



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**

Villamosmérnöki és Informatikai Kar

Méréstechnika és Információs Rendszerek Tanszék

Kooperáció és gépi tanulás labor (vimim223)

## **Labor jegyzőkönyv**

MÁTYÁS-BARTA CSONGOR (VYW0YR)

2014. OKTÓBER 16.

## Tartalomjegyzék

<b>1. A mérés bemutatása</b>	<b>2</b>
<b>2. Otthoni feladat 1</b>	<b>2</b>
2.1. Leírás . . . . .	2
2.2. Megoldás . . . . .	2
<b>3. Otthoni feladat 2</b>	<b>4</b>
3.1. Leírás . . . . .	4
3.2. Megoldás . . . . .	4
<b>4. Otthoni feladat 3</b>	<b>5</b>
4.1. Leírás . . . . .	5
4.2. Megoldás . . . . .	6
<b>5. Otthoni feladat 4</b>	<b>6</b>
5.1. Leírás . . . . .	6
5.2. Megoldás . . . . .	7
<b>6. Labor feladat 1</b>	<b>9</b>
6.1. Leírás . . . . .	9
6.2. Megoldás . . . . .	10
<b>7. Labor feladat 2</b>	<b>11</b>
7.1. Leírás . . . . .	11
7.2. Megoldás . . . . .	11
<b>8. Labor feladat 3</b>	<b>12</b>
8.1. Leírás . . . . .	12
8.2. Megoldás . . . . .	12
<b>9. Labor feladat 4</b>	<b>12</b>
9.1. Leírás . . . . .	12
9.2. Megoldás . . . . .	13
<b>10. Összefoglalás</b>	<b>13</b>

## 1. A mérés bemutatása

A mérésen játékelméleti problémák leírását, megoldását és szimulációját végeztük. Ehhez a GAMBIT játékleíró/megoldó szoftvert és JADEX es ágenseket használtunk.

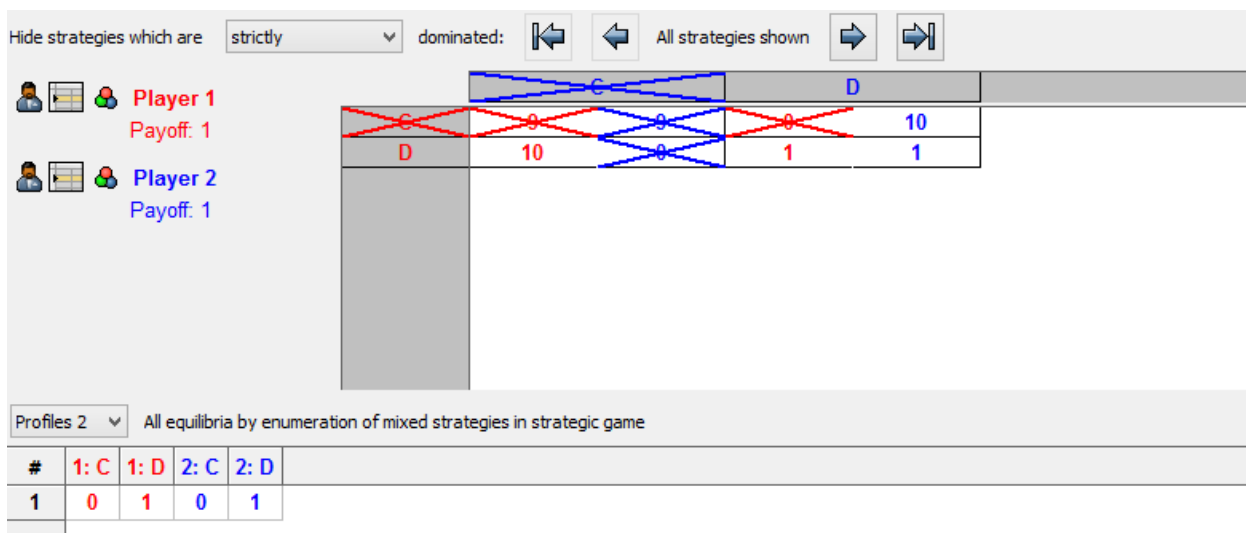
## 2. Otthoni feladat 1

### 2.1. Leírás

A laborsegédlet és az egyéb segédanyagok maradéktalan elolvasását és megértését követően tölts le a labor weblapjáról (<http://www.mit.bme.hu/oktatas/targyak/vimim223/feladat/4-Jatekelmelet>) a labor forráskódjait és a GAMBIT (Software Tools for Game Theory) alkalmazást. Bontsa ki a forráskódokat a msclab01 könyvtárba, illetve telepítse a GAMBIT-et. Ezt követően nyissa meg, értelmezze, és oldja meg GAMBIT-tel a games könyvtárban található... a. pd.nfg b. hd2v4.0c2.0.nfg c. hd2v2.0c4.0.nfg d. hd3v6.0c3.0.nfg e. totc10.nfg

