AI player for board game Diplomacy

Sancho de Mascarenhas, and Pedro Santos

Instituto Superior Técnico, Universidade de Lisboa
sancho.mascarenhas@ist.utl.pt

Abstract. Diplomacy is a strategy board game in which 2 to 7 players compete for their supremacy over Europe in the turn of the 20th century. It is a game where luck or randomness has no role and players negotiate with each other in order to gain advantage. The board is divided into 75 provinces where 34 are called supply centers. Each turn is divided in phases dedicated to negotiation, the movement of pieces and board adjustments. At the end of each turn, each player player adjusts the number of pieces to the number of supply centers they have. The player who, by the end of a season, detains 18 out of the 34 supply centers available in the board wins the game. In this work, it was created an automated player (bot) dedicated to play the game presented above. Tagus is a bot developed for DAIDE's game platform, using the offered development kit. By negotiating with other players and using the opening libraries built in it, the bot gains advantage in the game, always with the goal of winning. The type of negotiation is simple which included only peace treaties and alliances based upon the tension and trust with other players. The experiments made reveal that negotiation brings advantage to the involved players. It was also clear that the opening libraries strengthen its users at the start of the game.

Keywords: Diplomacy, Negotiation, Opening Libraries, Artificial Intelligence, Automated Player, DAIDE, Multi-Agent System.

1 Introduction

Diplomacy is a board game created by Allan Calhamer in 1954 and it was playtested and tuned until 1958, and since then many companies have published it[2]. With the arrival of the internet, Diplomacy also started being played via e-mail and later in 1984, the first Diplomacy video game was released.

The main characteristic of Diplomacy that makes it so interesting to the scientific community is the fact that it includes no luck, no random factors and there is no hidden information on the board (for example, there are no cards) other than the agreements made between players as the game is played. The main component of this game is negotiation and persuasion. Players make deals with each other in order to satisfy their interests and are free to break those deals anytime they see fit. These deals may include, for example, alliances, peace treaties or information exchange. Negotiations are done in a specific phase of the game referred to as Diplomacy period. Also, during this phase, players

are allowed to make their deals in different rooms, so that the remaining players are kept out. Thus, the only information considered hidden in this game are the deals between players.

Comparing to other zero-sum games, such as chess, the negotiation factor in Diplomacy as well as the number of players and units make its branching factor, i.e., the number of possible plays in one turn, larger than the other multiplayer games. This means that it is almost impossible to get the optimal solution. Even with the Monte Carlo Tree Search (MCTS), the solution would not be trivial to find. Also, the non-determinism of Diplomacy make the use of heuristics not very effective. Furthermore, it is agreed by the scientific and gamer community that there is no heuristic that can evaluate with precision the state of the game [6]. Players only have access to the information the board offers and their own deals, which are hidden and are not accessible to everyone.

Overall, Diplomacy's seems a rather simple game but in theory it can be quite complex. This is the reason why it is an interesting target of study in artificial intelligence. As this game offers a good environment to test negotiation, the main goal of this project will be to build an automated player, i.e, an artificial intelligence (or bot) capable of playing Diplomacy including the taking advantage of negotiation. The bot will make use of the tension created between oneself and other players to decide what, when and with whom will it negotiate with. Besides negotiation, the bot will also take advantage of strategic moves in an initial phase of the game in order to be strong right from the start. Thus, this project can be divided in the following subgoals:

Subgoal #1 Create a functional automated player for board game Diplomacy;
Subgoal #2 Elaborate strategic moves that the player can follow to be stronger in the start of the game;
Subgoal #3 Enhance the player with negotiation abilities to gain advantage over the others as the game progresses;

2 Related work

2.1 Diplomacy: the game

In Diplomacy, preferably, 7 players control a European power set in the turn of the 20th century. The rules presented below were extracted from Diplomacy games rule book [11]

These powers include Russia, Turkey, Austria, Italy, Germany, England and France and each powers compete against the others to establish supremacy over the continent. Also, the map includes 75 provinces where 34 are supply centres. Some supply centres are called "home" supply centres as they are controlled by the specific power even when it is not occupied. For example, the home supply centres of Russia are are Moscow, St. Petersburg, Warsaw and Sevastopol and are controlled by Russian even if no Russian unit occupies it. However, it stops being controlled, when another power (e.g. Turkey, Austria) occupies it. When a player controls 18 of the 34 supply centres, he/she wins the game. In order to

control these provinces, each player possess armies and fleets that he/she must maneuver throughout the map.

