Diplomacy Literature Search

Steve Smith

Contents

1 Introduction **5**

2 The Diplomatic Pouch **5**

3 Play Diplomacy Websites **5**

4 Software and Tools **6**

4.1 Apps . . . 6

4.2 HANA and DipGame . . . 6

4.3 DAIDE . . . 6

4.4 Diplomacy Bots . . . 6

4.4.1 DAIDE and Albert . . . 6

4.4.2 Other Bots . . . 7

4.5 DipTool . . . 7

4.6 JDiplomacy . . . 7

4.7 Diplomacy Strategy Editor . . . 7

4.8 webDiplomacy . . . 7

5 Articles **8**

5.1 Designing and Building a Negotiating Automated Agent . . . 8

5.2 Automated Negotiation and Decision Making in Multi-agent Environments . . . 8

5.3 Learning a Game Strategy Using Pattern-Weights and Self-Play . . . 8

5.4 Building Automated Agents for Multi-agent Games: A Case Study on "Diplomacy" . . . 8

5.5 Learning, Resource Allocation and the Art of War . . . 9

5.6 Tactical Coordination in No-Press Diplomacy . . . 9

5.7 MARS - A Multi-Agent System Playing Risk . . . 9

5.8 On Using Multi-Agent Systems in Playing Board Games . . . 9

5.9 Using Multi-agent Potential Fields in Real-time Strategy Games . . . 9

5.10 Agent Architecture in Social Games - The Implementation of Subsumption Architecture in Diplomacy . . . 10

5.11 Diplomacy Game: The Testbed . . . 10

5.12 A Testbed for Multi-agent Systems . . . 10

5.13 DarkBlade: A Program That Plays Diplomacy . . . 10

5.14 Is Human-like and Well Playing Contradictory for Diplomacy Bots? . . . 10

5.15 Evolving Robust Strategies for an Abstract Real-time Strategy Game 11
5.16 Influence Graphs: a Technique for Evaluating the State of the World 11
5.17 Automated Negotiations 11
5.18 DipGame: A Testbed for Multi-Agent Systems 11
5.19 An Agent Architecture for Simultaneous Bilateral Negotiations 11
5.20 A Balanced Diplomacy Tournament 12
5.21 RedTNet: A Network Model for Strategy Games 12
5.22 A Selective Move Generator for the Game Axis and Allies 12
5.23 Developing Intelligent Bots for the Diplomacy Game 12
5.24 DipGame: A Challenging Negotiation Testbed 12
5.25 DipTools: Experimental Data Visualization Tool for the DipGame Testbed 12
5.26 Nested Look-Ahead Evolutionary Algorithm Based Planning for a Believable Diplomacy Bot 13
5.27 Modeling Social Preferences in Multi-Player Games 13
5.28 Running Experiments on DipGame Testbed 13
5.29 A Comparison of Diplomacy Gameboard Graph Search Algorithms 13
5.30 HANA: A Human-Aware Negotiation Architecture 13
5.31 Abridgement of HANA: a Human-Aware Negotiation Architecture 14
5.32 Branch and Bound for Negotiations in Large Agreement Spaces 14
5.33 Comparing Co-evolution, Genetic Algorithms, and Hill-Climbers for Finding Real-Time Strategy Game Plans 14
5.34 Finding Robust Strategies to Defeat Specific Opponents Using Case-Injected Co-evolution . 14
5.35 A Review of Computational Intelligence in RTS Games 14
5.36 Designing Competitive Bots for a Real Time Strategy Game using Genetic Programming . . 15
5.37 Monte Carlo Tree Search Variants for Simultaneous Move Games 15
5.38 Linguistic Harbingers of Betrayal: A Case Study on an On-line Strategy Game 15
5.39 DipBlue: A Diplomacy Agent with Strategic and Trust Reasoning 15
5.40 Strategic Negotiation and Trust in Diplomacy - The DipBlue Approach 16
5.41 Learning About the Opponent in Automated Bilateral Negotiation: a Comprehensive Survey of Opponent Modeling Techniques 16

6 Theses **16**

6.1 Diplomacy AI 16
6.2 Multi Agent Diplomacy 17
6.3 Creating a Diplomat 17
6.4 Automated Negotiation in the Game Of Diplomacy 17
6.5 Mind Game: (Axis and Allies) 17
6.6 Automated Negotiation in the Game of Diplomacy 18
6.7 Optimizing a Diplomacy Bot Using Genetic Algorithms 18
6.8 Facing the Challenge of Automated Negotiation with Humans 18

6.9 Multi-Agent Potential Field Based Architectures for Real-Time Strategy Game Bots 19
6.10 Study on Intelligent Negotiation using Genetic Network Programming Evolving based on Agreement Conditions 19
6.11 Co-evolutionary Approaches to Generating Robust Build-Orders for Real-time Strategy Games 20
6.12 On-line Build-Order Optimisation for Real-time Strategy Agents Using Multi-Objective Evolutionary Algorithms 20
6.13 Negotiations Over Large Agreement Spaces 21

7 Books **21**
7.1 The Game of Diplomacy, by Richard Sharp 21
7.2 Diplomacy: Prima's Official Strategy Guides, by Rex Martin and Michael Knight 21
7.3 The Art of Correspondence in the Game of Diplomacy, by Conor Kostick 21
7.4 Calhamer on Diplomacy: The Boardgame "Diplomacy" and Diplomatic History, by Allan Calhamer 22

8 Variants **22**

9 My Economic Variant **22**

10 Listed Articles **23**

11 Listed Theses **24**

Prerequisites

None.

Notes

None.

Document History

Date	Version	Comments
13th February 2016	1.0	Initial creation of the document.

1 Introduction

This document is my collection of Diplomacy-related discoveries from around the internet. I started this search with the idea of finding somewhere I could play the game. My ideas have evolved considerably as I went along, and discovered more and more.

I now want to make by own bot (whatever that is), research automated negotiation (whatever that means), write my own Diplomacy-playing program in Java, and write my own economic Diplomacy variant.

When am I going to get the time to do all this?

Anyway. So, in no particular order, I present my current jumble of thoughts on all things Diplomacy related.

2 The Diplomatic Pouch

The Diplomatic Pouch, DPCouncil (2015 (accessed January 25, 2016), is your one-stop shop for most things Diplomacy related. There are loads of links to sites related to:

- general resources (rule books, the variant bank)
- Diplomatic Pouch resources (the openings library, tactics, strategies, maps, stalemate lines)
- Judge resources (being a Games Master, etc)
- Player communities
- Diplomacy organisations
- Online Zines
- The Diplomacy ring

3 Play Diplomacy Websites

The following is a short list of websites that enable the playing of the game.

- Backstabbr BS (2016 (accessed February, 2016)
- Dip 2000 Dip2000 (2016 (accessed February, 2016)
- Diplomatic Corp DC (2016 (accessed February, 2016)
- Play Diplomacy Online PDO (2016 (accessed February, 2016)
- Web Diplomacy WD (2016 (accessed February, 2016)

4 Software and Tools

A set of test cases for analysing the performance of automated adjudicators can be found at Kruijswijk (2009).

4.1 Apps

It looks like the only app worth considering is "Droidippy" (Oort Cloud (2016 (accessed January 27, 2016)).

4.2 HANA and DipGame

Angela Fabregues and Carles Sierra at The Artificial Intelligence Research Institute (IIIA) in Barcelona have developed a human-computer negotiation language called HANA and a tool to test it using Diplomacy as a vector. Their published work on this can be found at Fabregues (2012b) and Fabregues et al. (2016).

dipGame is also a test bed for developing diplomacy-playing bots (see also Section 4.4). Instructions on developing your own bots can be found at the dipGame website.

Amongst the publications released by this group are: Fabregues and Sierra (2009b), Fabregues and Sierra (In press), Fabregues et al. (2012), Fabregues and Sierra (2011), Fabregues et al. (2011), Fabregues et al. (2010), Fabregues and Sierra (2009a), and Fabregues (2012a).

Dave de Jonge (de Jonge (2016 (accessed January 27, 2016)) was also involved in this work.

See Sections 5 and 6 for more information on these articles and theses.

4.3 DAIDE

From the website (van Hal (2013 (accessed January 27, 2016)):

"Welcome to the Diplomacy AI Centre. Here you will find a wealth of information relating to the development of computer programs designed to play the board game Diplomacy. For those with a theoretical leaning, you'll find academic papers and articles that are relevant to Diplomacy. If you just want to dive in there and play with the state-of-the-art AIs, you'll find them in the downloads section. If you're serious about writing your own AI then there are various tool kits and libraries that do a lot of the legwork for you, leaving you to the interesting parts."

This is an amazing site. DAIDE is a software system, consisting of a number of modules. The DAIDE System is generally used for two purposes:

- To play a game of diplomacy against computer opponents on a single PC
- To play a game of diplomacy against human (and maybe also computer) opponents across the Internet.

The main idea behind the DAIDE system is that of creating your own bots (short for *robot*: computer programs, in this case those that play diplomacy) and getting them to play against other bots and humans!

4.4 Diplomacy Bots

4.4.1 DAIDE and Albert

From the website (van Hal (2016 (accessed January 27, 2016)):

"In January 2002, a group of programmers got together to produce an environment in which several Diplomacy AI's could compete. This environment was named the Diplomacy AI Development Environment (DAIDE)."

