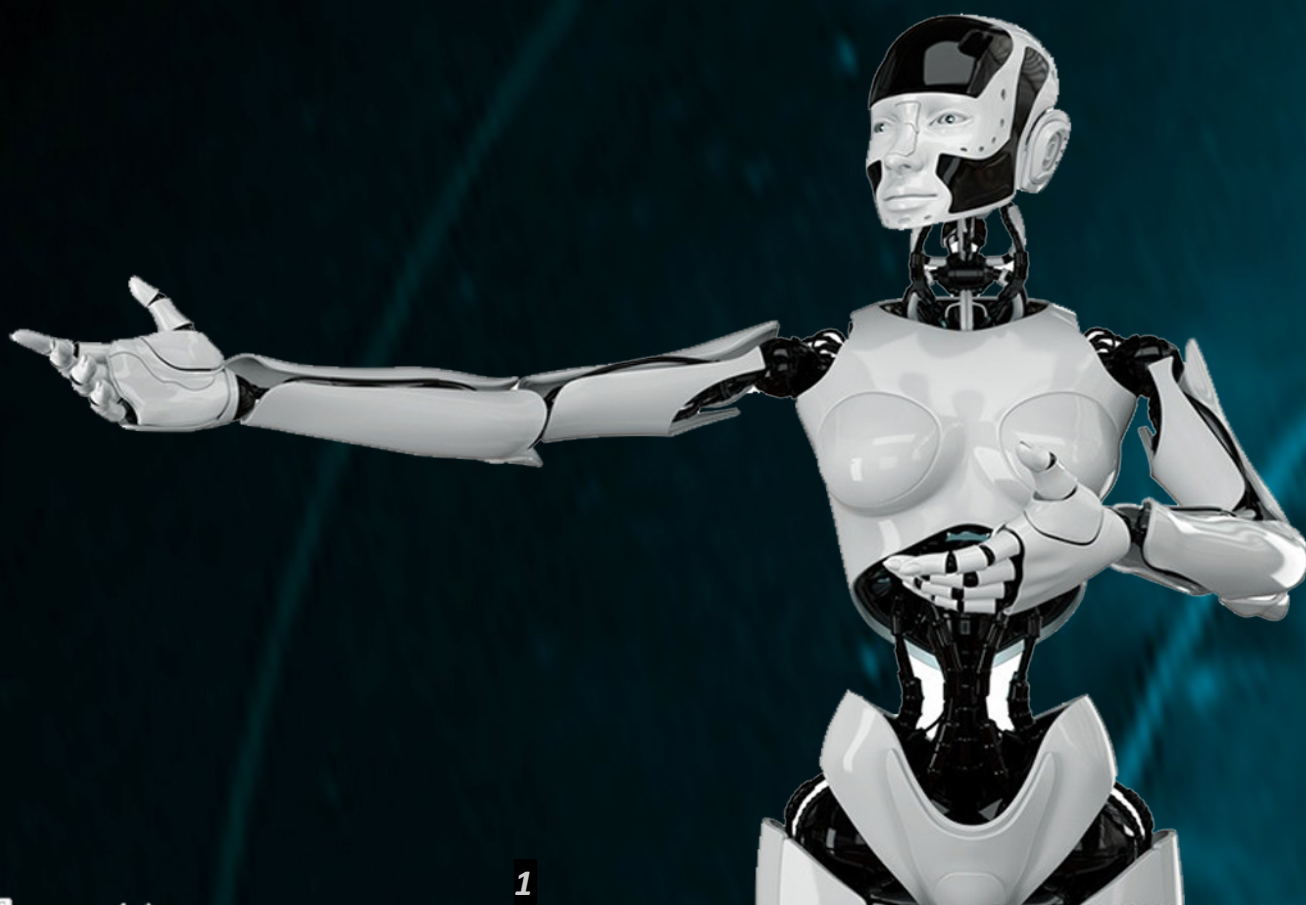




ΠΛΗ 517  
ΠΟΛΥΠΡΑΚΤΟΡΙΚΑ ΣΥΣΤΗΜΑΤΑ

# AgentBD

Φλέγγας Γεώργιος 2014030161  
Χατζηπέτρος Αλέξανδρος 2013030161



## Περιεχόμενα:

1. Εισαγωγή
2. Άντληση ιδεών
3. Στρατηγική
4. Πειραματικά αποτελέσματα
5. Αποτελέσματα διαγωνισμού
6. Αλλαγές μετά τον διαγωνισμό
7. Πειραματικά αποτελέσματα 2.0
8. Brainstorming
9. Βιβλιογραφία

## 1.Εισαγωγή:

Στα πλαίσια του μαθήματος Πολυπρακτορικά Συστήματα [ΠΛΗ517] και του διαγωνισμού ANAC2018 υλοποιήθηκε ο πράκτορας agentBD. Ο πράκτορας αναπτύχθηκε στο περιβάλλον GENIUS 8.0.5 με σκοπό τη συμμετοχή τους στην κατηγορία Repeated Multilateral Negotiation League .