### 2.2. Megoldás

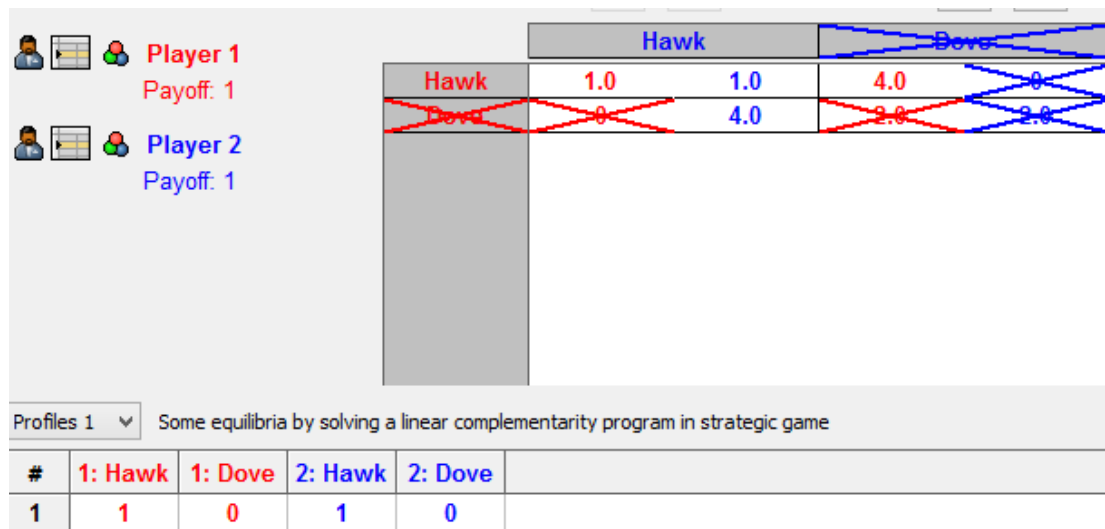
1. pd.nfg A fogolydilemma szimmetrikus leírását tartalmazza, mint látható a domináns stratégia a ha egy



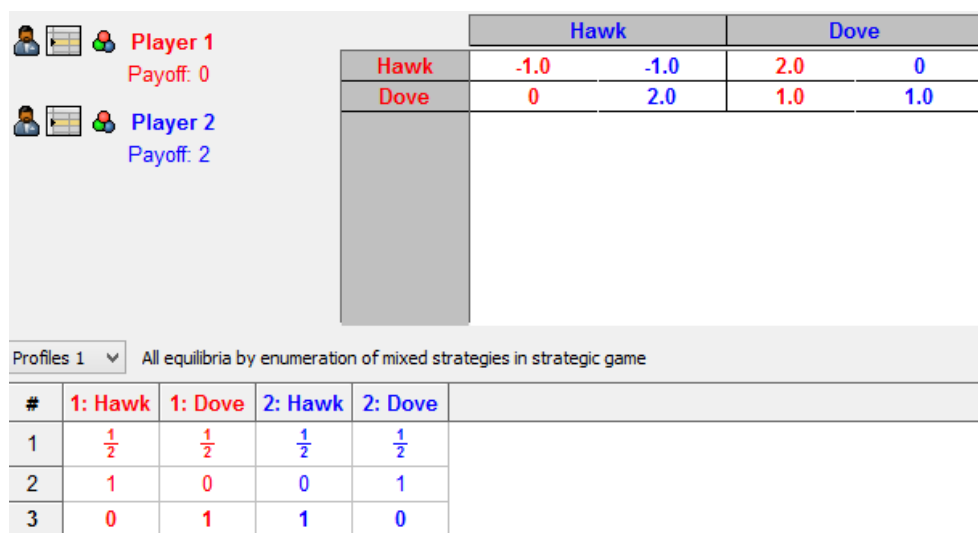
1. ábra. A pd.nfg megoldása

ágens vall, és a Nash egyensúly az amikor mindketten vallanak.

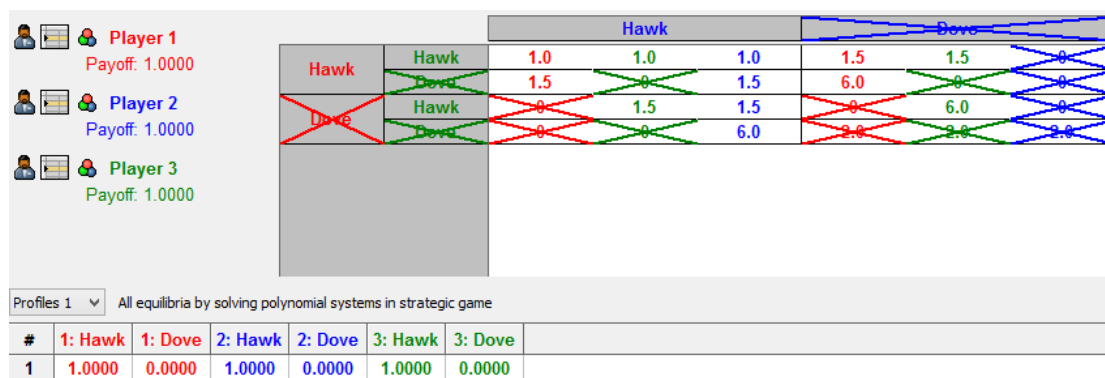
2. hd2v4.0c2.0.nfg A galamb-héja probléma egy olyan leírását tartalmazza, amelynek a domináns stratégiája a héja választás, ami egyben a Nash egyensúly is.
3. hd2v2.0c4.0.nfg A galamb-héja probléma egy olyan leírását tartalmazza, amelynek nincsen domináns stratégiája, de van 3 Nash egyensúlya.
4. hd3v6.0c3.0.nfg A galamb-héja probléma 3 játékos változata, amiben a domináns stratégia a héja választás, a Nash egyensúly pedig akkor áll be, ha mindhárom játékos a héját választja.
5. totc10.nfg no dominance A közös legelő problémát írja le 10 játékos esetén. Nem rendelkezik domináns stratégiával, sem Nash egyensúlyal.



