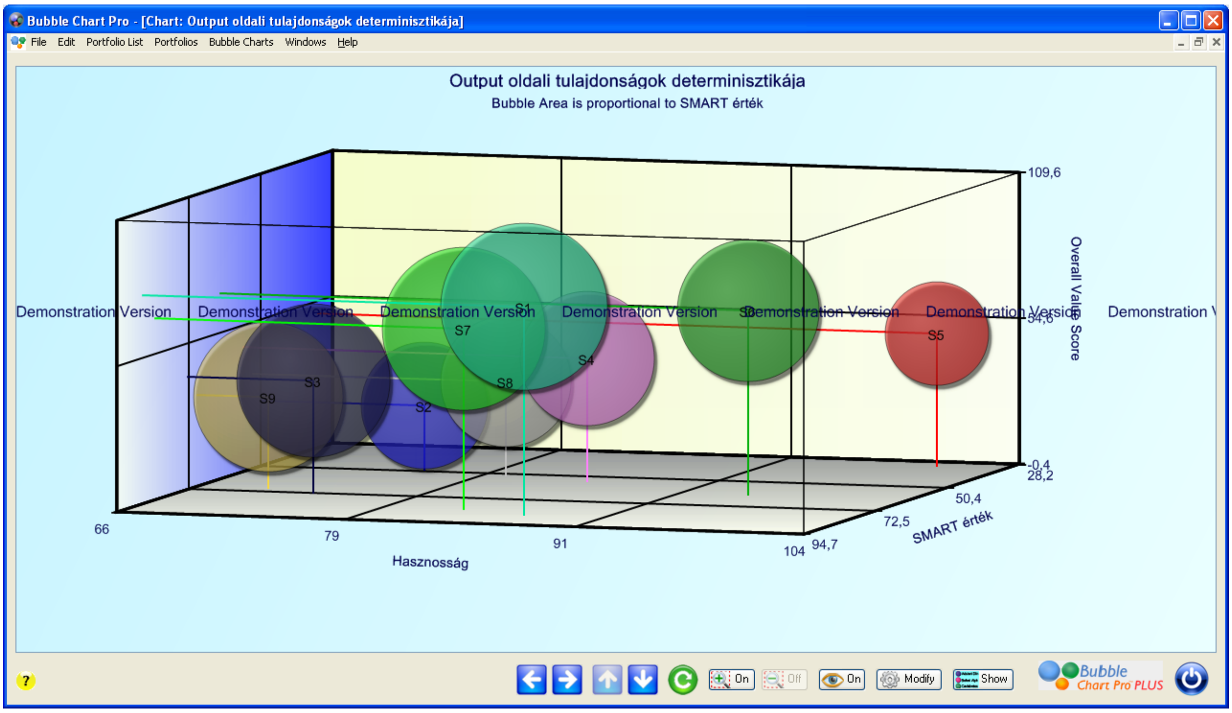
Forrás: saját kutatás

Output oldal karaktertáblája a SMART programban:



Sel.	No.	Name	Pic	Value	*A (100)	*B (100)	C (0)	D (0)	E (0)	F (0)	G (0)	H (0)	I (0)	J (0)	K (0)
Sel.	No.	Name	Pic	Value Score	Hasznosság	SMART érték	C	D	E	F	G	H	I	J	K
✓	1	S1		78,33	87	87	0	0	0	0	0	0	0	0	0
✓	9	S2		24,07	75	50	0	0	0	0	0	0	0	0	0
✓	6	S3		42,04	73	73	0	0	0	0	0	0	0	0	0
✓	5	S4		46,3	85	56	0	0	0	0	0	0	0	0	0
✓	4	S5		50	100	33	0	0	0	0	0	0	0	0	0
✓	2	S6		69,44	95	63	0	0	0	0	0	0	0	0	0
✓	3	S7		67,96	83	83	0	0	0	0	0	0	0	0	0
✓	7	S8		35,19	80	53	0	0	0	0	0	0	0	0	0
✓	8	S9		34,26	70	70	0	0	0	0	0	0	0	0	0
No.	9	Sum		447,6	748	568	0	0	0	0	0	0	0	0	0
		Median		46,3	83	63	0	0	0	0	0	0	0	0	0
		Mean		49,7	83	63,1	0	0	0	0	0	0	0	0	0
		Standard Deviation		18,4	10	17,1	0	0	0	0	0	0	0	0	0
		Maximum		78,3	100	87	0	0	0	0	0	0	0	0	0
		Minimum		24,1	70	33	0	0	0	0	0	0	0	0	0

