

SZÉCHENYI ISTVÁN EGYETEM

MŰSZAKI TUDOMÁNYI KAR



OTDK-DOLGOZAT

Tudás, blöff és szerencse gépi játékokban

Boros Norbert

Tanár-Mérnökstanár
(Mérnök Informatikus)

MA

2013. tavaszi forduló

Tudás, blöff és szerencse gépi játékokban

Készült:

Széchenyi István Egyetem
Műszaki Tudományi Kar
Matematika és Számítástudomány tanszék

Készítette:

Boros Norbert
Tanár-Mérnökstanár (Mérnök Informatikus) MA
2. évfolyam

Témavezető:

Dr. Kallós Gábor egyetemi docens

REZÜMÉ

Boros Norbert

Tudás, blöff és szerencse gépi játékokban

Sok híres játék fontos jellemzője, hogy a résztvevők tudása, rutinja mellett a szerencse és a blöffölési képesség is meghatározó a végkimenetel szempontjából. Az ilyen típusú stratégiák algoritmizálása a gépi játékelmélet nagy kihívásai közé tartozik.

A Perudo a blöff- és szerencsejátékok egy érdekes, ugyanakkor kevésbé ismert és egyszerűbb képviselője, ezért választottam az adott osztály játéakai közül a gépi stratégiák megvalósítására.

A dolgozatban a Perudo játék szimulációja mellett különböző gépi játékosokat (stratégiákat) terveztem és programoztam. Elemeztem, hogy melyik stratégiának mik az erősségei és a fogyatékosságai. További vizsgálat foglalkozik azzal a problémakörrel, hogy emberi játékosok ellen melyik gépi stratégia lehet sikeres, és az eredmény hogyan függhet össze az emberek felkészültségével, játékrutinjával.

RESUME

Norbert Boros

Knowledge, bluff and luck with game machines

Many famous games rely on luck and bluffing skills besides the participants' knowledge and practice to win the game. Creating algorithms for such strategies is one of the biggest challenges of game theory.

Perudo is an interesting but not very well known variant of the bluffing and gambling games, that was the reason I've chosen it for implementing computer strategies.

In this thesis I designed and implemented computer based players (strategies), as well as creating the simulation of the Perudo game. I analyzed the strengths and drawbacks of the different strategies. Further investigations were conducted with the problem of computer strategies performing against human players and how the human player's skills and practice affect the results.

Tartalomjegyzék

1. BEVEZETÉS.....	3
2. A PERUDO JÁTÉKSZABÁLYA	4
3. JÁTÉKELMÉLET.....	6
3.1. JÁTÉKELMÉLET BEVEZETŐ.....	6
3.2. A SZÁMÍTÓGÉP ÉS AZ AGY	6
3.3. JÁTÉKELMÉLETI KATEGORIZÁLÁS	7
3.4. ELSZÁMOLÁSI MÁTRIX.....	10
3.5. NASH-EGYENSÚLY A PERUDOBAN	13
4. A PROGRAM FELÉPÍTÉSE, FELÜLETE	17
5. STRATÉGIÁK/1.....	21
5.1. ALAP GÉPSTRATÉGIA.....	21
5.2. KIEGÉSZÍTETT GÉPSTRATÉGIA	24
5.3. ELŐRELÁTÓ GÉPSTRATÉGIA.....	25
6. ELEMZÉSEK.....	28
7. EMBER A GÉP ELLEN	31
8. STRATÉGIÁK/2.....	33
8.1. ÓVATOS STRATÉGIA	33
8.2. ESETALAPÚ STRATÉGIA.....	33
8.3. VEGYES ÉS PUHATOLÓZÓ STRATÉGIA.....	34
9. A BLÖFF, MINT STRATÉGIA	35
10. PROGRAM ÉS ALGORITMUS	40
11. BEFEJEZÉS.....	42
IRODALOMJEGYZÉK.....	43

1. BEVEZETÉS

Azokat a játékokat, amelyeknél a játék végkimenetelét nagyban meghatározza a szerencse, összefoglaló módon szerencsejátékoknak nevezzük. A szerencsejátékok többségénél a nyeresé esélyét jelentősen növelheti egy jó stratégia, amely tapasztalati vagy elméleti tudást integrál. (Pl. totó: otthon általában eredményesebb egy csapat, mint idegenben, a tabellán előrébb álló csapat általában jobban szerepel, mint egy hátul álló, stb.)

Ilyen szerencsejátéknak számítanak a különböző kártya- és kockajátékok. Az egyik legismertebb és legnépszerűbb kártyajáték a póker. Minden évben rendeznek nagy póker versenyeket és világbajnokságot is. Ezeken a bajnokságokon sokszor ugyanazokat a neveket látjuk a legjobbak között, de nyilvánvalóan nem csak azért jelennek meg ezek a nevek évről-évre, mert ők a legszerencsésebb emberek a világon, hanem mert van egy jó taktikájuk, aminek a segítségével nagyobb esélyük van nyerni. Például Amarillo Slim Preston 1971-ben, 1972-ben, 1985-ben és 1990-ben is ott volt a legjobbak között a WSOP-n.

A szerencsejátékokhoz tartozó stratégiák algoritmizálása a gépi játékelmélet nagy kihívásai közé tartozik. A BSc-s szakdolgozatom során egy kockajátékhoz fejlesztettem ki különböző stratégiákat. A témában nagy fantáziát láttam és szívesen foglalkoztam vele, ezért a TDK munkám során ezt a dolgot fejlesztettem tovább. Új ötleteim felhasználásával még újabb és hatékonyabb stratégiákat alakítottam ki, és ezek kapcsán elméleti és gyakorlati vizsgálatokat végeztem.

Az alapul vett Perudo a kockadobálás blöff- és szerencsejátékok egy érdekes, ugyanakkor kevésbé ismert és egyszerűbb képviselője, ezért választottam az adott osztály játékaik közül a gépi stratégiák megvalósítására.

A dolgozatom során egy Perudo játékot fejlesztettem ki JAVA-ban, és ehhez a játékhoz különböző gépstratégia családokat terveztem és programoztam. A játékban a felhasználó a különböző gépstratégia családokkal küzdhet meg. A stratégiák fejlesztése során kutatásokat is végeztem a játékelmélet és blöff témakörökben, hogy minél jobb algoritmust tudjak készíteni hozzájuk. Internetes és hagyományos irodalomkutatás során nem találtam olyan dolgot vagy publikációt, amely témájában akárcsak közelítő egyezést mutatott volna az általam leírtakkal. Ebből arra következtetek, hogy eredményeim újszerűek, és érdeklődésre tartanak számot.

A program kétféle szimulációs üzemmódot is támogat, amelyek segítségével a különböző gépstratégiák egymás ellen is játszhatnak.

A téma még további lehetőségeket is tartogat, ezért terveim között szerepel a további fejlesztés.

2. A PERUDO JÁTÉKSZABÁLYA

Az új világrészre, Amerikába indított egyik kincskereső spanyol expedíció Francisco Pizarro vezetése alatt a mai Peru északi partvidékén szállt partra 1531 januárjában. Kalandos hadjárat után 1532-ben Pizarro és hódító vezérei legyőzték és foglyul ejtették az inka Atahualpát. A szóbeszéd szerint Atahualpa bebörtönzése alatt megtanította Pizarót Perudozni és így Spanyolországban ugyanúgy elterjedt a játék, mint Peruban.

A Peruból származó Perudo a szerencse a logika a blöffölés és a találgatás izgalmas játéka. Ezt az ősi perui kockajátékot kettő vagy több játékos játszhatja. Mindegyik játékos egy keverőpohárral és öt azonos színű kockával kezdi a játékot. A játék során a kockák száma fokozatosan csökken. A játékot végül az a játékos nyeri meg, akinek a végére is marad legalább egy kockája.

A játék körökből áll, minden kör kockadobással kezdődik és vagy a következő kockadobásig vagy a játék végéig tart. A játék kezdetekor az összes játékos dob egyet egy kockával. A játékot az a játékos kezdheti, aki a legnagyobbat dobta. Ha a legnagyobb pontszámot több játékos is eléri, akkor az egyformát dobók mindaddig újra dobnak, amíg nem lehet meghatározni a kezdő licitálót.

Ezután mindenki belehelyezi a kockáit a poharába, majd megkeverik és végül mindannyian egyszerre dobnak úgy, hogy a poharaikat fejjel lefelé az asztalra fordítják arra vigyázva, hogy a többi játékos ne láthassa meg a kockáikat. Az a játékos, aki az induláskor a legnagyobbat dobta kockáit a többiek elől eltakarva megnézi a saját dobásait. Ezután megkezdheti a licitálást.

A licit bemondása két részből áll egy darabszámból és a darabszámhoz tartozó kocka értékéből. A bemondáskor a játékos azt becsüli meg, hogy az adott értéket mutató kockából legalább mennyi van az asztalon. A darabszámhoz hozzáadódik az egyesek darabszáma is, mert azok jokerként (más néven ászként, nálunk a továbbiakban joker) szerepelnek. A licit bemondásához természetesen figyelembe kell venni az asztalon lévő kockák számát. Ha pl. hat játékos játszik, akkor a kockák száma harminc. Ilyenkor várható értékben minden dobott számból öt lesz, de nagyobb eséllyel lesz egy-egy számból ötnél több illetve kevesebb. Előfordulhatnak szélsőséges helyzetek is pl. minden kocka négyesre fordul, de ennek nagyon kicsi a valószínűsége.

Érdekes azzal is számolni, hogy a jokerek azt az értéket veszik fel, amit a játékosok az utolsó bemon-dásban megneveztek. Így például az utolsó játékos bemondja, hogy hat ötös akkor, ha az asztalon csak négy ötös van és van két joker, akkor a jokereket hozzá kell tenni az ötösökhöz és így az hat ötösnek számít. A jokerek akkor is ötösnek számítanak, ha úgy mondja be a játékos a négy darab ötöst, hogy az asztalon egy ötös sincs, de vannak jokerek. A kezdőtől balra ülő játékos folytatja a lici-tálást, majd a további játékosok tovább licitálnak.

Minden licitnek nagyobb-nak kell lennie összpontszám tekintetében az előző licitnél. A licit összpontszáma úgy áll elő, hogy össze kell adni az azonos kockák darabszámának a tízszeresét és a kockák értékét, például hat négyes 64-nek felel meg. Darabszám növelés esetében az egy kockára eső pontértéket akár csökkenteni is lehet, mert hat négyes (64) kevesebbet ér, mint kilenc hármas, mely-nek 93 az értéke. Tehát a hét négyes licitet is követheti a kilenc hármas, de a hat hatos nem, mert a 66 kisebb a 74-nél. A licitáló játékosnak figyelembe kell venni, hogy mennyi kocka van az asztalon, és az előzőek által bemondott licitértékből célszerű következtetni arra, hogy az egyes kockákból hány darab lehet.

Ha a következő játékos úgy érzi, hogy nem tudja tovább növelni a licitet, akkor blöffölhet, vagy ha úgy gondolja, hogy az előző játékos által bementett licit lehetetlen, akkor választhatja a „pinokkió” vagy „dudo” (spanyol szó, ami magyarul azt jelenti, hogy kétkedem) bementését is. A „dudo” (vagy „pinokkió”) bementésével a licitalás véget ér és az utolsónak bementett licit válik érvényessé. Ezután sorban minden játékos a „dudo”-t bementő játékoskal kezdve felfedi kockáit, majd az utolsó licitben meghatározott értékű kockák darabszámát a jokerekkel együtt megszámlálják. Ha az asztalon lévő megszámlált kockák száma kisebb, mint amit az utolsó érvényes licitben meghatároztak akkor az utolsó licitaló elveszít egy kockát. Ha a licit érvényes volt - azaz a kockák darabszáma ugyanannyi vagy több, mint ami a licitben elhangzott -, akkor a „dudo”-t bementő játékosnak kell az egyik kockáját bedobnia. Például, ha az utolsó licit hat négyes volt, és az asztalon a jokerekkel együtt is csak két négyes van, akkor az utolsó licitaló veszít; de ha hat vagy annál több négyes van, akkor a „dudo”-t bementő játékos lesz a vesztes. A „dudo” bementésakor csak az utolsó licitet kell figyelembe venni még akkor is, ha az utolsó licitaló előtti játékosok is blöfföltek. Az elveszített kockákat egy zsákba kell tenni, hogy a játékban maradó dobókockák száma rejtve maradjon azon játékosok előtt, akik nem figyeltek rendszeren.

A „dudo” bementése után az a játékos folytatja a játékot, amelyik a kockáját a zsákba helyezte. Ha az egyik játékos az összes kockáját elvesztette, akkor a tőle balra ülő játékos kezdi az új licitet.

A „dudo” bementésakor a jokerek mindig az utolsó licitben megnevezett értéket veszik fel. Csavar a játékban, hogy a soron következő játékosoknak lehetősége van a jokerek bementésére is.

Ilyenkor a licitaló játékos megbecsüli a játékban lévő jokerek darabszámát. A jokerek bementésakor legalább az előző licit felét vagy annál nagyobbat kell mondani. Például ha az előző licitben nyolc hatos szerepelt, akkor legalább négy jokert kell bementeni ahhoz, hogy a licit érvényes legyen.

Páratlan licit esetén a kerekítési szabályoknak megfelelően mindig felfelé kell kerekíteni, így hét hatos esetében is minimum 4 jokerrel lehet az előzőt túllicitálni. A jokerlicitet követő játékosnak két lehetősége van vagy a jokerek darabszámát növeli meg vagy visszatér a számok bementéséhez úgy, hogy a jokerek számát megduplázza és hozzáad egyet. Így például a négy jokert követheti az öt vagy hat... joker, de követheti a kilenc kettes, hármas, négyes... is.

Természetesen a joker bementést is követheti „dudo”. Minden körben csak egyszer lehet jokert bementeni, és az első licit nem lehet jokerbementés.

Az a játékos, aki elveszíti az utolsó kockáját is kiesik a játékból, ilyenkor a tőle balra ülő játékos kezdi a licitalást a következő körben. Végül az nyeri meg a játékot, akinek utoljára is marad kockája.

3. JÁTÉKELMÉLET

3.1. Játékelmélet bevezető

A gazdasági életben lefolytatott üzletek nagyon hasonlítanak a játékokra, mert

- mindegyikben általában kettő vagy több személy vesz részt,
- minden üzlet és játék véges számú lépés után befejeződik,
- általában a játék és üzlet végén mindig van egy győztes és egy vesztes (kivétel azok a játékok, ahol a döntetlen is megengedett)
- a játékosok, üzletkötők valami stratégia alapján cselekednek.

A játékelmélettel azért fontos foglalkozni, mert a játékok során olyan gondolatok stratégiák jelennek meg, amelyek a gazdasági életben is előfordulnak. A játékelmélet olyan többszereplős döntési problémákat tanulmányoz, amelyek gyakran előfordulnak a közgazdaságtanban. A játékok és a gazdaság közötti összefüggések miatt a játékelmélet eredményeit, gondolkodásmódjait az üzleti és a politikai életben is felhasználják.