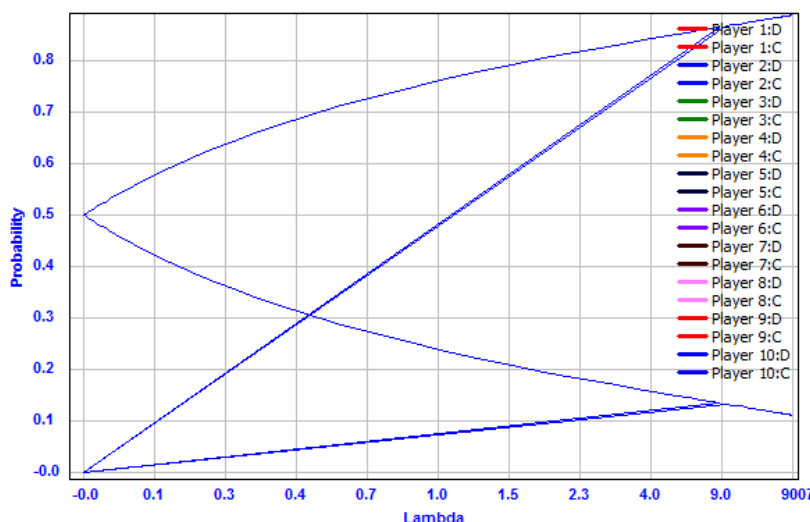
2. ábra. A hd2v4.0c2.0.nfg megoldása



3. ábra. A hd2v2.0c4.0.nfg megoldása



4. ábra. A hd3v6.0c3.0.nfg megoldása



5. ábra. A totc10.nfg megoldása

### 3. Otthoni feladat 2

#### 3.1. Leírás

A GAMBIT segítségével... a. ...építse fel alapvető játékok (Fogolydilemma, Gyáva nyúl, Nemek harca, Vezérürü, Érmepárosítás, Közlegelők tragédiája, 2-lapos póker, stb) extenzív és/vagy normál alakját majd pedig elemezze az elkészült játékokat (találja meg bennük a domináns stratégiákat, illetve a Nash-egyensúlyokat)! b. Az elkészült játékok normál alakját mentse el a games könyvtárba új fájlnevekkel, és egészítse ki őket egy megfelelő .nfo fájlal! Minta gyanánt használhatja a pd.nfo fájlt. Lényegében tehát egy azonos nevű, .nfo kiterjesztésű szövegfájl (meta-információt) kell létrehoznia minden egyes előbb létrehozott és mentett normál-formájú játék mellé. Ez egy tab-delimited szövegfájl legyen, aminek első eleme (0 vagy 1) adja meg, hogy a kapcsolódó játék szimmetrikus-e, második eleme (0 vagy 1) azt mondja meg, hogy van-e ebben a játékban minden játékosnak ún. kooperatív tiszta stratégiája, és amennyiben ez igaz, úgy harmadik eleme az egyes játékosok kooperatív tiszta stratégia-azonosítóinak (0, 1, 2, ...) szóközzel elválasztott listája, ahol a játékosok sorrendje a kapcsolódó .nfg fájlban megadott sorrendnek feleljen meg.

#### 3.2. Megoldás

A segédletben leírtak szerint felépítettem a következő problémák normál alakját, és megoldottam őket.

- Fogolydilemma Amint láthat a fogolydilemma ezen formájának a domináns stratégiája a vall, és a Nash egyensúlya amikor mindkét fogoly vall. A játék szimmetrikus, ennek értelmében az nfo fájl tartalma: 1 1 1 1.
- Gyáva nyúl A gyáva nyúl játéknak nincs domináns stratégiája, de van két Nash egyensúlya. A játék szimmetrikus, tehát az nfo fájl tartalma: 1 1 1 1
- Nemek harca A nemek harcának ezen leírásában, nincs domináns stratégia, van 2 Nash egyensúly és a játék szimmetrikus. Az nfo fájl tehát: 1 1 0 0
- Vezérürü A vezérürü játéknak nincs domináns stratégiát, van 2 Nash egyensúlya és szimmetrikus. Az nfo fájl tehát: 1 1 1 0
- Érmepárosítás Az érmepárosítás játék asszimmetrikus, nincs domináns stratégia és nincs Nash egyensúlya.