Each round is composed of 3 distinct phases: 1) the diplomacy phase, 2) the writing orders phase and 3) the execution of the written orders. The diplomacy phase is the time when players are allowed to negotiate and bargain whatever deal they desire. However, no agreement is binding: each player may or may not choose to later betray the players he/she made a deal with. After diplomacy is made, the orders for the units to take must be written. Each player writes what each of his/hers controlled units will do. Once written, the orders cannot be altered. After the order have been written, they are passed to the player next to the owner of the orders and are read to all players. The orders have to be written in a specific format: they must indicate the round they will be performed in, the unit itself (army or fleet), the province the unit is occupying, the action and the province which the action will take place. This format ensures that the orders are clear and not ambiguous.

2.2 Negotiation Testbeds

Regarding multi agent systems, there are several testbeds, a platform for testing, that allow automated negotiation experimentation. Some testbeds for MAS include:

- Multi AGent NEgotation Testbed, MAGNET for short[3]
- Trading Agent Competition, TAC[10]
- the Agent Reputation and Trust Testbed, ART [7]
- the Colored Trails Game, CT[8]
- Generic Environment for Negotiation with Intelligent multipurpose Usage Simulation, GENIUS[9]

2.3 Diplomacy Testbeds

Initially, Diplomacy had a few platforms where players could play against each other or against low capability bots. As an example, the community has the digital version released by Avalon Hill in 1984 in which the computer opponents are not fully intelligent [1]. With the increase in popularity of the game among players, and equally by the scientific community, some testbeds created especially for Diplomacy started to appear and being the ones that will be discussed in more detail below. I will focus in two particular testbeds: the Diplomacy AI Development Environment (DAIDE) and the Dipgame.

DAIDE The Diplomacy AI Development Environment, is and environment created in January 2002 by a group of programmers with the purpose of allowing several Diplomacy bots to compete against each other. This environment consists of a communication model, a communication protocol and a specific language, including syntax and semantics in which negotiations and instructions are expressed. It also provides some useful utilities such as an arbiter that serves as

a judge, several libraries that facilitate the creation of new bots and a diversity of bots to test new AIs[1]. This platform (server and client) was developed in C/C++ language.

In order to categorize the communication capabilities of the bots, the group of DAIDE created several levels of communication called Press levels. The higher the level, the more complex is the communication capability of the bot. There are 15 levels but the highest level reached so far is the 4th level (e.g., Level 30 Multi-part Offers in Table 1).

Table 1. Press Levels in DAIDE

Level	Category
Level 0	No Press
Level 10	Peace and Alliances
Level 20	Order Proposals
Level 30	Multi-part Offers
Level 40	Sharing out the Supply Centers
Level 50	Nested Multi-part Offers
Level 60	Queries and Insistences
Level 70	Requests for suggestions
Level 80	Accusations
Level 90	Future Discussions
Level 100	Conditionals
Level 110	Puppets and Favours
Level 120	Forwarding Press
Level 130	Explanations
Level 8000	Free Text

Dipgame The Dipgame testbed is the newest testbed that allows competition between Diplomacy bots. It was developed by IIIA-CSIC of Barcelona and it is very well documented in phD thesis of Angela Fabregues [5]. Dipgame uses a similar infrastructure to DAIDE with the exception that it provides a parallel server exclusively for negotiation and it is built in Java. Also, to complement this addition, it also includes a negotiation library to allow the use of the L language. The L language is a layered communication standard (Figure 1) that defines the way agents can communicate. The communication between clients and server is done by TCP/IP which allows the players to set a game using different computers but the same network.

Dipgame does not support convoy moves. This is because "adding convoys increases the adjudicator's (module that applies rules) complexity 100% while not providing any benefit from the negotiating point of view", stated by Angela Fabregues[4].