Όσον αφορά την υλοποίησή του ο πράκτορας βασιστήκαμε σε διάφορες προϋπάρχουσες ιδέες, καθώς και σε κάποιες δικές μας. Σκοπός μας ήταν η δημιουργία ενός πράκτορα ο οποίος :

- Επιτυγχάνει το μεγαλύτερο δυνατό utility
- Είναι hardheaded αρχικά, αλλά στο τέλος ρίχνει τις απαιτήσεις του
- Έχει σκοπό το fairness και την αύξηση του social welfare
- Κάνει βέλτιστη μοντελοποίηση του αντιπάλου
- Καταλήγει πάντα σε συμφωνία

Σε αυτή την αναφορά θα παρουσιάσουμε την θεωρητική προσέγγιση της στρατηγικής του πράκτορα, την υλοποίησή του και τέλος κάποια από τα πειράματα που έγιναν πριν και μετά τον εσωτερικό διαγωνισμό .

## 2.Άντληση ιδεών:

Για την ανάπτυξη της στρατηγικής του πράκτορα μας, λάβαμε υπόψιν όλα τα papers που υπάρχουν στη σελίδα του μαθήματος, καθώς και κάποια ακόμα που βρήκαμε στο διαδίκτυο. Αναλυτικότερα:

- Για τον αλγόριθμο υπολογισμού του threshold, βασιστήκαμε στο πράκτορας AgentK [1], αφού η στρατηγική του είχε καλά αποτελέσματα και ως ιδέα παρουσίαζε αρκετό ενδιαφέρον. Σκοπός είναι να προκύπτει ένα λογικό threshold για τα δικά μας bids, το οποίο θα είναι Hardheaded και προς το τέλος θα φθίνει γρηγορότερα.
- Χάρη στο paper [2] καταλήξαμε το είδος μοντέλου μάθησης αντιπάλου που χρησιμοποιήσαμε. Σε αυτό φαίνεται ότι το Frequency model δίνει καλύτερα αποτελέσματα σε σχέση με ένα Bayesian model, κάτι το οποίο φαίνεται και από την υλοποίηση του Hardheaded agent. Έγινε λοιπόν υλοποίηση του Hardheaded Frequency model, η οποία πρόσφερε αρκετά καλές εκτιμήσεις της evaluation function του αντιπάλου.
- Τέλος υλοποιήθηκε ο μηχανισμός των popular bids, ο οποίος εξασφαλίζει την επιτυχία συμφωνίας προς το τέλος της διαπραγμάτευσης, για το οποίο δεν αρκούσε μόνο το δικό μας accept.

### 3.Στρατηγική:

Με βάση τα προηγούμενα καταλήξαμε στην ακόλουθη βασική ιδέα με βάση την οποία θα ενεργεί ο πράκτορας μας:

```
if (time < 0.2 * tmax)
    send max_util_bid
else {
    if (acceptable_offer) {
        Accept
    }
    Offer
}
```

#### Bidding Strategy:

Όπως αναφέραμε η makeBid() είναι εμπνευσμένη από τον agentK και έχει την ακόλουθη λειτουργία:

Υπολογίζουμε το κατώτερο threshold της προσφοράς μας για κάθε αντίπαλο, χρησιμοποιώντας τη μέση τιμή και την απόκλιση των bids του καθενός και κρατάμε το χαμηλότερο threshold που προκύπτει.

Ο τρόπος υπολογισμού είναι ο εξής:

$$\begin{aligned}\text{threshold} &= 1 - (1 - e_{\max}) \times (\text{time} \times df)^r \\ e_{\max} &= \text{average} + (1 - \text{average}) \times \text{var}\end{aligned}$$

όπου:

- **e<sub>max</sub>** είναι το εκτιμώμενο μέγιστο utility που θα λάβουμε στο μέλλον
- **df** είναι ο discount factor του domain
- **r** είναι ο ρυθμός με τον οποίο φθίνει η συνάρτηση όσο περνάει ο χρόνος. Το r αλλάζει ανάλογα με το time interval στο οποίο βρισκόμαστε.

Αναλυτικότερα είναι :

<b>r =</b>	<b>3 ,</b>	<b>0,2 ≤ time &lt; 0.85</b>
	<b>2 ,</b>	<b>0.85 ≤ time &lt; 0.95</b>
	<b>1.5,</b>	<b>0.95 ≤ time ≤ 1</b>

το οποίο σημαίνει ότι φθίνει πολύ αργά μέχρι το 0.85, μετά επιταχύνει και μετά το 0.95 φθίνει πολύ πολύ γρήγορα.

Αφού υπολογιστεί το threshold, γίνεται η επιλογή του bid μας. Ψάχνουμε bids με utility μεταξύ threshold και  $1,1 \times \text{threshold}$ . Από τα bids αυτά, διαλέγουμε εκείνο με το μεγαλύτερο εκτιμώμενο utility του αντιπάλου, το οποίο υπολογίζουμε χρησιμοποιώντας ένα frequency model του αντιπάλου.