Vannak olyan játékok, melyben a kezdőjátékos egy jó gondolkodásmenettel, stratégiával mindig győzni tud, de a játékok többségének és a gazdaság területén megjelenő üzleteknek az optimális megoldása általában nem állapítható meg előre. Általában vannak olyan külső tényezők, amelyekkel a gazdasági élet szereplői előre nem tudnak pontosan számolni. Ilyenek az egyes játékokban is előfordulnak, például a Perudoban, ilyen ismeretlen tényező a kockák értéke vagy a másik szereplő blöffje. A blöff is egy olyan előre meg nem határozható esemény, ami nem csak a játékban, hanem az üzleti életben is előfordulhat. Ha a játékok kimenetele ilyen külső befolyásoló tényezőktől is függ, akkor csak a valószínűségek alapján tett feltevésekkel lehet meghatározni a játék optimális kimenetét. A játékelméletnek tulajdonképpen az a feladata, hogy ezt az **optimális megoldást** megtalálja.

A játékelmélet megalapozója Neumann János magyar matematikus volt, aki 1928-as publikációjában a matematika segítségével elsőként írt a játékok problémáiról és azoknak a megoldásáról. A játékelmélet tudományág kialakulásának időpontját hagyományosan Neumann János és Oskar Morgenstern Theory of Games and Economic Behaviour (magyarul: Játékelmélet és gazdasági viselkedés) című könyvének 1944-es megjelenéséhez kötik.

3.2. A számítógép és az agy

Neumann János nemcsak az optimális megoldás megtalálásával foglalkozott, hanem azzal is, hogy ezt a megoldást a számítógép hogyan tudná előállítani. A The Computer and the Brain (magyarul A számítógép és az agy) című munkájában azzal foglalkozott, hogy az ember agya a gondolkodás során hasonlóan viselkedik, mint a számítógép műveletvégzés közben.

Az ember agya idegsejtekből, neuronokból áll, ezek a neuronok úgy működnek, hogy idegimpulzusokat keltenek és továbbítanak. Ha az idegimpulzusokat 0-kal és 1-esekkel jelöljük, akkor az agyat ugyanez lehet modellezni, mint a számítógépet. Az emberi agy döntései logikai feltételekből tevődnek össze. Ezeket a logikai feltételeket is lehet az idegimpulzusokkal és a neuronokkal modellezni. Ha egy neuronba két idegimpulzus egyszerre érkezik be és erre a neuron csak akkor reagál, ha mindkét ingerlője aktív, akkor ez a logikai „és” műveletnek (konjunkciónak) felel meg. Ha pedig már egy aktív

idegimpulzus beérkezésére is reagál a neuron, akkor az „vagy” műveletnek (diszjunkciónak) felel meg. Az emberi agyban az egyes döntések során ezek a logikai műveletek összetetten jelennek meg. A döntések során szükség van az információ tárolására is, ezért az agynak memóriája is van.

A számítógép is rendelkezik memóriával és a számítógép is logikai műveleteket hajt végre a nullákkal és az egyesekkel. Programnyelvek segítségével az egyes döntéseknél fellépő emberi agyban megjelenő logikai műveleteket le lehet programozni a számítógép nyelvére. Az ember játékoknál megjelenő stratégiáit is le lehet programozni, ezáltal tesztelni is lehet és továbbfejleszteni, hogy az emberek jól tudjanak az adott szituációban dönteni.

3.3. Játékelméleti kategorizálás

A stratégia kialakítása és leprogramozása előtt először meg kell határozni, hogy az adott játék milyen játékelméleti kategóriába tartozik és akkor könnyebb a stratégiát kialakítani. Neumann János munkássága során a játékoknak két nagy csoportját különböztette meg: a **szerencsejátékokat** és a **stratégiai játékokat**. A szerencsejátékok esetében a nyereség csak a szerencsén múlik, a játékosoknak a játék kimenetelére semmilyen befolyásuk nincs. Ilyen szerencsejáték a lottó. Stratégiai játékoknak azokat a problémákat nevezzük, ahol a játékosok a fennálló feltételek és szabályok eldöntik, meghatározzák a játék kimenetelét. A stratégiai játékok esetében az a játékos nyer, aki jobban gondolkodik, akinek jobb a stratégiája, ilyen játék a sakk. A szakirodalmak egy része ezt a két kategóriát teljesen külön tárgyalja, pedig a gazdasági folyamatok és a játékok nagy része a két kategória között van.



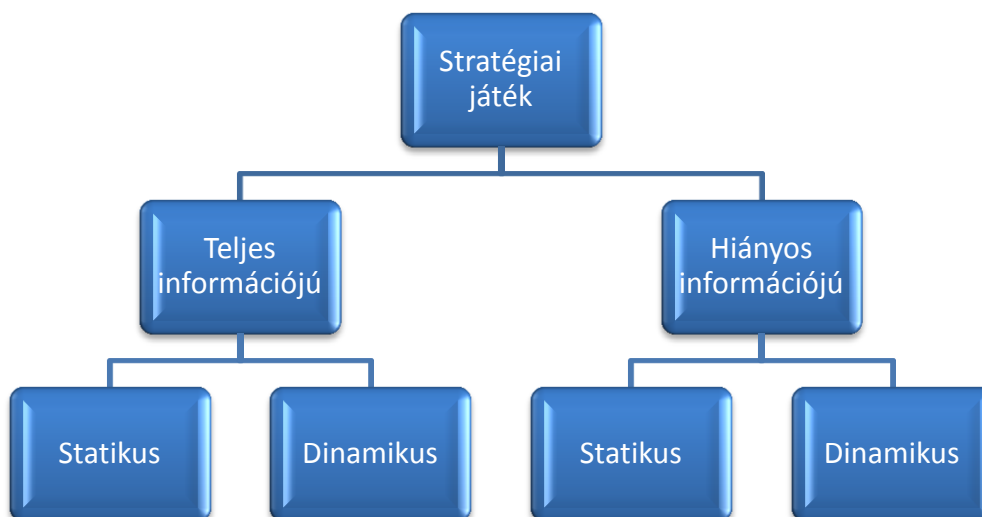
1. ábra

A Perudo is egy olyan szerencsejáték, amelyben egy jó stratégia befolyásolhatja a játék kimenetelét, ezért ezt a játékot részben stratégiai játéknak is tekinthetjük. A stratégiai játékokban a döntéshozókat játékosnak, a döntéseket pedig stratégiának nevezzük. A játékok során minden játékosnak van egy célja, ami maga a nyereség, ezt a célt egy célfüggvénnyel, az úgynevezett **kifizetőfüggvénnyel** jellemezhetjük. Az egyes játékosok számára a kifizetőfüggvény nagyobb értéke a játék előnyösebb kimenetelét jelenti.

A játékelméletben a játékokat a kifizetés szempontjából is megkülönböztetjük. Azokat a játékokat, ahol az egyes játékosok csak a másik játékosok kárára tudják növelni a kifizetésüket **zéróösszegű** játékoknak nevezzük. Zéróösszegű játék például a Sakk és a Póker is, ahol a nyertes annyit nyer, amennyit a vesztesek befizetnek. Azokat pedig, ahol külső forrásból is lehet növelni a nyereményt **nem-zéróösszegű** játékoknak hívjuk. Nem-zéróösszegű játék például a Gazdálkodj okosan és a Monopoly is, ahol a játékosok nem csak egymástól kapnak pénzt, tudják megvenni a megszerzett telkeket, eszközöket, hanem a banktól is tudnak vásárolni, illetve a bank is ajándékozhat és fizethet

nekik. A Perudo viszont zéróösszegű játék, mert ha a Perudot pénzben játsszák, akkor annyi pénzt fognak nyerni a győztesek, amennyi pénzt a vesztesek elveszítenek.

A játékelmélet többszereplős döntési problémákat vizsgál, és azokra keres megoldást. A játékelméletben megkülönböztetünk teljes információjú statikus, teljes információjú dinamikus, hiányos információjú statikus és hiányos információjú dinamikus játékokat.



2. ábra

A **teljes információjú játékok** esetén minden játékos kifizetőfüggvénye ismert minden játékos számára. Teljes információjú játéknak tekinthető minden olyan játék, amelyben látjuk az ellenfeleink összes lépését. A teljes információjú játékoknak az egyik legnépszerűbb fajtája az olyan táblás játékok, mint a Backgammon, a Sakk, a Dáma vagy az Amőba. A **hiányos információjú játékokban** egy vagy több játékos számára létezik valamilyen formájú bizonytalanság a többi játékos kifizetőfüggvényét illetően. A hiányos információjú játékok esetében nem látjuk, hogy az ellenfelünk hogyan áll, mit lép, illetve, hogy miért döntött úgy az egyes szituációkban. Ezeknek a játékoknak a legnépszerűbb képviselői a kártya és a kockajátékok. Ezeknél a játékoknál az ismeretlen, a bizonytalanság okozza a játék izgalmát, mert ha például egy kártyajátékban minden időpillanatban látnánk az ellenfeleink lapjait, akkor a kártyajáték egyszerű szerencsejáték lenne és nem stratégiai alapú, hiszen mindig az nyerne, akinek jobbak a lapjai. Neumann János is sokat foglalkozott ezzel a kategóriával a Póker elemzése során, a Theory of Games and Economic Behavior című művében bebizonyította, hogy a Pókerben meg lehet növelni a nyeresé valószínűségét egy jó, matematikailag alátámasztott stratégiával. Ehhez a stratégiához hozzátartozik az is, ami a Póker alaptételében is szerepel, hogy a játékot úgy kell játszani, hogy mi minél többet megtudjunk az ellenfelünkről, az ellenfelünk pedig minél kevesebbet tudjon meg rólunk. A Póker alaptételét David Sklansky hívatásos pókerjátékos fogalmazta meg: „Minden esetben, amikor eltérünk attól a játékmódtól, ahogyan akkor játszanánk, ha látnánk ellenfeleink lapjait, akkor ők nyernek és minden esetben, amikor pontosan úgy játszunk, mintha látnánk ellenfeleink lapjait, ők veszítenek. Megfordítva a dolgot: minden esetben, amikor az ellenfeleink eltérnek attól, ahogyan akkor játszanának, ha láthatnák a lapjainkat, akkor mi nyerünk és minden esetben, amikor pontosan úgy

játszanak, mintha látnák lapjainkat, akkor mi veszítünk.”¹ Az alaptétel bizonyítékeként lehet például a blöffre is gondolni, ha az ellenfelünk blöfföl és mi észrevesszük a blöfföt, akkor nyerünk, ha pedig nem figyelünk rendesen, akkor veszíteni fogunk. Nem csak a Pókert kell úgy játszani, ahogy David Sklansky megfogalmazta, hanem minden hiányos információjú játékot, ha nyerni akarunk. A Perudoban is megjelennek bizonytalanságok a játszmák során, ennél a játéknál se lehet tudni, hogy az ellenfél hogyan áll, ezért erre a játékra is érdemes alkalmazni a Póker alaptételét, ha nyerni akarunk. A gazdasági életben is sokszor előfordulnak ilyen hiányos információjú játékok, ilyen például az árverés is, ahol a licitálók nem tudják, hogy a többi résztvevő mennyit lenne hajlandó az eladásra bocsátott árura áldozni.² A Perudoban is folyamatosan licitálnak a játékosok és nem ismerik a többiek licitjeit, ezért ez a játék is tekinthető egy speciális árverésnek, ahol a nyereség az az ár, amiért a licitálás folyik. Mivel a Perudoban több ismeretlen is szerepet játszik, nem ismert az egyes játékosok kifizetőfüggvénye, nem lehet tudni, hogy az egyes játékosoknál milyen kocka van, ezért a Perudo hiányos információjú játék.

Statikus játékok esetén az egyes játékosok előre meghatározzák a stratégiájukat és a cselekedeteiket, amit az egész játék során alkalmaznak. Statikus játék például a Kő, papír, olló is, mert itt a játékosnak előre el kell döntenie, hogy mit fog mutatni. **Dinamikus játékoknál** pedig a játékosok egymás után döntenek és a játszma során megváltoztathatják a stratégiájukat. Dinamikus játék például a Monopoly, mert ugyan lehet, hogy a játékos eldönti előre, hogy csak a drága telkeket vásárolja fel és azokat fejleszti, a többit pedig hagyja, de a játék során (például vagy felvásárolták előle vagy nem jött neki össze rá elég pénz és mégsem tudja megvenni a drága telkeket) módosítani kell a stratégiáját. A Perudo is dinamikus játék, mert a játékosoknak lehetőségük van arra, hogy az előző liciteknek megfelelően alakítsák ki a licitjeiket, illetve az előző köröknek megfelelően módosítsák a stratégiájukat.

A játékok során minden játékosnak az a célja, hogy megnyerje a játékot. Bizonyos játékokban a nyereséghez az egyes játékosok összefoghatnak egy másik játékos vagy játékosok legyőzése érdekében. Neumann János az együttműködés szempontjából is megkülönböztette a játékokat és az általa használt megkülönböztetést a mai napig használjuk. Az olyan játékot, ahol a játékosoknak lehetőségük van az együttműködésre **kooperatív** játékoknak nevezzük. Ilyen kooperatív játék az interneten elterjedt Honfoglaló elnevezésű társasjáték. A Honfoglalóban három játékos játszik egymás ellen, de lehetőségük van arra a játékosoknak, hogy ketten összefogjanak a harmadik ellen és amikor a harmadikat legyőzték csak akkor küzdenek meg egymás ellen. Ennek a játéknak a végén mindig egy győztes lesz, de a lehetséges összefogások miatt nem biztos, hogy a legjobb játékos fogja megnyerni a játékot. Azokat a játékokat pedig, ahol nincs mód az együttműködésre **nemkooperatív** játékoknak hívjuk. A sakk is ilyen nemkooperatív játék. A sakk esetében mivel csak két játékos játszik egymás ellen a játékosoknak lehetőségük sincs az összefogásra. A Perudot amikor többen játszik a játékosok egymás elleni versengése során lenne lehetőség az együttműködésre, de a szabályai ezt tiltják. Mivel a szabályok alapján nincs mód az összefogásra a Perudot nemkooperatív játéknak tekintjük. Ha az egyes játékosok a játékszabály ellenére összefognak egy másik játékos ellen, akkor az csalásnak minősül.

A Perudo tehát játékelméleti szempontból a stratégiai és szerencsejáték határvonalán elhelyezkedő zéróösszegű nemkooperatív hiányos információjú dinamikus játék.

¹ David Sklansky: The Theory of Poker

² Robert Gibbons: A Primer in Game Theory

3.4. Elszámolási mátrix

Egy jó vagy lehetőség szerint optimális stratégia kialakításához, megválasztásához érdemes elemzéseket készíteni a játék lehetséges alakulásairól és az ellenfelekről. Célszerű az ellenfelek minden egyes döntési alternatíváját, illetve stratégiai lépését felírni és megfelelően kiértékelni. A táblázatos formában felrajzolt, a stratégiák egymással szembeállításával kialakított értékelési sémát **elszámolási mátrixnak** nevezzük

A következő ábrán a Perudonak azt a lehetőségét mutatjuk be elszámolási mátrix segítségével, amikor két játékos játszik. Az elszámolási mátrixban nem jelenítünk meg minden lehetőséget csak a két játékos egy-egy kockáját állítjuk egymással szembe.

		1. játékos					
		1	2	3	4	5	6
2. játékos	1	X	-	-	-	-	-
	2	+	X	+	+	+	+
	3	+	-	X	+	+	+
	4	+	-	-	X	+	+
	5	+	-	-	-	X	+
	6	-	-	-	-	-	X

3. ábra

Ha a játékosok csak a tényleges kocka értékével játszanának és nem élnének a blöff lehetőségével, akkor ennél a mátrixnál a plusz jel jelenti azt, hogy az első játékos nyert és a második veszített, a mínusz pedig azt, hogy az első veszített a második pedig nyert. Az X-ek pedig azokat az eseteket mutatják, amikor a kezdő játékos nyer, mert ebben a játékban nincs döntetlen és a másik játékosnak nagyobb licitet kell mondania az előzőnél.