[1] Introduction to DAIDE webpage: http://www.daide.org.uk/index.php?title=Introduction

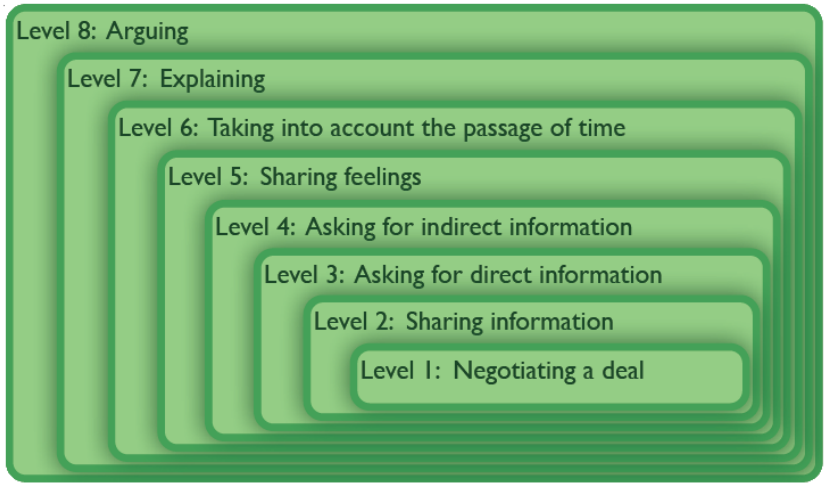

Fig. 1. Dipgame's L Language table.

2.4 Diplomacy bots

With the creation of the before mentioned testbeds, some bots were created and put into test against each other. Over the years, developers kept creating more and better bots than the ones previously created. Even before the creation of DAIDE, some bots were created with the goal of playing and winning a game of Diplomacy. Only the relevant bots for this work will be described while only mentioning others.

Dumbbot Dumbbot is a very popular bot in the Diplomacy community. It was created in just two hours, for a challenge, by David Norman[2]. It is a simple bot that does not negotiate but performs well given the circumstances in which it was created. Some of its parameters were chosen by chance, like the values of its heuristics.

The bot works in two distinct stages being the first stage used to calculate a value for each province and coast. To calculate such values, Dumbbot has the following criteria: if it's its supply center, the size (number of units) of the largest adjacent power; if it's not its supply center, the size of the owning power; if it is not a supply center, zero. The resulting values make it so that Dumbbot tends to make an attack or defend against the strongest player. These values are used to determine not only the moves but also the retreats, builds and disbands. In the second stage, it assigns an order to each of its units trying to move to the adjacent province or coast with the highest value.

[2] Dumbbot Algorithm webpage: http://www.daide.org.uk/index.php?title=DumbBot_Algorithm

RandBot RandBot was also created by David Norman and it is a simple Diplomacy bot that plays randomly: from the available moves each turn, it chooses one randomly.

Next, a description of exclusive bots for DAIDE will be made. The majority of these bots are described in the web page of their creator rather than on a refereed publication. Some of them were actually created as an hobby and not as an academic project.

Albert Albert was created by Jason van Hal and is considered by the community the best Diplomacy bot so far. It is able to play a level 30 press game, which means it can perform multi-part offers and it uses KissMyBot, another Jason van Hal's creation, as a basis. It is known that Albert deeply thinks about other power's best moves and, iteratively, adjusts the probability of selection of any order set. Also, the creator states that Albert tries "to guess what the other power is going to do"[3]. Albert does have a downside: its good performance comes with a price in term of time consume. The bot takes between five to ten minutes to submit its orders. Of course that this depends on the computer. As the game progresses and Albert gets more units, it gets even slower.

BlabBlot BlabBot was created by John Newbury. It is built over David Norman's Dumbbot. Therefore, we can consider it a Dumbbot capable of simple negotiation[4]. Though it is not clear what press level it is capable of playing, it is stated in [6] that it plays up to level 20 press. Blabbot's negotiation strategy consists of sending an initial peace proposal to all powers. The powers that accept the proposal are considered friends decreasing the value of that power's provinces. Also, if a power sends a peace proposal to Blabbot, it always accepts the proposal. From that point, Blabbot's actions depend on its Policy. The Policy is an argument passed via command-line and it is optional. If "Joe" is passed, Blabbot remains honest and it will offer a draw if all the remaining powers are friends. Another possible argument is "Mac" which means Blabbot will backstab the remaining players who agreed with peace and it will never offer draws, playing like DumbBot for the rest of the game.