Output oldal tulajdonságainak 3D ábrázolása a SMART programban:



A 3D ábrázolás alapján nem megengedett eltérést mutat az egyensúlyi állapottól az S9 illetve S3 tulajdonságokra. A vizsgálatok alapján a piaci feltételrendszer (pl. megfelelő piaci felület) és szolgáltatások egyensúlyba terelése, azaz játékelméleti optimumkeresése javasolt.

M5: SMART TÁBLÁZAT ÉS KIMENETI ÁBRÁK A KOCKAKÖZÉP OLDAL-TULAJDONSÁGOKRA

Kockaközép oldal-tulajdonságok determinisztikája:

SMART output értékek generálása az adatbevitelhez (’►X’ = hasznosság)

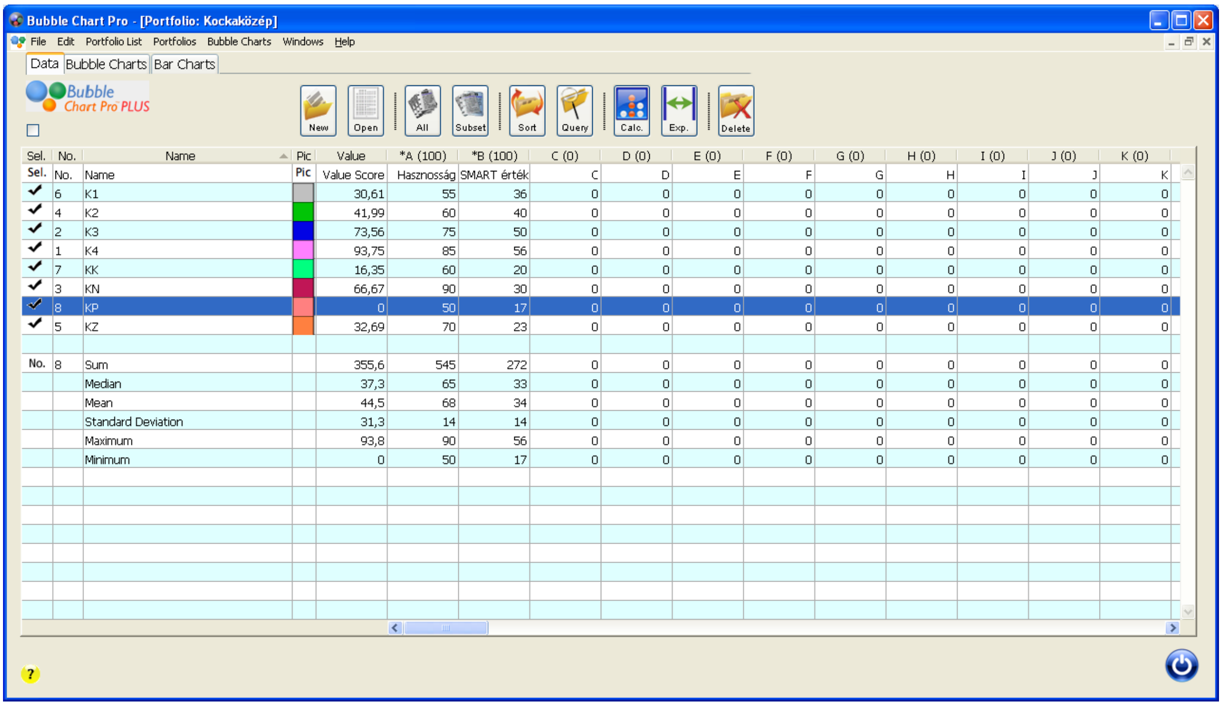
KÖZÉPKOCKÁK
KN- NARANCSKÖZÉP (1D) – megtérülési idő, cégérték (N) ►(90)
KZ- ZÖLDKÖZÉP (1D) – lokális/vállalati stratégia megvalósítása (Z) ►(70)
KK- KÉKKÖZÉP (1D) – kereslet és kínálati egyensúlyban tervezhető ár (K) ►(60)
KP- PIROSKÖZÉP (1D) – megfelelő technológiai alkalmazás (P) ►(50)