Ha jól megnézzük a mátrixot, akkor ebből is látszik, hogy amikor az egyik játékos nyer, akkor a másik veszít. Ez az egyszerű példa is jól szemlélteti, hogy a Perudo zéróösszegű játék.

Így ebben a formában, ha úgy játszanának, hogy látnák egymás dobásait és csak valós kockaértéket mondanának be, akkor a mátrix felírásának nem sok értelme lenne, mert az egyik játékos mindig úgy nyer, hogy a másik nem tudja befolyásolni a játék kimenetelét és csak egy sima szerencsejátékról lenne. De a Perudo játékban az egyes játékosok csak a saját dobókockáikat látják, és úgy kell licitet mondaniuk, hogy figyelembe kell venniük az ellenfél ismeretlen kockáit és akár blöffölhetnek is, így az egyes lépéseknél ennél sokkal összetettebb, bonyolultabb elszámolási mátrixot lehet felírni, amik már befolyásolhatják a játék kimenetelét.

A következő példában ennél egy bonyolultabb, elemezhetőbb mátrixot fogunk felrajzolni a Perudo egy adott szituációjához.

Példa:

A szituációban két játékos játszik és az egyiket **játékos1**-gyel a másikat **játékos2**-vel fogjuk jelölni. A játékban játékos1 egy jokert és egy négyest, játékos2 pedig két hatost dobott. A játék során a példa egyszerűsítése miatt a játékosok csak a valós kockákkal fognak játszani, nem blöffölnek. A szabályok szerint a játékosok egymás kockáit nem ismerik. Rajzoljuk fel a játék elszámolási mátrixát:

		játékos1 (1; 4)		
		1db: 1	1db: 4	2db: 4
játékos2 (6; 6)	1db: 6	+	-	+
	2db: 6	+	-	-

4. ábra: 0. lépés

A mátrixban plusz jel van ott, ahol a játékos1 licitje az erősebb és mínusz, ahol gyengébb. Az előbbi mátrix szerint játékos1 jokerlicite a legjobb licit, de készítsünk további mátrixokat, amiben azt elemezzük, hogy mi történik, ha játékos1 jokerlicitet mond. A jokerlicit után játékos2 megtudja, hogy az asztalon játékos1-nél van egy joker és a következő mátrix szerinti liciteket tudja mondani:

		játékos1 (saját: 1; 4)		
		1db: 1	1db: 4	2db: 4
játékos2 (saját: 6; 6) (ismert: 1)	1db: 1	X	-	-
	1db: 6	+	-	+
	2db: 6	+	-	-
	3db: 6	-	-	-

5. ábra: 1. lépés

Játékos2-nek mindenképp nagyobb licitet kell mondania az előző jokerlicitnél, ezért játékos1 licitjére az előző mátrix szerint csak 3db 6-ost tud mondani. (A mátrixban a kapott licit zölddel, amivel pedig felül tudja mondani pirossal van jelölve.) Ebből a licitből, mivel a játékosok blöff nélkül játszanak játékos1 megtudja, hogy játékos2-nél két hatos van. A 2db 6-os, 1db 1-es és 1db 4-es kockákból a maximum licit, amit ki lehet hozni a 3db 6-os (2db 6-os + 1db joker), ezért játékos1 hiába ismeri meg játékos2 kockáit felülmondani már nem tudja az előző licitet. Ha a játékosok így játszanak, akkor hiába volt játékos1-nél a joker, hiába tűnt úgy a 0. lépésnél, hogy játékos1 a nyerő **játékos2** fogja **megnyerni** a játékot.

Ha játékos1 nem az előző szisztéma szerint játszik, hanem egy kicsit óvatosabban és nem használja el egyből az első licitnél a jokert, hanem először 1db 4-es licitet mond, akkor játékos2 nem fogja megtudni, hogy játékos1-nél joker van és a következő mátrix szerint fog tudni licitet mondani:

		játékos1 (saját: 1; 4)		
		1db: 1	1db: 4	2db: 4
játékos2 (saját: 6; 6) (ismert: 4)	1db: 4	+	X	+
	1db: 6	+	-	+
	2db: 6	+	-	-

6. ábra: módosított 1. lépés

A mátrixból jól látszik, hogy játékos2-nek ahhoz, hogy felül tudja mondani az előző 1db 4-es licitet vagy 1db 6-ost vagy 2db 6-ost kell mondania. Ha játékos2 óvatosan licitál és az 1db 6-ost választja, akkor játékos1 megtudja, hogy van az asztalon 1db 6-os és a következőképpen tud licitálni:

		játékos1 (saját: 1; 4) (ismert: 6)			
		1db: 1	1db: 4	2db: 4	2db: 6
játékos2 (saját: 6; 6) (ismert: 4)	1db: 4	+	X	-	+
	1db: 6	+	-	+	+
	2db: 6	+	-	-	X

7. ábra: 2. lépés

Ha játékos1 erre egy jokerrel licitál, akkor visszatérünk az előző példához, amikor játékos2 az első licitnél megismerte játékos1 jokerét és ismét fel tudjuk rajzolni az 1. lépés című mátrixot, mely szerint **játékos2** 3db 6-ossal nyer. A példa az előzőekhez képest annyit módosult, hogy játékos1 később veszített és játékos2 nem csak az egyik, hanem az összes kockáját megismerte játékos1-nek.

Ha pedig játékos1 mégse joker, hanem 2db 6-os licitet mond, akkor játékos2 megtudja, hogy játékos1-nél van egy 6-os vagy egy joker. Ennél az esetnél is az előző szituáció áll elő és **játékos2** 3 db 6-os bemondásával **megnyeri** a játékot.

De ha játékos1 még ennél a licitnél se használja fel a jokerét és óvatosan 2db 4-essel mondja felül az 1db 6-os licitet, akkor a következő mátrix szerint játékos2-nek már nincs más választása, mindenképpen ki kell játszania a 2db 6-osát.

		játékos1 (saját: 1; 4) (ismert: 6)			
		1db: 1	1db: 4	1db: 6	2db: 4
játékos2 (saját: 6; 6) (ismert: 4; 4)	1db: 4	+	X	+	+
	2db: 4	+	-	+	X
	1db: 6	+	-	X	+
	2db: 6	+	-	-	-

8. ábra: 3. lépés

Ha játékos2 2db 6-ost mond be, akkor kiderül, hogy 2db 6-os értékű kocka van nála, és akkor **játékos1** a joker kocka és a 2db 6-os segítségével be tudja mondani a 3db 6-ost és **megnyeri** a játékot.

Az előző példa jól mutatja, hogy miért érdemes használni az elszámolási mátrixot. A példából látszik, hogy ugyanazon a kockákkal történő játék során az egyes stratégiák alapján többféle lehet a játék kimenete és nem biztos az, hogy az a játékos nyeri a játékot, aki nagyobb értékű kockákkal rendelkezik. Ez is bizonyítja, hogy a Perudo nem egyszerű szerencsejáték, hanem stratégiai is. A bemutatott szituáció azt is jól szemlélteti, hogy egy adott mátrix eredményei után érdemes további elemzéseket is végezni, hogy jobban megismerjük a játék végkimenetét, mert nem biztos, hogy a játék végén az lesz a nyerő, ami egy adott lépésnél jó döntésnek tűnik. Ha a játékot nem blöffmentesen játszunk, akkor az elszámolási mátrixot azért is érdemes a felírni, hogy ha az adott játékos kicsit dobott, akkor is tudjon jó és hihető blöff licitet mondani.

A játékelmélet értelme és célja, hogy az elszámolási mátrix alapján kiderítse a nyerés optimális módját amennyiben ilyen létezik, azaz az egyes stratégiák bevetését oly módon optimalizálja, hogy ezek segítségével a nyerés ténylegesen be is következzen.³

3.5. Nash-egyensúly a Perudoban

Neumann János a játékelméleti munkássága során az egyensúlyi helyzetekkel is foglalkozott. Szerinte minden véges, záróösszegű, teljes információs játék rendelkezik olyan egyensúlyi helyzettel, amittől egyik játékosnak sem érdemes eltérni, mert eltérésükkel nem tudják növelni a nyereményüket, csak csökkenteni. Bizonyos megkötésekkel az egyes lépéseknél minden hiányos információjú játék visszavezethető teljes információjú játékká. Ha a Perudoban csak a döntéseket vesszük figyelembe, az egyes játékosok állásait, azaz az ismeretlen kockák értékét elhanyagoljuk, és úgy elemezzük, akkor a Perudo is teljes információs játék, mert ismerjük az összes döntési lehetőséget az egyes szituációkban.

³ M. Eigen – R. Winkler: A játék

John Nash is sokat elemezte a játékelméletben megjelenő szituációkat. Elemzései során sokat dolgozott az egyensúlyi helyzetekkel is és bebizonyította, hogy a véges játékok mindig rendelkeznek a később róla elnevezett Nash-egyensúlyi ponttal. A teljes információjú játék optimális megoldását a Nash-egyensúly segítségével kapjuk meg.

Vegyük a Perudonak azt az egyszerű esetét, amikor két játékos játszik és még mindegyik játékos licitálni szeretne, egyik se akar még dudo-t mondani. Ezen felül tudjuk még, hogy a játékosok kockái között az egyiknél van a másiknál pedig nincs joker. Az a játékos, akinek nincs jokere, jobban szeretne normál licitet mondani, mint jokerlicitet. A jokerrel rendelkező játékos pedig inkább jokerlicitet mondana, mint normál licitet. A példánkban mindkét játékos úgy akar játszani, hogy csak olyan licitet akar mondani, mint a másik, ha joker licit kör kezdődött, akkor joker licitet, ha pedig normál licitet mond a másik, akkor normál licitet. Ennek a játékstílusnak matematikai oka nincs, ilyen előfordulhat a valóságban is, de most csak a példa kedvéért alakítottuk ki. A példában megjelenő probléma nagyon hasonlít a gyáva nyúl és a nemek harca elnevezésű játékokhoz. (Lásd: Mérő László: Mindenki másképp egyforma)

A játékok felírására Neumann János a ma is használt **normálalakot** alkalmazta. Az előbb tárgyalt Perudo helyzet felvázolásához és is a normálalakot fogjuk alkalmazni. Ha a játékot normál formában írjuk le, akkor a következő mátrixot kapjuk, a mátrixban az első játékosnak van a másodiknak nincs jokere. A mátrixban nulla súllyal jelöljük azokat a helyzeteket, amiket a játékosok nem szeretnének, eggyel amit kicsit és kettővel pedig amit nagyon akarnak.

		2. játékos (nincs joker)	
		Joker licit (J')	Normál licit
1. játékos (van joker)	Joker licit (J)	J;J' 2;1	J;N' 0;0
	Normál licit (N)	N;J' 0;0	N;N' 1;2

9. ábra

Egy n-személyes játék **normál formájú reprezentációja** megadja a játékosok $S_1, \dots, S_n$ stratégiahalma-
zait és $u_1, \dots, u_n$ kifizetőfüggvényeit. Jelölje ezt a játékot⁴ $G = \{S_1, \dots, S_n; u_1, \dots, u_n\}$.

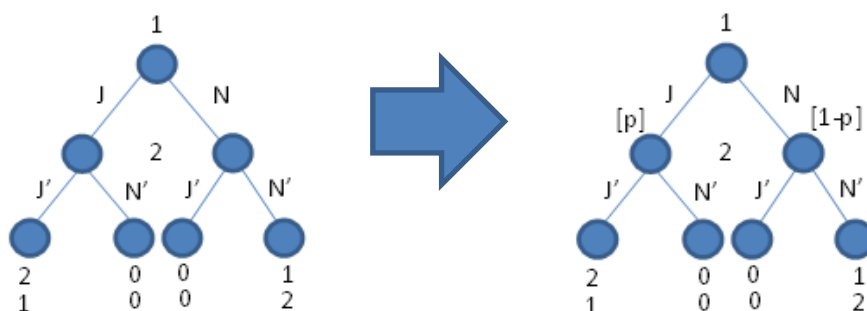
Az n-személyes $G = \{S_1, \dots, S_n; u_1, \dots, u_n\}$ normál formájú játékban az $(S^*_1, \dots, S^*_n)$ stratégiák **Nash-egyensúlyt** alkotnak, amennyiben minden i-edik játékos számára az S^*_i stratégia az i-edik játékos legjobb válasza (vagy legalábbis egy a legjobbak közül) a másik n-1 játékos adott $\{S^*_1, \dots, S^*_n; u^*_1, \dots, u^*_n\}$ stratégiájára.⁵

⁴ Robert Gibbons: A Primer in Game Theory

⁵ Lsd. előző pont

A kétszemélyes zérőösszegű játékoknak két egyenértékű egyensúlyi pontja van. Ennek a játéknak is két Nash-egyensúlya létezik a (J;J') és az (N;N') pontban. A mátrix alapján mindkét egyensúlyi pont azonos esélyekkel alakul ki a játék során, de ez a valóságban általában nem így van.

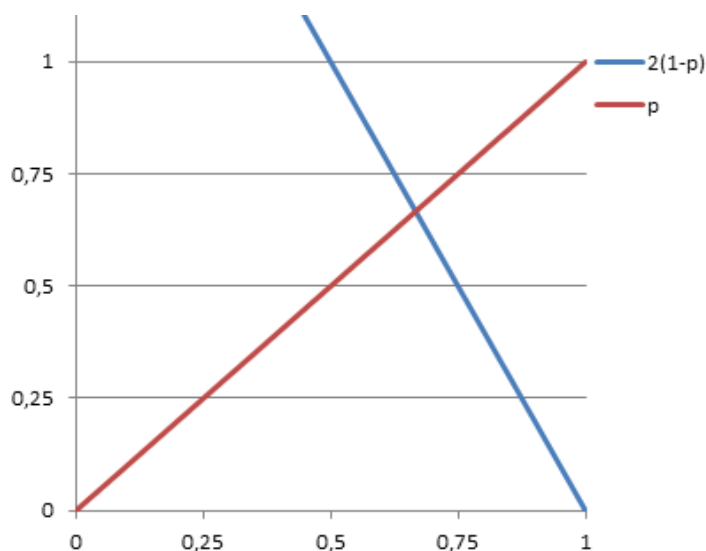
Ahhoz, hogy a Nash-egyensúlyt fel tudjuk oldani, a valószínűségek figyelembe vételével tovább kell elemezni az adott szituációt. Azokat a játékokat, amiknek a megoldásában a valószínűség is szerepet játszik **Bayesi játékoknak**, a valószínűségekkel kiegészített Nash-egyensúlyt pedig **Bayesi Nash-egyensúlynak** nevezzük. A hiányos információjú játékokat a Bayesi Nash-egyensúly segítségével lehet megoldani. A további elemzéshez a táblázatos normál forma helyett használjuk az **extenzív formát** gráfokkal. Az előző normál formájú mátrixot a következő extenzív formában lehet felírni:



10. ábra

Az ábrán az egyes ágak bekövetkezésének a valószínűségét p -vel jelöltük. A p az egy 0 és 1 közötti szám. Ha az adott esemény sose fog bekövetkezni, akkor a p értéke 0, és 1 az értéke, ha mindenképp be fog következni. Az első játékosnak a döntése során két lehetősége van: vagy joker licitet vagy normál licitet mondhat be. A játékos a joker licitet — ahogy a második gráfon is látszik — p valószínűséggel, a normál licitet pedig $1-p$ valószínűséggel választja.