Player 1  
Payoff: -2

Player 2  
Payoff: 2

	Vall	Hallgat
Vall	-2, 2	3, 3
Hallgat	3, 3	2, 2

Profiles 1 ▾ All equilibria by enumeration of mixed strategies in strategic game

#	1: Vall	1: Hallgat	2: Vall	2: Hallgat
1	1	0	1	0

6. ábra. A fogolydilemma megoldása

Player 1  
Payoff: -1

Player 2  
Payoff: 1

	Vakmero	Gyava
Vakmero	-10, -10	1, -1
Gyava	-1, 1	0, 0

Profiles 1 ▾ All equilibria by enumeration of mixed strategies in strategic game

#	1: Vakmero	1: Gyava	2: Vakmero	2: Gyava
1	$\frac{1}{10}$	$\frac{9}{10}$	$\frac{1}{10}$	$\frac{9}{10}$
2	1	0	0	1
3	0	1	1	0

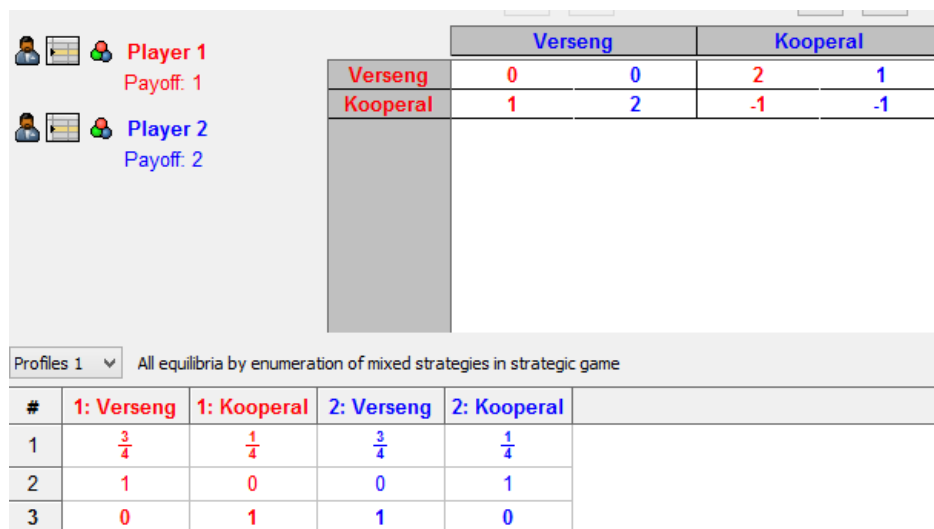
7. ábra. A gyáva nyúl megoldása

Az nfo fájl tehát: 0 0

## 4. Otthoni feladat 3

### 4.1. Leírás

Futtassa az `msclab01.gametheory-lab.Game` osztályt az elkészült játékokkal (az osztály egyetlen argumentuma a normál-formájú játék fájlneve legyen kiterjesztés nélkül), és részletesen értelmezze a console kimenetet a 2/a feladatban látottak viszonylatában! Miket ír ki ez a program a konzolra? Minden adat helyes, amit kiír? Megjegyzés (FONTOS!!!): amennyiben egy eredetileg extenzív alakban létrehozott játékot normál formájú játékként exportálunk ki a GAMBIT-ből, de a Game osztály (vagy alább a GameAgent) nem hajlandó ezt rendesen beolvasni, úgy nyissuk meg külön ezt a kiexportált normál formájú játékot (.nfg fájl) a GAMBIT-tel, és exportáljuk ki újra! Ennek hatására a kiexportált játék meg kell, hogy javuljon.

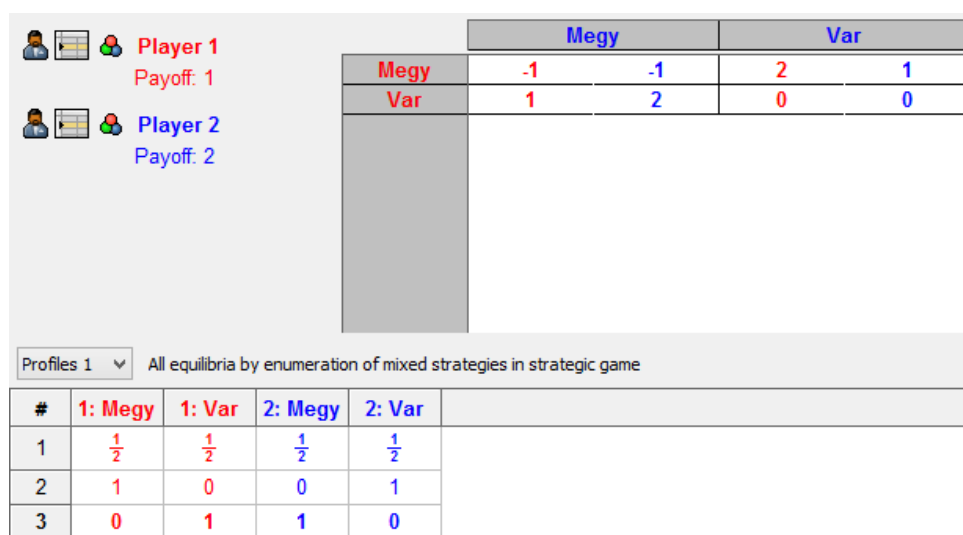


	Verseng		Kooperál	
Verseng	0	0	2	1
Kooperál	1	2	-1	-1

#	1: Verseng	1: Kooperál	2: Verseng	2: Kooperál
1	$\frac{3}{4}$	$\frac{1}{4}$	$\frac{3}{4}$	$\frac{1}{4}$
2	1	0	0	1
3	0	1	1	0

8. ábra. A nemek harca megoldása



	Megy		Var	
Megy	-1	-1	2	1
Var	1	2	0	0

#	1: Megy	1: Var	2: Megy	2: Var
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
2	1	0	0	1
3	0	1	1	0

9. ábra. A vezérürü megoldása

## 4.2. Megoldás

A Game osztály beolvassa a problémát és meghatározza a játék Nash egyensúlyát.