"The project has a fully established language syntax and communications protocol."

"To support this project David Norman has created a diplomacy server (AIserver) and a human interface program (AImapper). This allows for games to be played between bots, as well as humans against bots, and finally all humans."

"The AI Server is known to be well suited for human real-time (RT) games, with the added benefit that bots can fill in any missing positions. It works over simple TCP/IP protocol and has allowed players all around the world to easily conduct games."

"Furthermore, if you are at all interested in programming an AI, David Norman has created a C++ Client Framework that makes it very easy to get started and create a bot. All of the basic communications code and game information is readily available in the framework."

"I was not involved in any of the establishment of DAIDE back in 2002. But it is the environment where bots can be made, and best of all compete."

4.4.2 Other Bots

A list of other diplomacy-playing bots can be found at Rose et al. (2014 (accessed January 27, 2016).

4.5 DipTool

From the website (Hagenah (2016 (accessed January 27, 2016)):

"DipTool parses the output of the diplomacy judges (results, lists, histories, summaries, and press) and history files of Keith Schneider. It then adds the games, turns, and messages to a tree and displays the maps. DipTool can:

- read the judge output from files and the clipboard,
- connect to POP3 and IMAP servers to directly get the e-mail messages from the judges, and
- import games from Alain TÃ©sios online mapper www.floc.net.

"You can enter the orders using Drag & Drop or popup menus and send the orders directly to the judges using SMTP. DipTool also has a build in adjudicator so that you can use it during a FTF game."

4.6 JDiplomacy

From the website (Baptiste (2016 (accessed January 27, 2016)):

"JDiplomacy is a multiplayer, online version of the famous strategy game. 1 To 7 players. A Java client connected to a Google Appengine application."

The source code is available for this project.

4.7 Diplomacy Strategy Editor

From the website (Labatut (2016 (accessed January 27, 2016)):

"This standalone editor allows designing diplomacy board game strategies. It allows editing strategy trees with multiple branches."

4.8 webDiplomacy

From the website (Kuliukas (2016 (accessed January 27, 2016)):

"A multiplayer web implementation of the popular turn-based strategy game Diplomacy."

5 Articles

I've done a literature search of academic articles and masters and PhD theses related to the game of Diplomacy. The text in the following sections comprises the abstracts for the given article. The articles are meant to be in roughly chronological order.

5.1 Designing and Building a Negotiating Automated Agent

Negotiations are very important in a multi-agent environment, particularly in an environment where there are conflicts between the agents, and cooperation would be beneficial.

We have developed a general structure for a *Negotiating Automated Agent* that consists of five modules: a Prime Minister, a Ministry of Defence, a Foreign Office, a Headquarters, and Intelligence. These modules are implemented using a dynamic set of local agents belonging to the different modules.

We used this structure to develop a Diplomacy player, *Diplomat*. Playing Diplomacy involves a certain amount of technical skills as in other board games, but the capacity to negotiate, explain, convince, promise, keep promises or break them, is an essential ingredient in good play. *Diplomat* was evaluated and consistently played better than human players. Kraus and Lehmann (1995).

5.2 Automated Negotiation and Decision Making in Multi-agent Environments

This paper presents some of the key techniques for reaching agreements in multi-agent environments. It discusses game-theory and economics based techniques: strategic negotiation, auctions, coalition formation, market-oriented programming and contracting. It also presents logical based mechanisms for argumentations. The focus of the survey is on negotiation of self-interested agents, but several mechanisms for co-operative agents who need to resolve conflicts that arise from conflicting beliefs about different aspects of their environment are also mentioned. For space reasons, we couldn't cover all the relevant works, and the papers that are mentioned only demonstrate the possible approaches. We present some of the properties of the approaches using our own previous work. Kraus (2001).

5.3 Learning a Game Strategy Using Pattern-Weights and Self-Play

This paper demonstrates the use of pattern-weights in order to develop a strategy for an automated player of a non-cooperative version of the game of Diplomacy. Diplomacy is a multi-player, zero-sum and simultaneous move game with imperfect information. Pattern-weights represent stored knowledge of various aspects of a game that are learned through experience. An automated computer player is developed without any initial strategy and is able to learn important strategic aspects of the game through self-play by storing pattern-weights and using temporal difference learning. Shapiro et al. (2003).

5.4 Building Automated Agents for Multi-agent Games: A Case Study on "Diplomacy"

We will be building an autonomous non-cooperative agent for Diplomacy. Diplomacy is a multi-player zero-sum simultaneous move game with imperfect information. In other words, hard. The goal of this project is to research the capabilities of self-learning as a means of exploring the massive strategy search space that is Diplomacy. Our approach involves abstracting the search space into local strategies to help reduce the complexity of movement choice. Rather than choosing particular movements, we focus first on evaluating the importance of a particular region. Once a region has been decided on details of actual movements are determined. Learning is used to help determine the importance of each region on the board. Since every player in Diplomacy will disagree on that importance, an agent is learned for each country in the game. McMahan and Bekris (2004a).

5.5 Learning, Resource Allocation and the Art of War

Recently, multi-agent games have received attention because of their high complexity and close association with real life problems. We are studying multi-agent games by building an automated player for the non-cooperative version of the game "Diplomacy", a multi-player, simultaneous move game with imperfect information. "Diplomacy" is a good test bed for algorithms that have to coordinate multiple agents with non-trivial interactions so as to produce an effective policy in a hostile stochastic environment. We model our problem as a multi-agent Markov decision process where the optimum policy attempts to maximize the reward of controlled provinces. The reward function is learned through self-play by using temporal difference learning. The most important aspect of the learning procedure is the proper selection of features for the distinction of provinces. To effectively explore the search space of possible moves we break up the problem into a series of province-focused matrix games. We then compute the expected probability of success of each strategy for the corresponding province. Finally, we formulate a constrained stochastic policy optimization problem that yields optimal policies among the class of realizable ones given the resource limitations, the learned reward function and the probabilities of success. We have implemented our algorithm on the DAIDE platform and tested the algorithm versus available agents. Although our agent does not yet successfully compete with all the possible AI opponents, the results show the positive prospectives of our approach. We also provide a short discussion on the directions that future work on this problem could follow. McMahan and Bekris (2004b).

5.6 Tactical Coordination in No-Press Diplomacy

While there is a broad theoretic foundation for creating computational players for two-player games, such as Chess, the multi-player domain is not as well explored. We make an attempt to apply a multi-agent approach to a multi-player game with huge search spaces and multiple adversaries, namely no-press Diplomacy. We tested our solution against other available bots in an open competition and show that our solution outperforms its competitors in score while being competitive in speed. Johansson and Håård (2005).

5.7 MARS - A Multi-Agent System Playing Risk

We present a multi-agent architecture for playing the game of Risk which is a multi-player game where the players control armies and try to conquer the world through attacking each others territories. Our solution puts an agent in every territory, and let them negotiate about what actions to prioritize in the phases of placing armies, attack, and fortify. The results of a tournament of 13 participating bots, shows outstanding results for our solution, MARS which ends first or second (out of six participants) in 507 out of 792 matches. Johansson and Olsson (2005).

5.8 On Using Multi-Agent Systems in Playing Board Games

Computer programs able to play different kinds of games (aka bots) is a growing area of interest for the computer game industry as the demand for better skilled computerized opponents increase. We propose a general architecture of a Multi-agent System (Mas) based bot able to play complex board games and show that this solution is able to outperform other bots in two quite different games, namely no-press Diplomacy and Risk. Based on these results, we formulate a hypothesis of the applicability of Mas based bots in the domain of board games and identify the need for future investigations in the area. Johansson (2006).

5.9 Using Multi-agent Potential Fields in Real-time Strategy Games

Bots for Real Time Strategy (RTS) games provide a rich challenge to implement. A bot controls a number of units that may have to navigate in a partially unknown environment, while at the same time search for enemies and coordinate attacks to fight them down. Potential fields is a technique originating from the area of robotics where it is used in controlling the navigation of robots in dynamic environments. Although attempts have been made to transfer the technology to the gaming sector, assumed problems

with efficiency and high costs for implementation have made the industry reluctant to adopt it. We present a Multi-agent Potential Field based bot architecture that is evaluated in a real time strategy game setting and compare it, both in terms of performance, and in terms of softer at- tributes such as configurability with other state-of-the-art solutions. Although our solution did not reach the performance standards of traditional RTS bots in the test, we see great unexploited benefits in using multi-agent potential field based solutions in RTS games. Hagelbäck and Johansson (2008).

5.10 Agent Architecture in Social Games - The Implementation of Subsumption Architecture in Diplomacy

Social games are challenging for AI research because they involve not only the mechanism of game (taking actions, planning, etc), but also the social aspects including communicating, cooperating and reaching agreements. Some social games also require cunning, duplicity, or bad faith. In this paper we present the implementation of a prototype that utilizes the properties of the subsumption architecture to provide an interesting computer opponent in a social board game – Diplomacy. The evaluation result indicates that the subsumption architecture is appropriate for social games. Krzywinski et al. (2008).

5.11 Diplomacy Game: The Testbed

A rich shared-application domain is helping to further current research on negotiation and trust. Fabregues and Sierra (2009a).