A második játékosnak is két lehetősége van, ha a joker licitet választja, akkor a várható kifizetése $p \cdot 1 + (1-p) \cdot 0 = p$, míg a normál licitet megjátszásával $p \cdot 0 + (1-p) \cdot 2 = 2-2p$.



11. ábra

A p -nek mint valószínűségi változónak az értéke 0 és 1 között lehet. Ha a $p < 2/3$, akkor a normál licit megjátszásával, ha pedig $p > 2/3$, akkor a joker licit megjátszásával lesz nagyobb a kifizetése a második játékosnak. Mivel kétszer annyi olyan lehetőség van, hogy a $p < 2/3$, minthogy $p > 2/3$, ezért ha a második játékos racionálisan játszik, akkor optimális esetben a normál licitet fogja választani. Ha a játékosok előre optimalizálással eldöntik a stratégiájukat és a későbbiekben ezzel az előre meghatározott stratégiával játszanak, akkor ebben az esetben ezzel a megoldással kiküszöböltük a példa (J;J') pontban lévő Nash-egyensúlyát.

4. A PROGRAM FELÉPÍTÉSE, FELÜLETE

A programot alapos megfontolás után JAVA nyelven készítettem el. A döntés okai között meghatározó volt, hogy a nyelv objektumorientált tulajdonságai egyszerűbbé teszik a különböző játékosok együttes kezelését, a swing osztályok pedig megkönnyítik a grafikus felület kialakítását. A Perudo JAVA program főbb osztályai a következők: **Perudo**, **PerudoJatek** és **Jatekosok**.

A **PerudoJatek** osztály irányítja magát a játékot, ez az osztály figyeli a játékosokat, dönti el, hogy mikor melyik következik, és hogy ki nyerte meg a játékot. Az osztály érdekessége, hogy két konstruktorral rendelkezik: egy az időre menő és egy a kattintással tovább lépő játékok számára.

A **Jatekosok** **őszosztályban** jelenik meg a különböző Jatekosok felépítése, hogy azoknak milyen metódusokat és hogyan kell megvalósítaniuk.

A Jatekosok szülőosztály gyerekei a különböző gépi stratégiák és a felhasználó által megvalósított Ember osztály.

Minden játékosnak a Jatekosok osztály leszármazottjának kell lennie, és a következő metódusokat tartalmazniuk kell: **dontes()**, **dobas()**, **kezdolicit()**.

Minden új játék (akár szimuláció, akár normál játék) a PerudoJatek **jatekKezd()** metódusának meghívásával kezdődik. Ez a metódus teszi alapállapotba a játékosokat és készíti fel a őket a küzdelemre. Ezt követően a kezdőjátékos gépszámával meghívásra kerül a PerudoJatek **licitKapKuld(int kezd)** metódusa. Igazából ez a metódus irányítja az egész játékot, mindaddig ez a metódus fut és hív meg különböző más metódusokat, amíg legalább két játékos játékban van. Mielőtt az első kör elkezdődne, a PerudoJatek **licitKapKuld()** metódusa meghívja az összes játékos **dobas()** metódusát, amelynek segítségével a játékosok dobnak a kockáikkal, majd a **licitKapKuld()** meghívja a kezdő játékos **kezdolicit()** metódusát. Ez gépi stratégiák esetében a stratégiának és a játékszabálynak megfelelően meghatározza a kezdőlicitet, emberi játékos esetén pedig kialakítja a megfelelő interfészt a kezdő licit beírásához és elküldéséhez. Az emberi játékos liciténél az elküldés előtt ellenőrzi, hogy az adott licit megfelel-e a szabályoknak, és ha nem, akkor újra kéri a kezdő licitet.

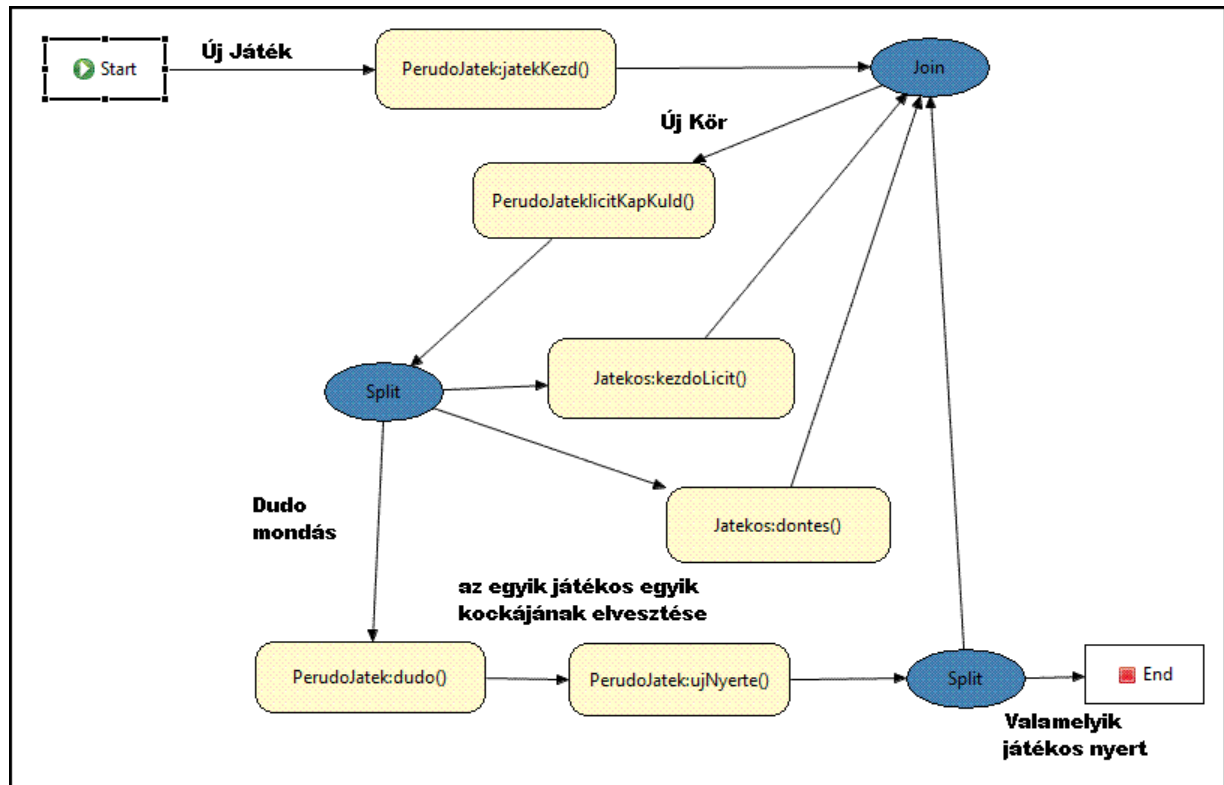
A **kezdolicit()** visszatérési értékeként a PerudoJatek osztály **licitKapKuld()** metódusa továbbítja a következő játékos **dontes()** metódusának az adott licitet. A **dontes()** metódus hasonlóan működik, mint a **kezdolicit()**. Gépi játékos esetében a stratégiának megfelelően eldönti, hogy rálicitál-e az előző licitre vagy „dudo”-t mond. Ha a felülmondást választja, akkor kiszámítja a következő licitet. Végül visszatérési értéként visszaküldi a **LicitKapKuld()**-nek a licitet, ami „dudo” mondás esetében mínusz egy.

Ha a következő játékos maga a felhasználó, akkor a **Dud_Lic form** megjelenítése után, a felhasználó a form nyomógombjai segítségével választhat, hogy „dudo”-t akar küldeni vagy licitálni szeretne. Ha a játékos a „dudo” lehetőséget választja, akkor a metódus visszaküldi a mínusz egyet, ha pedig a licitálást, akkor az eljárás a **Licit form** segítségével ellenőrzés után továbbítja a megfelelő licitet.

A **licitKapKuld()** mindaddig továbbítja a következő játékos **dontes()** metódusa számára az adott licitet, amíg „dudo”-t nem kap. Ha „dudo”-t kap akkor a **licitKapKuld()** osztály a játékszabályoknak megfelelően a **dudo()** metódusa segítségével eldönti, hogy a „dudo”-t mondó vagy az azt megelőző játékostól vesz el egy kockát. Ha egy játékosnak elfogyott az összes kockája, akkor ő kiesett a játékból és

akkor az őt követő játékos **kezdolicit()** metódusát fogja meghívni a PerudoJatek, ha pedig van még kockája, akkor az ő kezdőlicitével fog kezdődni a következő kör. Minden új kör a játékosok **dobas()** metódusának meghívásával kezdődik.

Ez mindaddig így megy, még végül egy játékos marad, aki a győztes lesz.



12. ábra: Perudo játékmenet workflow

A programnál használható egy **szimulációs üzemmód**. Ennek két fajtája van, az egyszerű és a gyors szimuláció. Mindkét üzemmód elérését az **Elemzes osztály** teszi lehetővé. Az Elemzes osztály egy Frame, ami a szimulációk beállításának megkönnyítését segíti elő.

A gyors szimulációhoz tartozik még egy **Szimulator osztály**, ami az elemzéshez jeleníti meg a különböző eredményeket. A Szimulator osztály külön szálon fut. Erre azért van szükség, hogy a szimuláció futása közben a felhasználó elérhesse a játékelületet, és ha úgy gondolja, ki tudjon lépni a programból, ne fagyjon le a GUI.

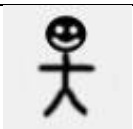
A **Perudo osztály** felel a felhasználói felület kialakításáért, segítségével jelennek meg a különböző elemek a Formon. A felhasználói felület három fő részből áll. Jobb oldalon található a gépstratégiákat jelző ikonok és az adott gépi játékoshoz tartozó dobókockák. A dobókockáknak háromfajta állapotuk lehet, a még nem ismert, elrejtett kockák, a felfordított kockák és a már játékban nem szereplő törölt kockák.

		
felfordított dobókocka	ismeretlen kocka	elvesztett kocka

1. táblázat: Dobókocák

Jobb oldalon felül található a játékasztal, körben a játékosok képeivel, középen az aktuális licittel. Itt ugyanazok a képek látszanak, mint a játékosok dobókockái mellett, kivétel az aktív játékosnál, mert az utolsó licitet bemondó aktív játékos képének a tetején a gomb pirossá változik.

Ha egy helyre nem ült le senki játszani vagy az adott játékos elvesztette az összes kockáját, akkor az üres gép ikon jelenik meg az adott játékos képe helyett.

				
ember játékos	alap gépstratégia	kiegészített gépstratégia	előrelátó gépstratégia	üres hely
				
aktív ember játékos	aktív alap gépstratégia	aktív kiegészített gépstratégia	aktív előrelátó gépstratégia	

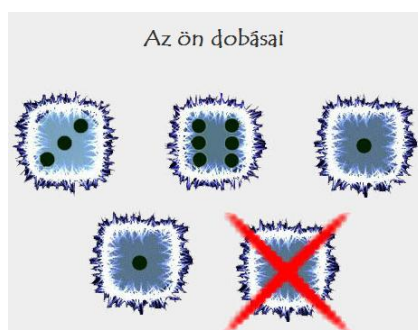
2. táblázat: Ikonok

A játékasztal mellett elhelyezkedő mezőben az aktuális kör érvényes licitjei jelennek meg. Ennek segítségével a felhasználó nyomon követheti, hogy melyik játékos hogyan gondolkodik, hogyan mondják felül az adott licitet.



13. ábra: Játékasztal

A játékasztal alatt normál játék esetében a felhasználó dobókockái helyezkednek el.



14. ábra: Játékos kockái

5. STRATÉGIÁK/1.

5.1. Alap gépstratégia

Az első általam fejlesztett stratégiacsalád az alap gépstratégia. Ennek a módszernek a legegyszerűbb az algoritmus, ebből fejlődtek ki és erre épülnek a további stratégiák.

A stratégia a kezdőlicitet úgy határozza meg, hogy megnézi, melyik kockából van neki a legtöbb (az egyesekkel együtt) és azt a kockát válassza ki. Ha több kockájából is ugyanannyi van neki, akkor a nagyobb értékűt veszi figyelembe.

```
for (kocka_érték = 0; kocka_érték < 6; kocka_érték++) {  
    if (darab[kocka_érték] >= max_kocka_db) {  
        max_kocka_db = darab[kocka_érték];  
        max_kocka_érték = kocka_érték;  
    }  
}
```

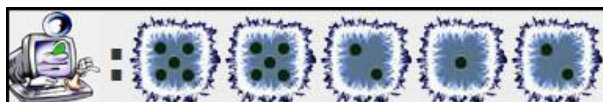
Ezután az asztalon lévő kockák darabszámából kiszámolja, hogy a többieknek mennyi lehet az adott típusú kockákból és az egyesekből várható értékben.

```
várható_érték(max_kocka_érték) {  
    for (i = 0; i < összes_játékos; i++)  
        összes_kocka_db += játékos[i].kocka_db;  
    if (max_kocka_érték == joker)  
        várható_érték = összes_kocka_db/6+max_kocka_db;  
    else  
        várható_érték = (összes_kocka_db*2)/6+max_kocka_db;  
    return (várható_érték);  
}
```

Végül a várható értéket és a kiválasztott kockájának a darabszámát összerakja és elküldi kezdőlicitként.

Példa:

Ha hárman játszanak, és még mindenkinek megvan az összes kockája és, ha a kezdőjátékosnak a jokerekkel együtt van három ötöse, akkor öt darab ötöst fog küldeni kezdőlicitként.



15. ábra: Kezdő játékos kockái



16. ábra: Kezdő licitje

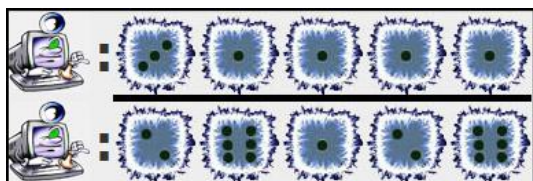
Ha a stratégia kap egy licitet úgy dönti el, hogy „dudo”-t mond vagy felülmondja az előzőt, hogy megnézi, mennyi lehet a várható értéke az adott licitben szereplő kockának.

Ezt az értéket a kezdőlicithez hasonlóan kapja meg: az asztalon lévő kockák darabszámából megsaccolja a megfelelő értékű kockák és az egyesek számának várható értékét, majd ehhez az összeghez hozzáadja a saját egyeseinek és licitben szereplő kockáinak az összegét. Ezután összehasonlítja a kapott licit összeget és az itt kiszámolt várható értéket. Ha a kapott darabszám kisebb, mint a várható érték akkor felülmondja. Ha a darabszám kisebb, mint a várhatóérték plusz a tűréshatár (a stratégia egyik előre beállított paramétere), akkor, ha még lehetséges megfelezi a darabszámot és jokerlicitet mond, de ha már nem mondhat jokerlicitet, akkor egyszerűen felüllicitálja. Ha pedig a kapott érték több mint a tűréshatárral kiegészített várhatóérték, akkor „dudo”-t mond.