## 5. Otthoni feladat 4

### 5.1. Leírás

Amennyiben megbizonyosodtunk arról, hogy az msclab01.gametheory-lab.Game osztály console kimenete helyes az általunk 2/a feladatban létrehozott összes játék esetén (vagy korrigáltuk az említett játékokat, és így lett helyes az output) Eclipse alól indítsunk el egy JADE platform-ot, majd pedig végezzünk egy-egy pár tucat körös tesztfuttatást mindegyik játék esetén! Magyarán minden játék kapcsán indítsuk el a szükséges számú felhasználói UserAgent és/vagy gépi Player ágenszt (utóbbiaknak mindig legyen bekapcsolva a GUI-ja, és programjuk

		Fej		Iras	
Fej	Iras	1	-1	-1	1
		-1	1	1	-1

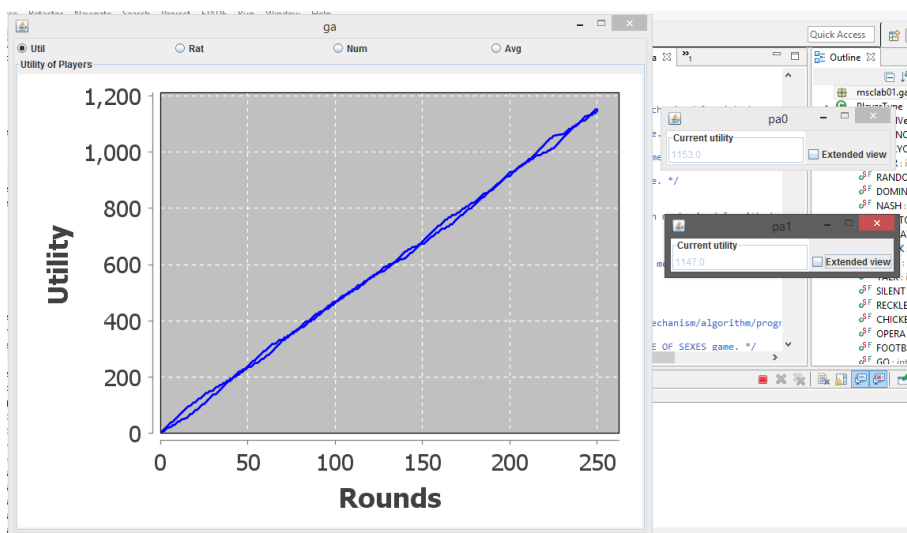
#	1: Fej	1: Iras	2: Fej	2: Iras
1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$

10. ábra. Az érmepárosítás megoldása

legyen a TFT), majd utána egy adott játékot host- oló GameAgent ágenst is, és regisztráljuk/dokumentáljuk a történeteket! Mit látunk? Helyes a működés? ... az egyes futtatásokhoz a segédletben látottaknak megfelelően Eclipse-es launch configuration-őket hozunk létre (és ezeket is dokumentáljuk)!

## 5.2. Megoldás

A jadex ágensek implementációja miatt átírtam a játékleírásokat egy olyan formára ami csak szigorúan pozitív jutalmakat tartalmaz. Erre azért volt szükség, mert az ágensek leállnak ha  $\leq 0$  a hasznuk.

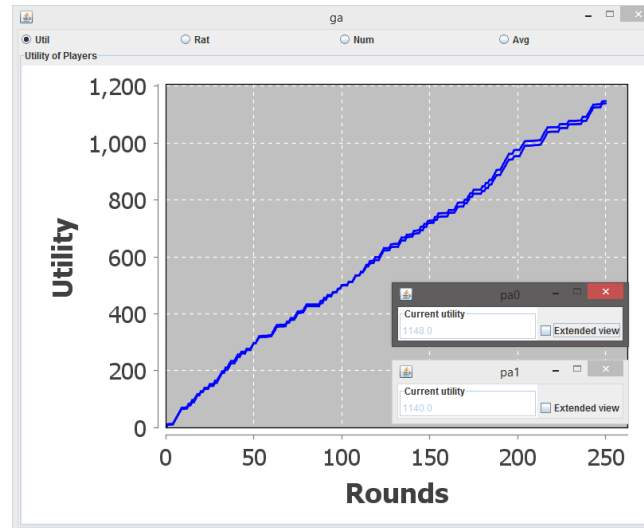


11. ábra. A fogolydilemma ágensekkel



Listing 1. Fogolydilemma run config

```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(fogolydilemma 2 2)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"fogolydilemma\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"fogolydilemma\\\\" gui=true myType=7")
```



12. ábra. A gyáva nyúl ágensekkel

Listing 2. Gyáva nyúl run config

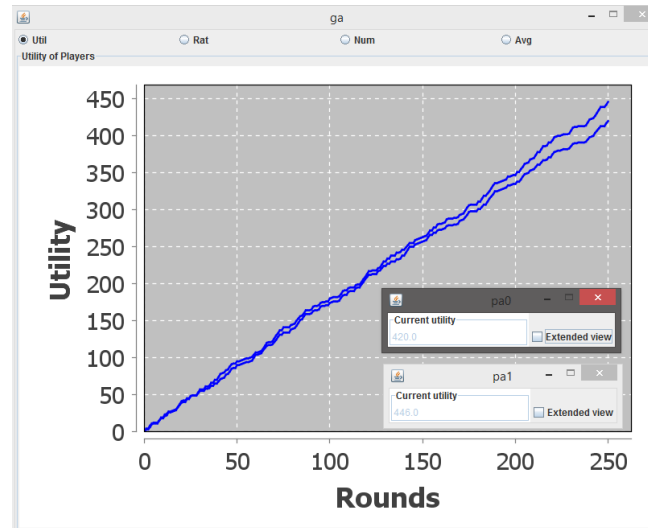
```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(gyavanyul 2 2)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"gyavanyul\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"gyavanyul\\\\" gui=true myType=7")
```