### Opponent Model:

Υλοποιήθηκε ένα Hardheaded Frequency Model, όπου έχουμε έναν πίνακα με βάρη για το κάθε value κι έναν με βάρη για το κάθε issue. Πιο συγκεκριμένα, μετράμε τη συχνότητα με την οποία αλλάζουν οι τιμές στα values και στα issues. Για κάθε bid που λαμβάνουμε, ελέγχουμε ποια issues/values δεν άλλαξαν τιμή σε σχέση με το προηγούμενο bid και αυξάνουμε τη συχνότητά τους (κατά 1 για τα values, κατά 0.2 για τα issues).

Με βάση αυτό το μοντέλο, μπορούμε να υπολογίσουμε το utility ενός bid για τον αντίπαλο, παίρνοντας τα βάρη των values που εμφανίζονται στο bid και πολλαπλασιάζοντάς τα με τα βάρη των αντίστοιχων issues:

$$\text{oppUtility} = \sum \text{weightI} \times \text{weightV}$$

Ο μηχανισμός popular bids λειτουργεί ως εξής: Κάθε φορά που δεχόμαστε ένα bid, ελέγχουμε από πόσα parties γίνεται accept. Αυτό χρησιμοποιείται στο μοντέλο του αντιπάλου, ανανεώνοντάς το έτσι ώστε να παίζουν μεγαλύτερο ρόλο τα «δημοφιλή» bids, κι επίσης μπορούμε να χρησιμοποιήσουμε το πλέον δημοφιλές bid σαν πρόταση προς το τέλος της διαπραγμάτευσης για να εξασφαλίσουμε τη συμφωνία.

### Acceptance Strategy:

Η στρατηγική αποδοχής χωρίζεται σε 2 κομμάτια με βάση την χρονική περίοδο στην οποία βρίσκεται η διαπραγμάτευση μας.

Κατά το αρχικό 20% του χρόνου, ο agentBD θα προσφέρει το max utility bid και θα εμφανίζεται αδιάλλακτος, ενώ παράλληλα θα συλλέγει πληροφορίες και θα χτίζει το opponent model, μια στρατηγική που εμπνευσμένη από πράκτορες όπως TheNegotiator, BRAMAgent [3].

Στην συνέχεια θα πράττει ως εξής:

```
Bid ourBid = makeBid()
if (acceptBid(ourBid, lastReceivedBid)) {
    return new Accept(getPartyId(), this.lastReceivedBid);
}
return new Offer(getPartyID(), ourBid);
```

Με την χρήση της acceptBid ελέγχει εάν το bid του αντιπάλου είναι καλύτερο από το χαμηλότερο bid που έχουμε προσφέρει και άρα θεωρεί την πρόταση αποδεκτή, ενώ σε αντίθετη περίπτωση κάνει counteroffer.

#### 4. Πειραματικά αποτελέσματα:

Για να καταλήξουμε στο τελικό πράκτορα μας πειραματιστήκαμε με αρκετές διαφορετικές στρατηγικές. Στο ακόλουθο κομμάτι θα παρουσιαστούν κάποια από τα θετικά αποτελέσματα, καθώς και κάποιες περιπτώσεις που μας προβλημάτισαν.

Αρχικά θα παραθέσουμε τα αποτελέσματα κάποιων τουρνουά που τρέξαμε και στην συνέχεια μεμονωμένα τεστ .

##### Tournaments:

Domain: Party	
<i>Agent</i>	<i>Average Utility</i>
Random	0.499
Boulware	0.625
agentBD	0.633

Domain: JapanTrip	
<i>Agent</i>	<i>Average Utility</i>
Random	0.597
Boulware	0.614
agentBD	0.603

Domain: JapanTrip	
<i>Agent</i>	<i>Average Utility</i>
Boulware	0.609
Caduceus	0.499
agentBD	0.453

Domain: Party	
<i>Agent</i>	<i>Average Utility</i>
Boulware	0.660
Caduceus	0.705
Terra	0.734
Atlas3	0.568
agentBD	0.523

Domain: Japan	
<i>Agent</i>	<i>Average Utility</i>
Boulware	0.373
Caduceus	0.399
Terra	0.338
Atlas3	0.151
agentBD	0.251

Παρατηρώντας τα παραπάνω αποτελέσματα, ο πράκτορας μας φαίνεται να επιτυγχάνει καλά αποτελέσματα όταν το Domain είναι πιο απλό και έχει λιγότερους και σχετικά πιο αδύναμους αντιπάλους. Όταν όμως αυξηθεί ο αριθμός των αντιπάλων και ταυτόχρονα γίνει επιλογή πιο δυνατών πρακτόρων, η αποτελεσματικότητα του πέφτει κατακόρυφα.