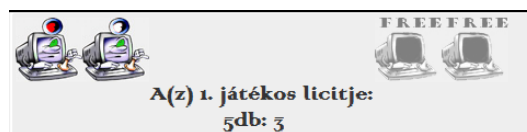
```
dontes(kapott_licit) {
    várható=várható_érték(kapott_licit_kocka_érték);
    if (kapott_licit_kocka_db < várható)
        licitálás();
    else if (kapott_licit_kocka_db <= várható+tűrés_határ)
        joker_licit();
    else
        dudo();
}
```

Példa:

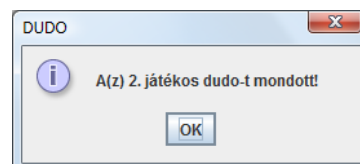
Ha az első játékosnak az egyesekkel együtt van öt hármasa és ezt el is küldi licitként, akkor ha a másodiknak csak egy hármasa van, „dudo”-t mond.



17. ábra: Az első két játékos kockái



18. ábra: Az első játékos licitje



19. ábra: A második játékos döntése

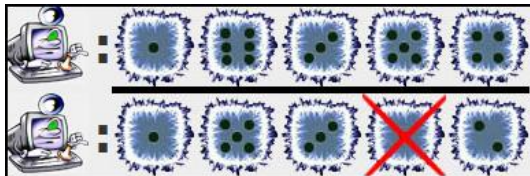
Az alapstratégia háromféle módon mondhatja felül az előző licitet: vagy a darabszámot, illetve a kockaértéket mondja felül, vagy mond egy teljesen új licitet.

Ha a kezdőlicitje nagyobb, mint az előző licit, akkor a kezdőlicittel mondja felül az előzőt. Ha pedig az előző licit több mint a kezdőlicit, akkor a stratégia megnézi, hogy milyen értékű kockából van neki a legtöbb az egyesekkel kiegészítve. Ha a kapott kockaérték kisebb, mint a saját kockaérték, akkor az előző licit darabszámát és a saját kockaértéket küldi tovább. De ha a saját kockaérték a kevesebb, akkor a darabszámot növeli és végül a növelt darabszámot és a kapott kockaértéket küldi tovább licitként.

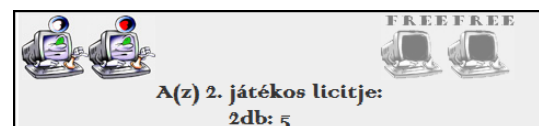
```
licit(kapott_licit) {
    if (saját_kezdő_licit > kapott_licit)
        licit_küld(saját_kezdő_licit);
    else {
        if (max_kocka_érték > kapott_licit_kocka_érték)
            licit_küld(kapott_licit_kocka_db, max_kocka_érték);
        else
            licit_küld
                (kapott_licit_kocka_db++, kapott_licit_kocka_érték);
    }
}
```

Példa:

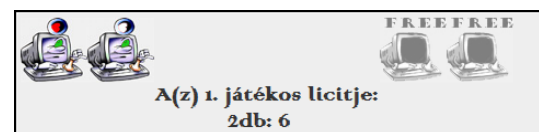
Két játékos játszik egymás ellen. A második játékos kezdi a licitkört két darab ötössel (egy ötös és egy joker). Az első játékosnak küldte a licitet, aki mivel annyi neki is van elfogadta és mivel van két hatosa, ami többet ér, mint a két ötös, így azt küldi tovább licitként.



20. ábra: Játékosok kockái



21. ábra: A második játékos kezdőlicitje



22. ábra: Az első játékos rálicitál

Ha az alapstratégia joker licitet kap, akkor kiszámítja az egyesek számának várható értékét majd megnöveli eggyel.

Ha a megnövelt várható érték nagyobb, mint a kapott jokerek darabszáma, akkor licitként eggyel több jokert küld tovább, mint amennyit kapott. Ha a megnövelt érték egyenlő a kapottal, akkor a stratégia kilép a joker bemondásból, megduplázza a kapott összeget, hozzáad egyet, majd ezt az új összeget kiegészíti azzal a kockaértékkel, amelyikből neki van a legtöbb és tovább küldi licitként. Minden más esetben pedig „dudo” licitet ad tovább.

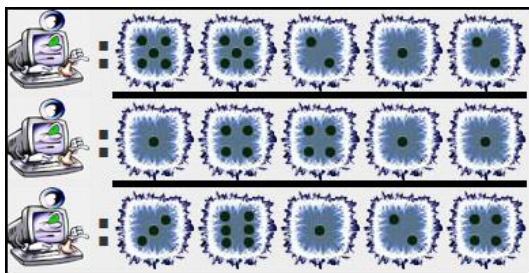
```

joker_licit(kapott_joker_licit) {
    if (várható_érték(joker)++ > kapott_joker_licit_db)
        licit_küld(kapott_joker_licit_db++);
    else if (várható_érték(joker)++ == kapott_joker_licit_db)
        normál_licit();
    else
        dudo();
}

```

Példa:

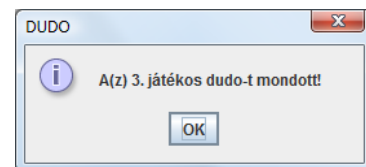
Ha hárman vannak játékban és mindenkinek megvan az összes kockája és a második játékos utolsó licitje öt darab egyes és a következő játékosnak csak egy egyese van akkor „dudo”-t fog mondani.



23. ábra: Játékosok kockái



24. ábra: Második játékos licitje



25. ábra: A harmadik játékos „dudo”-ja

Az alapstratégia előnye, hogy olyan liciteket mond, ami nagy valószínűséggel megvan az asztalon, de ez egyben a hátránya is, mivel nem licitál nagy összegekkel, ezért könnyen visszaér hozzá a kör.

5.2. Kiegészített gépstratégia

Ez a stratégia az alap stratégia első módosított és továbbfejlesztett változata. A kezdőlicit összegét ugyanúgy határozza meg, mint az alap gépstratégia.

Ennél a stratégiánál alkalmaztuk azt is, amit David Skalnisky a Póker alaptételében megfogalmazott. Az ellenfél helyzetének az átlátása a stratégia döntési algoritmusában jelenik meg. A döntés annyiban változott az alapstratégiához képest, hogy kiegészült egy licitBetolt() metódussal, ami arra szolgál, hogy a licitekből visszakövetkeztet a játékosok kockáinak az értékére és azt egy tömbben eltárolja. A visszakövetkeztetést még nem olyan pontosan végzi, mint az előrelátó stratégia.

```

licitBetolt(licit[], licitek_db) {
    for(i = 0; i < licitek_db; i++)
        ellenfelek_kockái[licit[i].kockaérték-1] +=
        mennyi_kocka(licit[i]);
}

```

A rálicitálás módja az alap gépstratégiához nagymértékben hasonlít. Csak az a különbség, hogy a kockák várható értékének kiszámolásakor a saját ismert kockáin kívül figyelembe veszi a többiek ismert kockáit is.

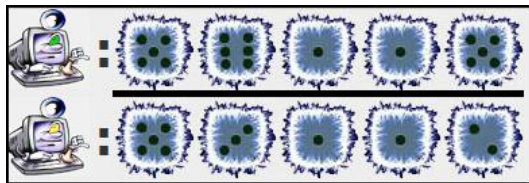
```

várható_érték(max_kocka_érték) {
    for (i = 0; i < összes_játékos; i++)
        összes_kocka_db += játékos[i].kocka_db;
    for (i = 0; i < 6; i++)
        összes_kocka_db -= ellenfelek_kockái[i];
    if (max_kocka_érték == joker)
        várható_érték = összes_kocka_db/6+max_kocka_db;
    else
        várható_érték = (összes_kocka_db*2)/6+max_kocka_db;
    várható_érték += ellenfelek_kockái[max_kocka_érték];
    return (várható_érték);
}

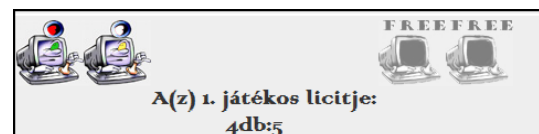
```

Példa:

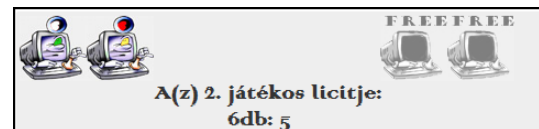
Az első játékosnak van négy ötöse, amit el is küld kezdőlicitként, mivel a másodiknak is van három, ezért emel a liciten. A licitet nem úgy emeli, mint ahogy az alapgép tenné, aki erre öt ötöst mondana, hanem mivel tudja, hogy az első játékosnak is van ötöse, ezért hat ötössel licitál tovább.



26. ábra: Játékosok kockái



27. ábra: Az alapgép kezdőlicitje



28. ábra: A kiegészített gépstratégia rálicitál

A joker licit majdnem ugyanúgy működik, mint az alapstratégiánál. A változás hasonló az előző ponthoz: akkor lép ki a joker licitból és mondja újra a saját kockáit, ha az ismert kockák és a várható értékek segítségével felül tudja mondani a licitet. Ellenkező esetben „dudo”-t mond.

A kiegészített gépstratégia előnye az alapstratégiával szemben, hogy erősebb liciteket tud bementeni, mint az alap. Hátránya, hogy ha nem alapgéppel játszik, akkor könnyen félrevezethető.

5.3. Előrelátó gépstratégia

Ez a stratégia is az alap stratégia fejlesztéseként jött létre. A lényege, hogy nyilvántartást vezet az ellenfelei dobókockáiról és ennek a segítségével megpróbálja előre tervezni a játékot.

A kezdőlicit összegét ugyanúgy határozza meg, mint az alap gépstratégia.

A Kiegészített gépstratégiahoz hasonlóan ez a gép is rendelkezik egy licitBetolt() metódussal. Ez a metódus arra szolgál, hogy nyomon követi az asztalon lévő licitbementéseket és ezekből a bementá-

sokból az egyes gépstratégiák licitemelési szokásai szerint visszakövetkeztet az adott játékosnál lévő kockákra.

```
licitBetolt(licit[], licitek_db) {
    játékos[licit[0].játékos_azonosító][licit[0].kockaérték-1]
    += mennyi_kocka(kezdőlicit);
    for(i = 1; i < licitek_db; i++)
        játékos[licit[i].játékos_azonosító][licit[i].kockaérték-1]
        += mennyi_kocka(licit[i], licit[i-1]);
}
```

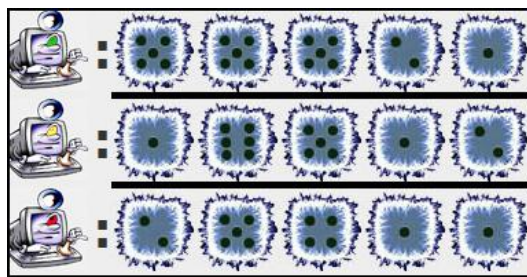
Ezekről a kockákról nyilvántartást vezet, amit szükség esetén a döntésnél vagy a rálicitálásnál használ is. Ez a stratégia az alaptól eltérően a döntésnél nem csak azt nézi meg, hogy a kezdőlicittel felül tudja-e mondani az előzőt, hanem hogy az ismert kockák segítségével felül tudja-e mondani. Ha felül tudja, akkor nem gondolkodik, hogy meglehet-e az előző licitösszeg, hanem felülmondja.

```
if (kapott_licit < saját_kezdő_licit ||
    kapott_licit < ismert_kockák_licit)
    licitálás();
```

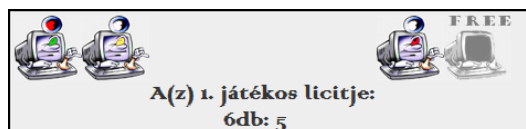
A licit annyiban változott az előző stratégiákhoz képest, hogy az új licitnél nemcsak a saját kockáit veszi figyelembe, hanem az ellenfelei ismert kockáit is.

Példa:

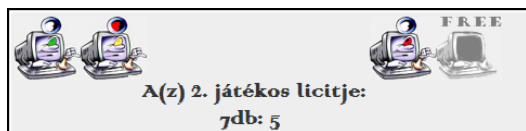
Ha három játékos játszik és az első játékos kezdőlicitje hat db ötös akkor abból ki tudja számítani, hogy az első játékosnak van négy darab ötöse. Mivel a következő játékos ezt a licitet elfogadta és felüllicitálta minimum három ötöse neki is van. Tehát az asztalon az előző két játékosnak biztos, hogy van hét darab ötöse, az előrelátónak pedig van három ötöse, így nyugodtan felüllicitálhatja az előzőt 10 darab ötössel.



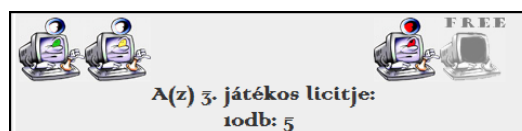
29. ábra: Játékosok kockái



30. ábra: Kezdőlicit



31. ábra: Második licit



32. ábra: Előrelátó játékos licitje

A stratégia jokerlicitálása teljesen ugyanúgy működik, mint a kiegészített stratégiáé.

Az előrelátó stratégia előnye a többivel szemben, hogy a licitekből vissza tud következtetni ellenfelei kockáinak az értékére. A stratégia által használt, kifinomult kockákra való visszakövetkezés, azért is fontos, mert ennek a segítségével jobban átlátja az ellenfeleit és az ellenfeleknek megfelelően tud licitálni is. Az ellenfélre szabott licitálást ez a stratégia még nem alkalmazza, de a jövőben tervezzük ennek a megalkotását is.

Hátránya, hogy a Póker alaptételének azt a részét nem tartja be, hogy nem szabad úgy játszania, hogy az ellenfelei ismerjék az állását, ezt az eddig készített stratégiák egyike se tartja be. Az előzőekből is adódik a másik hátránya, hogy túlságosan hiszékeny, ha olyan stratégia ellen játszana, aki blöfföl, akkor az előrelátó stratégia rosszul következtetne vissza. A hiszékenységet kihasználva blöffölő stratégiát is készítettem, amiről a dolgozat egy későbbi részében lesz szó.

6. ELEMZÉSEK

Az elemzésekben három különböző játékos vett részt. A három játékos a három különböző gépstratégia volt. A három stratégia úgy foglalt helyet egymás mellett az asztalnál, hogy alapstratégia kezd, őt követi kiegészített stratégia majd az előrelátó stratégia és ezután visszakerül a licitálás joga az alapra. Mindhárom stratégiánál a licitemelés és a túréshatár paraméter is egyre lett állítva.

Az első elemzés egy parti elemzés volt, amelyben meg lehetett figyelni a játékosok gondolkodásmódját, hogy hogyan reagálnak az adott szituációban.

Az alapstratégia kezdőlicitje gyenge volt és a másik két stratégia ki tudta következtetni, hogy az alapnak milyen kockái vannak, így az alapstratégia esett ki először a játékból. A második helyezet a kiegészített stratégia lett. Az elemzést az előrelátó játékos végül úgy nyerte meg, hogy egy kockát se veszített el.

Ez az elemzés nagyon jól bemutatja a három stratégia egymáshoz való viszonyát. Igaz, hogy a szerencsén is nagyon sok múlik ebben a játékban, mert a kiegészített gépstratégia többször is olyan licitet mondott be, ami könnyen meglehetett volna, de a szerencse az előrelátó mellé állt és így meg is nyerte a játékot.

Egy-egy parti végeredményét nagyban befolyásolja az adott partiban a játékosok szerencséje, ezért egy játék lejátszása után nem lehet eldönteni, hogy melyik stratégia a jobb. A játékosok hatékonyságának megismeréséhez egymás után több partit is le kell játszani.

A parti elemzésnél beállított három stratégiával elvégeztem 5 darab 100 ezer fordulós gyorselemzést is. A gyorselemzésekhez is úgy ültek le egymás után a játékosok, hogy alapstratégia, kiegészített stratégia és előrelátó stratégia.