Listing 3. Nemek harca run config

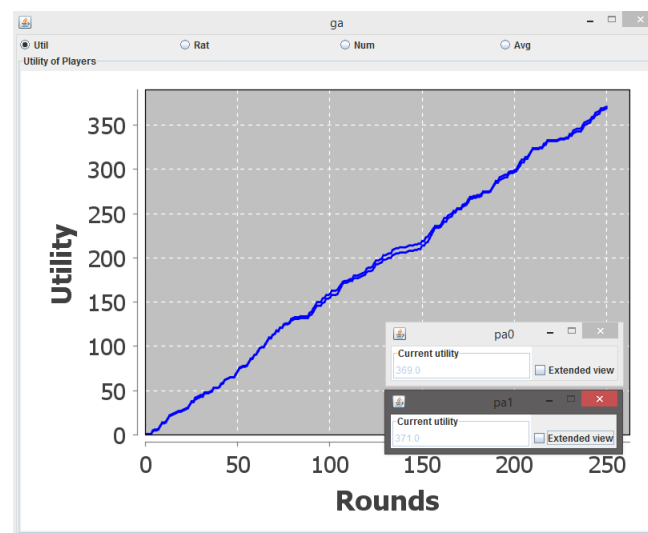
```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(nemekharca 2 2)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"nemekharca\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"nemekharca\\\\" gui=true myType=7")
```

Listing 4. Vezérürü run config

```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(vezeruru 2 2)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"vezeruru\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"vezeruru\\\\" gui=true myType=7")
```



13. ábra. A nemek harca ágensekkel



14. ábra. A vezérürü ágensekkel

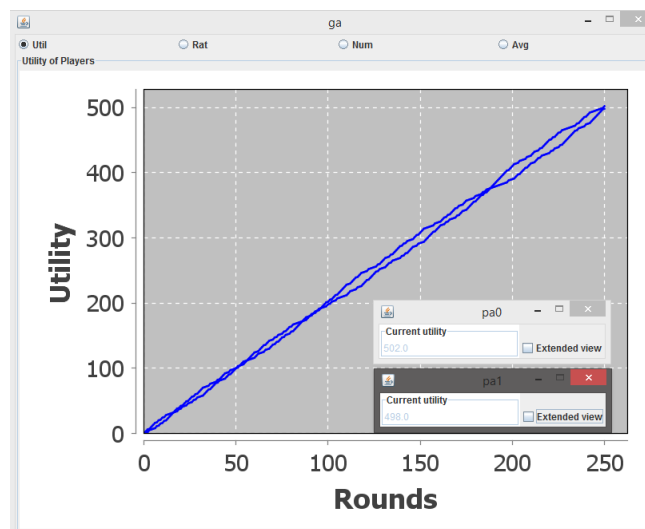
Listing 5. érme párosítás run config

```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(ermeparositas 2 2)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"ermeparositas\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"ermeparositas\\\\" gui=true myType=7")
```

## 6. Labor feladat 1

### 6.1. Leírás

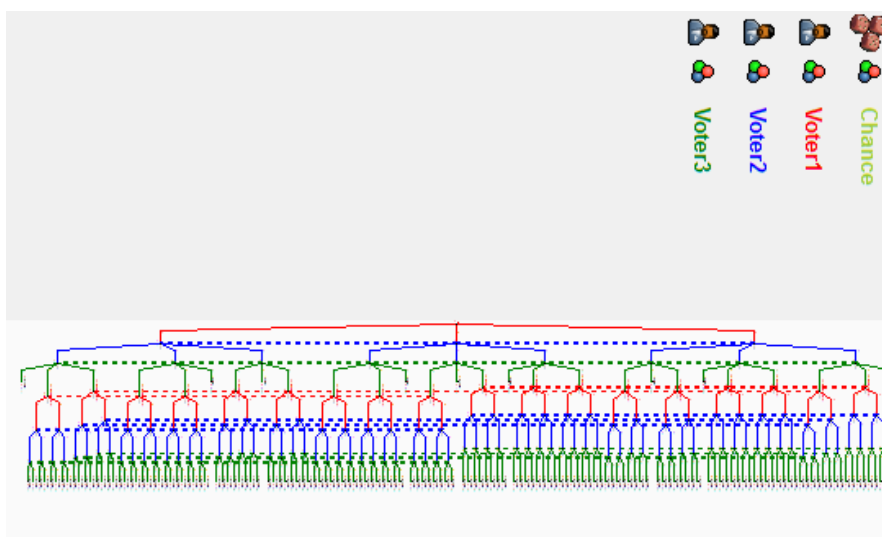
Alakítsa ki GAMBIT-ben egy legalább 3 szereplős, legfeljebb néhány lépésből álló szavazás (pl. több forduló (runoff) többségi szavazás) extenzív alakját, majd konvertálja ezt át normál alakba, elemezze, és mentse a games könyvtárba! Ellenőrizze az elkészült játékot az msclab01.gametheory\_lab.Game osztály segítségével is!



15. ábra. Az érmepárosítás ágensekkel

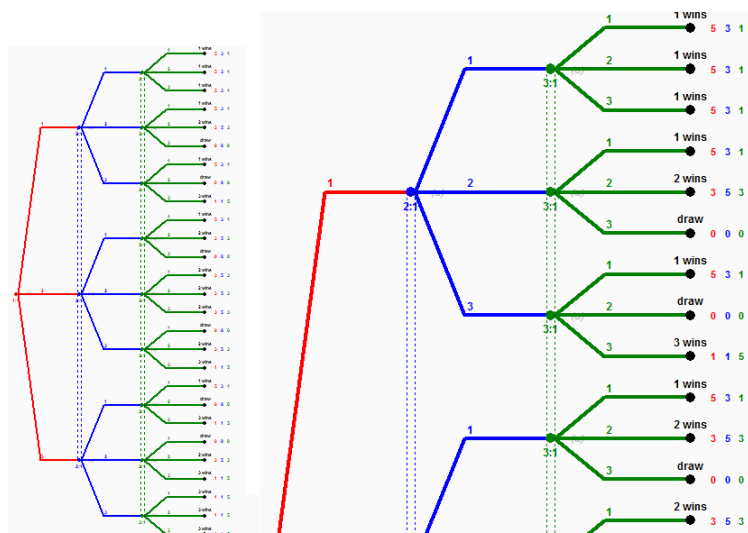
## 6.2. Megoldás

Az első lépésben egy két fordulós, többségi szavazást definiáltam a GAMBIT-ben, 3 játékoskal és 3 jelöltel. Az ötlet az volt hogy csak akkor megy végbe a második forduló ha két játékos egy közös jelöltre szavazott, a harmadik pedig egy másikra. Ekkor az a jelölt akire senki sem szavazott kiesik és a játékosok preferencia listájuknak megfelelően újra szavaznak a bentmaradt jelöltekre. Ekkor akinek a legtöbb szavazata lett az a győztes. Az ágensek hasznát aszerint határoztam meg, hogy a győztes hol állt az adott ágens preferencia listáján. Ha minden játékos másra szavazott vagy ugyanarra, akkor nem volt újabb forduló. Az extenzív alak eléggé komplex lett, az egyszerű probléma ellenére is, mint az ábra mutatja. A második feladatot nem sikerült



16. ábra. A többfordulós szavazás extenzív alakja

megoldani, mivel az ágens kifutott a memóriából a normál alak óriási mérete miatt. Ezért egy egyszerűbb szavazó sémát hoztam létre, ami csak egy fordulóból állt, 3 szavazó és 3 jelöltel. A leírt szavazó játéknak nincs domináns stratégiája és jópár Nash egyensúlya van.



17. ábra. Az egyfordulós szavazás extenzív alakja

		1			2			
1	1	5	1	3	5	1	3	5
	2	5	1	3	3	3	5	0
	3	5	1	3	0	0	0	1
2	1	5	1	3	3	3	5	0
	2	3	3	5	3	3	5	3
	3	0	0	0	3	3	5	1
3	1	5	1	3	0	0	0	1
	2	0	0	0	3	3	5	1
	3	1	5	1	1	5	1	1

18. ábra. Az egyfordulós szavazás normál alakja

## 7. Labor feladat 2

### 7.1. Leírás

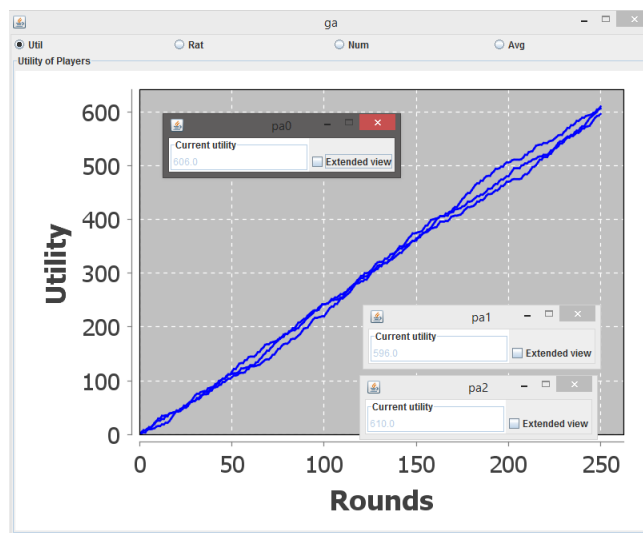
Az 1-es feladatban kapott normál alakot adja meg a JADE-es játékvezető (msclab01.lab04.GameAgent), és néhány gépi játékos ágens (msclab01.lab04.PlayerAgent) számára, majd végezzen velük kísérleteket! Milyen lejátszások adódnak a játékosok különböző beállításai (pl. stratégia-választó programja) mellett? Elérhető-e, illetve stabil-e a Nash-egyensúly? Mi a legjobb stratégia/program (ha pl. UserAgent ágenssel játszunk adott gépi ágensek, vagy akár más hallgatók által vezérelt UserAgent ágensek ellen)?

### 7.2. Megoldás

Az előbb definiált játékot futtatva a JADEx platformon a következő eredményt kaptam.

Listing 6. Szavazás run config

```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(voting_game 3 3)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"voting_game\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"voting_game\\\\" gui=true myType=7")
pa2:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"voting_game\\\\" gui=true myType=7")
```



19. ábra. Ágens szavazó játékeredmény

## 8. Labor feladat 3

### 8.1. Leírás

Alakítsa ki GAMBIT-ben egy legalább 3 szereplős, legfeljebb néhány lépésből álló egyszerűbb aukció (pl. angol, holland, vagy japán aukció) extenzív alakját, majd konvertálja ezt át normál alakba, elemezze, és mentse a games könyvtárba! Ellenőrizze az elkészült játékot az msclab01.gametheory-lab.Game osztály segítségével is!

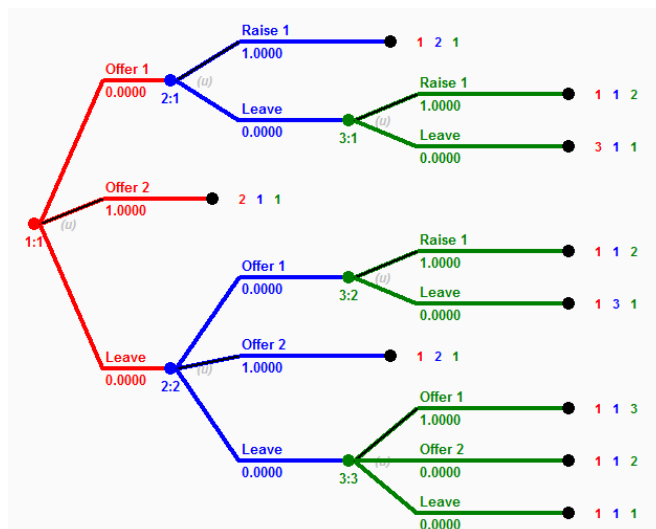
### 8.2. Megoldás

A létrehozott aukció játéknak 3 játékos lehet, egy termékre licitálnak és 2 pénzzel rendelkeznek. A termék valós értéke 4, tehát ha egy játékos megveszi 1 ért, akkor a haszna 3. Komplexebb aukció leírását sajnos most sem sikerült a JADEX platformon futtatni.

## 9. Labor feladat 4

### 9.1. Leírás

A 3-as feladatban kapott normál alakot adja meg egy JADE-es GameAgent játékevezető és néhány Player gépi játékos ágens számára, majd végezzen velük kísérleteket! Milyen lejátszások adódnak a játékosok különböző



20. ábra. Az aukció játék extenzív alakja

beállításai (pl. stratégia-választó programja) mellett? Elérhető-e, illetve stabil-e a Nash-egyensúly? Mi a legjobb stratégia/program (ha pl. UserAgent ágenssel játszunk adott gépi ágensek, vagy akár más hallgatók által vezérelt UserAgent ágensek ellen)?

## 9.2. Megoldás

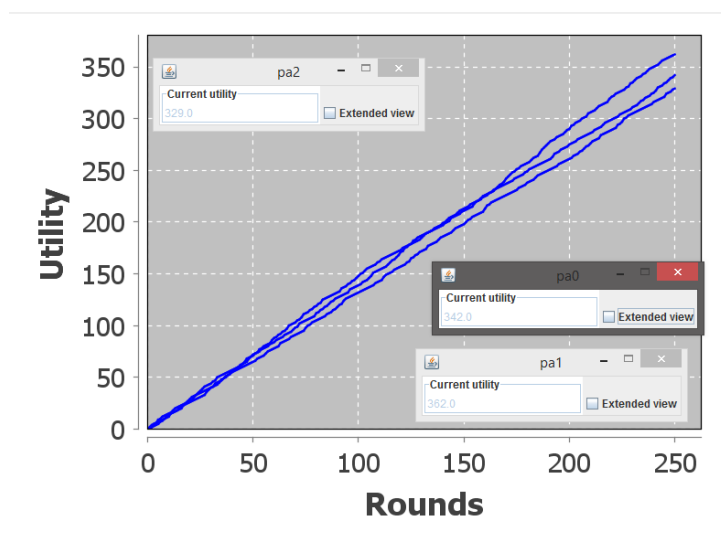
A létrehozott aukció játékot JADEx ágensekkel lejátszva a következő eredményt kaptam. A stratégiák változtatása nem járt különösebb eredménnyel, hisz a játék csak egy lépéses.

Listing 7. Aukció run config

```
-container -port 1099 -host localhost
ga:msclab01.gametheory_lab.GameAgent.GameAgent(auction_game 3 3)
pa0:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"auction_game\\\\" gui=true myType=7")
pa1:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"auction_game\\\\" gui=true myType=7")
pa2:jadex.adapter.jade.JadeAgentAdapter(msclab01.gametheory_lab.PlayerAgent.Player
"default gid=\\\\"auction_game\\\\" gui=true myType=7")
```

## 10. Összefoglalás

A labor során megismerkedtem a játékok GAMBIt-al való leírásával, megoldásával. Ez követően szimulációkat végeztem JADEx ágensekkel, akik lejátszották az adott játékokat. Tapasztaltam hogy a játékok világában a leírás exponenciálisan nő a játékosok, lehetséges lépések számával. Kis lépésszám mellett a GAMBIt alkalmas volt a játék modellezésére, de nagy lépésszámnál a Nash egyensúly számolás és domináns stratégia keresés már sikertelen volt.



21. ábra. Ágens aukció játékeredmény