## Negotiations:

Στο Party Domain τρέξαμε ένα negotiation μεταξύ Boulware, Terra, Atlas3, agentBD και είχαμε το ακόλουθο αποτέλεσμα :

Agreement?: Yes

Discounted?: No

4

Min. utility: 0.44818

Max. utility: 1.00000

Distance to pareto: 0.00000

Distance to Nash: 0.47954

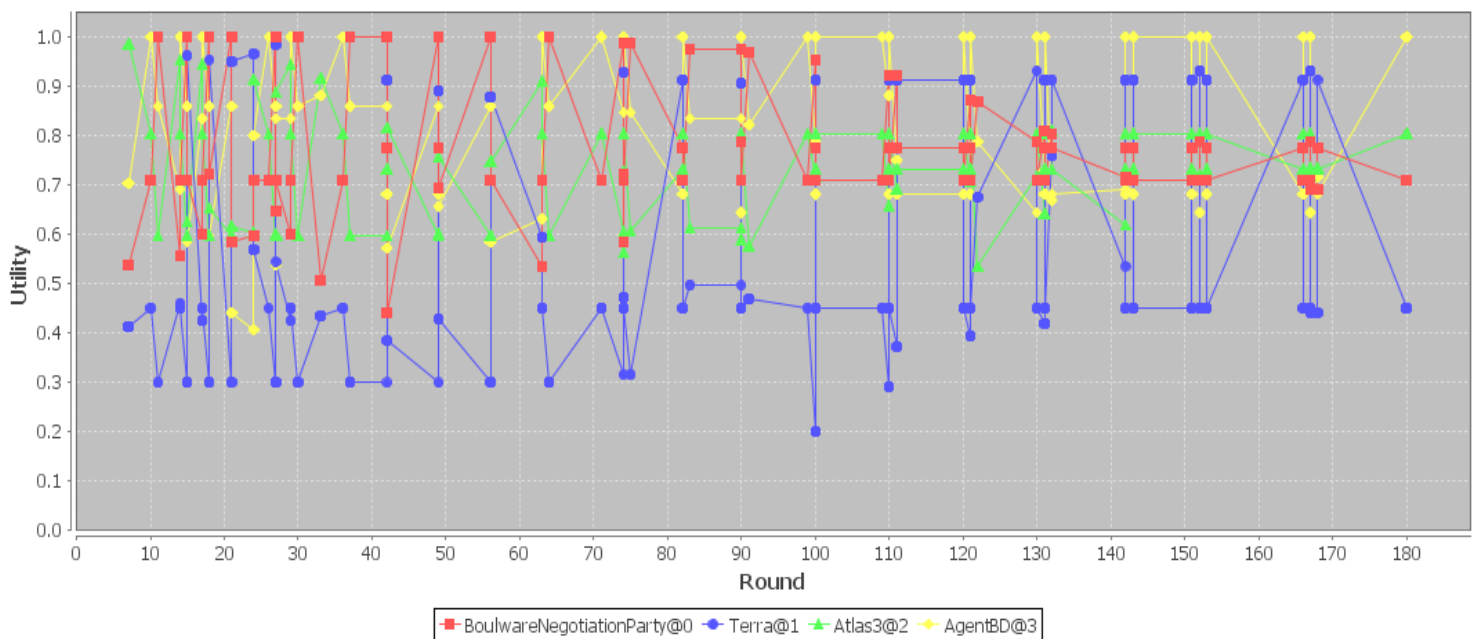
Social welfare: 2.95736

Agent utility: 0.70917 (BoulwareNegotiationParty@0)

Agent utility: 0.44818 (Terra@1)

Agent utility: 0.80002 (Atlas3@2)

Agent utility: 1.00000 (AgentBD@3)



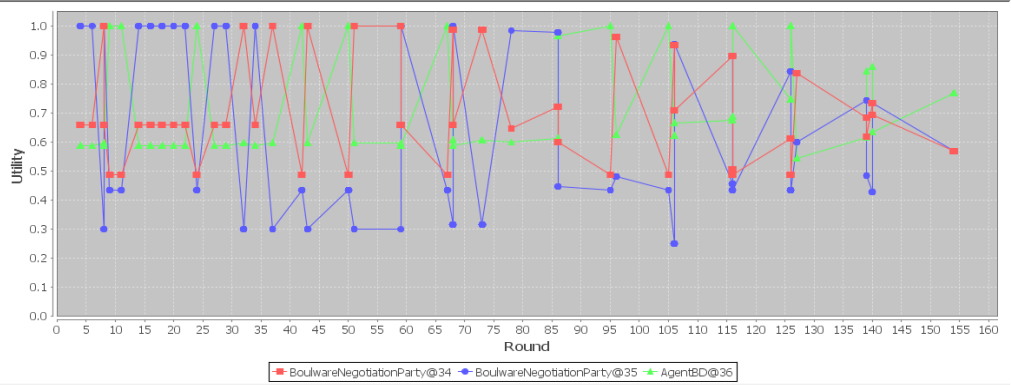
Παρατηρούμε ότι σε μεμονωμένες διαπραγματεύσεις ο agentBD έχει μια στατιστική η οποία καταλήγει σε βέλτιστα αποτελέσματα και καταφέρνει να πετύχει utility 1. Αυτό δεν ήταν ένα απλό τυχαίο αποτέλεσμα, καθώς επαναλήφθηκε σε αρκετά διαφορετικά test που κάναμε όπως φαίνεται παρακάτω:



## Session Progress

## Negotiation log

Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Accept bid  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Handmad  
 Round 148  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Offer bid:  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 149  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Offer bid:  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 150  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Offer bid:  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 151  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Offer bid:  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 152  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Accept bid  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 153  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Offer bid:  
 Turn 3:AgentBD@36 (Offer bid:Bid[Food: Chips a  
 Round 154  
 Turn 1:BoulwareNegotiationParty@34 (Offer bid:  
 Turn 2:BoulwareNegotiationParty@35 (Accept bid  
 Turn 3:AgentBD@36 (Accept bid:Bid[Food: Chips  
 Found an agreement: Bid[Food: Chips and Nuts, Dri  
 Finished negotiation session in 1.44904357 s.  
 Time (s): 1.44904357  
 Rounds: 154  
 Agreement?: Yes  
 Discounted?: No  
 3  
 Min. utility: 0.56833  
 Max. utility: 0.76702  
 Distance to pareto: 0.17292  
 Distance to Nash: 0.42301  
 Social welfare: 1.90397  
 Agent utility: 0.56833 (BoulwareNegotiat  
 Agent utility: 0.56862 (BoulwareNegotiat  
 Agent utility: 0.76702 (AgentBD@36)

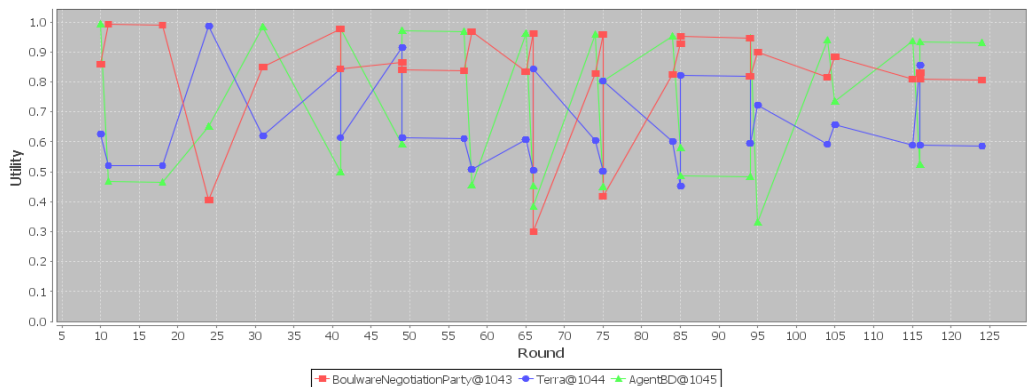


Round - Turn	BoulwareNegotiationParty@34	BoulwareNegotiationParty@35	AgentBD@36
146-1	0.671	0.38	0.83
146-2	0.549	0.67	0.854
146-3	0.487	0.432	1
147-1	0.658	0.916	0.668
147-2	0.658	0.916	0.668
147-3	0.389	0.461	0.814
148-1	0.646	0.482	0.576
148-2	0.644	0.647	0.809
148-3	0.487	0.432	1
149-1	0.634	0.465	0.598
149-2	0.771	0.634	0.727
149-3	0.487	0.432	1
150-1	0.622	0.415	0.717
150-2	0.549	0.622	0.763
150-3	0.487	0.432	1
151-1	0.609	0.571	0.736
151-2	0.641	0.609	0.725
151-3	0.487	0.432	1
152-1	0.597	0.681	0.899
152-2	0.597	0.681	0.899
152-3	0.487	0.432	1
153-1	0.582	0.568	0.72
153-2	0.713	0.582	0.669
153-3	0.487	0.432	1
154-1	0.568	0.569	0.767
154-2	0.568	0.569	0.767
154-3	0.568	0.569	0.767
154-3	0.568	0.569	0.767

## Session Progress

## Negotiation log

Turn 1:Terra@1044 (Accept bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 117  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Accept bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 118  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Offer bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 119  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Offer bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 120  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Offer bid:Bid[Place: Osaka  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 121  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Offer bid:Bid[Place: Osaka  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 122  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Accept bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 123  
 Turn 1:BoulwareNegotiationParty@1043 (Offer b.  
 Turn 2:Terra@1044 (Accept bid:Bid[Place: Tokyo  
 Turn 3:AgentBD@1045 (Offer bid:Bid[Place: Tok  
 Round 124  
 Turn 1:BoulwareNegotiationParty@1043 (Accept b.  
 Turn 2:Terra@1044 (Accept bid:Bid[Place: Tokyo  
 Found an agreement: Bid[Place: Tokyo, Budget: 90  
 Finished negotiation session in 0.37499261 s.  
 Time (s): 0.37499261  
 Rounds: 124  
 Agreement?: Yes  
 Discounted?: Yes  
 3  
 Min. utility: 0.58613  
 Max. utility: 0.93036  
 Distance to pareto: 0.00000  
 Distance to Nash: 0.63068  
 Social welfare: 2.32126  
 Agent utility: 0.80476 (BoulwareNegotia  
 Agent utility: 0.58613 (Terra@1044)  
 Agent utility: 0.93036 (AgentBD@1045)



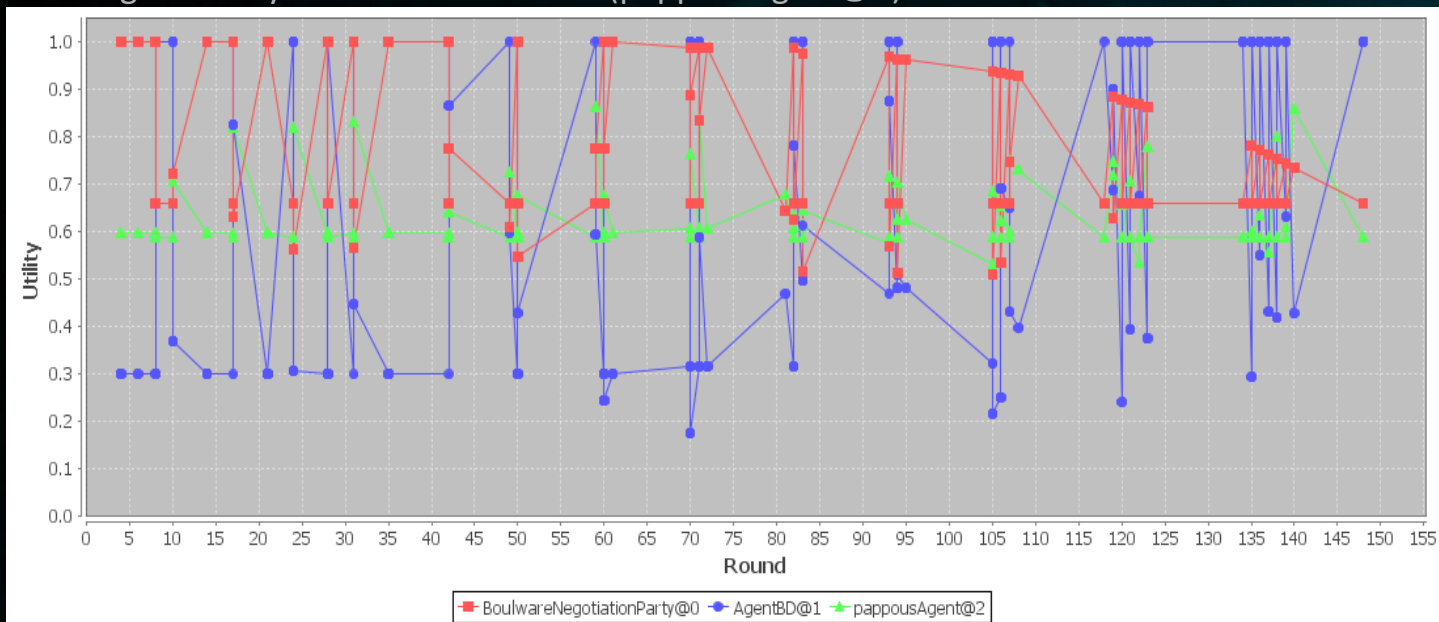
Round - Turn	BoulwareNegotiationParty@1043	Terra@1044	AgentBD@1045
1-1	0.999	0.864	0.51
1-2	0.41	0.999	0.61
1-3	0.864	0.63	0.999
2-1	0.999	0.524	0.469
2-2	0.41	0.999	0.659
2-3	0.864	0.629	0.999
3-1	0.998	0.524	0.469
3-2	0.409	0.998	0.609
3-3	0.863	0.629	0.998
4-1	0.998	0.524	0.469
4-2	0.409	0.998	0.658
4-3	0.863	0.629	0.998
5-1	0.997	0.523	0.469
5-2	0.409	0.997	0.608
5-3	0.862	0.628	0.997
6-1	0.997	0.523	0.468
6-2	0.409	0.997	0.658
6-3	0.862	0.628	0.997
7-1	0.996	0.523	0.468
7-2	0.408	0.996	0.657
7-3	0.861	0.627	0.996
8-1	0.995	0.523	0.468
8-2	0.408	0.995	0.657
8-3	0.861	0.627	0.995
9-1	0.995	0.522	0.468
9-2	0.408	0.995	0.607
9-3	0.86	0.627	0.995
10-1	0.994	0.522	0.467
10-2	0.408	0.994	0.606
10-3	0.86	0.626	0.994

## 5.Αποτελέσματα διαγωνισμού:

Το μεγάλο μειονέκτημα του πράκτορα μας, ήταν ότι σε μερικές περιπτώσεις δεν κατάφερνε να πετύχει συμφωνία μέχρι το τέλος της διαπραγμάτευσης, κάτι το οποίο έριχνε την αποδοτικότητα του σε ένα τουρνουά. Όμως η δυνατότητα του να επιτυγχάνει βέλτιστα αποτελέσματα σε μεμονωμένες περιπτώσεις μας έκανε να αποφασίσουμε να χρησιμοποιήσουμε αυτή την στρατηγική, έχοντας στο μυαλό μας ότι οι αντίπαλοι μας θα ήταν πιο αποφασισμένοι να πετύχουν συμφωνία.