Az első gyorselemzésnél jokerbemondást kezdeményezni körönként csak egyszer lehetett, a játékosok végig úgy ültek egymás mellett, ahogy az elemzés elején elhelyezkedtek és minden játék az alapgép kezdőlicitjével kezdődött. A szimuláció végére az előrelátó stratégia nyert a legtöbbször, 59167-szer, a második a kiegészített stratégia lett 31630 nyert körrel. Hiába kezdődött minden új játék az alapstratégia licitjével, ő csak 9203 kört tudott megnyerni, így az utolsó lett ebben az elemzésben.

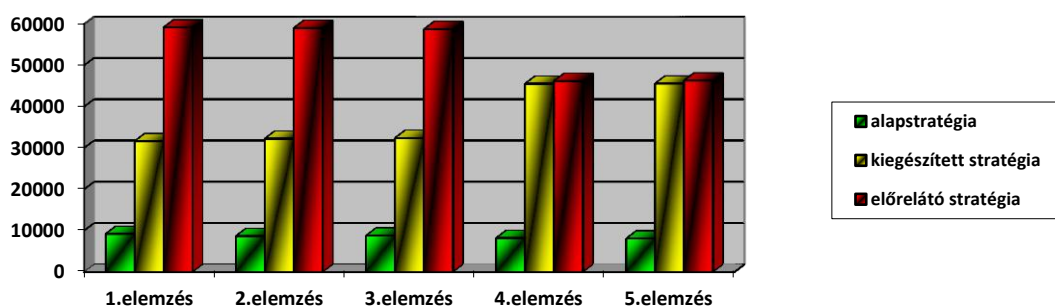
A második elemzésnél annyi változás történt az előzőhöz képest, hogy licitkört nem csak az első játékos, hanem sorban minden stratégia kezdhet. Ennek az elemzésnek is nagyon hasonlóhoz lett az előzőhöz a végeredménye. Itt is az előrelátó stratégia nyert a legtöbbször és az alapgép legkevesebbszer.

A következő szimulációnál a kezdés jogát a véletlenre bíztuk, minden új játéknál véletlenszerűen kiválasztott stratégia kezdte a játékot. Ez a kezdési mód matematikailag nagyban hasonlít az előzőre, mert annak a legnagyobb valószínűsége, hogy itt is minden játékos nagyjából ugyanannyiszor kezd. Az ülésrendben nem történt változás. Így végül az előző két elemzéshez nagyon hasonló statisztikát kaptunk eredményül, amiben az alap nyert legkevesebbszer és az előrelátó a legtöbbször.

A negyedik szimulációnál nem csak a kezdő játékos személye lett véletlenszerűre állítva, hanem a játékosok ülőhelye is. Minden játék végénél a játékosok felállnak, majd kezdetekor véletlenszerűen valahova leülnek a sorba. A játékot most is, mint már várni lehetett az alapgép veszítette el és az

előrelátó nyerte. De az eredmények tekintetében nagy változás történt, a kiegészített stratégia már nem maradt le annyira az előrelátótól, mint eddig. A százezer körből mindössze 646-tal nyert többet az előrelátó, mint a kiegészített stratégia.

Az utolsó elemzést azért készítettem el, hogy kiderüljön, mennyire van hatással az egyes stratégiák nyerési esélyére, ha egy körben többször is kezdeményezhetnek jokerlicitet. A szimulációnál mind az ülésrend, mind a kezdőjátékos személye véletlenszerűre lett állítva. A szimuláció végeredménye nagyon hasonlít az előző elemzésre. Az alapgép volt a leggyengébb, az előrelátó stratégia pedig egy kicsivel jobb volt a kiegészített gépstratégiánál. Az alapgép kevesebbszer nyert, de ez a végeredményt nem befolyásolta. Így azt a következtetést lehet levonni, hogy ezeknél a stratégiáknál, ha csak hárman játszanak egymás ellen, nem befolyásolja a nyerő játékos személyét a jokerlicit bemondása.

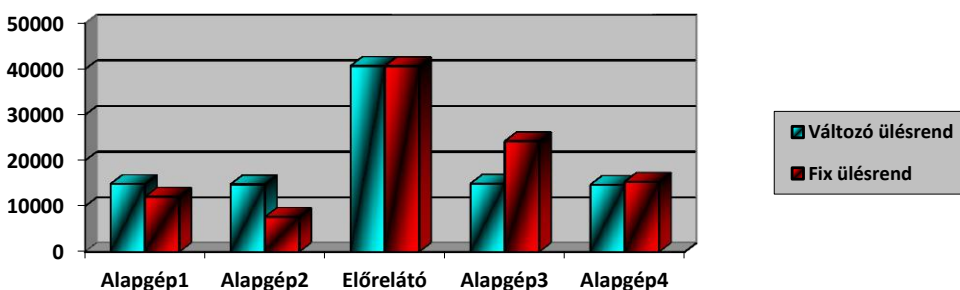


33. ábra: A szimulációk végeredménye

Igaz, hogy a Perudo alapvetően szerencsejáték, de az elemzésekből egyértelműen látszik, hogy nem mindegy milyen stratégiával játsszák a játékosok, mert ezeknél az elemzéseknél egyes stratégiákkal többször nyertek, mint a másokkal.

Az előző elemzésekből az is kiderült, hogy az ülésrend nagyban befolyásolja a játék végeredményét, de ezekből még nem lehetett megtudni, hogy hogyan.

Ahhoz, hogy kiderüljön, pontosan mi is történik, két új szimulációt futtattam le. Mindkét szimulációban öt játékos volt, négy alap és egy előrelátó. Az első szimulációnál véletlenszerű ülésrenddel, míg a másikban fix ülésrenddel úgy, hogy az előrelátó ült középen.



34. ábra: A szimulációk végeredménye

Mind a fix, mind a változó ülésrendes szimulációnál az előrelátó stratégia nyert a legtöbbször és az alapgépek a legkevesebbszer. Az alapgépek nyerési eloszlása nagyon érdekes. A véletlenszerű ülésrendnél mindegyik körülbelül 15000 kört nyert meg. A fix ülésrendnél lépcsőzetesen az előrelátó után elhelyezkedő a legtöbbet, majd mindegyik kevesebbet, és az előrelátó előtti a legkevesebbet. Ennek oka az lehet, hogy az előrelátó gép mindig a maga előtti gépnek tud jókor „dudo”-t mondani és ezzel kiejtetni a játékból, ha pedig a maga előttit kiejtette, akkor az azelőtti kerül eléje, majd az azelőtt és így tovább. Ezért amelyik játékosokhoz a legjobban el tud jutni az előrelátó gép „dudo”-ja, azok a játékosok esnek ki először, azok nyernek a legkevesebb kört.

7. EMBER A GÉP ELLEN

Érdekes a gépi játékosok egymás ellen elért teljesítménye, de az igazán fontos az, hogy a gépi játékos ember ellen is eredményes-e. Az a jó stratégia, amelyik ellenféltől függetlenül győzni tud a játszmák során.

A játék paramétereit a program úgy állítja be, hogy ember elleni játszmák esetén véletlenszerűen választ a különböző stratégiák közül, hogy ne lehessen könnyen kitalálni, hogy hogyan játszanak a stratégiák.

A teszt során 3 különböző ember játszott 10 kört egymás után a gép ellen, a körökben mindig egy ember játszott egy gép ellen. A játszmák során a gép nyert a legtöbbször és az emberek a legkevesebbszer. Elemzések szempontjából ez a 3-szor 10 kör kevésnek tűnhet, de elegendő volt ahhoz, hogy bizonyos következtetéseket levonjunk ezekről a stratégiákról.

	1. teszt	2. teszt	3. teszt
1. kör	GÉP	GÉP	GÉP
2. kör	GÉP	EMBER	GÉP
3. kör	GÉP	GÉP	GÉP
4. kör	GÉP	GÉP	EMBER
5. kör	EMBER	GÉP	GÉP
6. kör	GÉP	GÉP	EMBER
7. kör	GÉP	GÉP	EMBER
8. kör	EMBER	GÉP	EMBER
9. kör	GÉP	EMBER	GÉP
10. kör	EMBER	EMBER	EMBER
Nyert körök száma	3	3	5

3. táblázat: Körök

Mindhárom ember esetében megfigyelhető volt, hogy a játszmák során az első köröket többségében a gép nyerte, majd a későbbi köröket az ember. Ennek az az oka, hogy az emberek elkezdtek kiismerni a gépi játékosokat. A kísérlet mindhárom szereplője rájött, hogy a gépi játékosok hogyan adják meg a kezdőlicitet, és a táblázatban második játékosként jelölt ember arról is tudott következtetéseket levonni, hogy hogyan mondja a játékos a további licitjeit. Az eredményében ez azért nem feltétlenül látszik, mert a teszt során ő nem tudta, hogy különböző stratégiákkal játszik és így hiába voltak jók a következtetései, egy másik stratégiánál azt már nem tudta kihasználni.

Az eredmények alapján érdemes részletesebben is foglalkozni a harmadik emberi játékosal, aki a teszt során ugyanannyi kört nyert, mint a gép. Ezt a játékost nem érdekelte, hogyan játszik és licitál a gép, hanem azt használta ki, hogy ismerte hogyan adja meg a stratégia a kezdőlicitet. A kezdőlicit

megadása után tudta, hogy az adott stratégiának melyik kockájából van a legtöbb és abból mennyi, ennek felhasználásával adta meg a következő licitet.

Ha a harmadik játékosnak nem volt se jokere, se olyan kockája, ami a kezdőlicitben szerepelt, akkor a darabszámot megnövelte eggyel. A stratégia erre, mivel a bement kockák darabszáma pont annyi, amennyi az ő kockáinak a darabszáma plusz egy ezt elfogadja és jokerlicitet kezd. Majd a harmadik játékos mivel neki nem volt jokere a licitet nem fogadta el, hanem dudo-t mondott és ha szerencséje volt megnyerte a kört. A játékos dudo bementésénél a szerencse nagyban befolyásolta a játék kimenetelét, mert a bementott jokerek darabszáma sok esetben még meglehetett volna a bementónál.

Ha a harmadik játékosnak volt olyan kockája amelyet a stratégia megadott a kezdőlicitben, akkor azoknak a kockáknak a darabszámával emelte meg a licitet.

Amikor nem a gép, hanem a játékos kezdte a licitbementést, akkor a harmadik játékos egy trükkel átadta a gépnek a licitálás kezdetének a jogát. Mindig egy kicsi kezdőlicitet adott meg, általában egy darab kettest és ezt a gép elfogadta, majd az ő kezdőlicitével folytatta a játékot.

Az egyik körben, amikor a harmadik játékos éppen az előrelátó gépstratégia ellen játszott, kezdőlicitként bementott úgy egy darab kettest, hogy nem volt neki egy kettese se. A gépnek viszont a ketteseből volt a legtöbb és azt hitte, hogy a játékosnak is van kettese, ezért a licitet úgy adta meg, hogy bementta a saját ketteseinek és a hitt kettéseknek a darabszámát. Az emberi játékos a továbbiakban is úgy gondolkodott, hogy a kezdőlicitet kapta eredményül, ezért azt megnövelte eggyel. A gép pedig az ilyen sok kettest már nem tudta elfogadni és dudo-t mondott, ezért a játékos elvesztett egy kockát.

Ha egy játékos mindig ugyanazzal a stratégiával játszik, ugyanúgy adja meg a kezdőlicitet, akkor könnyen rá lehet jönni a stratégiájára, de ha egy játékos többfajta stratégiával játszik, akkor könnyen bele lehet futni olyan bakikba, mint amibe a harmadik ember belefutott.

8. STRATÉGIÁK/2.

8.1. Óvatos stratégia

Tanulságosak voltak az emberi játékosokkal a gépi játékosok ellen játszott partik. A tapasztalatok lehetőséget adtak arra, hogy fejlettebb gépi játékosokat állítsak elő. Ennek a stratégiának az ötletét az ember-gép játszmákban a harmadik játékos adta. Ezen gépi játékosok algoritmusának megtervezése és leprogramozása a TDK munkám egyik érdekes feladata.

Mivel az emberek a stratégiák kezdőlicitéből vonták le a legegyszerűbben a következtetéseket, ezért ez a stratégia óvatosan bánt a kezdőlicitek bemondásával.

Ha a stratégia mindig ugyanaz az algoritmus alapján mondaná be a kezdőlicitet, akkor könnyen ki lehetne találni a kezdőlicit bemondásának az algoritmusát.

Ha például mindig egy kettessel kezdene, akkor a következő játékos egy idő után két kettést vagy egy hármast mondana és azzal visszaadná a gépnek a licitálás jogát, ezért ezt a stratégiát úgy kell kifejleszteni, hogy ne legyen ennyire triviális a bemondása, ne lehessen ilyen könnyen visszafejteni.

Ennek a stratégiának a kialakításánál azt használtuk ki, hogy a Perudo dinamikus játék és így menet közben módosíthatja a stratégiáját. A stratégiát úgy terveztük meg, hogy több különböző algoritmusból véletlenszerűen választhatja ki a kezdőlicit bemondást.

Az egyik algoritmus lehetne az első három stratégia kezdőlicit bemondó algoritmus. A másik lehetne egy olyan algoritmus, ami nem annak a kockának mondja be a várható értékét amiből a legtöbb van neki, hanem annak amiből a legkevesebb. Lehetne olyan algoritmus is, ami ezeknek a várható értékeknek a felét mondaná be. Ezeknek a segítségével az ellenfél nem tudná, hogy melyik kockából van a legtöbb és hogy mennyi lehet ezekből.

Ha a stratégia csak ilyen algoritmusokkal kezdené a köröket, akkor az igaz, hogy az ellenfél nem jönne rá, hogy miből hány darab van neki, de abban biztos lehet, hogy az adott kockából van neki, ezért időnként a stratégiának blöffölnie is kell, azaz egy olyan licittel kell kezdeni, amiből nem lehet rájönni a kockáira. Érdemes időnként olyan kockaértéket bemondani, amilyen kockája neki nincs is. A blöffel vigyázni kell, ekkor is „értelmes” darabszámot kell bemondani, hogy az az ellenfél számára hihető legyen. A megtervezett, hihető blöffről a tanulmány egy későbbi részében részletesebben is írtunk.

Az itt felsorolt kezdőlicit bemondásokon kívül még lehet egyebeket is kitalálni, minél több algoritmus van, annál nehezebb kitalálni, visszafejteni a megoldásokat.

8.2. Esetalapú stratégia

A stratégia ötlete onnan jött, hogy az emberek bizonyos helyzeteket könnyebben elfogadnak. Ha például valaki bemond X darab kettést, akkor azt az ellenfél könnyebben elfogadja, mintha X darab hatost mondott volna be, mert könnyebb felülmondani. Pedig ugyanakkora valószínűséggel van meg az asztalon az X darab kettes, mint az X darab hatos.

Az alapötletet tehát az, hogy a gépi játékos a játék során megfigyeli az asztalon lévő kockák darabszámát, a bementő kockáinak a darabszámát, a saját kockáinak a darabszámát és az értékét, majd ezeket minden licitnél felveszi egy esetként. Sok lejátszott parti során tanulna, ha még az adott eset nem szerepelt az esetek között, akkor az esettel együtt felvenné azt, hogy hogyan kell reagálni az adott esetben, hogy nyerjen. Ha pedig már szerepelt, akkor az esetnek megfelelően bementja a dudo-t vagy a megfelelő licitet.