Other bots include:

- Israeli Diplomat
- Bordeaux Diplomat
- LA Diplomat
- D-Brane
- DipBlue
- Diplominator
- others

[3] ALbert webpage: https://sites.google.com/site/diplomacyai/albert
[4] Blabbot webpage: http://johnnewbury.me.uk/diplomacy/blabbot/

3 Tagus Bot's Architecture and Implementation

Tagus is a bot that plays Diplomacy and tries to gain advantage over its opponents by negotiating with them, either to form peace treaties or alliances against other players. Not only that but Tagus also makes use of opening libraries to help it gain some board advantage right at the beginning of the game. It uses some data structures to save information about how trustworthy its opponents are as well as how much tension or friction there is between it and the opponents.

In order to develop the bot, two different environments were considered, ending up with the Diplomacy AI Development Environment (DAIDE) being chosen. Furthermore, it was used a development kit created by David Norman and updated by John Newbury. This development kit contained all the necessary base material that allowed the created bot to communicate with the server, have a representation of the board and of the plays. This way, it was possible to focus on the implementation of Tagus itself.

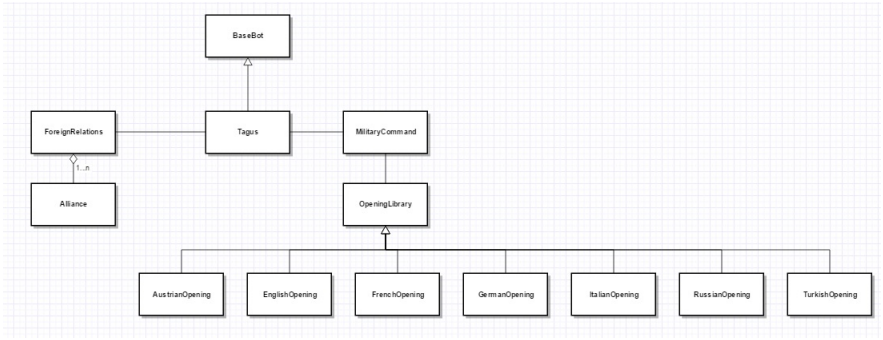

Fig. 2. Tagus' architecture diagram

3.1 Tagus

Tagus is the core class of the bot. It inherits from the BaseBot class provided by the development kit and it is where the other classes are initiated.

The BaseBot class contains the functions used to handle the incoming messages from the server, including press. But, as it is a basic bot, the content of the messages isnt, in fact, treated. It is Tagus that handles the information of each message regarding the game itself, i.e, not the connection part. Therefore, one the features of Tagus is to parse the content of the messages following DAIDE's syntax. To do so, a parser was built in Tagus class to process the content of the messages.

It is this class that also sends proposals to other players (via the server) with the help of the Foreign Relations. As mentioned before, Tagus is a bot for level

10 press games, which means that it is only capable of negotiating peace and alliances.

3.2 Military Command

One of Tagus component is called the Military Command. This component uses the DumbBot code provided in the Development Kit. Unlike BaseBot, which Tagus directly inherits from, the Military Command does not inherit from DumbBot. It is Tagus version of DumbBot.

As the name suggests, the Military Command is charged with generating orders to each unit every turn. Using DumbBots algorithm to analyze the board, the Military Command assigns a value to each supply center based upon the strength of the supply centers owner and blurs that value to the surrounding provinces. As such, Tagus, like DumbBot, tends to attack supply centers owned by the largest powers. Then, the orders are generated.

Although DumbBots code was re-utilized for this project, some changes were made.

Province evaluation The major change was regarding the calculation of the values for each province. Instead of considering only the strength of each supply centers owner, Tagus also considers the relation between itself and the owner of that center. When calculating the defense values, Tagus considers allied provinces as if they were its own. Therefore, Tagus does not discard the possibility of supporting an allied province. As far as attacking is concerned, when calculating the attack value for a province, Tagus also considers if that province belongs to an ally or to someone who it has a peace treaty with.

To accomplish this behavior, the calculated values are multiplied by a weight, depending on the situation: if it is a province occupied by an ally/peace treaty and the bot is calculating defense values, the value is calculated as if it would be province belonging to it; if it is calculating an attack value and it is and allied province, the value is multiplied by one half (0.5); if the province belongs to an enemy, the attack value for that province is multiplied by one and a half (1.5).

Betrayal Regarding betrayal, mentioned above, Tagus always checks if it should betray an ally (including peace treaties). The criteria for backstabbing includes the number of supply centers owned by that ally and the trust Tagus has towards that ally. If the ally has ten or more supply centers, the attack value for its provinces will be calculated normally, without multiplying it by 0.5, increasing the possibility of it being attacked. Later it will be explained how Tagus uses trust.

Determinism vs Non-Determinism To avoid being deterministic and predictable, DumbBot uses randomness when choosing the destination of a unit. After calculating each provinces value, for each unit that DumbBot possesses, it

would randomly choose one possible destination amongst the best ones, sometimes not being the best one. For this project, as Tagus would only face other automated players and it was not expected that its moves would be predicted, the random aspect regarding destination selection for each unit was removed. As such, Tagus would only move its units to provinces with the highest value.

3.3 Opening Library

A small addition to Tagus was the use of an opening library depending on the power played. There are seven libraries, one for each power, and they were based on articles written by Richard Hucknall [5].

This component simply consists of a set of powers that Tagus wishes to establish peace with in the beginning of the game and a set of provinces of interest for the beginning of the game as well. In a way, Tagus first two rounds have a scripted component. The goal is to make Tagus a stronger player in the first few rounds of the game.

The Opening Library class can be extended to seven different subclasses, one for each power. At the start of the game, after the assignment of the powers to each player, Tagus verifies which power it was assigned with and loads the adequate library. After that, Tagus sends a peace request to the powers contained in the library. Regarding the Military Command, it calculates the values for the provinces considering of interest contained in the library, i.e., multiplying the province value by an associated weight.

3.4 Foreign Relations

The Foreign Relations is the component that gathers the information about alliances and peace treaties between Tagus and the other players. Not only that but it also keeps track of any potential alliance or peace treaty between other players. It is also the Foreign Relations that deals with trust and tension.

Tagus considers that two or more players have an agreement (either peace or alliance in this level of press) if it witnesses a support from either of the players to any other.

Trust Factor Regarding trust, as the game progresses, the Tagus analyzes other players moves and updates a map with seven entries, each with a value between 5 and -5 associated, starting in 1. That value increases or decreases depending on the other players moves and what Tagus assumed to be an alliance or peace treaty. If Tagus witnesses an attack from a player to another and they were considered allies of each other, Tagus decreases the trust value associated to the attacker by 2. This value is also increased by 1 anytime Tagus witnesses a cooperative move between two players he previously considered allies of each other. This way, any betrayal would have a greater effect over Tagus judgement along the game.

[5] http://www.diplomacy-archive.com/resources/strategy/opening.htm

Tension Factor Another factor that helps in Tagus way of playing is the tension factor. Like the trust factor, tension is also a value between 5 and -5, starting in 1 as well. This value is used as way to look for peace treaties or alliances against someone and there is a value associated to each other player. As the game progresses, the tension value is decreased by 2 every time a hostile move is made towards Tagus. The value is also increased, by 1, at the beginning of every spring or fall season. When the value is lesser or equal then -1, Tagus will try to arrange an agreement of peace with the hostile power or an alliance against it. Thus, the more hostile moves in a single turn, the quicker Tagus will look for a solution to this tension.

The tension factor is modified this way so that Tagus wont look for immediate peace with a player as result of one single attack. It is meant to look for agreements in case of recidivist attacks.

Still regarding the tension factor, Tagus also keeps a 7x7 matrix to record the tension between other players. The values in this matrix are updated the same way as explained before. This is used to check which player would tend to form an alliance against a certain other player.

3.5 Breaking Agreements

Another feature that can be found in Tagus, and as a major aspect of the game, is the ability of breaking an agreement it made and not being bound to it until the end of the game.

In a game of Diplomacy, the criteria for backstabbing can be quite diverse. Tagus criteria when pondering backstabbing includes the number the number of supply center the other powers have. For instance, if Tagus has an agreement with a power that has a high number of supply centers, in this case 10 or more, a betrayal is most likely to happen. Another criterion involves the trust that Tagus has with that power. If Tagus has low trust in a power with whom it made an agreement with, Tagus rather betray instead of getting betrayed.