5.12 A Testbed for Multi-agent Systems

There is a chronic lack of shared application domains to test the research models and agent architectures on areas like negotiation, argumentation, trust and reputation. In this paper we introduce such a friendly testbed that we used for all such purposes. The testbed is based on the Diplomacy Game due to its lack of random moves and because of the essential role that negotiation and the relationships between the players play in the game. The testbed may also profit from the existence of a community of bot developers and a large number of human players that would provide data for our experiments. We offer the infrastructure and make it freely available to the MAS community. Fabregues and Sierra (2009b).

5.13 DarkBlade: A Program That Plays Diplomacy

Diplomacy is a 7-player game that requires coordination between players in order to achieve victory. Its huge search space makes existing search algorithms useless. In this paper we present DarkBlade, a player designed as a Multi-Agent System that uses potential fields to calculate moves and evaluate board positions. We tested our player against other recent players. Although there are some limitations, the results are promising. Ribeiro et al. (2009).

5.14 Is Human-like and Well Playing Contradictory for Diplomacy Bots?

This paper presents a non-player character (NPC, bot) for the strategy game Diplomacy. The bot is able to communicate with other players and thus shows a human-like behaviour. We investigate how far the playing abilities can be improved without corrupting the human-like behaviour. Is there a trade-off at all or do these skills complement one another? Different versions of the bot are tested against other bots and humans which requires means to automatically measure believability. We derive such a measure after a general approach and apply it for monitoring the believability criterion while improving the playing strength of our bot. Kemmerling et al. (2009).

5.15 Evolving Robust Strategies for an Abstract Real-time Strategy Game

This paper presents an analysis of evolved strate- gies for an abstract real-time strategy (RTS) game. The abstract RTS game used is a turn-based strategy game with properties such as parallel turns and imperfect spatial information. The automated player used to learn strategies uses a progressive refinement planning technique to plan its next immediate turn during the game. We describe two types of spatial tactical coordination which we posit are important in the game and define measures for both. A set of ten strategies evolved in a single environment are compared to a second set of ten strategies evolved across a set of environments. The robustness of all of evolved strategies are assessed when playing each other in each environment. Also, the levels of coordination present in both sets of strategies are measured and compared. We wish to show that evolving across multiple spatial environments is necessary to evolve robustness into our strategies. Keaveney and O'Riordan (2009).

5.16 Influence Graphs: a Technique for Evaluating the State of the World

This paper gives a brief overview of influence maps - an existing method of game state evaluation in strategy board games, explains why this method cannot be applied to state evaluation of certain board games, and proposes a new method for state evaluation of these games - influence graphs. The new method is explained and its use is demonstrated in the game of Diplomacy. The paper finally cites examples of other games and real world applications where influence graphs can be used and looks at possible future improvements to this new method. Utkin (????).

5.17 Automated Negotiations

In this paper we focus on the question of whether an automated agent can proficiently negotiate with human negotiators. To this end we define a proficient automated negotiator as one that can achieve the best possible agreement for itself. This, of course, also depends on the preferences of the other party and thus adds complexity to the design of such an agent. Lin and Kraus (2010).

5.18 DipGame: A Testbed for Multi-Agent Systems

There is a chronic lack of shared application domains to test the research models and agent architectures on areas like negotiation, argumentation, trust and reputation. In this demostration we introduce such a friendly testbed called dipGame that can be used for all such purposes. The testbed is based on the Diplomacy Game due to its lack of random moves and because of the essential role that negotiation and the relationships between the players play in the game. The testbed may also profit from the existence of a community of bot (player software agent) developers and a large number of human players that would provide data for experiments. We offer the infrastructure, including a bot, and make it freely available to the MAS community. Fabregues et al. (2010).

5.19 An Agent Architecture for Simultaneous Bilateral Negotiations

In this paper we introduce an agent architecture that is suitable for joint action plan negotiation among several agents in complex environments and with negotiation time bounds. The architecture is based on a graded BDI model to drive the decisions that the agent makes. The practical reasoning explores the space of possible deals with the help of a genetic algorithm that copes with the potentially sheer amount of joint plans. Beliefs represent the agent's view of the world. Before starting negotiating the agent summarises all what she knows into the five dimensions of the LOGIC negotiation model. This summary is used to decide what to say next and whom to say it to in order to sign deals on joint plans. This work does not yet include experimental results but it is going to be tested using the DipGame testbed. Fabregues and Sierra (2010).

5.20 A Balanced Diplomacy Tournament

We present a design for a seven game tournament of the 7-player board game Diplomacy, in which each player plays each country one time and each pair of players shares a border either 4 or 5 times. It is impossible for each pair of players to share a border the same number of times in such a tournament, and so the tournament presented is the most "balanced" possible in this sense. A similarly balanced tournament can be constructed for a generalized version of the game involving an arbitrary number of countries. The question of whether a balanced tournament can be found for any number of countries and any border configuration is considered. Such tournaments are found for some infinite families, but most cases remain open. Ash et al. (2010).

5.21 RedTNet: A Network Model for Strategy Games

In this work, we develop a simple, graph-based framework, RedTNet, for computational modeling of strategy games and simulations. The framework applies the concept of red teaming as a means by which to explore alternative strategies. We show how the model supports computer-based red teaming in several applications: real-time strategy games and critical infrastructure protection, using an evolutionary algorithm to automatically detect good and often surprising strategies. Hingston et al. (2010).

5.22 A Selective Move Generator for the Game Axis and Allies

We consider the move generation in a modern board game where the set of all the possible moves is too large to be generated. The idea is to provide a set of simple abstract tactics that would generate enough combinations to provide strong opposition. The reduced search space is then traversed using the $\alpha\beta$ search. We also propose a technique that allows us to remove the stochasticity from the search space. The model was tested in a game called Axis and Allies: a modern, turn-based, perfect information, non-deterministic, strategy board game. We first show that a tree search technique based on a restrained set of moves can beat the actual scripted AI engine - E.Z. FODDER. We can conclude from the experiments that searching deeper generates complex manoeuvres which in turn significantly increase the likelihood of victory. St-Pierre et al. (2010).

5.23 Developing Intelligent Bots for the Diplomacy Game

This paper describes the design of an architecture of a bot capable of playing the Diplomacy game, to be used within the dip framework - a testbed for multi-agent negotiations. The proposed SillyNegoBot, is an extension of the SillyBot. It is designed to be used in the level-1 negotiations (as defined within the dip framework) taking place during the Diplomacy game. Polberg et al. (2011).

5.24 DipGame: A Challenging Negotiation Testbed

There is a chronic lack of shared application domains to test advanced research models and agent negotiation architectures in Multi-agent Systems. In this paper we introduce a friendly testbed for that purpose. The testbed is based on The Diplomacy Game where negotiation and the relationships between players play an essential role. The testbed profits from the existence of a large community of human players that know the game and can easily provide data for experiments. We explain the infrastructure in the paper and make it freely available to the AI community. Fabregues and Sierra (2011).

5.25 DipTools: Experimental Data Visualization Tool for the DipGame Testbed

DipGame is a testbed for negotiation. It permits to test negotiation algorithms, even if enriched with argumentation, trust or reputation techniques. It is very appropriate to run experiments that mix humans and agents. In this demonstration we introduce a tool to visualise data obtained from DipGame experiments. Fabregues et al. (2011).

5.26 Nested Look-Ahead Evolutionary Algorithm Based Planning for a Believable Diplomacy Bot

With regard to literature, improved estimations for the number of possible moves and placements are provided, showing that the complexity of Diplomacy is enormous, making it a good candidate for machine learning and evolutionary learning techniques. To enhance the playing strength of an existing Diplomacy bot and alleviate the distance to the presumed best current bot, a look-ahead planning component based on nested evolutionary algorithms, is then implanted into an already existing bot. The experimental investigation shows that the resulting bot is significantly improved. Kemmerling et al. (2011).

5.27 Modeling Social Preferences in Multi-Player Games

Game-tree search algorithms have contributed greatly to the success of computerized players in two-player extensive-form games. In multi-player games there has been less success, partly because of the difficulty of recognizing and reasoning about the inter-player relationships that often develop and change during human game-play. Simplifying assumptions (e.g., assuming each player selfishly aims to maximize its own payoff) have not worked very well in practice.

We describe a new algorithm for multi-player games, Socially-oriented Search (SOS), that incorporates ideas from Social Value Orientation theory from social psychology. We provide a theoretical study of the algorithm, and a method for recognizing and reasoning about relationships as they develop and change during a game. Our empirical evaluations of SOS in the strategic board game Quoridor show it to be significantly more effective against players with dynamic interrelationships than the current state-of-the-art algorithms. Wilson et al. (2011).

5.28 Running Experiments on DipGame Testbed

DipGame is a testbed for MAS negotiation involving humans. It is very appropriate to run experiments that mix humans and agents. In this demonstration we introduce an application to facilitate the execution of experiments on several machines and with a friendly graphical user interface. Fabregues et al. (2012).

5.29 A Comparison of Diplomacy Gameboard Graph Search Algorithms

The boardgame Diplomacy has been used as a testbed for multiagent systems almost since the time of its introduction in 1959. The reason is that the game presents a number of interesting challenges to artificial intelligence researchers: a state space that is too large to be tackled by brute forces searches, imperfect information due to simultaneous movement, no random elements, and non-binding negotiations between the seven players. This paper looks at just one aspect of creating an agent for playing Diplomacy finding the fewest number of moves to achieve a victory in the game, if the player was unopposed. This planning function forms the basis for a more sophisticated move planner that also takes into account the game state an the other players. Three search algorithms are compared to determine which is the most effective (in terms of the number of map nodes expanded during the search). Stormont and Allan (2012).