Az eset mellett felvesz két változót, amikben a jó és a rossz döntések számát tárolja. Ha a bementésnél jól döntött, és az adott esettel megnyeri a kört, akkor az eset és a döntés mellett megnöveli a jó döntések változó értékét eggyel, ha rosszul döntött, akkor a rossz döntések változó értékét növeli eggyel és felvesz egy új esetet amiben ugyanezek a körülmények mellett a jó döntés szerepel. Majd a későbbiekben az egyforma körülményű esetek közül mindig azt fogja alkalmazni, amelyiknél a jó döntés változó értéke a legmagasabb. Az ilyen gépi játékos nagyon sok lejátszott kör után válna igazán „profivá”.

```
eset = new eset(össz_kocka_d, bementő_kocka_db, saját_kocka[]);
if (esetek.létezik(eset)) {
    döntés=esetek[eset].reagálás();
    if (döntés==siker)
        esetek[eset].döntés(helyes)++;
    else
        esetek[eset].döntés(hibás)++;
}
else
    esetek.addeset(eset);
```

8.3. Vegyes és puhatolózó stratégia

Az óvatos stratégia esetében a stratégia a kezdőlicitet adta meg kiszámíthatatlanul. A stratégia akkor fog jól működni, ha nem csak a kezdőlicit értékét mondja be oly módon, hanem a további licitálást is úgy végzi, hogy ne lehessen könnyen visszafejteni az algoritmust.

A vegyes stratégiának az a lényege, hogyha kap egy licitet, akkor az előző stratégiák valamelyikének az algoritmus szerint dönt és licitál. Az algoritmust mindig véletlenszerűen választaná ki, hogy ne lehessen egyértelműen kitalálni a stratégiáját.

Az előrelátó stratégia a licitbetöltésénél a mások kockái értékének kiderítéséhez csak azt veszi figyelembe, hogy az előző játékos az adott licitet miért fogadta el, és azt nem, hogy miért az adott licittel mondta felül az előző licitet. A puhatolózó stratégiába azt kellene beépíteni, hogy derítse ki miért mondta az adott licitet, milyen stratégiát használhat a másik játékos.

Ha az ellenfél vegyes stratégiát használ, akkor a puhatolózó stratégiának nehéz dolga lesz az algoritmus kitalálása során, de a második licitkörnél nagy valószínűséggel kiderül, hogy az ellenfél ugyanazt a stratégiát alkalmazza mindig vagy változtatgatja.

A puhatolózó stratégia az ellenfél megtévesztése érdekében lehetne egy speciális vegyes stratégia.

Ha egy stratégiában sikerül jól megoldani a puhatolózást és a kiszámíthatatlanságot, akkor megvalósítottuk, amit David Sklansky a Póker alaptételében leszögezett, mely szerint az a stratégia a jó, amelyik úgy játszik, mintha ő látná az ellenfelei lapjait, ellenfelei nem látnak át őrajta.

9. A BLÖFF, MINT STRATÉGIA

Az emberek ellen lejátszott játszmákból kiderült, hogy bizonyos szituációkban a racionális viselkedéssel nem lehet nyerni. A partikból az is kiderült, hogy a játékosoknak kiismerhetetlennek kell lenniük, mert ha mindig ugyanúgy, ugyanazon stratégia szerint döntenek, akkor könnyen le lehet őket győzni. A kiismerhetetlenség egyik formája a blöff.

A blöff olyan pszichológiai fogalom, ami a matematikában is megjelenik, már Neumann János is sokat foglalkozott vele a játékelmélet alapjául szolgáló művében. Neumann János nagyon szeretett póke-rezni, ezért a pókeren keresztül mutatta be a műben a blöff megtérülését.

A blöff nem más, mint az ellenfél megtévesztésének az egyik módja. A blöfföt, mint a tudatos megtévesztést tekinthetjük a játékban megengedett hazugságnak is. A hazugság tudatosan valótlan állítás, tágabb értelemben minden olyan, a valóságnak ellentmondó közlés, amelynek célja mások félrevezetése.⁶

A hazugságra és így a blöffre is vannak különböző árulkodó jelek. Ezek a támpontok nagyrészt az érzelmi és verbális jelzésekkel kapcsolatosak. Adrian Furnham a következő jeleket gyűjtötte össze:⁷

- Válasz-latencia, vagyis az az idő, amely a kérdés vége és a válasz kezdete között telik el. Aki hazudik, annál ez az idő valamivel hosszabb.
- A szünetek gyakorisága és hossza, vagyis hogy az illető hányszor tűnik bizonytalannak azzal kapcsolatban, hogy mit mondjon, vagy hogy mit mondott.
- Nyelvi távolságtartás – nem azt mondja, hogy „én” vagy „ő”, hanem személytelenül fogalmaz, még akkor is, amikor olyan eseményeket idéz fel, amelyeknek a részese volt.
- Beszédhibák-tétovázás, freudi elszólások, gyakori „ö-ö-ö-zés”, gyakori sebességváltások a beszédben.
- Lassú, de egyenetlen beszéd – arról árulkodik, hogy a vizsgált személy beszéd közben folyton gondolkodik, de olykor elakad. Hogy egy adott kérdés kapcsán valami nincs rendben, azt az jelzi, ha azzal a kérdéssel következik be a lelassulás vagy elakadás.
- Túlságos törekvés arra, hogy kitöltse a szüneteket, vagyis hogy folyamatosan beszéljen akkor is, amikor nem szükséges. Akik hazudnak, azok általában túlkompensálnak, és a többnyire észrevehetetlen, rövid csendek is zavarják őket.
- Túl sokszor emelkedik meg a beszédhang magassága – vagyis ahelyett, hogy a hang egy válasz végén lemenne, emelkedik, mintha kérdése lenne. Ez úgy hangzik, mintha a beszélő azt kérdezné: „Ezt most elhiszi?”
- A hang rezonanciája csökken – laposabb, kevésbé mély, monoton lesz.
- Fészkelődés – az illető túl sokat mocorog a székekben. Számos szokatlan fej-, láb- és törzsmozgást végez.
- Túl folyamatos szemkontaktust tart. Akik hazudnak, általában ezen a téren is túlkompensálnak.

⁶ Christian Leszczynski, Wilfried Schuhmann és társai: Pszichológiai lexikon

⁷ Adrian Furnham: 50 psychology ideas you really need to know

- Mikrokifejezések, vagyis villanások (meglepetés, sértődés, harag jelei). A felvétel kimerevítése nélkül ezek nehezen észrevehetők.
- Az önvigasztaló gesztusok megszaporodása – saját test, arc és felsőtest megérintése.
- Az arckifejezés, különösen a mosolygás, a pislogás és a pillantás irányának változtatásai.

Ha a számítógép emberi játékos ellen játszik, akkor a különböző arcfelismerő és mimikaelemző szoftverek segítségével a gép is rá tud jönni, hogy mikor blöfföl az adott személy. Igaz, nem blöff elemzésre, de ilyen mimikaelemző alkalmazást a magyar Facereading Kft. is készített 2007-ben. Az általuk készített program az embereket személyiségét vizsgálta.

Ahhoz, hogy a megtévesztés sikeres legyen az emberi játékosok között, illetve az ember-gép játékok során, a játékosok viselkedésén nem szabad látszani, hogy mikor blöffölnek. A blöfföt is „pókerarcot” alkalmazva ugyanolyan „fapofával” kell bemondani, mint a többi licitet. A gépi stratégiák esetében a viselkedésből, a gesztusokból, arcvonásokból nem lehet a blöffre rájönni, mert ezekkel a számítógép nem rendelkezik, ezért a gépek esetében valami más megközelítést kell alkalmazni. A blöff a számítógép stratégiájának a része. A stratégia a gépi játékos esetében különböző szabályszerűségek alapján valamilyen algoritmus segítségével van leírva. Ahhoz, hogy ki tudjuk deríteni, mikor blöfföl az adott gépi játékos, ezeket a szabályszerűségeket kell elemezni. Az algoritmusok elemzésénél sok kör lejátszása után ki lehet deríteni, hogy mikor blöfföl az adott játékos.

A blöffre a Perudoban azért van szükség, mert ha egyik játékos se élne vele, akkor egy egyszerű szencsejáték lenne, amit az nyerne meg, aki nagyobbat dob.

A játéknak a blöff egy nagyon fontos eleme, a játékosnak tudnia kell, hogy mikor és hogyan kell blöffölni ahhoz, hogy a játékot megnyerje. Itt a nyerés nem egy játszma szó, hanem hosszú távra. Egy játszma esetében az ellenfél vagy elhiszi a blöfföt vagy nem, így 50% az esély arra, hogy a blöffölő nyer vagy veszít. A blöfföt is meg kell tervezni és úgy kell kialakítani, hogy hosszú távon nyerő legyen. Ha egy játékos mindig blöfföl, akkor az első egy-két menetben jobb eséllyel fog nyerni, de hosszú távon, amikor az ellenfelei kiismerik a szokásait biztos, hogy veszíteni fog. A blöfföt is úgy kell megtervezni, hogy kiszámíthatatlan legyen. Az eddig bemutatott óvatos és vegyes stratégiákban, mint a megtévesztés egyik módja már megjelent a kiszámíthatatlan blöff, de ez a blöff még csak véletlenszerű volt és nem előre megtervezett tudatos. Ezeknek a stratégiáknak a blöffje csak egy-egy stratégia ellen volt hatásos, a következőkben olyan matematikailag megtervezett blöfföt fogunk készíteni, ami játékostól függetlenül nyereséges lesz.

A blöff, mint hosszú távú stratégia kialakításához vegyük azt a példát, amikor a Perudot pénzben játsszák. A pénzben való játék azért fontos, mert ha a játéknak van tétje, akkor a játékosok megfontoltabban blöffölnek. Most itt a példában azért is fontos a pénzes játék, mert a fizetés szabályaihoz fogjuk úgy kialakítani a blöfföt, hogy több játszma után mindenképp nyereséges legyen.

Példa:

Most a példában az egyszerűség kedvéért a játékot két játékos fogja játszani. A fizetést úgy alakítottuk ki, hogy a kezdőlicitet bemondó játékos befizet 1 egység pénzt a bankba, majd a második játékos, ha felüllicitálja 2 egység pénzt és ezután minden játékos a felüllicitálásnál annyi pénzt fizet be, amennyi a bankban összesen van. Pl.: a második felüllicitálásnál 3 egységet (1+2), a harmadiknál pedig 6 egységet (1+2+3). Ehhez a játékhoz a fizetést nem csak így lehet kialakítani. Mi azért választottuk ezt a módszert, hogy a játékosoknak minden egyes lépésnél

mérlegelnie kelljen, hogy érdemes-e tovább mennie, illetve azért mert így különböző szabályszerűségeket is fel fogunk tudni írni a példa következő részében, ami megkönnyíti a számolást.

A következő táblázatban a pénz alakulását foglaltuk össze, hogy az előbb említett szabályszerűségekre fény derüljön. A „Dudo bemondás” oszlopban azok az összegek szerepelnek, amit az adott licitálásnál a dudot bemondó játékos nyer vagy veszít, a „Licit bemondás”-ánál pedig az az összeg szerepel, amit az adott licitet bemondó játékosnak be kell fizetni.

Licit	Dudo bemondás		Licit bemondás Befizet
	Ha veszít	Ha nyer	
Kezdőlicit			1x
1. Rálicitálás	0x	1x	2x
2. Rálicitálás	1x	2x	3x
3. Rálicitálás	2x	4x	6x
4. Rálicitálás	4x	8x	12x
5. Rálicitálás	8x	16x	24x
6. Rálicitálás	16x	32x	48x
7. Rálicitálás	32x	64x	96x
8. Rálicitálás	64x	128x	192x
9. Rálicitálás	128x	256x	384x
n. Rálicitálás	$2^{n-2}x$	$2^{n-1}x$	$2^n - 2^{n-2}x$

4. táblázat

A példában azt a szituációt fogjuk elemezni, amikor a játék a már a végéhez közeledik és az egyik játékosnak már csak egy dobókockája van. Ezt a játékost a későbbiekben **játékos1**-nek jelöljük. Játékos1 $n-1$ -dik licitként kap egy olyan érvényes licitet, amit csak jokerrel lehet felülmondani. Ha a játékos1 az n -dik licitként nem mondja felül az előző licitet, azaz dudo-t mond, akkor mivel az előző licit érvényes volt elveszít az előző táblázat szerint $2^{n-2}x$ pénzt.

Ha pedig felülmondja egy jokerrel és a következő játékos, akit **játékos2**-nek fogunk jelölni $n+1$ -dik licitként dudot mond, akkor, ha játékos1-nek van jokere nyer $2^{n-1}x$ -et, ha pedig nincs, akkor veszít $2^n x$ -et. Mivel zéróösszegű játékról van szó, ezeket az eredményeket is ki lehet olvasni táblázatból úgy, hogy az $n+1$ -dik körben annyit nyer a játékos1, amennyit a játékos2 veszít és annyit veszít, amennyit a másik nyer.

Ha pedig a játékos2 nem dudot mond, hanem az $n+1$ -dik licitnél ráemel játékos1 joker licitére, akkor az $n+2$ -dik licitnél játékos1-nek mindenképp dudo-t kell mondania, mert tovább racionálisan nem tudja növelni a licitet. A dudo után, ha játékos1-nek van jokere vagy az első vagy a második játékos nyer attól függően, hogy játékos2-nek is van-e jokere. Ebben az esetben, ha játékos2 nyer, akkor játékos1 elveszít $2^n x$ -et, ha pedig játékos1 nyer, akkor $2^{n+1}x$ -et fog nyerni. Ha pedig játékos1-nek nincs jokere, akkor a kört mindenképpen játékos1 fogja nyerni, mert játékos2 nem tud érvényes licitet mondani. Ebben a szituációban az $n+2$ -dik licitnél a táblázat szerint játékos1 $2^{n+1}x$ -et fog nyerni. Innen is látszik, hogy szükség van a blöffre, mert ha az egyik játékos el tudja hitetni a másik játékoskal, hogy neki van jokere, akkor ebben a szituációban mindenképp nyer.

```

if (játékos1.dudo())
    játékos1=-2^(n-2)*x;
else if (játékos1.joker_licit()){
    if (játékos2.dudo()){
        if (játékos1.kocka==joker)
            játékos1=+2^(n-1)*x;
        else
            játékos1=-2^n*x;
    }
    else if (játékos2.emel()){
        if (játékos1.kocka==joker)
            játékos1=+2^(n+1)*x || játékos1=-2^n*x; /*50%*/
        else
            játékos1=+2^(n-1)*x;
    }
}

```

Nézzük azt az esetet, amikor a játékos1 és a játékos2 egymás után játszik **24** kört. A 24-nek különösebb jelentősége nincs, csak a könnyebb számolás miatt választottuk ezt a számot. A példában tételezzük fel, hogy ebben a 24 játékban a játék vége mindig az előző szisztéma szerint fog alakulni, az egyéb esetekkel most ne foglalkozunk. A valószínűségek alapján 1/6-szor lesz joker a játékos1 kockáján az utolsó körben, mivel a kocka 6 féle értéket vehet fel. Tehát a 24 menetből **4-et** fog úgy befejezni, hogy joker van nála (24/6) és **20-at** úgy, hogy valami más.