4 Experiments

To test the developed bot, Tagus, it was first necessary to decide how it would be tested. Therefore, it was created five series of one thousand games each. In each series, Tagus would play against a certain known bot and it was evaluated the negotiation made, the supply center count and the final result of each game.

In order to make it possible to run a total of five thousand games, it was necessary to automatize the process of launching the server and the seven bots that were to participate in each series five thousand times. The chosen method was the use of Powershell scripts due to its modern and sophisticated attributes. Also, all the results were stored in comma separated values files (.csv), for easier treatment.

The five series were the following:

- Series 1: 6 RandBot and 1 Tagus

- Series 2: 6 DumbBot and 1 Tagus
- Series 3: 4 DumbBot and 3 Tagus
- Series 4: 4 BlabBot and 3 Tagus
- Series 5: 4 Albert and 3 Tagus

5 Results

The results of the five series are translated in the five tables bellow

The first series had the expected good results. Tagus won 987 games out of 1000, a percentage of 98,7, and there were no draws. Regarding the number of supply centers, there was a significant increase in the first 8 turns of the game, having some games ending between the 8th and 10th turn.

	Tagus		
Power	# of Games	# of Wins	% of Wins
Austria	147	146	99,32
England	145	141	97,24
France	131	129	98,47
Germany	138	136	98,55
Italy	131	130	97,24
Russia	159	159	100
Turkey	149	146	97,98

Table 2. Results of series 1 with.

In this series, Tagus, as expected, showed a poor performance regarding win percentages. Out of 1000 games, Tagus was able to win only 59 games. By not being able to negotiate with other players, Tagus was tied to its opening strategies and to its "DumbBot behavior" in order to win. The poor results of this series are also related to the "nature" of DumbBot as its algorithm focuses on moving to supply centers owned by the stronger players, i.e, players with higher number of supply centers.

	Tagus			DumbBot		
Power	# of Games	# of Wins	% of Wins	# of Games	# of Wins	% of Wins
Austria	133	0	0	866	74	8,54
England	146	21	14,38	852	299	35,1
France	153	12	13,27	845	132	15,62
Germany	127	0	0	885	4	0,45
Italy	141	5	11,97	857	111	12,95
Russia	150	13	12,28	848	44	5,19
Turkey	150	8	7,45	835	107	12,81

Table 3. Results of series 2 with.

In the third series, Tagus accomplished only 340 wins in this series and 133 draws. Although it was not the minimal number of wins, which would be 3/7 of 1000, around 429 wins, 340 is not a value too far from the such minimal, considering the total number of games. The fourth series of test to be run was

	Tagus			DumbBot		
Power	# of Games	# of Wins	% of Wins	# of Games	# of Wins	% of Wins
Austria	450	24	5,33	551	50	9,07
England	472	113	23,94	529	168	31,76
France	437	58	13,27	563	93	16,52
Germany	391	13	3,32	610	5	0,81
Italy	426	51	11,97	575	83	14,43
Russia	399	49	12,28	602	33	5,48
Turkey	429	32	7,45	572	67	11,71

Table 4. Results of series 3 with.

the first to have seven negotiating bots. It is know that BlabBot's negotiating capacities are limited to only peace and draw proposals. Nevertheless, it was enough to run a series where Tagus could negotiate with any of the players, contrary to the previous. Tagus was able to get a total of 361 wins, 79 draws and 560 losses. Something that was possible to identify in this series was the use of the betrayal feature. The last series of tests was the one with the most

	Tagus			BlabBot		
Power	# of Games	# of Wins	% of Wins	# of Games	# of Wins	% of Wins
Austria	452	70	15,48	548	28	5,11
England	419	6	1,43	581	32	5,51
France	419	113	26,97	582	137	23,54
Germany	450	39	8,66	550	51	9,27
Italy	408	20	4,90	592	97	16,39
Russia	431	66	15,31	569	123	21,62
Turkey	422	47	11,14	578	46	7,96

Table 5. Results of series 4 with.

curious results. Tagus achieved 236 victories which, again, is below the minimal but, considering the high number of games played, Tagus only lost 316. This means the series had a total of 448 draws, making it the one with the highest number. These results were surprising considering the ones from series 4, where Tagus played against a supposedly weaker bot.