5.30 HANA: A Human-Aware Negotiation Architecture

In this paper we propose HANA, a software architecture for agents that need to bilaterally negotiate joint plans of action in realistic scenarios. These negotiations may involve humans and are repeated along time. The architecture is based on a BDI model that represents the uncertainty on the environment as graded beliefs, desires and intentions. The architecture is modular and can easily be extended by incorporating different models (e.g. trust, intimacy, personality, normative, ...) that update the set of beliefs, desires or intentions. The architecture is dynamic as it monitors the environment and updates the beliefs accordingly. We introduce an innovative search & negotiation method that facilitates HANA agents to cope with huge spaces of joint plans. This method implements an anytime search algorithm that generates partial plans to feed the negotiation process. At the same time the negotiation guides the search towards joint plans that are more likely to be accepted. Fabregues and Sierra (In press).

5.31 Abridgement of HANA: a Human-Aware Negotiation Architecture

HANA is an agent architecture suitable for multiple bilateral negotiations in realistic problems involving humans. The architecture deals with pre- negotiation and provides a new search and negotiation technique where search and negotiation go hand in hand: the former providing offers to propose, and the later providing commitments for pruning the search space, and information for fine-tuning the evaluation of offers. The architecture represents graded beliefs, dynamic desires and general intentions. It can be extended incorporating new behavioural models that can enrich the negotiation strategy with new information. Fabregues and Sierra (2012).

5.32 Branch and Bound for Negotiations in Large Agreement Spaces

We introduce a new multi-agent negotiation algorithm for domains where the space of joint plans is intractably large, utility is non-linear and time is limited, so an exhaustive search for the best solution is not feasible. The algorithm is called NB^3 and applies Branch & Bound to search for good plans to negotiate on. The search for good solutions and the negotiation process happen simultaneously and strongly influence each other. To analyse the performance of the algorithm we present a new problem called the Negotiating Salesmen Problem. We have performed a series of experiments with an implementation of NB^3 from which we conclude that it is able to decrease the costs of the agents significantly, that it outperforms random search and that it scales well with the complexity of the problem. De Jonge and Sierra (2012).

5.33 Comparing Co-evolution, Genetic Algorithms, and Hill-Climbers for Finding Real-Time Strategy Game Plans

This paper evaluates a co-evolutionary genetic algorithm's performance at generating competitive strategies in the initial stages of real-time strategy games. Specifically, we evaluate co-evolution's performance against an exhaustive search of all possible build orders. Three hand coded strategies out- side this exhaustive list provide a quantitative baseline for comparison with other strategy search algorithms. Earlier work had shown that a bit-setting hill-climber only finds the best strategies six percent of the time but takes significantly less time compared to a genetic algorithm that routinely finds the best strategies. Our results here show that co-evolved strategies win or tie against hill-climber and genetic algorithm strategies eighty percent of the time but routinely lose to the three hand coded baselines. This work in- forms our research on improving co-evolutionary approaches to real-time strategy game player design. Ballinger and Louis (2013a).

5.34 Finding Robust Strategies to Defeat Specific Opponents Using Case-Injected Co-evolution

Finding robust solutions that are also capable of beating specific opponents presents a challenging problem. This paper investigates solving this problem by using case-injection with a co-evolutionary algorithm. Specifically, we recorded winning strategies used by a human player against a co-evolved strategy and then injected the player's strategies into the co-evolutionary teach-set. We compare the strategies produced by case-injected co-evolution to strategies produced by a genetic algorithm that only evaluated against the player's strategies. In this paper, our results show that genetic algorithms do not work well against sufficiently difficult opponents. However, co-evolution eventually learns to defeat these opponents by first bootstrapping strategies that work well in general, which drives the population closer to strategies that can defeat the challenging opponent. This work informs our research on finding robust real-time strategy game players that also defeat specific opponents. Ballinger and Louis (2013b).

5.35 A Review of Computational Intelligence in RTS Games

Real-time strategy games offer a wide variety of fundamental AI research challenges. Most of these challenges have applications outside the game domain. This paper provides a review on computational intelligence in real-time strategy games (RTS). It starts with challenges in real-time strategy games, then it reviews different tasks to overcome this challenges. Later, it describes the techniques used to solve this

challenges and it makes a relationship between techniques and tasks. Finally, it presents a set of different frameworks used as test-beds for the techniques employed. This paper is intended to be a starting point for future researchers on this topic. Lara-Cabrera et al. (2013).

5.36 Designing Competitive Bots for a Real Time Strategy Game using Genetic Programming

The design of the Artificial Intelligence (AI) engine for an autonomous agent (bot) in a game is always a difficult task mainly done by an expert human player, who has to transform his/her knowledge into a behavioural engine. This paper presents an approach for conducting this task by means of Genetic Programming (GP) application. This algorithm is applied to design decision trees to be used as bots AI in 1 versus 1 battles inside the RTS game Planet Wars. Using this method it is possible to create rule-based systems defining decisions and actions, in an automatic way, completely different from a human designer doing them from scratch. These rules will be optimised along the algorithm run, considering the bot's performance during evaluation matches. As GP can generate and evolve behavioural rules not taken into account by an expert, the obtained bots could perform better than human-defined ones. Due to the difficulties when applying Computational Intelligence techniques in the video games scope, such as noise factor in the evaluation functions, three different fitness approaches have been implemented and tested in this work. Two of them try to minimize this factor by considering additional dynamic information about the evaluation matches, rather than just the final result (the winner), as the other function does. In order to prove them, the best obtained agents have been compared with a previous bot, created by an expert player (from scratch) and then optimised by means of Genetic Algorithms. The experiments show that the three used fitness functions generate bots that outperform the optimized human-defined one, being the area-based fitness function the one that produces better results. Fernández-Ares et al. (2014).

5.37 Monte Carlo Tree Search Variants for Simultaneous Move Games

Monte Carlo Tree Search (MCTS) is a widely-used technique for game-tree search in sequential turn-based games. The extension to simultaneous move games, where all players choose moves simultaneously each turn, is non-trivial due to the complexity of this class of games. In this paper, we describe simultaneous move MCTS and analyse its application in a set of nine disparate simultaneous move games. We use several possible variants, Decoupled UCT, Sequential UCT, Exp3, and Regret Matching. These variants include both deterministic and stochastic selection strategies and we characterize the game-play performance of each one. The results indicate that the relative performance of each variant depends strongly on the game and the opponent, and that parameter tuning can also not be as straightforward as the purely sequential case. Overall, Decoupled UCT performs best despite its theoretical shortcomings. Tak et al. (2014).

5.38 Linguistic Harbingers of Betrayal: A Case Study on an On-line Strategy Game

Interpersonal relations are fickle, with close friendships often dissolving into enmity. In this work, we explore linguis- tic cues that presage such transitions by studying dyadic interactions in an on- line strategy game where players form alliances and break those alliances through betrayal. We characterize friendships that are unlikely to last and examine temporal patterns that foretell betrayal.

We reveal that subtle signs of imminent betrayal are encoded in the conversational patterns of the dyad, even if the victim is not aware of the relationship's fate. In particular, we find that lasting friend- ships exhibit a form of balance that manifests itself through language. In contrast, sudden changes in the balance of certain conversational attributes - such as positive sentiment, politeness, or focus on future planning - signal impending betrayal. Niculae et al. (2015).

5.39 DipBlue: A Diplomacy Agent with Strategic and Trust Reasoning

Diplomacy is a multi-player strategic and zero-sum game, free of random factors, and allowing negotiation among players. The majoriry of existing artificial players (bots) for Diplomacy do not exploit the strategic opportunities enabled by negotiation, instead trying to decide theor moves through solution search and the use of complex heuristics. We present DipBlue, an approach to the development of an artificial player

that uses negotiation in order to gain advantage over its opponents, through the use of peace treaties, formation of alliances and suggestion of actions to allies. A simple trust assessment approach is used a means to detect and react to potential betrayals by allied players. DipBlue was built to work with DipGame, a multi-agent systems testbed for Diplomacy, and has been tested with other players of the same platform and variations of itself. Experimental results show that the use of negotiation increases the performance of bots involved in alliances, when full trust is assumed. In the presence of betrayals, being able to perform trust reasoning is an effective approach to reduce their impact. Ferreira et al. (2015a).

5.40 Strategic Negotiation and Trust in Diplomacy - The DipBlue Approach

The study of games in Artificial Intelligence has a long tradition. Game playing has been a fertile environment for the development of novel approaches to build intelligent programs. Multi-agent systems (MAS), in particular, are a very useful paradigm in this regard, not only because multi-player games can be addressed using this technology, but most importantly because social aspects of agenthood that have been studied for years by MAS researchers can be applied in the attractive and controlled scenarios that games convey. Diplomacy is a multi-player strategic zero-sum board game, including as main research challenges an enormous search tree, the difficulty of determining the real strength of a position, and the accommodation of negotiation among players. Negotiation abilities bring along other social aspects, such as the need to perform trust reasoning in order to win the game. The majority of existing artificial players (bots) for Diplomacy do not exploit the strategic opportunities enabled by negotiation, focusing instead on search and heuristic approaches. This paper describes the development of DipBlue, an artificial player that uses negotiation in order to gain advantage over its opponents, through the use of peace treaties, formation of alliances and suggestion of actions to allies. A simple trust assessment approach is used as a means to detect and react to potential betrayals by allied players. DipBlue was built to work with DipGame, a MAS testbed for Diplomacy, and has been tested with other players of the same platform and variations of itself. Experimental results show that the use of negotiation increases the performance of bots involved in alliances, when full trust is assumed. In the presence of betrayals, being able to perform trust reasoning is an effective approach to reduce their impact. Ferreira et al. (2015b).