Ha a játékos1 **blöff nélkül** játszik, akkor 20 körben leáll és 4 körben megy tovább. Azokban a körökben, amikben leáll veszíteni fog a $2^{n-2}x$ -et. A maradék 4 körben, amikor játékos1-nek van jokere és továbbmegy, játékos2 50%-ban fog emelni és dudot mondani. Amikor játékos2 dudot mond $2^{n-1}x$ -et fog játékos1 nyerni, amikor pedig játékos2 emel játékos1 50%-ban vagy $2^n x$ -et fog veszíteni vagy $2^{n+1}x$ -et fog nyerni. Összesítve 24 körben játékos1 blöff nélkül $20(-2^{n-2}x)+4((2^{n+1}x-2^n x)/2+2^{n-1}x)/2$ -t fog nyerni. Az egyszerűség kedvéért n helyére helyettesítsünk **nullát**. Ha n helyére nullát írunk akkor az előző egyenletből a következő egyenletet kapjuk: $20(-2^{-2}x)+4((2x-x)/2+2^{-1}x)/2=-3x$. Ezek szerint blöff nélkül játékos1 **3x-et fog veszíteni** összesen a 24 menet során.

A következőben azt vizsgáljuk meg, hogy mi történik akkor, ha játékos1 **blöfföl** is. A példában 25%-os valószínűséggel fog játékos1 blöffölni, de csak akkor, amikor nem jokert dob. Tehát a 24 menetből a valószínűségek alapján ismét 4-szer lesz játékos1-nél joker és 20-szor valami más. A 20 valami másból 1/4-szer fog blöffölni. Az 1/4 nem úgy fog kijönni, hogy amikor nem jokert dob minden 4.-nél blöfföl, mert az hosszútávon kiszámítható, hanem mert a blöfföknek véletlenszerűeknek kell lenniük generál egy egész számot 1-től 4-ig és ha 4-est generál, akkor blöfföl egyéb esetben igazat mond. Ezt a véletlen szám generálást gépi játékos esetében könnyen meg lehet oldani, de az emberi játékosnak valami egyéb trükkre van szüksége. Ilyen trükk lehet az, hogy vesz egy négyoldalú „dobókockát” és dob azzal is egyet. Ha 1/4 valószínűséggel blöfföl, akkor a 20 esetből 15-ször fogja blöff nélkül befejezni a játékot és 5-ször fog jokert blöffölni. Abban a 15 körben, amikor leáll most is el fog veszíteni $2^{n-2}x$ -et. Ugyanúgy, mint amikor blöff nélkül játszott, abban a 4 esetben, amikor van nála joker vagy nyerni fog $2^{n-1}x$ vagy $2^{n+1}x$ -et vagy el fog veszíteni $2^n x$ -et. Amikor blöfföl a játékos1 a táblázat szerint $2^n x$ -et fog veszíteni, ha játékos2 nem fogadja el a blöfföt és $2^{n+1}x$ -et nyerni, ha elfogadja. Játékos2 a valószínűségek alapján, mert nem tudja játékos1 mikor blöfföl és mikor nem 50%-ban fogja a blöfföt elfogadni. Összesítve 24 körben játékos1 blöffel $15(-2^{n-2}x)+4((2^{n+1}x-2^n x)/2+2^{n-1}x)/2+5(-2^n x+2^{n+1}x)/2$ -t fog nyerni. Ha az egyszerűsítéshez n helyére ismét **nullát** írunk, akkor azt kapjuk, hogy $15(-2^{-2}x)+4((2x-x)/2+2^{-1}x)/2+5(-x+2x)/2=0,75x$. Kis számolás alapján kiderült, hogy hosszútávon ezzel a 25%-os valószínűségű blöffel 24 körben játékos1 **0,75x-et fog nyerni**.

Ebből is látszik, hogy a blöffre szükség van a játék során ahhoz, hogy nyerni tudjunk, mert egy olyan helyzetből is lehet nyerő pozíciót kialakítani, amiben blöff nélkül veszítenénk. A 0,75x nem biztos, hogy a maximum amit itt nyerni lehet, ez az az összeg, amit egy 1/4-es blöffel el lehet érni.

Ez a példa is azt mutatja, hogy a Perudo hiába szerencsejáték is, egy jól átgondolt stratégiával hosszútávon nyerni lehet benne. Itt lehet gondolni a bevezetőben említett Póker bajnokságokra is, ahol többször nyernek ugyanazok az emberek. A pókerbajnokok nyerő stratégiájához hozzátartozik a blöff is, mert a bajnokságot csak akkor tudják megnyerni többször egymás után, hogyha a rossz pozícióból is előnyt tudnak kovácsolni maguknak, azaz úgy játsszák a játékot, ahogy David Sklansky a Póker alap-tételében megfogalmazta.

Az előbb bemutatott blöff a Perudonak egy adott szituációjához készült, ha a játékos csak ezt a blöfföt alkalmazza, akkor még nem biztos, hogy hosszútávon nyerni fog. Ahhoz, hogy sikeres legyen a többi gyakran előforduló szituációhoz is ki kell alakítani egy blöff stratégiát, amit be lehet vetni a nyelés érdekében.

10. PROGRAM ÉS ALGORITMUS

A számítógépes játékokat sokszor úgy készítik el, hogy a gépi ellenfél látja a játékos pillanatnyi pozícióját és véletlenszerűen dönti el, hogy engedi-e a játékosnak megnyerni a játszmát vagy kihasználja azt az előnyét, hogy mindent lát és legyőzi az ellenfelét. A TDK dolgozat során készített programnál én ilyen módszereket nem alkalmaztam. Figyeltem arra, hogy a gépi játékosok ne csaljanak, hanem egy stratégia alapján nyerjék meg a játékot.

A dolgozat célja nem egy izgalmas játék létrehozása, hanem egy adott játékhoz való használható stratégiák kialakítása volt. Így a játék tulajdonképpen a dolgozatnak egy hasznos mellékterméke lett, mert a program elsősorban a teszteléshez készült. A stratégiákat muszáj volt folyamatosan elemezni, hogy a gyengeségeiket ki tudjam javítani és egy szinte legyőzhetetlen stratégiát tudjak kifejleszteni.

A stratégiák fejlesztésénél a tervezés és a tesztelés vette el a legtöbb időt. A játék során szituációtól függően döntenek az emberek, a programba nehéz az összes lehetséges szituációra jó döntést alkalmazni. A fejlesztés során ezt a problémát úgy oldottam meg, hogy először készítettem egy egyszerű, de logikusan gondolkodó valós életben is használható stratégiát. Majd a következő lépésekben ennek a stratégiának a kijátszását programoztam bele a következő stratégiákba. Az első stratégia tervezése során még nem terveztem meg minden lehetséges kimenetet, hanem a későbbi stratégiákat terveztem úgy, hogy az előzőeket le tudják győzni.

A stratégiák kifejlesztésénél a legnagyobb problémát az okozta, hogy ezekkel a stratégiákkal gépek játszanak, a gépeknek pedig nincsenek érzéseik, nem tudják eldönteni az ellenfélről, hogy blöfföl vagy komolyan gondolja az adott licitet.

Ha valós emberekkel játszunk, akkor ellenféltől függően vannak olyan szituációk, amikor merünk nagyobb licitet mondani, mint ami abban a szituációban várható lenne, és vannak olyan esetek is, amikor sokkal óvatosabbak vagyunk.

Egy számítógépes játékba érzéseket nem lehet programozni, a mesterséges intelligencia annyira még nem fejlődött ki, hogy a gépek érezzenek és véleményem szerint nem is fognak soha. Mivel a gépek nem éreznek, ezért nem láthatják az ellenfél mozgulataiból, szemrebbenéseiből, hogy szórakoznak-e vele vagy komolyan gondolja-e az adott liciteket.

Ezt a problémát nehéz áthidalni, de azt mindenképpen megtehetjük, hogy olyan stratégiákat alakítunk ki, amelyek nem foglalkoznak a megérzésekkel, hanem még a gondolkodó embernél is taktikusabbak és kiszámíthatatlanabbak. Konkrétan a vegyes stratégia kialakításánál az volt a legfőbb szempont, hogy kiszámíthatatlan legyen.

Az igaz, hogy a gépek az ellenfél mozgulataiból nem tudnak következtetni semmire, de a tetteiből igen. Ha például egy játékos mindig ugyanaz a stratégia szerint játszik, akkor a licitjeiből vissza lehet következtetni a kockáinak az értékére. A puhatolózó stratégia ezt próbálja kihasználni és a licitjeit ennek segítségével mondja be.

A stratégiák tesztelésénél gondot okozott, hogy mivel szerencsejátékról van szó, azért a véletleneknek is nagy szerepe van egy-egy parti kimeneténél. Ezt a problémát a játék elemző üzemmódjaival oldottam meg. Minél több partit játszanak le a játékosok, annál kevésbé befolyásolja az összesített eredményt a szerencse.

Igaz, hogy a program a stratégiák tesztelésére készült, de mivel egy játékot csináltam a teszteléshez, ezért fontosnak tartottam, hogy olyan mindenki számára érthető és használható legyen a felhasználói felülete.

A stratégiák tesztelésekor én már ismertem a szabályokat, ezért nekem a teszteléshez nem volt szükségem arra, hogy a játék tartalmazza ezt a leírást. De mivel a játék úgy készült, hogy mások is könnyen tudják használni, ezért a jól felépített felhasználói felület mellett azt is fontosnak tartottam, hogy a játék tartalmazza a szabályokat is.

```
protected int[] dontes(int licit, Map licitek, int licszam, int[] kilic) {  
  
    int x = licit % 10;  
    kapdb = (licit - x) / 10;  
    var = varhato(x);  
    int tova[] = new int[2];  
    if (x == 1) {  
        tova = egyesdontes(kapdb);  
    } else {  
        if (kapdb < var) {  
            tova = licit(licit);  
        } else if (kapdb <= var + turesHatar && kapdb >= var - turesHatar) {  
            if (PerudoJatek.joker) {  
                if (kapdb % 2 > 0) {  
                    kapdb++;  
                }  
                tova[0] = kapdb / 2;  
                tova[1] = 1;  
                if (Perudo.joker) {  
                    PerudoJatek.joker = false;  
                }  
            } else {  
                tova = licit(licit);  
            }  
        } else {  
            tova[0] = tova[1] = -1;  
        }  
    }  
  
    return tova;  
}
```

35. ábra: Kódrészlet

11. BEFEJEZÉS

A TDK dolgozatomban során egy szerencsejátékhoz terveztem különböző stratégiákat, amelyek segítségével nagyobb eséllyel lehet az adott szerencsejátékot megnyerni. A dolgozat elkészítése során a stratégiák többségét le is programoztam JAVA nyelven és ezeket a stratégiákat teszteltem is. Egyes stratégiákat még csak elméletben terveztem meg, de készülök ezeknek a stratégiáknak is a leprogramozására, hogy tesztelni lehessen ezeket is, illetve tervezem a dolgozatomban továbbfejlesztését és stratégiákkal való bővítését is.

A stratégiai alapú szerencsejátékok többségéhez lehetne a dolgozatban szereplő stratégiákhoz hasonló algoritmusú stratégiákat kidolgozni a nyerési esélyek növelése céljából.

Mivel szerencsejátékról van szó és a szerencsét sok olyan tényező befolyásolja, amit nem tudunk algoritmizálni, ezért olyan stratégiát nem lehet létrehozni, amelyik biztosan minden esetben nyer, csak a nyerési esélyeket lehet egy jó stratégiával megnövelni.

A program futtatásával kapcsolatos néhány további érdekességet, tapasztalatot és egyéb továbbfejlesztési lehetőségeket a prezentációban mutatok be.

A dolgozat végén szeretnék köszönetet mondani konzulenseimnek Kallós Gábornak és Takács Gábornak, hogy lehetőséget biztosítottak a dolgozat megírásához és külön köszönöm a sok segítséget, amit nyújtottak a sikeres TDK munka elkészítéséhez.

IRODALOMJEGYZÉK

1. **Ady Krisztián:** Az arcfelismerés a HR-esek munkáját is könnyebbé teheti
HWSW Informatikai Hírmagazin, 2007. december 04.
2. **Alison Cawsey:** Mesterséges intelligencia - alapismeretek
Panem, Budapest 2002.
3. **Daniel J. Berg:** Java felsőfokon
Kiskapu, Budapest 1999.
4. **Yiling Chen és társai:** Gaming Prediction Markets: Equilibrium Strategies with a Market Maker
Algorithmica, Vol. 58, No. 4, 930-969. o., 2010.
5. **M. Eigen – R. Winkler:** A játék
Gondolat, Budapest, 1981.
6. **Adrian Furnham:** 50 psychology ideas you really need to know
Quercus Publishing, London, 2008.
7. **Robert Gibbons:** A Primer in Game Theory
Princeton University Press, 1992.
8. **Harsányi János:** Games with incomplete information played by bayesian players part I-III
Management Science, Vol. 14 No. 3 November, 1967.
9. **Tatsuro Ichiishi, Akira Yamazaki:** Cooperative Extensions of the Bayesian Game
Series on Mathematical Economics and Game Theory,
World Scientific Publishing Company, 2006.
10. **Kóczy Á. László:** A Neumann-féle játékelmélet
Közgazdasági Szemle, LIII. évf., 2006. január, 31–45. o.
11. **George Leitmann:** Cooperative and non-cooperative many players differential games
CISM Monograph No. 190, Springer Verlag, Vienna, 1974.
12. **JAVA API 6**
<http://java.sun.com/javase/6/docs/api>
Letöltés időpontja: 2011. szeptember 3.
13. **Jason J. Manger:** A Java programozási nyelv
Panem, Budapest 1996.

- 14. Mérő László:** A blöff
Café Babel, 1993/1, 37-45.o.
- 15. Mérő László:** Mindenki másképp egyforma
Tericum, Budapest 2007.
- 16. Molnár Sándor, Szidarovszky Ferenc:** Játékelmélet
ComputerBooks, Budapest 2011.
- 17. John F. Nash:** Equilibrium Points in n-Person Games
Proceedings of the National Academy of Sciences, Vol. 36 No. 1, 1950, 48-49.o.
- 18. John F. Nash:** Noncooperative games
Annals of Mathematics, Vol. 54 No. 2, September, 1951, 286-295.o.
- 19. Christian Leszczynski, Wilfried Schuhmann és társai:** Pszichológiai lexikon
Helikon kiadó, 2007.
- 20. Neumann János:** The Computer and the Brain
Yale University Press, New Haven, 1958.
- 21. Neumann János:** Zur Theorie der Gesellschaftsspiel
Mathematische Annalen, No. 100, 295-320. o., 1928.
- 22. Neumann János, Oskar Morgenstern:** Theory of Games and Economic Behavior
Princeton University Press, Princeton, 3rd edition, 1953.
- 23. Piatnik:** Perudo játékszabály.
- 24. Stuart J. Russel, Peter Norvig:** Mesterséges intelligencia – modern megközelítésben
Panem, Budapest 2005.
- 25. David Sklansky:** The Theory of Poker
Creel Printing Co., Las Vegas, Nevada, 2007.
- 26. Szép Jenő – Forgó Ferenc:** Bevezetés a játékelméletbe
Közgazdasági és Jogi Könyvkiadó, Budapest 1974.
- 27. Végh László:** Algoritmus és játékelmélet
<http://www.cs.elte.hu/~veghal/orak/alg-jatek-v2.11.pdf>
Letöltés időpontja: 2011. szeptember 5.
- 28. World Series of Poker Winners**
<http://www.playwinningpoker.com/wsop/>
Letöltés időpontja: 2011. szeptember 5