Εν τέλει, ο πράκτορας είχε μέτρια αποτελέσματα και έλαβε την 7<sup>η</sup> θέση στο Pool A με απόκλιση 0.0552 από την 3<sup>η</sup> θέση που θα μπορούσε να του προσφέρει μια θέση στον τελικό. Το αποτέλεσμα δεν ήταν το αναμενόμενο, αλλά θα μπορούσε με κάποιες μικρές αλλαγές να επιτευχθεί κάποια καλύτερη θέση. Μια απλή σύγκριση με τον rarpousAgent που πήρε την 3<sup>η</sup> θέση στο pool μας, είχε την εξής κατάληξη:

Agreement?:	Yes
Discounted?:	No
3	
Min. utility:	0.58562
Max. utility:	1.00000
Distance to pareto:	0.00000
Distance to Nash:	0.19642
Social welfare:	2.24396
Agent utility:	0.65833 (BoulwareNegotiationParty@0)
Agent utility:	1.00000 (AgentBD@1)
Agent utility:	0.58562 (rappousAgent@3)



## 6.Αλλαγές μετά τον διαγωνισμό:

Μετά από το τέλος του εσωτερικού διαγωνισμού και των παρουσιάσεων, είχαμε αρκετές ιδέες για την βελτίωση του αλγορίθμου μας, κάποιες από τις οποίες θα αναφερθούν και στην συνέχεια. Για την καλύτερη αποτελεσματικότητα του αλγορίθμου μας, αποφασίσαμε να επαναφέρουμε και να εφαρμόσουμε μια μικρή αλλαγή του acceptance strategy, που είχαμε δοκιμάσει στα αρχικά στάδια της ανάπτυξης του πράκτορα μας.

Όπως αναφέραμε, το μεγαλύτερο πιθανόν ελάττωμα του πράκτορα μας ήταν ότι σε περιπτώσεις που αντιμετώπιζε άλλους hardheaded agents, δεν ερχόντουσαν σε συμφωνία. Για να αποφύγουμε αυτό, θα χρησιμοποιήσουμε την συνάρτηση acceptImmediately(), η οποία υπάρχει είδη ως υλοποίηση στον κώδικα μας απλά δεν καλείται κατά το acceptance strategy. Η συνάρτηση αυτή ουσιαστικά ελέγχει :

- α) αν φτάνουμε στο deadline των διαπραγματεύσεων
- β) αν η προσφορά του αντιπάλου είναι καλύτερη από το κατώτερο bid που έχουμε προσφέρει
- γ) αν η προσφορά του αντιπάλου είναι καλύτερη από το προηγούμενο bid μας

και σε κάθε μια από της παραπάνω περιπτώσεις κάνει αποδοχή της πρότασης του αντιπάλου.

Το νέο acceptance strategy μας έχει την ακόλουθη μορφή:

```
if(acceptImmediately()) return new Accept(getPartyId(), this.lastReceivedBid);
Bid ourBid = makeBid()
if (acceptBid(ourBid,lastReceivedBid)){
    return new Accept(getPartyId(), this.lastReceivedBid);
}
return new Offer(getPartyID(), ourBid);
```

Με την αλλαγή αυτή, όπως θα δούμε στην συνέχεια όχι μόνο βελτιώνεται η επίδοση μας στα tournaments, αλλά δεν επηρεάζεται και απόδοση του πράκτορα σε απλά negotiations.



## 7.Πειραματικά αποτελέσματα 2.0:

Αρχικά ξανατρέξαμε ένα negotiation με έναν Boulware και τον pappousAgent, για να ελέγξουμε την πιθανή επίδραση της αλλαγής στην αποτελεσματικότητα του agentBD:

Agreement?: Yes

Discounted?: No

3

Min. utility: 0.65833

Max. utility: 0.91634

Distance to pareto: 0.05472

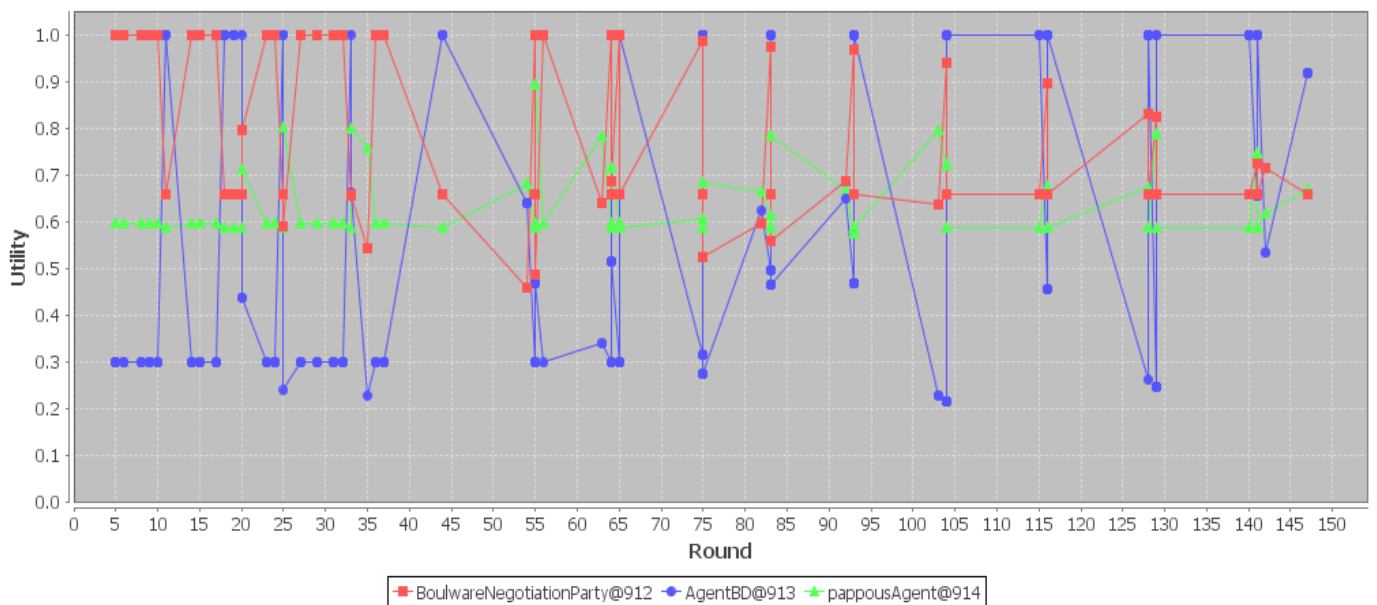
Distance to Nash: 0.13643

Social welfare: 2.24315

Agent utility: 0.65833 (BoulwareNegotiationParty@912)

Agent utility: 0.91634 (AgentBD@913)

Agent utility: 0.66847 (pappousAgent@914)



Βλέπουμε ότι η απόδοση του πράκτορα μας έχει μειωθεί, αλλά και πάλι είναι πάρα πολύ καλή και απομένει να ελέγξουμε αν όντως με αυτή την στρατηγική θα έχουμε θετικά αποτελέσματα σε tournaments.



Domain: Party	
<i>Agent</i>	<i>Average Utility</i>
Random	0.577
Boulware	0.683
agentBD	0.798

Domain: JapanTrip	
<i>Agent</i>	<i>Average Utility</i>
Random	0.606
Boulware	0.709
agentBD	0.883

Domain: Party	
<i>Agent</i>	<i>Average Utility</i>
Boulware	0.668
Terra	0.750
Atlas3	0.687
pappousAgent	0.666
agentBD	0.737

Domain: Japan	
<i>Agent</i>	<i>Average Utility</i>
Boulware	0.360
Terra	0.318
Atlas3	0.061
pappousAgent	0.512
agentBD	0.405

Είναι εμφανές ότι η απόδοση του agentBD σε tournaments θα εκτοξευτεί με την προσθήκη που εφαρμόσαμε. Αποδεικνύεται λοιπόν, ότι η απόφαση μας να κάνουμε λίγο πιο hardheaded τον πράκτορα μας ήταν εσφαλμένη, καθώς οδήγησε σε χειρότερη απόδοση σε tournaments, παρότι βελτίωσε ελαφρά την επίδοση του σε απλά negotiations.

## 8. Brainstorming:

Όπως αναφέρθηκε, είχαμε κάποιες ακόμα ιδέες για την περαιτέρω βελτίωση του πράκτορα μας, τις οποίες δεν είχαμε χρόνο να προσαρμόσουμε και να εκτιμήσουμε την αποτελεσματικότητά τους.

Στην νέα έκδοση του agentBD, χρησιμοποιήσαμε έναν διαφορετικό acceptance strategy. Μια ακόμα προσθήκη που θα μπορούσαμε να εφαρμόσουμε, θα ήταν η δημιουργία ενός μηχανισμού concession, με βάση τον οποίο άμα παρατηρούσαμε, ότι ο αντίπαλος πράκτορας είναι υπερβολικά hardheaded, θα παραδινόμασταν ώστε να έρθουμε σε πιο εύκολη συμφωνία.

Πέρα από το acceptance strategy, θα μπορούσαμε να πειραματιστούμε με την αλλαγή του υπολογισμού του threshold και να εφαρμόσουμε μια Boulware τακτική παρόμοια με το paper[4].

Τέλος, μια ενδιαφέρουσα ιδέα θα ήταν να αλλάξουμε ελαφρά το opponent module, σύμφωνα με το paper[2] και να μην συγκρίνουμε κάθε bid με το προηγούμενο, αλλά κρατάμε τα bids στη μνήμη και συγκρίνουμε με X προηγούμενα bid με τα X τωρινά bid. Αυτό σύμφωνα με τους συγγραφείς βελτιώνει το κλασσικό Hardheaded frequency model. Το X σαν μέγεθος παραθύρου θα επιλέγονταν με βάση πειραματισμούς.

## 9.Βιβλιογραφία:

- [1] AgentK: Compromising Strategy Based on Estimated Maximum Utility for Automated Negotiating Agents, Kawaguchi, Fujita, Ito
- [2] Rethinking Frequency Opponent Modeling in Automated Negotiation, Tunalı, Aydoğan, Sanchez-Anguix
- [3] TheNegotiator: A Dynamic Strategy for Bilateral Negotiations with Time-Base Discounts, Dirkzwager A.S.Y., Hendrikx M.J.C., De Ruiter J.R.
- [4] Optimal Negotiation Strategies for Agents with Incomplete Information, S.S Fatima, M. Wooldridge, N.R. Jennings
- [5] Σελίδα του Genius: <http://ii.tudelft.nl/genius/>