5.41 Learning About the Opponent in Automated Bilateral Negotiation: a Comprehensive Survey of Opponent Modeling Techniques

A negotiation between agents is typically an incomplete information game, where the agents initially do not know their opponent's preferences or strategy. This poses a challenge, as efficient and effective negotiation requires the bidding agent to take the other's wishes and future behaviour into account when deciding on a proposal. Therefore, in order to reach better and earlier agreements, an agent can apply learning techniques to construct a model of the opponent. There is a mature body of research in negotiation that focuses on modeling the opponent, but there exists no recent survey of commonly used opponent modeling techniques. This work aims to advance and integrate knowledge of the field by providing a comprehensive survey of currently existing opponent models in a bilateral negotiation set- ting. We discuss all possible ways opponent modeling has been used to benefit agents so far, and we introduce a taxonomy of currently existing opponent models based on their under- lying learning techniques. We also present techniques to measure the success of opponent models and provide guidelines for deciding on the appropriate performance measures for every opponent model type in our taxonomy. Baarslag et al. (2015).

6 Theses

The text in the following sections comprises the abstracts for the given thesis.

6.1 Diplomacy AI

Diplomacy involves seven players negotiating and fighting across Europe as they attempt to conquer the continent. It is an entirely deterministic game, but players move simultaneously producing a game tree too large to be searched by normal methods.

Traditional search methods struggle with imperfect information, and fail with simultaneous moves.

This report describes a program that plays a simplified five player no-press variant of the game at a novice level, being moderate tactically but weak at the strategic level.

It details a simplified 2 player and 5 player version of Diplomacy, which the program was tested against, and examines in depth the moves made compared to what moves would be expected from a human player. The program does not implement fleets. Ritchie (2003).

6.2 Multi Agent Diplomacy

While there is a broad theoretic foundation for creating artificial intelligence based solutions for two-player games, such as Chess, the multi-player domain is not as well explored and artificial intelligence solutions for multi-player games is often flawed. This report is an attempt to apply a multi-agent approach to a multi-player game, and use distributed problem solving to create viable plans in an environment of huge search spaces and multiple adversaries. An automated player (bot) for the game Diplomacy was created using distributed methodologies, and tested against other existing bots. Although the bot developed proved flawed, it did well in key aspects of the game. Häärd (2004).

6.3 Creating a Diplomat

Many programs have been written to play two-player games, but few for multi-player games in which negotiation becomes important. Diplomacy is a popular board game in which players assume the roles of the major protagonists of world war one. It is a game of negotiations; alliances, promises kept and promises broken. In order to survive a player needs help from others. Knowing whom to trust, when to trust them, what to promise, and when to promise it is at the heart of the game. In this, it provides an exciting and unique environment for negotiation, and one which can generalise to an enormous number of applications.

This report describes a novel approach to creating an automated Diplomacy player, using auctions as a means of negotiation, and simple algorithms to help a Diplomacy player determine if and when to lie. Shaheed (2004).

6.4 Automated Negotiation in the Game Of Diplomacy

Diplomacy is a strategic board game with relatively simple tactics but with a rich negotiation element between players. The Diplomacy AI Development Environment provides a framework within which automated Diplomacy players can be developed. We create a simple negotiating Diplomacy player within this framework and use it to show that negotiating players outperform non-negotiating ones. Webb et al. (2008).

6.5 Mind Game: (Axis and Allies)

Games are present in every society and play an important role. In terms of research, games have significant potential when compared to other real-world applications, because it is easy to control most of the variables under study. Evidently, almost every game today needs strong artificial intelligence (AI) to provide the player with a suitably challenging and enjoyable experience, from board game like checkers to Real Time Strategy computer games (RTS) such as Creative Assembly's Empire: Total War. However, for most recent games, creating an AI that does not have to cheat to give a human player a strong adversity is a challenge yet to be overcome.

In this regard, this project is about the design of a "smarter" AI for an open source version of a game called Axis and Allies. This game possesses a lot of interesting characteristics that has yet to be studied in the literature. First, there are multiple decisions to take per turn. Second, albeit perfect information zero sum game, there are multiple simultaneous choices to make at some point of the game.

A formalisation of the research problem is provided and links between the game (Axis and Allies) and other games is studied. Following this, a background survey is executed under two main sub-sections;

namely, "analysis technique" and "utility function". Different techniques to analyse the moves are discussed and a brief survey of the technique most likely to be chosen is done. A review on the utility function is offered.

Subsequently, a description of the experimental protocol, hypothesis and the variables are discussed. After, the design and implementation are described along with the key decisions made throughout these sections. In the evaluation chapter, it is statistically proven that the AI created in this project (davidlsAI) is better than the actual one (oldAI). An ANOVA is presented to improve the evaluation function. A student test revealed that the depth of the search increases the strength of davidlsAI. Even if the split attack considered by davidlsAI slows him down, this attack is particularly effective against human players. The future work section discuss on the next step to improve davidlsAI. Mainly, the objective is to generalise davidlsAI to allow it to play on any map and to implement Monte Carlo Upper Confidence bound applied for Tree (MCTS-UCT). St-Pierre (2009).

6.6 Automated Negotiation in the Game of Diplomacy

Diplomacy is a strategy game involving seven players, each representing one of the Great Powers of Europe just before the beginning of World War I. Players command military units in order to fight for control of key supply centres and ultimately all of Europe.

The distinguishing factor that separates Diplomacy from any other game is the aspect of negotiation. A player must forge alliances with opponents to make progress in the game, and at the same time, work towards an opening where a stab in the back of an ally would propel our player towards victory. The assessment of an opponents trustworthiness is therefore fundamental to success in Diplomacy.

In this report, we will describe our approach to building an artificially intelligent diplomat that is able to negotiate and has a concept of trust. Lim et al. (2009).

6.7 Optimizing a Diplomacy Bot Using Genetic Algorithms

We give a short explanation of the game Diplomacy and an overview of the currently existing projects regarding artificial intelligence applied to Diplomacy. Then we make an analysis of the DumbBot, which is one of the simplest existing bots that can play Diplomacy at a reasonable level; we analyze its strategy and its parameters. Our goal is to optimize the parameters of the DumbBot, and therefore we repeat the basics of optimizing functions of many variables and show when and why one can use a genetic algorithm to do this. Also we propose a new alternative variant of the genetic algorithm that might lead to better results in our specific optimization problem. This alternative increases the exploration of lower order schemas, at the expense of a decrease of the exploitation of higher order schemas. We argue that this decrease is not a loss, because it is impossible for us to explore higher order schemas sufficiently in the first place. Finally we optimize the parameters using three different methods: hill climbing, a standard genetic algorithm and our alternative genetic algorithm. De Jonge (2010).

6.8 Facing the Challenge of Automated Negotiation with Humans

The research field of negotiation has been studied from many different perspectives, among them: game theory, psychology, business, neuro-economics, and psycho-pharmacology. The computational study of negotiations is denoted by automated negotiation. Most works on automated negotiation assume rational agents and static negotiation problems. However, humans are rationally bounded, and their negotiations are usually dynamic. It is often impossible to explore the complete negotiation space due to time limitations and the dynamics of the problem. By the time that an optimal solution is found, the solution is not optimal any more. Currently available testbeds on automated negotiation share the same shortcomings. Those testbeds that intend to involve humans in experiments assume that humans are rational, or are designed over artificial domains that require intense instruction of experiment participants. This thesis contributes to automated negotiation designing an agent architecture suitable to negotiate with humans, and a testbed that allows for an easy participation of humans in experiments.

We denote the agent architecture by HANA. It allows multiple bilateral negotiations about actions, and deals with pre-negotiation looking for good enough sets of actions and orders. It is a modular architec-

ture based on an ecological model of rationality. The mental state of the agent is represented as graded beliefs, dynamic desires and general intentions. We use a novel search and negotiation technique where search and negotiation go hand in hand: the former providing orders to propose, and the later providing commitments for pruning the search space, and information for re-tuning the evaluation of orders. Several negotiation strategies are provided that can be dynamically combined. The architecture is extensible, allowing the incorporation of new behavioural models.

The name of the testbed is DipGame. It is based on a popular board game where being a skilled negotiator is crucial for winning. DipGame allows the study of relationships, emotions, and coalitions that take place during successive negotiations involving humans. There are many research opportunities in different topics all of them connected to negotiation. The study of a topic or another is selected constraining the negotiation language used during the game. The testbed provides a framework for agent development, and several nego- tiation utilities for the representation of messages and communication among agents. It assists the execution of experiments using a graphical software application called GameManager. It facilitates the inclusion of humans with another application called ChatApp. Moreover, the analysis of results is supported by a different application called DipTools. This thesis is completed with a formal definition of the problem, a formal specification of the game, and the application of the work to the game industry. Fabregues (2012a).