	Tagus			Albert		
Power	# of Games	# of Wins	% of Wins	# of Games	# of Wins	% of Wins
Austria	442	22	4,98	582	46	7,9
England	389	7	1,80	537	58	10,8
France	434	72	16,59	566	61	10,78
Germany	459	21	4,58	563	35	6,22
Italy	408	25	6,18	599	42	7,01
Russia	431	54	12,53	574	40	6,97
Turkey	423	35	8,27	579	34	5,87

Table 6. Results of series 5 with.

6 Conclusions

The main goal was achievable by successfully completing the three subgoals presented in the first section.

6.1 Subgoal #1

The first subgoal, to "create a functional automated player for board game Diplomacy", was successfully achieved with the help of David Norman's development kit. It provided the basic but essential material to build a simple Diplomacy bot and start the project. Tagus' main class was created, extending the provided BaseBot class and, by using DumbBot, Tagus become a playable bot though with nothing new. This was not an issue regarding the subgoal as, overall, the project focused more in its negotiation abilities.

6.2 Subgoal #2

With the first subgoal complete, i.e, with a fully functional bot, it was possible to start working on its enhancements. The first one, which was the second subgoal of this project, consisted on the implementation of the opening libraries. These turned out be a very flexible component of the bot as more can easily be created, as well as more complex ones.

The results show that some opening libraries were more effective than others. Taking France as an example, the library helped Tagus grow stronger in the first turns of the game by focusing on key supply centers. With some other powers, for example Italy, the library was not very helpful mainly because the supply centers it would focus on were contested most of the times.

Overall, this subgoal was successful as in some cases Tagus' was able to take advantage of the libraries.

6.3 Subgoal #3

The third subgoal of this project, and the one in which more effort was spent to complete, focused on implementing negotiation abilities to the bot. This subgoal

would make the difference between having a bot solely capable of playing level 0 press games or higher, in this case, level 10. Presently, Tagus is a successful negotiating bot for Diplomacy, being able to play up until level 10 press games. It negotiates peace and alliances with other players as either an effort to protect itself or in an offensive way.

Still regarding this subgoal, Tagus successfully ponders betrayals and executes them when it sees fit.

References

1. Brooks, M.E.: An annoted paiktograpgy ("game"ography). Computer Gaming World: The Definitive Computer Game Magazine (75), 12 (1990), retrieved from http://www.cgwmuseum.org/galleries/issues/cgw_75.pdf
2. Calhamer, A.: The invention of diplomacy. diplomacy-archive.com (1974), retrieved from http://www.diplomacy-archive.com/resources/calhamer/invention.htm
3. Collins, J., Tsvetovat, M., Mobasher, B., Gini, M.: Magnet: A multi-agent contracting system for plan execution. In: Proceedings of SIGMAN 98 (1998)
4. Fabregues, A.: personal contact via email, february 11, 2016
5. Fabregues, A.: Facing the Challenge of Automated Negotiation with Humans. Ph.D. thesis, Universitat Autnoma de Barcelona (2012)
6. Ferreira, A.: DipBlue: a Diplomacy Agent with Strategic and Trust Reasoning. Master's thesis, Faculdade de Engenharia da Universidade do Porto (2014)
7. Fullam, K.K., Klos, T.B., Muller, G., Sabater, J., Schlosser, A., Topol, Z., Barber, K.S., Rosenschein, J.S., Vercouter, L., Voss, M.: A specification of the agent reputation and trust (art) testbed: Experimentation and competition for trust in agent societies. In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (2005)
8. Grosz, B.J.: Colored trails: a formalism for investigating decision-making in strategic environments. In: In IJCAI Workshop on Reasoning, Representation, and Learning in Computer Games (2005)
9. Lin, R., Kraus, S., Baarslag, T., Tykhonov, D., Hindriks, K., Jonker, C.M.: Genius: An integrated environment for supporting the design of generic automated negotiators. Comput. Intell. 30(1), 48–70 (2014)
10. Wellman, M., Wurman, P.R.: A trading agent competition for the research community. In: In IJCAI-99 Workshop on Agent-Mediated Electronic Trading (1999)
11. Wizards of the Coast: Diplomacy rules book (2014)