ÉLKOCKÁK
K1- piaci stratégia követése és összehangolása a technológiai prioritásokkal (KP) 60/50 ►55
K2- gazdaságos tech. rendszerek paramétereinek illesztése a szabályozáshoz (ZP) 70/50 ►60
K3- piaci változások, deviza kockázati tényezők és globális hatások elemzése (NK) 90/60 ►75
K4- leggazdaságosabb stratégiai cél kijelölése a hosszú távú hasznokkal (NZ) 90/70 ►85

KISKOCKA SZÁMA	HASZNOSSÁG (1-100)	KAPCSOLAT, DIMENZIÓ ÉRTÉK	SMART ÉRTÉK (kerekített)	KOCKATÍPUS ÉS DOMINANCIA	FŐÁGENS BELSŐ TULAJDONSÁGOK
KN	**90**	1/3	**30**	KÖZÉPKOCKA (1D) – megtérülési idő, cégérték (N)	PÉNZÜGYI HATÁSOK ÖSSZESÍTÉSE
KZ	**70**	1/3	**23**	KÖZÉPKOCKA (1D) – lokális/vállalati stratégia megvalósítása (Z)	STRATÉGIAI PROGRAMILLESZTÉS/ JOGI ÉS SZABÁLYOZÁSI FELTÉTELEK
KK	**60**	1/3	**20**	KÖZÉPKOCKA (1D) – kereslet és kínálati egyensúlyban tervezhető ár (K)	PIACI LEHETŐSÉGEK VIZSGÁLATA
KP	**50**	1/3	**17**	PIROSKÖZÉP (1D) – megfelelő technológiai alkalmazás (P)	TECHNOLÓGIAI FELTÉTELRENDSZER
K1	**55**	2/3	**36**	ÉLKOCKA piaci stratégia követése és összehangolása a technológiai prioritásokkal (KP)	PIACI LEHETŐSÉGEK VIZSGÁLATA / TECHNOLÓGIAI FELTÉTELRENDSZER
K2	**60**	2/3	**40**	ÉLKOCKA gazdaságos tech. rendszerek paramétereinek illesztése a szabályozáshoz (ZP)	STRATÉGIAI, JOGI ÉS SZABÁLYOZÁSI FELTÉTELEK / TECHNOLÓGIAI FELTÉTELRENDSZER
K3	**75**	2/3	**50**	ÉLKOCKA piaci változások, deviza kockázati tényezők és globális hatások elemzése (NK)	PÉNZÜGYI HATÁSOK ÖSSZESÍTÉSE/ PIACI LEHETŐSÉGEK VIZSGÁLATA
K4	**85**	2/3	**56**	ÉLKOCKA leggazdaságosabb stratégiai cél kijelölése a hosszú távú hasznokkal (NZ)	PÉNZÜGYI HATÁSOK ÖSSZESÍTÉSE / STRATÉGIAI, JOGI ÉS SZABÁLYOZÁSI FELTÉTELEK

Forrás: saját kutatás

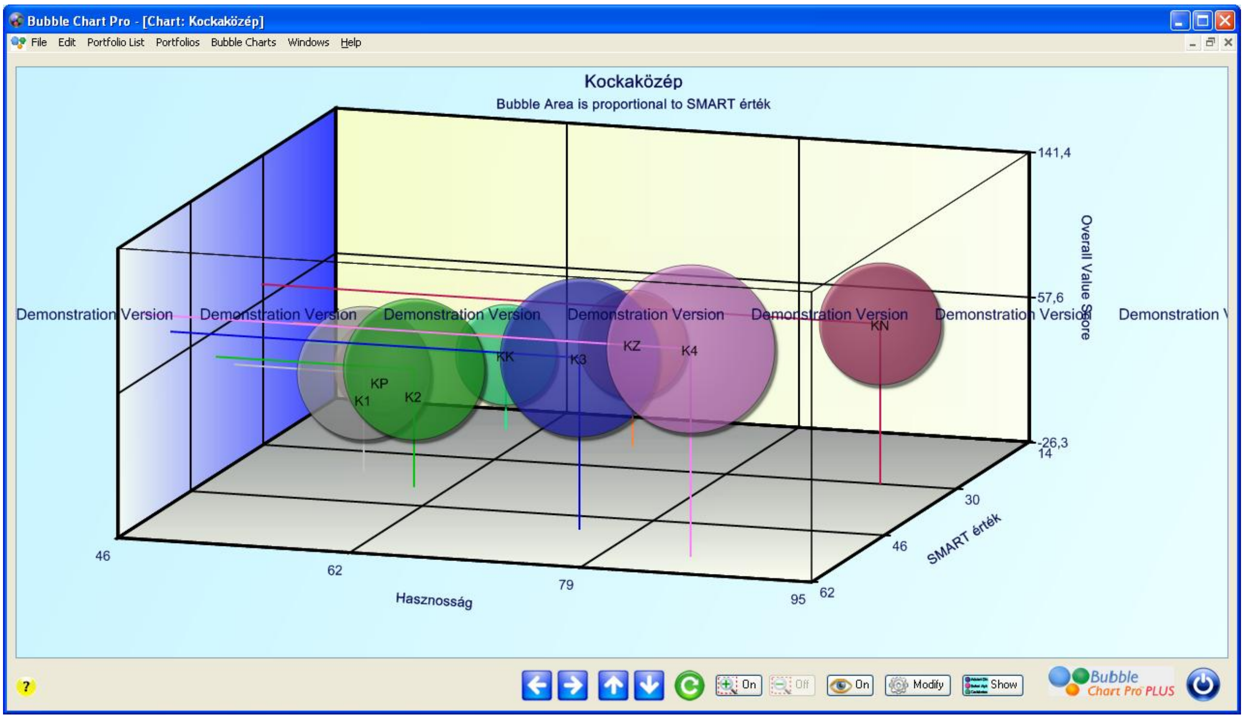
Kockaközép oldal-tulajdonságok karaktertáblája a SMART programban:



Bubble Chart Pro - [Portfolio: Kockaközép]

Sel.	No.	Name	Pic	Value Score	Hasznosság	SMART érték	C	D	E	F	G	H	I	J	K
✓	6	K1		30,61	55	36	0	0	0	0	0	0	0	0	0
✓	4	K2		41,99	60	40	0	0	0	0	0	0	0	0	0
✓	2	K3		73,56	75	50	0	0	0	0	0	0	0	0	0
✓	1	K4		93,75	85	56	0	0	0	0	0	0	0	0	0
✓	7	KK		16,35	60	20	0	0	0	0	0	0	0	0	0
✓	3	KN		66,67	90	30	0	0	0	0	0	0	0	0	0
✓	8	KP		0	50	17	0	0	0	0	0	0	0	0	0
✓	5	KZ		32,69	70	23	0	0	0	0	0	0	0	0	0
No.	8	Sum		355,6	545	272	0	0	0	0	0	0	0	0	0
		Median		37,3	65	33	0	0	0	0	0	0	0	0	0
		Mean		44,5	68	34	0	0	0	0	0	0	0	0	0
		Standard Deviation		31,3	14	14	0	0	0	0	0	0	0	0	0
		Maximum		93,8	90	56	0	0	0	0	0	0	0	0	0
		Minimum		0	50	17	0	0	0	0	0	0	0	0	0

Kockaközép oldal-tulajdonságainak 3D ábrázolása a SMART programban:



A 3D ábrázolás alapján nem megengedett eltérést mutat az egyensúlyi állapottól az KN illetve K4 tulajdonságokra. A vizsgálatok alapján a Narancsközép oldal a jelenlegi főtulajdonság halmazban „a projekt pénzügyi értéket, a megtérülési idő” egyensúlyba terelése, azaz játékelméleti optimumkeresése javasolt.

M6: ÁBRÁK JEGYZÉKE

M7: TÁBLÁZATOK JEGYZÉKE

KÖSZÖNETNYILVÁNÍTÁS

Köszönettel tartozom a dolgozat elkészítése során nyújtott messzemenő támogatásukért témavezetőimnek:

Dr. Szűcs István professzor úrnak és

Dr. Molnár Sándor professzor úrnak.