6.9 Multi-Agent Potential Field Based Architectures for Real-Time Strategy Game Bots

Real-Time Strategy (RTS) is a sub-genre of strategy games which is running in real-time, typically in a war setting. The player uses workers to gather resources, which in turn are used for creating new buildings, training combat units, build upgrades and do research. The game is won when all buildings of the opponent(s) have been destroyed. The numerous tasks that need to be handled in real-time can be very demanding for a player. Computer players (bots) for RTS games face the same challenges, and also have to navigate units in highly dynamic game worlds and deal with other low-level tasks such as attacking enemy units within fire range.

This thesis is a compilation grouped into three parts. The first part deals with navigation in dynamic game worlds which can be a complex and resource demanding task. Typically it is solved by using path-finding algorithms. We investigate an alternative approach based on Artificial Potential Fields and show how an APF based navigation system can be used without any need of path-finding algorithms.

In RTS games players usually have a limited visibility of the game world, known as Fog of War. Bots on the other hand often have complete visibility to aid the AI in making better decisions. We show that a Multi-Agent PF based bot with limited visibility can match and even surpass bots with complete visibility in some RTS scenarios. We also show how the bot can be extended and used in a full RTS scenario with base building and unit construction. In the next section we propose a flexible and expandable RTS game architecture that can be modified at several levels of abstraction to test different techniques and ideas. The proposed architecture is implemented in the famous RTS game StarCraft, and we show how the high-level architecture goals of flexibility and expandability can be achieved.

In the last section we present two studies related to game-play experience in RTS games. In games players usually have to select a static difficulty level when playing against computer opponents. In the first study we use a bot that during runtime can adapt the difficulty level depending on the skills of the opponent, and study how it affects the perceived enjoyment and variation in playing against the bot.

To create bots that are interesting and challenging for human players a goal is often to create bots that play more human-like. In the second study we asked participants to watch replays of recorded RTS games between bots and human players. The participants were asked to guess and motivate if a player was controlled by a human or a bot. This information was then used to identify human-like and bot-like characteristics for RTS game players. Hagelbäck (2012).

6.10 Study on Intelligent Negotiation using Genetic Network Programming Evolving based on Agreement Conditions

...the aims of this research are to propose, design and develop the advanced, automated and intelligent negotiation solutions considering all kinds of possible practical real world negotiation situations. Toward

these objectives, in the first step, a basic bilateral negotiation system has been proposed by Genetic Network Programming, where a buyer and a seller have been participated in the negotiation. The main objective of this bilateral negotiation system is to examine the intelligent decision making ability of agents preciously and concretely in the negotiation through GNP evolution without being influenced by any other values, such as coefficient values and private values that used in the next chapters. In addition, to make the negotiation system automated and intelligent, the effectiveness of GNP can be confirmed in the negotiation. Therefore, the negotiation system has been kept simple, where objective of the buyer and seller is to obtain higher fitness for their own in the negotiations. In addition, a Negotiation Protocol has been devised aiming to develop a set of rules for agents' behavior during the negotiation. Hossain et al. (2013).

6.11 Co-evolutionary Approaches to Generating Robust Build-Orders for Real-time Strategy Games

We aim to find winning build-orders for real-time strategy games. Real-time strategy games provide a variety of challenges, from short-term control to longer-term planning. We focus on a longer-term planning problem: which units to build and in what order to produce the units so a player successfully defeats the opponent. Plans which address unit construction scheduling problems in real-time strategy games are called build-orders. A robust build-order defeats many opponents, while a strong build-order defeats opponents quickly. However, no single build-order defeats all other build-orders, and build-orders that defeat many opponents may still lose against a specific opponent. Other researchers have only investigated generating build-orders that defeat a specific opponent, rather than finding robust, strong build-orders. Additionally, previous research has not applied co-evolutionary algorithms towards generating build-orders. In contrast, our research has three main contributions towards finding robust, strong build-orders. First, we apply a co-evolutionary algorithm towards finding robust build-orders. Compared to exhaustive search, a genetic algorithm finds the strongest build-orders while a co-evolutionary algorithm finds more robust build-orders. Second, we show that case-injection enables co-evolution to learn from specific opponents while maintaining robustness. Build-orders produced with co-evolution and case-injection learn to defeat or play like the injected build-orders. Third, we show that co-evolved build-orders benefit from a representation which includes branches and loops. Co-evolution will utilize multiple branches and loops to create build-orders that are stronger than build-orders without loops and branches. We believe this work provides evidence that co-evolutionary algorithms may be a viable approach to creating robust, strong build-orders for real-time Strategy games. Ballinger (2014).

6.12 On-line Build-Order Optimisation for Real-time Strategy Agents Using Multi-Objective Evolutionary Algorithms

This investigation introduces a novel approach for on-line build-order optimisation in real-time strategy (RTS) games. The goal of our research is to develop an artificial intelligence (AI) RTS planning agent for military critical decision-making education with the ability to perform at an expert human level, as well as to assess a players critical decision-making ability or skill-level. Build-order optimisation is modeled as a multi-objective problem (MOP), and solutions are generated utilizing a multi-objective evolutionary algorithm (MOEA) that provides a set of good build-orders to a RTS planning agent. We define three research objectives: (1) Design, implement and validate a capability to determine the skill-level of a RTS player. (2) Design, implement and validate a strategic planning tool that produces near expert level build-orders which are an ordered sequence of actions a player can issue to achieve a goal, and (3) Integrate the strategic planning tool into our existing RTS agent framework and an RTS game engine. The skill-level metric we selected provides an original and needed method of evaluating a RTS players skill-level during game play. This metric is a high-level description of how quickly a player executes a strategy versus known players executing the same strategy. Our strategic planning tool combines a game simulator and an MOEA to produce a set of diverse and good build-orders for an RTS agent. Through the integration of case-base reasoning (CBR), planning goals are derived and expert build-orders are injected into a MOEA population. The MOEA then produces a diverse and approximate Pareto front that is integrated into our AI RTS agent framework. Thus, the planning tool provides an innovative on-line approach for strategic planning in RTS games. Experimentation via the Spring Engine Balanced Annihilation game reveals that the strategic planner is able to discover build-orders that are better than an expert scripted agent and thus achieve faster strategy execution times. Blackford (2014).

6.13 Negotiations Over Large Agreement Spaces

In this thesis we investigate negotiation algorithms for domains with non-linear utility functions and where the space of possible agreements is so large that the application of exhaustive search is impossible. Furthermore, we explore the relationship between the fields of Automated Negotiations, Game Theory, Electronic Institutions, and Constraint Optimization.

We present three case studies with increasing complexity. Firstly, we introduce an automated negotiator based on Genetic Algorithms, which is applied to a domain where the set of possible agreements is explicitly given as a vector space and, although the utility functions are non-linear, the utility value of any given deal can be calculated quickly by solving a linear equation. Secondly, we introduce a general purpose negotiation algorithm called NB^3, which is based on Branch & Bound. We apply this to a new negotiation test case in which the value of any given deal can only be determined by solving an NP-hard problem. Our third case involves the game of Diplomacy, which is even harder than the previous test cases, because a given deal usually does not entirely fix the agent's possible actions. The utility obtained by an agent thus also depends on the actions it performs after making the deal. Moreover, its utility also depends on the actions chosen by the other agents, so one needs to take Game Theoretical considerations into account. We argue that in this Game Theoretical model there no longer exists a satisfactory definition of a reservation value, unlike the models commonly used in classical bargaining theory.

Furthermore, we argue that negotiations require a mechanism, known as an Electronic Institution, to ensure that agreements are obeyed. One framework for the development of Electronic Institutions is EIDE and we introduce a new extension to EIDE that provides a user interface so that humans can interact within such Electronic Institutions. Moreover, we argue that in the future it should be possible for humans and agents to negotiate which protocols to follow in an Electronic Institution. This could be especially useful for the development of a new kind of social network in which the users can set the rules for their own private communities. Finally, we argue that the EIDE framework is too complicated to be used by average people who do not have the technical skills of a computer scientist. We therefore introduce a new language for the definition of protocols, which is very similar to natural language so that it can be used and understood by anyone. Jonge et al. (2015).

7 Books

7.1 The Game of Diplomacy, by Richard Sharp

Richard Sharp, Sharp (1978). One review of this book states: "This is probably the best, most enlightening book on Diplomacy that I've ever read. Any time I flip to it to reference something - particularly the excellent guide to opening moves - I end up re-reading the entire thing. Just a very well written, amazingly entertaining guide to the world's finest board game. Trumps even Mr. Sharps writings in the classic zines of the 1970s-1990s - a true classic, and available online at Sharp (2016 (accessed January 27, 2016).

7.2 Diplomacy: Prima's Official Strategy Guides, by Rex Martin and Michael Knight

Martin and Knight (1999). From Amazon.com: "General strategies for advanced and beginning players. Comprehensive coverage of each Great Power. Sneaky opening, midgame, and endgame moves. In-depth strategies and sample games. Gain negotiation tactics through a psychological study of the multiplayer game. Strategy for variations of the game. Over 40 years of insight by masters of the game."

7.3 The Art of Correspondence in the Game of Diplomacy, by Conor Kostick

Kostick (2015). One review stated: "This book is a great read. Great if you play Diplomacy, but also for an insight into the art of persuasion generally. I didn't expect such a book to be a page turner - but it is. Loved it."

7.4 Calhamer on Diplomacy: The Boardgame "Diplomacy" and Diplomatic History, by Allan Calhamer

Written by the great man himself. Impossible to get hold of a copy.

Calhamer (1999). A reviewer of this book stated: "The author of this book, Allan Calhamer, is the game's inventor. He devised the prototype during his student days at Harvard and refined it through playtesting over a period of many years. Now he has authored a book in which he relates the game to its historical foundations, explaining in wonderful depth where the game mirrors history and where it diverges. He offers a wealth of ideas for successful play, but it is of special interest that the game's inherent flexibility is such that these 'suggestions' are only several of nearly infinite possibilities, guidelines rather than strictures; the game lends itself so well to development of individual styles of play that it would be impossible to offer anything more concrete, and that fact is one of the greatest features of the game! In addition Mr. Calhamer covers in summary form the actual events of the replicated period, and while this is by no means a history text in any sense, it does offer the spark for further exploration for anyone interested in pursuing the matter. Calhamer also draws parallels and divergences with modern diplomatic and military situations, again inspiring further investigation for those so inclined. The book derives from a lecture Calhamer delivered in Japan in 1997, and this fact gives rise to its only serious weakness: Frequently it is clear that the text has not been fully edited into book form, and often comes across as the mere expansion of lecture notes. It needs to be turned more fully into prose. This does not detract in any way from the informational value, but it does make for a number of awkward transitions and seemingly random points scattered into the body of the work. Still, for anyone interested in the game, it is absolutely essential. And for anyone with interest in the history of 20th century diplomacy and international relations, it is a fascinating and wide-ranging overview and an excellent place to begin one's in-depth approach."

8 Variants

The Variant Bank, Agar (2016 (accessed January 27, 2016), provides the most comprehensive list of Diplomacy variants. Some even made it to manufacture, like *Hundred*, for example.

9 My Economic Variant

I came up with my own variant many years ago. I can't remember much about it now: I'll have to give it some thought in the near future.

What I do remember, though, is that it had an economic slant. Essentially, each province (not just the supply centres) earned income at the end of each year. Let's call the unit of currency the *Stevo*. Maybe different provinces earned different amounts of Stevos, I can't remember. But provinces earned Stevos. And to move pieces on the board cost Stevos. As did being at war with another country (what *being at war* means I can't remember).

So the idea was that at the beginning of the game, you wouldn't want to do anything militarily, as you needed to save Stevos to spend later in the game attacking someone. And if you ran out of Stevos, you would automatically be placed at peace with everyone else. You then couldn't move any pieces until you could pay do do so.

Winning the game had more to do with Stevos than supply centres, I recall. So theoretically you could win the game by doing nothing throughout!

It needs a bit more thought, as you can see!

10 Listed Articles

Year	Author	Title	Cited By Checked?	Section
1995	Kraus	Designing and building...	✓	5.1
2001	Kraus	Automated Negotiation and...	✓	5.2
2003	Shapiro	Learning a game strategy...	✓	5.3
2004	McMahan	Building automated agents...	✓	5.4
2004	McMahan	Learning, resource allocation...	✓	5.5
2005	Johansson	Tactical coordination...	✓	5.6
2005	Johansson	Mars - a multiagent...	✓	5.7
2006	Johansson	On using multi-agent systems...	✓	5.8
2008	Hagelbäck	UsingMulti-agent potential...	✓	5.9
2008	Krzywinski	Agent architecture...	✓	5.10
2009	Fabregues	Diplomacy game...	✓	5.11
2009	Fabregues	A testbed for multi-agent...	✓	5.12
2009	Ribeiro	DarkBlade: A program...	✓	5.13
2009	Kemmerling	Is human-like and...	✓	5.14
2009	Keaveney	Evolving robust strategies...	✓	5.15
2009	Utkin	Influence graphs...	✓	5.16
2010	Lin	Automated Negotiations	✓	5.17
2010	Fabregues	DipGame: A testbed...	✓	5.18
2010	Fabregues	An agent architecture...	✓	5.19
2010	Ash	A balanced diplomacy...	✓	5.20
2010	Hingston	RedTNet: A network model...	✓	5.21
2010	St-Pierre	A selective move generator...	✓	5.22
2011	Polberg	Developing intelligent...	✓	5.23
2011	Fabregues	DipGame: A challenging...	✓	5.24
2011	Fabregues	DipTools: Experimental data...	✓	5.25
2011	Kemmerling	Nested look-ahead...	✓	5.26
2011	Wilson	Modeling social preferences...	✓	5.27
2012	Fabregues	Running experiments...	✓	5.28
2012	Stormont	A comparison of diplomacy...	✓	5.29
2012	Fabregues	HANA: Human aware...	✓	5.30
2012	Fabregues	Abridgement of HANA...	✓	5.31
2012	deJonge	Branch and bound...	✓	5.32
2013	Ballinger	Comparing coevolution...	✓	5.33
2013	Ballinger	Finding robust strategies...	✓	5.34
2013	Lara-Cabrera	A review of computational...	✓	5.35
2014	Fernández-Ares	Designing competitive bots...	✓	5.36
2014	Tak	Monte Carlo...	✓	5.37
2015	Niculae	Linguistic harbingers...	✓	5.38
2015	Ferreira	DipBlue: A diplomacy agent...	✓	5.39
2015	Ferreira	Strategic Negotiation...	✓	5.40
2015	Baarslag	Learning about the opponent...	✓	5.41

Table 1: Listed Articles

11 Listed Theses

Year	Author	Title	Cited By Checked?	Section
2003	Richie	Diplomacy AI	✓	6.1
2004	Häärd	Multi-agent diplomacy	✓	6.2
2004	Shaheed	Creating a diplomat	✓	6.3
2008	Webb	Automated negotiation...	✓	6.4
2009	St-Pierre	Mind Games...	✓	6.5
2009	Lim	Automated negotiation...	✓	6.6
2010	de Jonge	Optimising a diplomacy bot...	✓	6.7
2012	Fabregues	Facing the challenge...	✓	6.8
2012	Hagelbäck	Multi-agent potential field.....	✓	6.9
2013	Hossain	Study on intelligent.....	✓	6.10
2014	Ballinger	Coevolutionary approaches...	✓	6.11
2014	Blackford	Online build order optimisation...	✓	6.12
2015	de Jonge	Negotiations over large...	✓	6.13

Table 2: Listed Theses

References

Agar, S. (2016 (accessed January 27, 2016)). The Variant Bank. http://www.variantbank.org/.

Ash, A., Ash, J. M., McMurry, T. L. and Tenner, B. E. (2010). A balanced diplomacy tournament .

Baarslag, T., Hendrikx, M. J., Hindriks, K. V. and Jonker, C. M. (2015). Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques. *Autonomous Agents and Multi-Agent Systems* , 1–50.

Ballinger, C. (2014). *Coevolutionary Approaches to Generating Robust Build-orders for Real-time Strategy Games*. Ph.D. thesis, University of Nevada, Reno.

Ballinger, C. and Louis, S. (2013a). Comparing coevolution, genetic algorithms, and hill-climbers for finding real-time strategy game plans. In *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*. ACM.

Ballinger, C. and Louis, S. (2013b). Finding robust strategies to defeat specific opponents using case-injected coevolution. In *Computational Intelligence in Games (CIG), 2013 IEEE Conference on*. IEEE.

Baptiste, J. (2016 (accessed January 27, 2016)). JDiplomacy. http://sourceforge.net/projects/jdiplomacy/.

Blackford, J. M. (2014). Online build-order optimization for real-time strategy agents using multi-objective evolutionary algorithms. Technical report, DTIC Document.

BS (2016 (accessed February, 2016)). Backstabbr. http://www.backstabbr.com/.

Calhamer, A. B. (1999). *Calhamer on Diplomacy: The Boardgame 'Diplomacy' and Diplomatic History*. Bloomington, USA: AnchorHouse, 2nd edition. ISBN 1585007587.

DC (2016 (accessed February, 2016)). Diplomatic Corp. http://www.diplomaticcorp.com/.

De Jonge, D. (2010). *Optimizing a diplomacy bot using genetic algorithms*. Ph.D. thesis, Master thesis, UAB, 2010. 10.

de Jonge, D. (2016 (accessed January 27, 2016)). Dave de Jonge Home Page. http://www.iiia.csic.es/~davedejonge/homepage/index.php.

De Jonge, D. and Sierra, C. (2012). Branch and bound for negotiations in large agreement spaces. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems.

Dip2000 (2016 (accessed February, 2016)). Dip 2000. http://www.dip2000.com/index.php.

DPCouncil (2015 (accessed January 25, 2016)). The Diplomatic Pouch Quick Reference Page. http://diplom.org/Online/quickref.html.

Fabregues, A. (2012a). *Facing the Challenge of Automated Negotiation with Humans*. Ph.D. thesis, Universitat Autònoma de Barcelona, Institut d'Investigaciò en Intel ligència Artificial Consejo Superior de Investigaciones Científicas (IIIA-CSIC).

Fabregues, A. (2012b). Personal Web Page. The Artificial Intelligence Research Institute Website. http://www.iiia.csic.es/~fabregues/publications.html.

Fabregues, A., Biec, S. and Sierra, C. (2012). Running experiments on dipgame testbed (demonstration). In *Proc. of the 11th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2012)*.

Fabregues, A., López-Paz, D. and Sierra, C. (2011). Diptools: Experimental data visualization tool for the dipgame testbed (demonstration). In *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*. http://www.iiia.csic.es/files/pdfs/aamas2011demoSubmission.pdf.

Fabregues, A., Navarro, D., Serrano, A. and Sierra, C. (2010). Dipgame: a testbed for multiagent systems. van der Hoek, Kaminka, Lesperance and Luck and Sen, Toronto, Canada: van der Hoek, Kaminka, Lesperance and Luck and Sen. Http://www.dipgame.org/.

Fabregues, A., Navarro, D., Serrano, A. and Sierra, C. (2016). Dipgame website. http://www.dipgame.org/.

Fabregues, A. and Sierra, C. (2009a). Diplomacy game: the test bed http://www.perada-magazine.eu/view.php?source=1761-2009-08-03.

Fabregues, A. and Sierra, C. (2009b). A testbed for multiagent systems. Technical report, IIIA-CSIC, Bellaterra, Barcelona.

Fabregues, A. and Sierra, C. (2010). An agent architecture for simultaneous bilateral negotiations. In *CCIA*.

Fabregues, A. and Sierra, C. (2011). Dipgame: a challenging negotiation testbed. *Engineering Applications of Artificial Intelligence* **24**, 1137–1146. http://www.iiia.csic.es/files/pdfs/eaai2011.pdf.

Fabregues, A. and Sierra, C. (2012). Abridgement of hana: a human-aware negotiation architecture .

Fabregues, A. and Sierra, C. (In press). Hana: a human-aware negotiation architecture. *Journal of Decision Support Systems* .

Fernández-Ares, A., García-Sánchez, P., Mora, A. M., Castillo, P. A. and Guervós, J. J. M. (2014). Designing competitive bots for a real time strategy game using genetic programming. In *CoSECivi*.

Ferreira, A., Cardoso, H. L. and Reis, L. P. (2015a). Dipblue: A diplomacy agent with strategic and trust reasoning. In *7th International Conference on Agents and Artificial Intelligence (ICAART 2015)*.

Ferreira, A., Cardoso, H. L. and Reis, L. P. (2015b). Strategic negotiation and trust in diplomacy–the dipblue approach. In *Transactions on Computational Collective Intelligence XX*. Springer, pages 179–200.

Häärd, F. (2004). *Multi-agent diplomacy*. Ph.D. thesis, Master's thesis, Department of Software Engineering and Computer Science Blekinge Institute of Technology.

Hagelbäck, J. (2012). Multi-agent potential field based architectures for real-time strategy game bots .

Hagelbäck, J. and Johansson, S. J. (2008). Using multi-agent potential fields in real-time strategy games. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*. International Foundation for Autonomous Agents and Multiagent Systems.

Hagenah (2016 (accessed January 27, 2016)). Diptool 2.7.1. http://www.hagenah.de/diptool/.

Hingston, P., Preuss, M. and Spierling, D. (2010). Redtnet: A network model for strategy games. .

Hossain, M. et al. (2013). *Study on Intelligent Negotiation using Genetic Network Programming Evolving based on Agreement Conditions*. Ph.D. thesis.

Johansson, S. J. (2006). On using multi-agent systems in playing board games. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. ACM.

Johansson, S. J. and Håård, F. (2005). Tactical coordination in no-press diplomacy. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. ACM.

Johansson, S. J. and Olsson, F. (2005). Mars a multi-agent system playing risk. In *Proceedings of Pacific Rim International Workshop on Multi-agents (PRIMA)*. Citeseer.

Jonge, D. d. et al. (2015). *Negotiations over large agreement spaces*. Ph.D. thesis.

Keaveney, D. and O'Riordan, C. (2009). Evolving robust strategies for an abstract real-time strategy game. In *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*. IEEE.

Kemmerling, M., Ackermann, N., Beume, N., Preuss, M., Uellenbeck, S. and Walz, W. (2009). Is human-like and well playing contradictory for diplomacy bots? In *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*. IEEE.

Kemmerling, M., Ackermann, N. and Preuss, M. (2011). Nested look-ahead evolutionary algorithm based planning for a believable diplomacy bot. In *Applications of Evolutionary Computation*. Springer, pages 83–92.

Kostick, C. (2015). *The Art of Correspondence in the Game of Diplomacy*. Curses & Magic. ISBN 0993415105.

Kraus, S. (2001). Automated negotiation and decision making in multiagent environments. In *Multi-agent systems and applications*. Springer, pages 150–172.

Kraus, S. and Lehmann, D. (1995). Designing and building a negotiating automated agent. *Computational Intelligence* **11**, 132–171.

Kruijswijk, L. B. (2009). Diplomacy Adjudicator Test Cases. Website. `http://web.inter.nl.net/users/L.B.Kruijswijk/`.

Krzywinski, A., Chen, W. and Helgesen, A. (2008). Agent architecture in social games-the implementation of subsumption architecture in diplomacy. In *AIIDE*.

Kuliukas, K. (2016 (accessed January 27, 2016)). Web Diplomacy. `http://webdiplomacy.net/`.

Labatut, P. (2016 (accessed January 27, 2016)). Diplomacy Stategy Editor. `https://code.google.com/archive/p/diplomacy-strategy-editor/`.

Lara-Cabrera, R., Cotta, C. and Fernández-Leiva, A. J. (2013). A review of computational intelligence in rts games. In *Foundations of Computational Intelligence (FOCI), 2013 IEEE Symposium on*. IEEE.

Lim, C.-U., Paula, I. S., Cheng, J., Ng, N. and Gong, Z. (2009). Automated negotiation in the game of diplomacy .

Lin, R. and Kraus, S. (2010). Can automated agents proficiently negotiate with humans? *Communications of the ACM* **53**, 78–88.

Martin, R. and Knight, M. (1999). *Diplomacy: Prima's Official Strategy Guides*. Roseville, California: Prima Games. ISBN 076152634X.

McMahan, B. and Bekris, K. (2004a). Comp540 project proposal: Building automated agents for multi-agent games: A case study on "diplomacy" .

McMahan, B. and Bekris, K. (2004b). Learning, resource allocation and the art of war .

Niculae, V., Kumar, S., Boyd-Graber, J. and Danescu-Niculescu-Mizil, C. (2015). Linguistic harbingers of betrayal: A case study on an online strategy game. *arXiv preprint arXiv:1506.04744* .

Oort Cloud (2016 (accessed January 27, 2016)). Droidippy. `https://play.google.com/store/apps/details?id=cx.ath.troja.droidippy&hl=en_GB`.

PDO (2016 (accessed February, 2016)). Play Diplomacy Oline. `http://www.playdiplomacy.com/`.

Polberg, S., Paprzycki, M. and Ganzha, M. (2011). Developing intelligent bots for the diplomacy game. In *FedCSIS*. Citeseer.

Ribeiro, J., Mariano, P. and Lopes, L. S. (2009). Darkblade: a program that plays diplomacy. In *Progress in Artificial Intelligence*. Springer, pages 485–496.

Ritchie, A. (2003). Diplomacy ai. *Unpublished Master Thesis. University of Glasgow, Scotland* , 20–21.

Rose, A., Norman, D., Newbury, J. and Williams, H. (2014 (accessed January 27, 2016)). Diplomacy Playing Bot Clients. `http://www.daide.org.uk/index.php?title=Clients`.

Shaheed, J. (2004). Creating a diplomat. *Master's thesis, Imperial College, London, UK* .

Shapiro, A., Fuchs, G. and Levinson, R. (2003). Learning a game strategy using pattern-weights and self-play. In *Computers and Games*. Springer, pages 42–60.

Sharp, R. (1978). *The Game of Diplomacy*. London: Arthur Barker. ISBN 0213166763.

Sharp, R. (2016 (accessed January 27, 2016)). The Game of Diplomacy. `http://www.diplomacy-archive.com/god.htm`.

St-Pierre, D. L. (2009). Mind game:(axis and allies) .

St-Pierre, D. L., Winands, M. H. and Watt, D. A. (2010). A selective move generator for the game axis and allies. In *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*. IEEE.

Stormont, D. P. and Allan, V. H. (2012). A comparison of diplomacy gameboard graph search algorithms. In *Proceedings of the 4th International Conference on Agents and Artificial Intelligence*. ISBN 978-989-8425-96-6.

Tak, M. J., Lanctot, M. and Winands, M. H. (2014). Monte carlo tree search variants for simultaneous move games. In *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*. IEEE.

Utkin, M. (????). Influence graphs: a technique for evaluating the state of the world .

van Hal, J. (2013 (accessed January 27, 2016)). DAIDE. `http://www.daide.org.uk/index.php?title=Main_Page`.

van Hal, J. (2016 (accessed January 27, 2016)). Diplomacy AI: Albert. `https://sites.google.com/site/diplomacyai/`.

WD (2016 (accessed February, 2016)). Web Diplomacy. `http://webdiplomacy.net/`.

Webb, A., Chin, J., Wilkins, T., Payce, J. and Dedoyard, V. (2008). Automated negotiation in the game of diplomacy. *Master's thesis, January* .

Wilson, B., Zuckerman, I. and Nau, D. (2011). Modeling social preferences in multi-player games. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems.