

# INDIVIDUAL VERSUS GROUP SUCCESS IN ROUND-ROBIN TOURNAMENT OF AN ABILITY-BASED TWO-PLAYER GAME

by

**Jonathan Asante Nyantakyi**

B.Sc. Computer Science,  
Valley View University, Ghana, 2007

THESIS SUBMITTED IN PARTIAL FULFILMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF  
MASTER OF SCIENCE  
IN  
COMPUTER SCIENCE

UNIVERSITY OF NORTHERN BRITISH COLUMBIA

August 2018

© Jonathan Asante Nyantakyi, 2018

# Abstract

This thesis is about the development of cooperation and measures of success among self-interested agents in a defined ability-based two-player asymmetric game that is structured as a round-robin tournament. Our research is motivated by the notion that in many systems cooperative behaviour depends on some parameters that are usually not considered in existing research. These include: the balance between individual activity and interaction with others; the impact of agents' ability levels; and the need to maintain balance between individual and group performance. In this thesis, we examine all these issues by using a defined game-theoretic modelling and simulation framework. Our simulation experiments on six agent group compositions establish some patterns of how an agent's ability and strategy impact its individual and overall group performance. The results demonstrate that the design framework supports methodical comparative studies of strategy profiles with respect to specific individual and group performance measures.

# Contents

<b>Abstract</b>	<b>i</b>
<b>List of Figures</b>	<b>v</b>
<b>Acknowledgements</b>	<b>vii</b>
<b>Dedication</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Background and Related Work</b>	<b>5</b>
2.1 Multiagent Systems . . . . .	5
2.2 Evolution of Cooperation . . . . .	7
2.3 Game Theory . . . . .	7
2.4 PD Tournament Structures . . . . .	11
2.4.1 Axelrod's Tournament . . . . .	11
2.4.2 Random Scheduling Tournaments . . . . .	12
2.4.3 Round-Robin Tournament Scheduling . . . . .	13
2.4.4 Iterated Round-Robin Tournament Scheduling . . . . .	14
2.5 PD Evolutionary Models . . . . .	15

<b>3</b>	<b>Problem Statement</b>	<b>17</b>
<b>4</b>	<b>The Modelling and Simulation Framework</b>	<b>21</b>
4.1	The Ability-Based Two-Player Games . . . . .	22
4.1.1	Introduction . . . . .	22
4.1.2	Motivation . . . . .	22
4.1.3	The Ability-based Cooperation (ABC) game . . . . .	23
4.1.4	The Extended Ability-based Cooperation (EABC) game . . . . .	25
4.1.5	The Design of ABC Payoff Matrix . . . . .	25
4.1.6	Game analysis . . . . .	26
4.2	Simulator Architecture . . . . .	28
4.2.1	The Purpose of the Simulator . . . . .	28
4.2.2	Simulator Requirements . . . . .	28
4.2.3	Structure of the Simulator . . . . .	31
4.2.4	Simulation Engine (SE) . . . . .	35
4.2.5	Behaviour of the Simulator . . . . .	39
4.2.6	Approaches to Experimentation . . . . .	51
<b>5</b>	<b>Experimental Results and Evaluation</b>	<b>52</b>
5.1	Definitions and Parameter Settings . . . . .	53
5.1.1	Composition of Agent Group . . . . .	53
5.1.2	Game Model . . . . .	54
5.1.3	Performance Measures . . . . .	54
5.2	Observation and Analysis of Results . . . . .	57
5.2.1	Group Composition 1 . . . . .	57
5.2.2	Group Composition 2 . . . . .	59
5.2.3	Group Composition 3 . . . . .	61

5.2.4	Group Composition 4 . . . . .	63
5.2.5	Group Composition 5 . . . . .	65
5.2.6	Group Composition 6 . . . . .	67
5.3	Inter-group Comparisons . . . . .	68
5.3.1	Failure Percentage . . . . .	68
5.3.2	Group Total Score . . . . .	70
5.3.3	Group Average Score . . . . .	71
5.3.4	Group Adjusted Average Scores . . . . .	72
5.4	Overall Observation and Analysis . . . . .	73
<b>6</b>	<b>Conclusions and Future Work</b>	<b>74</b>
	<b>Bibliography</b>	<b>77</b>
<b>A</b>	<b>Appendix</b>	<b>81</b>
A.1	Graphs and Tables of Results . . . . .	81
A.1.1	Group Composition 1 . . . . .	81
A.1.2	Group Composition 2 . . . . .	85
A.1.3	Group Composition 3 . . . . .	89
A.1.4	Group Composition 4 . . . . .	93
A.1.5	Group Composition 5 . . . . .	97
A.1.6	Group Composition 6 . . . . .	101

# List of Figures

2.1	The Payoff Matrix of the Prisoner's Dilemma (PD) game . . . . .	10
4.1	A Use Case representation of Simulator . . . . .	29
4.2	High-Level Structure of the Simulator . . . . .	32
4.3	Structure of the User Interface . . . . .	33
4.4	Structure of the Experiment Manager . . . . .	35
4.5	Structure of the Simulation Engine . . . . .	37
4.6	Agent Reasoning Architecture . . . . .	39
4.7	<i>Nested Simulation Control Structure</i> . . . . .	41
4.8	<i>Control Structure of Simulation Experiment</i> . . . . .	43
4.9	<i>Control Structure of a Tournament</i> . . . . .	44
4.10	<i>Control Structure of a Round</i> . . . . .	45
4.11	<i>Control Structure of an encounter</i> . . . . .	46
4.12	<i>About component of the Simulator</i> . . . . .	47
4.13	<i>SetUp pane of the Simulator</i> . . . . .	49
4.14	<i>Agent pane of the Simulator</i> . . . . .	50
5.1	Agents' performance for different ability levels in group composition 1 .	57
5.2	Agents' performance for different ability levels in group composition 2 .	59
5.3	Agents' performance for different ability levels in group composition 3 .	61

5.4	Agents' performance for different ability levels in group composition 4 .	63
5.5	Agents' performance for different ability levels in group composition 5 .	65
5.6	Agents' performance for different ability levels in group composition 6 .	67
5.7	Failure Percentage distribution for different group composition . . . . .	69
5.8	Group Total Score: 3 vs. 4 . . . . .	70
5.9	Group Total Score: 5 vs. 6 . . . . .	70
5.10	Group Avg Score: 3 vs. 4 . . . . .	71
5.11	Group Avg Score: 5 vs. 6 . . . . .	71
5.12	Group adjusted average scores . . . . .	72
A.1	Agents' performance for different ability levels in composition group 1 .	83
A.2	Agents' performance for different ability levels in composition group 2 .	86
A.3	Agents' performance for different ability levels in group composition 3 .	90
A.4	Agents' performance for different ability levels in group composition 4 .	94
A.5	Agents' performance for different ability levels in group composition 5 .	98
A.6	Agents' performance for different ability levels in group composition 6 .	102

# Acknowledgements

Foremost, I gratefully thank my supervisor, Dr Jernej Polajnar, for his continuous support of this research, for his patience, motivation, and immense knowledge in this field of study. The constructive comments and guidance greatly helped to make this research possible.

My sincere thanks also go to Dr Desanka Polajnar, for her invaluable advice and insightful suggestions to this work.

I am thankful to my committee members Dr Chris Opio and Dr David Casperson for their interest in serving on my supervisory committee.



This Thesis is dedicated to my family and friends

Ophelia Akyiaa, Felonie Johnson, Stephen A Takyi,  
and Mehul Solanki

for their endless advice, support, and motivation during the difficult  
times of this research.

# Chapter 1

## Introduction

An important aspect of contemporary technological developments is the rise of artificial intelligent systems that are built to autonomously perform increasingly complex and responsible tasks delegated to them. They are often built as multiagent systems (MAS) [Wooldridge, 2009], that is, they consist of interacting intelligent and autonomous units, called agents, which perform individual subtasks but also work together towards common objectives of the system as a whole. Specialized individual agents may be independently designed, and the system may be placed in an operating environment whose characteristics are not entirely known in advance. In anticipation of such circumstances, robust design should enable individual agents to develop cooperation in real time, choosing appropriate strategies depending on the task requirements, perceived environment, and observed behaviour of other agents.

These considerations lead in part to a more general question of how and when cooperation develops in a society of self-interested intelligent individuals. This problem, with some variations, is of common interest to many fields, such as evolutionary biol-

ogy and its generalizations to human society, psychology, political science, economics, and philosophy. The primary mathematical tool employed is game theory, and much of existing research is centred around variations of a single two-player game, known as Prisoner’s Dilemma (PD). The paradox of PD is that both players receive higher rewards (‘payoffs’) when they choose to cooperate with each other than when they both refuse to cooperate (‘mutual defection’), and yet the game-theoretic optimality criteria (dominant strategy, Nash Equilibrium) lead to mutual defection. This is caused by the fact that the highest payoff is provided to the player who defects against a cooperator, and the lowest for unreciprocated cooperation [Axelrod, 2006]. As aptly observed in [Grinberg et al., 2010], “because of this the PD game represents a conflict between individual and collective rationality.” However, subsequent research revealed that the pessimistic social implications of the initial PD analysis can be overcome by many different techniques, including iterated play among multiple agents with different strategies, controlled acquisition of knowledge about strategies of other players, etc. [Kretz, 2011, Takahashi, 2010, Axelrod, 2006].

The present study is motivated by the observation that cooperation in many natural and artificial systems depends on additional parameters that are largely absent from existing PD-centred research literature. One such aspect is the balance between individual activity and interaction with others. In many areas, such as academic study or research, one can make progress by working individually and by working with others, with the balance often guided by rational factors other than abstract ‘selfishness’ or ‘altruism’.

Furthermore, the agents may differ in their levels and types of ability. The decisions on whether, when, and with whom to cooperate may well be informed by one’s own ability and that of the prospective partner. However, the idea that payoffs may

depend on characteristics of the players and not only on the game itself does not fit with the fixed-payoff matrix of PD and is largely absent from PD-based research.

A practical multiagent system design usually needs to maintain a balance between the individual success of self-interested agents in addressing the individual objectives associated with their subtasks, and the success of the group as a whole in completing the overall task. This type of balance is insufficiently explored as most studies either exclusively focus on maximizing individual payoffs ([Takahashi, 2010, Kretz, 2011]) or emphasize group interest ([Grinberg et al., 2010]).

This thesis investigates all three of the above issues within a unified game-theoretic modelling and simulation framework. The framework implements and uses the ABC and EABC games defined in [Polajnar and Polajnar, 2018] specifically for use in this research. ABC is a two-player game in which players can cooperate or defect. It is derived from a base PD game. In every encounter, the players' payoffs depend on the individual abilities of the two participating players. EABC combines an agent's ABC payoffs with gains from the agent's individual activity to form the total individual score.

The framework simulator allows the experimenter to define a group composed of players with different abilities, that employ a number of different strategies. They play a round-robin tournament that may be iterated if required. We adopt the iterated round-robin tournament (IRRT) structure as opposed to the more commonly used Axelrod tournament structure. The calculated tournament scores reflect individual success and group success.

One purpose of the framework is to investigate the properties of different strategies by which agents decide to cooperate or defect. The 'naive' strategies include naive

cooperators, who always cooperate, and naive defectors, who always defect. The ‘advanced’ strategies have a certain knowledge about the partner. For instance, they may involve knowledge of the partner’s strategy type, or of the partner’s ability level. In this work, we are not concerned with how such knowledge is acquired (there are different techniques explored in the literature) but rather what strategies one can formulate using a specific type of knowledge and how such strategies impact the performance of individuals and groups.

The framework allows methodical experimental comparative studies across differently composed groups of agents with varying ability levels and individual strategies. Individual performance can be measured by individual average scores from a completed tournament, in case of groups of same size and similar structure, or by an approximate measure called ‘adjusted average’ that is readily comparable across groups of different sizes. Group performance may be measured by a ‘social welfare’ function such as the group average or the group adjusted average, or by a ‘failure rate’ based on an adopted threshold representing the minimum acceptable individual performance. This allows the experimenter to examine the profile of each individual strategy with respect to its impact upon the individual and group performance. As an illustration, we provide experimental studies of six different group compositions and observation of how individual strategies perform in different contexts.

The rest of this thesis is structured as follows. We review the background and other related work together with their limitations in Chapter 2. In Chapter 3, we outline and motivate the problem that we intend to address in this research. We present the models and the simulation framework including its requirements definitions, structure, and behaviour in Chapter 4. The experimental results, analysis and evaluation are presented in Chapter 5. Chapter 6 presents the conclusions and future work.

# Chapter 2

## Background and Related Work

This chapter presents the necessary background information and an overview of related previous work in multiagent systems, the evolution of cooperation, game theory, normal-form games, Prisoner’s Dilemma (PD), PD tournament structures, and PD evolutionary models.

### 2.1 Multiagent Systems

Shoham and Leyton-Brown [2009] define multiagent systems as “systems that include multiple autonomous entities with either diverging information or diverging interests or both.” In another study by Wooldridge [2009] that focused mostly on artificial systems, multiagent systems are defined as “systems in which many rational and intelligent agents interact with each other in an environment.”

By definition, an agent according to [Ferber and Weiss, 1999] “can be a physical or virtual entity that can act, perceive its environment (in a partial way) and communicate with others, is autonomous and has skills to achieve its goals and tendencies.” Russell and Norvig [2010] in another study consider agents to be rational entities and therefore define a rational agent as “one that acts so as to achieve the best outcome or, when there is uncertainty, the best expected outcome.”

In general, agents are autonomous entities that act upon their environment to satisfy their design objectives. Apart from their autonomous behaviour, there are other characteristics that define agents. Wooldridge and Jennings [1999] identify common characteristics of agents such as responding to changes in their environment (*reactivity*), ability to initiate goals, plans and perform goal-oriented actions (*proactiveness*) and the capability to communicate with other agents or human entities within or outside their environments (*social ability*).

According to Russell and Norvig [2010], the environments that agents might be situated in can have some general properties that are used in their classification. For instance, an environment may be deterministic or non-deterministic. A deterministic environment is one in which the next state of the environment is completely determined by the current state and the action executed by the agent. A non-deterministic environment is one in which the actions of agents on the environment in a given state may result in different possible outcomes, but no probabilities are attached to them. An environment is static if it can change only as a result of agent’s action, or dynamic if it can change due to other processes that influence the environment beyond the agent’s control [Russell and Norvig, 2010]. An environment is considered accessible if the agent can obtain complete information about the environment, and inaccessible

otherwise. An environment is classified as discrete when the sets of possible actions and percepts are finite and fixed, or continuous otherwise [Parsons and Wooldridge, 2002].

## **2.2 Evolution of Cooperation**

Political scientist Robert Axelrod and evolutionary biologist Hamilton used game-theoretic modelling to demonstrate how cooperation arises among members of the same species and even among different species [Axelrod and Hamilton, 1981]. Evolution of cooperation studies how within a population of self-interested individuals cooperation can emerge and persist [Axelrod, 2006]. Over the years, studies on the evolution of cooperation have seen applications in fields such as economics [Friedman, 1998], political science [Axelrod, 2006], evolutionary biology [Axelrod and Hamilton, 1981], and multiagent systems. A formal framework to study the evolution of cooperation is game theory, and in particular its developing sub-discipline of evolutionary game theory [Weibull, 1997].

## **2.3 Game Theory**

Game theory, formally introduced by von Neumann and Morgenstern [1944], studies interactions between self-interested agents and further provides the necessary analysis to determine how a player should act in a given situation [Aumann, 1989]. Applications of game theory are being recognized in fields such as economics [Owen, 2013], evolutionary biology [Ernst, 2009], and more recently in multiagent encounters where



the overall outcome depends critically on the choices made by all agents in the scenario [Shoham and Leyton-Brown, 2009]. Also, principles from game theory could be applied in developing cooperative and competitive multiagent systems [Pendharkar, 2012].

Games discussed in this study belong to the class of non-cooperative games, which means they model interactions of individual self-interested players which receive individual payoffs as a result of the game. In Non-Cooperative Game Theory, we consider the decision theory for more than one agent and all agents act autonomously without any binding agreements. The key assumption is that every player’s goal is to maximize its individual payoff. Depending on the game structure this may lead to adversarial relations (as in zero-sum games), cooperation (as in coordination games), or in most cases to interactions with elements of both conflict and cooperation.<sup>1</sup>

**Definition 2.3.1: (Normal-Form Game)** According to Shoham and Leyton-Brown [2009] A (*finite, n - person*) normal-form game is a tuple  $(N, A, u)$ , where :

- $N$  is a finite set of  $n$  players, indexed by  $i$ ;
- $A = A_1 \times \dots \times A_n$ , where  $A_i$  is a finite set of actions available to player  $i$ .  
Each vector  $a = (a_i, \dots, a_n) \in A$  is called an action profile.
- $u = (u_1, \dots, u_n)$  where  $u_i : A \mapsto \mathbb{R}$  is a real-valued utility (or payoff) function for player  $i$

A *two-player* normal-form game with two strategy options, called *Cooperate* and *Defect*, can be represented by a  $2 \times 2$  *payoff matrix*, as shown in Table 2.1. The row

---

<sup>1</sup>In this study, games are not necessary viewed as adversarial, and so we prefer to use the term “partner” instead of “opponent.”

		<i>Player j</i>	
		<b>Cooperate</b>	<b>Defect</b>
<i>Player i</i>	<b>Cooperate</b>	$(b_i, b_j)$	$(d_i, a_j)$
	<b>Defect</b>	$(a_i, d_j)$	$(c_i, c_j)$

Table 2.1: Payoff-matrix representation of a 2 x 2 game in normal-form

player  $i$  and column player  $j$  receive the payoff  $b_i$  and  $b_j$  respectively when they both cooperate. Similarly, when they both defect player  $i$  receives  $c_i$  and player  $j$  receives  $c_j$ . In the case where player  $i$  defects while player  $j$  is trying to cooperate, player  $i$  gets the payoff  $a_i$  and player  $j$  gets the payoff  $d_j$ . On the other hand, if player  $j$  defects while player  $i$  is cooperating then player  $i$  gets  $d_i$  as the payoff and player  $j$  gets  $a_j$  as his payoff [Beckenkamp et al., 2007].

The two-player matrix game in normal-form shown in Table 2.1 becomes the Prisoner's Dilemma game if and only if the following conditions are met for both player  $i$ 's and  $j$ 's payoffs:

$$a > b > c > d \quad (2.1)$$

and

$$2b > a + d \quad (2.2)$$

A normal-form game can exist as either *symmetric* or *asymmetric*. According to Shoham and Leyton-Brown [2009] a two-player two-action normal-form game is called a symmetric game if it has the form:

	$A$	$B$
$A$	$x, x$	$u, v$
$B$	$v, u$	$y, y$

The requirements of symmetric games are that players do not have distinct roles in the game and that the payoffs of individual players do not depend on their identities. Based on this definition, one can say that the general  $2 \times 2$  game in normal-form presented in Table 2.1 is symmetric when the indexed payoffs are equivalent to each other (example,  $a_i = a_j, \forall i \neq j$ ) [Beckenkamp et al., 2007].

However, in the asymmetric case, the payoff for a player does depend on their identities and players do have distinct roles. Considering the general  $2 \times 2$  game in normal-form presented in Table 2.1, one possible way it can exist in asymmetric form is when at least one of the payoffs  $a_i$  to  $d_i$  differs from the corresponding payoff in  $a_j$  to  $d_j$  [Beckenkamp et al., 2007].

**Prisoner's Dilemma:** The narrative of the Prisoner's Dilemma describes two arrested criminal suspects that are placed in separate rooms with no way to communicate with each other. Without knowing the other's intention, each player has to make a decision whether to betray (Defect) or remain silent (Cooperate) [Wooldridge and Jennings, 1999]. The rewards and conditions for the Prisoner's Dilemma game are shown in Figure 2.1. The highest reward is defection when partner cooperates and

	C(2)	D(2)	
C(1)	R R	S T	where $T > R > P > S$ $2R > (T + S)$
D(1)	T S	P P	

Figure 2.1: The Payoff Matrix of the Prisoner's Dilemma (PD) game

the lowest rewarded is cooperation when the partner defects. The rewards for mutual cooperation (R, R) are higher than the rewards for mutual defection (P, P). When one player cooperates and the other defects, the defector gets the highest payoff, T, and the cooperator gets the lowest, S. Defection is the *dominant strategy*, which means that regardless of the other player's choice, the defecting player will do as well or bet-

ter by defecting than by cooperating. Mutual defection is the only *Nash Equilibrium*, meaning that defection is always the best response to defection by the other player. The only *Pareto Optimal* strategy profile will be to mutually cooperate; this means that we cannot move from mutual cooperation to any other strategy profile without making any player worse off.

An iterated version of the PD game is called *Iterated Prisoner's Dilemma* (IPD). In the classical IPD game, players face each other in repeated encounters and therefore have to choose their mutual strategic actions (Cooperation or Defection) repeatedly with the sole motive of increasing their payoffs. The main assumption of the IPD game is that each competing player is totally unaware of the partner's thoughts except for the moves made by the partner during their previous encounters, which the player may or may not keep in its internal memory depending on its strategy.

## 2.4 PD Tournament Structures

### 2.4.1 Axelrod's Tournament

The IPD was first formalized as a public tournament in 1980 by Axelrod to study how cooperation will evolve among self-interested agents. The Axelrod's tournament has since been perceived as the standard model for the evolution of cooperation [Axelrod, 2006]. In the tournament, each of the  $n$  players meet each other player exactly once and in that meeting plays a long series of PD games [Axelrod, 2006]. In the

Axelrod’s tournament, varied participating strategies competed against one another and, “Tit-For-Tat”, a simple strategy where a player cooperates first, and then for subsequent games duplicates the partner’s last known action emerged as the winner. Analyses have shown that the success of “Tit-For-Tat” could be mostly attributed to the presence of other strategies in the tournament that were more inclined to cooperate [Axelrod, 2006].

Axelrod’s tournament scheduling, when compared to real life situations where PD game theory might apply has two unrealistic assumptions. These are the absence of information about the behaviour of players other than the current partner, and the scheduling of games where each participant plays a long series of PD games against a single partner and never meets the same partner again [Axelrod, 2006].

## **2.4.2 Random Scheduling Tournaments**

Variations of IPD redefine the settings under which players compete with one another. Random scheduling abstraction proposes an arrangement where players are randomly matched against one another to compete in an IPD tournament. Random scheduling was implemented by Takahashi [2010] to study the sustainability of cooperation in a large community when players are uniformly randomly matched in repeated encounters. Players in each round decide whether to cooperate or defect independent of their own past actions but refer to the given immediate past action of their scheduled partners. The results of the study by Takahashi was that in uniformly randomly matched encounters cooperation can be sustained by an equilibrium [Takahashi, 2010].

In another study, Heller and Mohlin [2016] showed that cooperation can still be sus-

tained in randomly matched repeated PD environments when in every encounter players are given a fixed number of random partner’s past actions without an assumed time zero at which interaction starts. Observations from the study by Heller and Mohlin was that in random encounters cooperation could evolve and persist even among players that are committed to specific strategies.

An actual experiment was conducted at Jiatong University’s Smith Economic Lab, Shanghai, by Chong et al. [2007] to investigate how information and reputation help induce cooperation in a random-matched IPD game. In their research, 144 undergraduate students were divided into groups of 12 and randomly matched to play 60 rounds of IPD with a payoff structure similar to the classic Axelrod’s tournament but with a higher incentive for defection and an ability to request up to 10 levels of partner’s preceding actions. The experimental results indicate that players are more cooperative towards partners that have a high reputation to cooperate.

### 2.4.3 Round-Robin Tournament Scheduling

A complete round-robin tournament of  $n$  agents is a tournament in which every agent plays the remaining  $n - 1$  agents. Compared to other scheduling abstractions such as *Single Elimination tournaments* where pairs of players are matched according to an initial seeding, and the winners of these pairs advance to the next round, while the losers are eliminated after a single loss [Kim et al., 2017], or the *Double Elimination tournament* where no player is eliminated until he has lost two games [Glenn, 1960], round-robin tournament scheduling provides equal advantage or opportunity for every player to meet all other partners in the game without any form of elimination.

Also, the round-robin tournament scheduling is considered to be the fairest way to schedule competitions since there are no elements of luck nor arbitrariness. The final results of round-robin tournaments are accurate because the scheduling abstraction presents results of long period interactions against equal competitions. [Rasmussen and Trick, 2008].

#### 2.4.4 Iterated Round-Robin Tournament Scheduling

The *Iterated Round-Robin tournament* (IRRT) scheduling abstraction describes a repeated version of the round-robin tournament where each participant meets all other players in turns for a fixed number of times and for each round they play a single PD game. Various studies into repeated round-robin tournaments have been performed to analyze different forms of repeated Prisoner Dilemma games. One of such is the study by Kretz [2011] which explored the development of cooperation among different agent strategies with varying memory capacities and payoff structures.

Kretz [2011] implements the Iterated Round-Robin Tournament scheduling to investigate how cooperation and defection will emerge for different numbers of iterations, payoff matrices, and memory sizes of strategies in a study on the emergence of cooperative behaviour. Through computer simulations, the scheduling discipline is a modified Iterated Round-Robin Tournament where strategies play with partners and themselves but with knowledge of up to three levels of the most recent preceding actions of future partners and up to two levels of their own. The conclusion is that payoff matrices have much influence on strategies' performances and also affect the evolution of cooperation. The study restricts the look-up of a partner's past moves to only three levels, and strategies lack the opportunity to autonomously decide on the

number of partner's past moves required to make a decision.

## 2.5 PD Evolutionary Models

Different studies of the Prisoner's Dilemma game consider the impact of information gathering on the evolution of cooperation. The Axelrod's tournament assumes that each competing player is only concerned with the current partner and once the iterated encounter with the same partner ends the player loses all information acquired from the interactions and move on to compete against another competitor. However, there are other models that study the impact of information gathering.

Kretz [2011] in his research considered a model that studies the evolution of cooperation among artificial agent strategies in a repeated round-robin tournament of the PD game. A strategy with a memory size  $n$  has  $n + 1$  sub-strategies to define the action in the first, second,  $\dots n^{th}$ , and any further iteration.

Players are given knowledge of up to three levels of the most recent preceding actions of their future partners and up to two levels of their own most recent actions. At the end of every tournament, agents whose total payoffs are below the overall average payoff are eliminated while the remaining agents evolve into the next tournament of the competition. The evolutionary tournament ends if only one strategy remains or all remaining strategies have the same cumulative payoffs.

Kretz [2011] in his study observed the results of evolution of different strategies for different levels of *Own-to-Opponent* memories.



In all simulation experiments, there were 10,000 iterations of the round-robin tournament and the results showed that different agent strategies emerged winners for different levels of memory combinations and utility payoffs.

Grinberg et al. [2010] also propose a model to study the evolution of cooperation among human participants. The main aim of that model is to represent and to predict existing game dynamics and also study the evolution of cooperation among 10 participants randomly paired to play 100 rounds of the PD Game.

These individuals are randomly assigned 5 agent strategies and competed at different *Cooperative Index* (CI) levels from 0.1 to 0.9. The CI level of a game is defined to be  $(R - P) / (T - S)$ . Using agents' total payoffs and the number of mutual cooperations as measures of success, agents evolve after every 100 rounds of play and the top 5 agents procreate and proceed to the next tournament whilst the bottom 5 agents are eliminated. Grinberg et al. [2010] argue that with the ability for agents to predict future subjective payoffs for both cooperation and defection moves there will be a strong positive correlation between CI levels and evolution of cooperation among independent self-interested agents.

# Chapter 3

## Problem Statement

Advancements in distributed intelligent systems have enabled us to delegate increasingly complex tasks to heterogeneous multiagent systems that may involve software agents and robots of various types in interactions that usually include cooperative behaviour. In particular, this motivates investigation into how a society of independently designed artificial agents can be constructed to perform a specific task in a dynamic environment whose behaviour is not fully predictable. Such agents should be self-interested enough to ensure that their design objectives are achieved (individual success) but must be enabled to proactively develop cooperation through interactions with other members of the society in order to achieve the overall group goal (group success).

The predominant part of modelling and simulation research on the development of cooperation in multiagent systems has been based on game-theoretic concepts, primarily on the Prisoner's Dilemma (PD), and to a lesser extent on other two-player games (for

example, public goods [Santos et al., 2008], snow drift [Souza et al., 2009]). To the best of our knowledge most such studies have emphasized either individual success ([Axelrod, 2006], [Kretz, 2011]) or group success ([Santos et al., 2008]). There have not been many studies that balance these two performance criteria. Yet in practical task-oriented systems individual and group success are often closely related, complement and influence each other. Our study aims at combining both measures of success within a single model and exploring both types of system performance within the same setup of a simulation experiment.

In many practical systems, the participating agents will have different levels of ability and methods of decision making. This is not adequately reflected in existing systems of game theoretic studies. Some models (for example, [Ridinger and McBride, 2016]) include abilities conducive to better strategy choices but do not model the skills relevant to efficiency in mainstream activities. Yet we observe that in many agent groups such mainstream abilities of individual agents impact the interaction patterns and the resulting individual and group success.

An illustrative example that motivates our research is a class of students in a software course, where each individual develops a piece of software to the same specification (for example, a compiler for a specified language), done as a sequence of phases. The students are allowed to cooperate by exchanging testing strategies and test cases, with the intent to speed up the group progress. Thus, the progress in design and coding is measured individually and reflects the individual’s ability, while the progress in testing depends also on the patterns of cooperation among players of varying individual abilities. The total progress of an individual in class is a balanced combination of the two components.

In each phase of the project, any two students have an opportunity to interact, analogous to a round-robin tournament; and since the project has a sequence of phases, an Iterated Round-Robin Tournament (IRRT) arises as a natural encounter schedule. This is one of many examples where for realistic modelling of repetitive interactions IRRT emerges as preferable to the widely used Axelrod tournament [Axelrod, 2006] pattern, in which each pair of players meet once and play a long series of games.

We further note that the example motivates efforts to develop strategies that enhance both individual and group success. While individual success may naturally be measured by players' individual scores, group success could be measured in a number of ways. Immediate possibilities in the current example are the class average score (that is, the 'social welfare' concept of game theory) and a low failure rate with respect to an established failure threshold modelling the minimum acceptable level of individual success.

In realistic agent societies where there are cooperation and defection interactions, it is possible that agents may acquire a certain level of knowledge about their partners and may use such information to achieve their objectives. We intend to introduce and study the impact of information gathering among different combinations of agent strategies. Most models in the reviewed literature implement information gathering through direct observance of partner's action during interactions (for example, [Axelrod, 2006]) or access to a central historical repository (for example, [Takahashi, 2010], [Kretz, 2011], [Grinberg et al., 2010]). Our study includes strategies based on knowledge of other agents' properties but does not address the underlying knowledge acquisition mechanisms. We are interested in the strategic effect of information gath-

ering and not concerned with the mechanisms and techniques of acquiring information.

When pursuing the above research ideas in a game-theoretic setting, we note specific limitations of the Prisoner’s Dilemma as the basic underlying game structure. The first observation is that all scoring in PD-based models typically comes from interaction, while progress in real systems may also be achieved through individual activity without interaction. Second, central to the dilemma in question is the fact that mutual cooperation is more rewarding than mutual defection while an even higher reward is collected by defecting against the cooperator, with the latter facing the lowest payoff. This ranking of interaction payoffs may not be realistic in all contexts. In particular, when agents perform their mainstream tasks individually, with varying abilities, they may or may not benefit from a specific instance of mutual cooperation (as might occur in the illustrative example above).

The main objective of this research is to establish a game-theoretic modelling and simulation framework that supports methodical experimental exploration of how different strategies of cooperation vs. defection impact both individual and group success in multiagent systems with different agent abilities. The agents perform individual activities and also interact with other agents, with both components contributing to their individual scores. The two-player game chosen to model the basic encounter must incorporate the influence of individual abilities upon player payoffs. The design of agents’ behaviour should allow advanced strategies that take advantage of knowledge about partners’ properties (notably, their strategies or individual abilities). The framework should let the experimenter evaluate the performance of different strategies with respect to their impact upon individual or group success, and also compare how a specific strategy performs in groups with different strategy compositions.

# Chapter 4

## The Modelling and Simulation Framework

This chapter presents an asymmetric two-player cooperation game, its model design, and the architecture of the software simulator used to implement the game. Section 4.1 describes the Ability-based Cooperation (ABC) game and the Extended ABC (EABC) game as defined in [Polajnar and Polajnar, 2018]. The architecture of the simulator in terms of its requirements (purpose, definition, functional and non-functional, and component requirements) is explained in Section 4.2. Following that, we consider the structure of the simulator in Section 4.3 and present the behaviour of the different components of the simulator architecture in Section 4.4. Section 4.5 details some notes on simulator implementation while Section 4.6 elaborates our different approaches to experimentation.

## 4.1 The Ability-Based Two-Player Games

### 4.1.1 Introduction

The *Ability-based Cooperation* (ABC) game is an asymmetric two-payer game derived from the Prisoner's Dilemma (PD), in which the payoffs depend on the abilities of participating players. In general, an ABC game is derived from a “base” PD game and an assignment of abilities to players. As in PD, the players of ABC choose between cooperation and defection, but the payoffs depend on players' abilities, and thus the ability distribution impacts the player motivation and outcomes. An ABC's payoff matrix may itself behave like PD in parts of the game's parameter space. We further introduce and study the notion of *Extended ABC* (EABC) game in which players pursue ability-based individual activities and also interact through the ABC game.

### 4.1.2 Motivation

The general idea is that in a system of  $n$  agents with different abilities, each agent performs an individual activity whose reward is proportional to the agent's ability. In addition, the agents interact with each other playing an asymmetric two-player game in which they can cooperate (C) or defect (D). The payoffs for cooperation or defection are influenced by the abilities of the individual partners. The total agent's score is a weighted sum of its earnings through individual activity and its payoffs from the interactions with other players.

For an illustrative example consider a class of students in a software course, where

each individual develops a piece of software to same specification (for example, a compiler for a specified language), done as a sequence of phases. The progress in design and coding is individual and reflects the individual ability. The students are allowed to cooperate by exchanging testing strategies and test cases, with the intent to speed up the group progress and add a cooperative dimension to the software development activity. The progress in testing thus partly depends on cooperation patterns.

In game-theoretic terms, the class activity can be represented as an iterated tournament of scheduled mutual EABC encounters, with each project phase corresponding to an iteration. In each encounter, a student can receive test information from the other player (C) or receive nothing (D); likewise, the student can provide test information to the other player (C) or provide nothing (D). The model establishes a framework for comparative studies of how different ability-based strategies impact cooperation and how they affect the individual and group success in a multiagent system.

### 4.1.3 The Ability-based Cooperation (ABC) game

#### Definition

An ABC game is constructed from a given Prisoners Dilemma (PD) game  $G$ , with ability values assigned to its players. The construction is as follows:

We adopt as the *base PD game* any Prisoner's Dilemma game for players  $i$  and  $j$ :

		<i>Player j</i>	
		<i>C</i>	<i>D</i>
<i>Player i</i>	<i>C</i>	$R, R$	$S, T$
	<i>D</i>	$T, S$	$P, P$



that satisfies the following two conditions:

$$T > R > P > S = 0 \quad (4.1)$$

$$2R > S + T \quad (4.2)$$

Next, we assign positive real values  $a_i$  and  $a_j$ , called *abilities*, to the players  $i$  and  $j$  respectively.

Then the ABC game  $G_{ij}$  is the two-player game with the following payoff matrix:

		<i>Player j</i>	
		<i>C</i>	<i>D</i>
<i>Player i</i>	<i>C</i>	$a_j R, a_i R$	$S, a_i R + a_j P$
	<i>D</i>	$a_j R + a_i P, S$	$a_i P, a_j P$

## Interpretation

The ABC game represents a situation in which each player pursues an individual activity and also interacts with the other player. The ABC game formally describes the interaction and resulting payoff. The Extended ABC (EABC) game, to be defined next, also includes the scores from the individual activity, and defines a balanced sum of individual activity score and the ABC game payoff as the total score. The intuitive motivation and interpretation for the ABC game definition are situated in that context.

#### 4.1.4 The Extended Ability-based Cooperation (EABC) game

##### Definition

An *EABC game*  $E_{ij}$  is a quadruple

$$E_{ij} = (G, a_i, a_j, \alpha)$$

where  $G$  is a base PD game,  $a_i$  and  $a_j$  are positive real numbers representing player abilities, and  $\alpha \in [0, 1]$  is the *balancing factor*.

The triple  $(G, a_i, a_j)$  is used to construct the ABC game  $G_{ij}$ . The *score* of player  $i$  is then defined as

$$s_i^{(j)} = \alpha a_i + (1 - \alpha) p_i^{(j)} \quad (4.3)$$

where  $p_i^{(j)}$  is the  $i$ 's payoff from its  $G_{ij}$  game with  $j$ , and the value of  $\alpha$  is used to balance the relative impact of the two components.

#### 4.1.5 The Design of ABC Payoff Matrix

Let  $G$  be a PD game with payoff parameters  $T > R > P > S = 0$ . For each pair of players  $(i, j)$ , the asymmetric game  $G_{ij}$  is defined with payoffs for  $i$  as follows:

$R_{ij} = a_j R$	The idea is that the higher the ability of the cooperation partner $j$ , the higher is the $i$ 's reward for mutual cooperation with that partner
$P_{ij} = a_i P$	By not engaging in cooperation, the player $i$ saves some resources (for example, time) that can be used for individual progress, commensurate with $i$ 's ability (as expressed by the coefficient $a_i$ ).
$T_{ij} = a_j R + a_i P$	When $i$ defects while $j$ cooperates, $i$ gets both the advantage of $j$ 's cooperation (as in $R_{ij}$ above) and the advantage of own non-engagement (as in $P_{ij}$ above).
$S_{ij} = 0$	$i$ does not benefit from this situation; all other payoffs are positive.

The definitions of  $R_{ji}$ ,  $P_{ji}$ ,  $T_{ji}$ , and  $S_{ji}$  (the payoffs for  $j$ ) are symmetric to the above.

#### 4.1.6 Game analysis

The ABC game has been derived from Prisoner's Dilemma and bears some similarity to it. Unlike the usual version of PD, ABC is asymmetric; however, asymmetric versions of PD have also been studied [Ahn et al., 2007, Sheposh and Gallo Jr., 1973, Murnighan, 1991].

The first question is when ABC is actually an asymmetric PD. According to the definition of asymmetric PD [Beckenkamp et al., 2007], ABC will be PD when all of the following conditions hold:

$$T_{ij} > R_{ij} > P_{ij} > S_{ij} \qquad T_{ji} > R_{ji} > P_{ji} > S_{ji} \qquad (4.4)$$

$$2R_{ij} > T_{ij} + S_{ij} \qquad 2R_{ji} > T_{ji} + S_{ji} \qquad (4.5)$$

After the substitution of values from the ABC payoff matrix definition, it immediately follows that all four conditions in (4.4, 4.5) hold if and only if the following two conditions hold:

$$a_j R > a_i P \qquad a_i R > a_j P \qquad (4.6)$$

One can thus formulate the following result:

**Proposition 1.** Let  $G_{ij}$  be an ABC game whose base payoff values for mutual cooperation and mutual defection are  $R$  and  $P$  respectively, and whose player abilities are  $a_i$  and  $a_j$ . Then  $G_{ij}$  is an instance of Asymmetric Prisoner's Dilemma if and only if

$$\frac{P}{R} < \frac{a_i}{a_j} < \frac{R}{P} \qquad (4.7)$$

Let us now consider a round-robin tournament with  $n$  players ( $n > 1$ ), where each player  $i$  plays one EABC game with every other player  $j$  according to a predefined tournament schedule. All players' ability values belong to an interval  $[\underline{a}, \bar{a}]$ , where  $\underline{a} > 0$ . The next proposition is immediate from (4.7).

**Proposition 2.** Consider a set of  $n$  agents ( $n > 1$ ), playing in a round-robin tournament based on EABC games. Let each player  $i$  have the ability  $a_i \in [\underline{a}, \bar{a}]$ , where  $\bar{a} > \underline{a} > 0$ . If

$$\frac{\bar{a}}{\underline{a}} < \frac{R}{P} \qquad (4.8)$$

then every ABC game played in the tournament is an Asymmetric Prisoner's Dilemma game.

## 4.2 Simulator Architecture

### 4.2.1 The Purpose of the Simulator

The purpose of our discrete-event simulator is to provide the framework necessary for the experimentation of a specific agent-based model involving the interaction of multiple agents engaged in a two-player asymmetric game structured in a round-robin tournament. This is a research simulator composed of different modules some of which have been developed and implemented in other previous studies.

### 4.2.2 Simulator Requirements

The list of requirements for the simulator is divided into two main categories: functional requirements, and non-functional requirements. In the rest of this section, different components of each category are described.

#### Requirements Definition

The requirements of the simulator outlined in this study provide an abstract description of the services provided by the simulator and the constraints under which the simulator operates. The requirements definition has been categorized into two different kinds: *functional requirements* and *non-functional requirements*. The remainder of this section elaborates each of these requirements definition in the context of the simulator described.

## Functional Requirements of the Simulator

The functional requirements of the simulator consider the requirements that are related to the conceptual functions of the simulator without considering how the features are interoperated. The *Use Case* diagram shown in Figure 4.1 describes the functional requirements of the simulator.

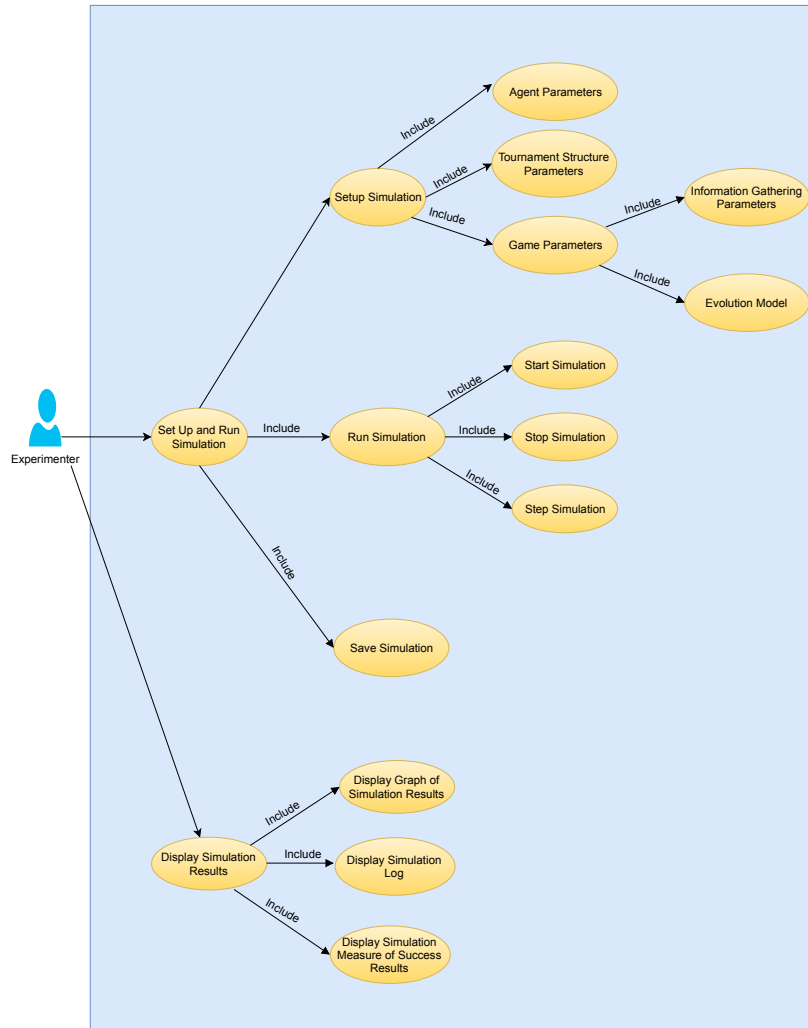


Figure 4.1: A Use Case representation of Simulator

1. *Set up and Run Simulation*: The experimenter should be able to set up and run a number of different simulation experiment to study the results of different measures of success. To set up and run simulation may include the following:

- (a) *Set up Simulation:* This includes;
    - i. *Agent Parameters:* The experimenter should be able to select the desired number of agents required to run in the simulation experiment and assign to each agent its level of ability.
    - ii. *Tournament Structures Parameters:* The experimenter of the simulator must be able to choose the number of tournament iterations for each experiment, base payoff matrix, and balancing factor.
    - iii. *Game Parameters:* The simulation environment shall comply with other game parameters such as information gathering approach and the approach to evolution specified per each experiment.
  - (b) *Run Simulation:* The experimenter will be able to run a simulation experiment. This functional requirement includes the ability of the simulator to start and stop the simulation experiment any time at the request of the experimenter in order to observe simulation results. Also, the simulator shall comply with the experimenter's decision to step through the simulation experiment. To step through the simulation run means the simulator simulates one round and then stores a snapshot for that round until the experimenter decides to perform another step, run, or stop simulation.
  - (c) *Save Simulation:* The experimenter shall be able to save the results of simulation experiments in terms of agents' performance as a statistics, progressive charts, or simulation logs.
2. *Display Simulation Results:* The experimenter should be able to display the results of each simulation experiment. The Use Case depiction of this functional requirement includes the following:
- (a) *Display Graph of Simulation Results:* The simulator shall be able to comply

with the experimenter’s request to display a graphical representation of the simulation results indicating for each tournament agents’ score, average score, maximum and minimum scores.

- (b) *Display Simulation Logs:* The simulator shall upon request from the experimenter provide access to a log of agents’ actions, strategies, and scores.
- (c) *Display Measures of Success Results:* The experimenter of the simulator should be able to view the results for different measures of success such as individual and group success.

### **Non-Functional Requirements of the Simulator**

The non-functional requirements of the simulator define the qualitative specifications including its properties and constraints on the simulator as a whole. These requirements ensure the simulator’s quality and maintainability [Sommerville, 2011]. The simulator described in this thesis meets the non-functional requirements of *Efficiency* and the potential ability to execute *Parallel Runs*. The efficiency of the simulator measures its productivity while parallel runs ensure faster simulation executions.

#### **4.2.3 Structure of the Simulator**

The high-level design of the simulator is described in this sub-section. We explore the organization of interrelated elements of the simulator to provide the basis for its detailed design and implementation.



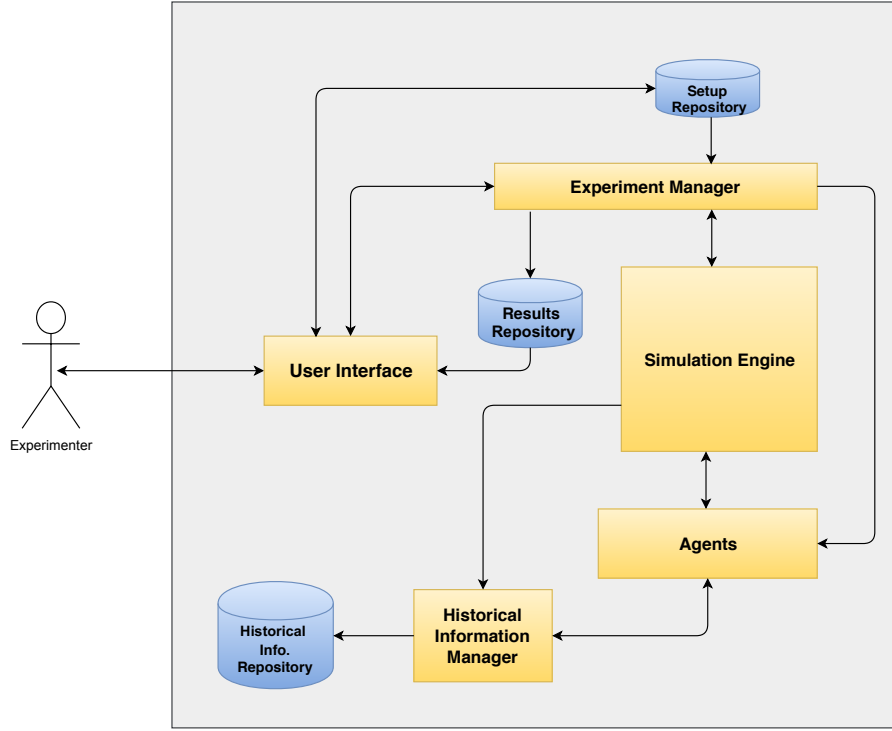


Figure 4.2: High-Level Structure of the Simulator

The structure of the simulator can be divided into eight main components. These are the *User Interface* (UI), *Experiment Manager* (EM), *Set Up Repository* (SR), *Simulation Engine* (SE), *Results Repository* (RR), *Historical Information Manager* (HIM), *Historical Information Repository* (HIR), and *Agent* as shown in Figure 4.2.

### User Interface (UI)

As a very important component of the simulator's design, the *User Interface* executes actively to ensure an easy, and interactive use of the simulator by the experimenter. Before a simulation experiment, the User Interface provides an *Input Interface* for the experimenter to set up the number of agents, assign agents' strategies, and initialize

or update other simulation setup parameters in the *Setup Repository*. At the end of simulation, the experimenter can use the *Results Interface* sub-component of User Interface to query and display specific results of the simulation. Figure 4.3 shows the interrelated components of the User Interface.

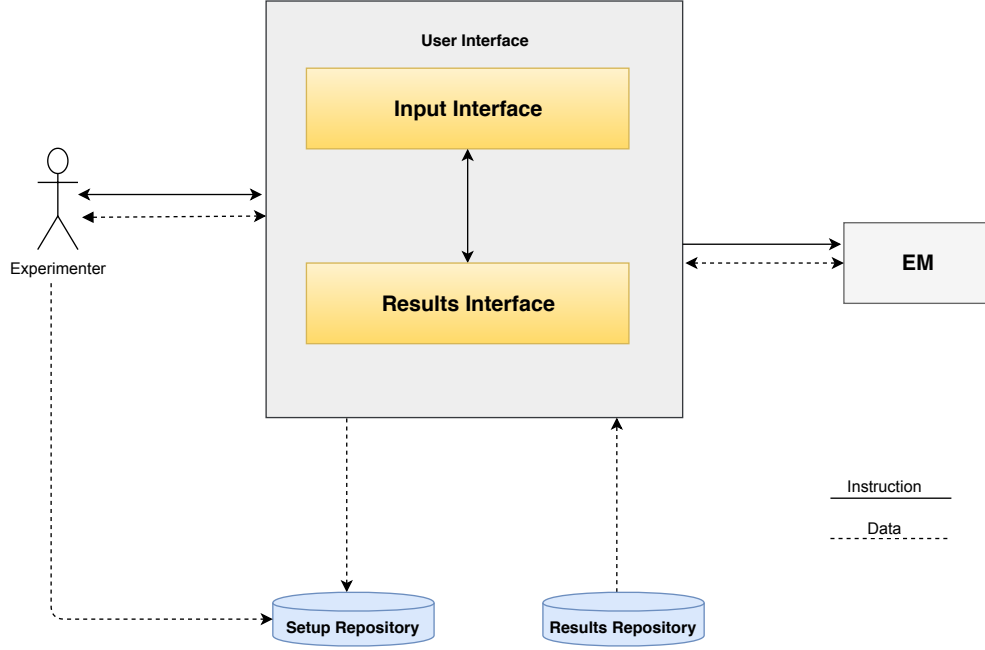


Figure 4.3: Structure of the User Interface

## Experiment Manager (EM)

The *Experiment Manager* upon activation from the User Interface performs the function of validating assigned strategies, setting up ability levels and other simulation input parameters stored in the *Setup Repository*. At the end of each experiment, the Experiment Manager retrieves the results from the Simulation Engine and stores in the *Result* repository to be displayed to the Experimenter.

The Experiment Manager can be functionally divided into two sub-components as

shown in Figure 4.4. These are *Agents Setup Manager* and *Parameter Configuration Manager*.

The Agents Setup Manager validates the number of agents in the experiment and also ensures that there is an accurate assignment of strategies and agents' ability levels as specified by the experimenter in the setup repository. The *Parameter Configuration Manager*, on the other hand validates and assigns all other parameter inputs of the simulation environment such as payoff matrix, number of tournaments iterations, and the balancing factor stored in the setup repository.

These validated inputs and assigned parameters are then sent to the *Simulation Engine* and *Agent* components respectively to begin the experiment. In the case of an invalid simulation input, the Experiment Manager alerts the experimenter through the User Interface to make the necessary corrections to invalid input parameters before the simulation begins.

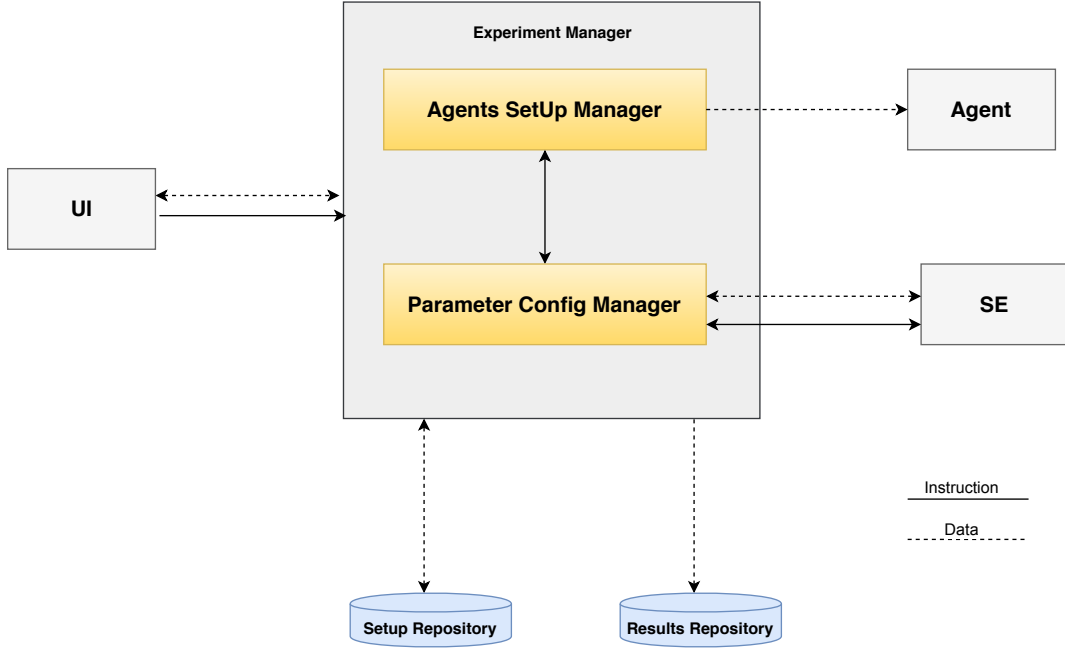


Figure 4.4: Structure of the Experiment Manager

### SetUp Repository (SR)

The *Setup Repository* (SR) serves as a storage component for configuration values. Inputs for the experiment are stored in the Setup Repository to be accessed by Experiment Manager for validation and assignments before simulation experiments begin.

#### 4.2.4 Simulation Engine (SE)

The design of the *Simulation Engine* (SE) shown in Figure 4.5 assumes an active simulation component capable of supporting concurrent discrete-event simulation of our model defined in Section 4.1. Its main responsibility is to begin and control the execution of simulation experiments after receiving the input and a signal from the

Experiment Manager.

For each simulation run, the Simulation Engine kickstarts by activating the *Scheduler*. The Scheduler plans events for execution and also provides the arrangement for agents to compete at their assigned time-steps while ensuring that the round-robin scheduling abstraction adopted for this model is maintained. The agent interaction schedule created by the scheduler is then made available to all other sub-components of the Simulation Engine.

For each tournament in the simulation experiment, the *Tournament Handler* component determines the total number of rounds of games that must be played based on the number of agents competing in the tournament. In addition, the *Tournament Handler* ensures that each agent participates in equal encounters before proceeding to the next tournament or end of the simulation. In every tournament, agents engage in  $r$  rounds of encounters. The *Round Handler* is responsible for ensuring that for each round in a specific tournament participants are paired with different partners. Also, the Round Handler manages agents' performance and pay-off profiles for each round.

The *Match Manager* provides the capacity for encounters scheduled in each round to be executed independent of one another. Also, it ensures that the right scores are assigned to every agent in the simulation experiment based on their actions and the payoff matrix. The *Match Manager* after every encounter stores information on agent ids, abilities, strategies, actions, and scores in the *Historical Information Repository* through the Historical Information Manager and in the *Results* repository through the Experiment Manager.

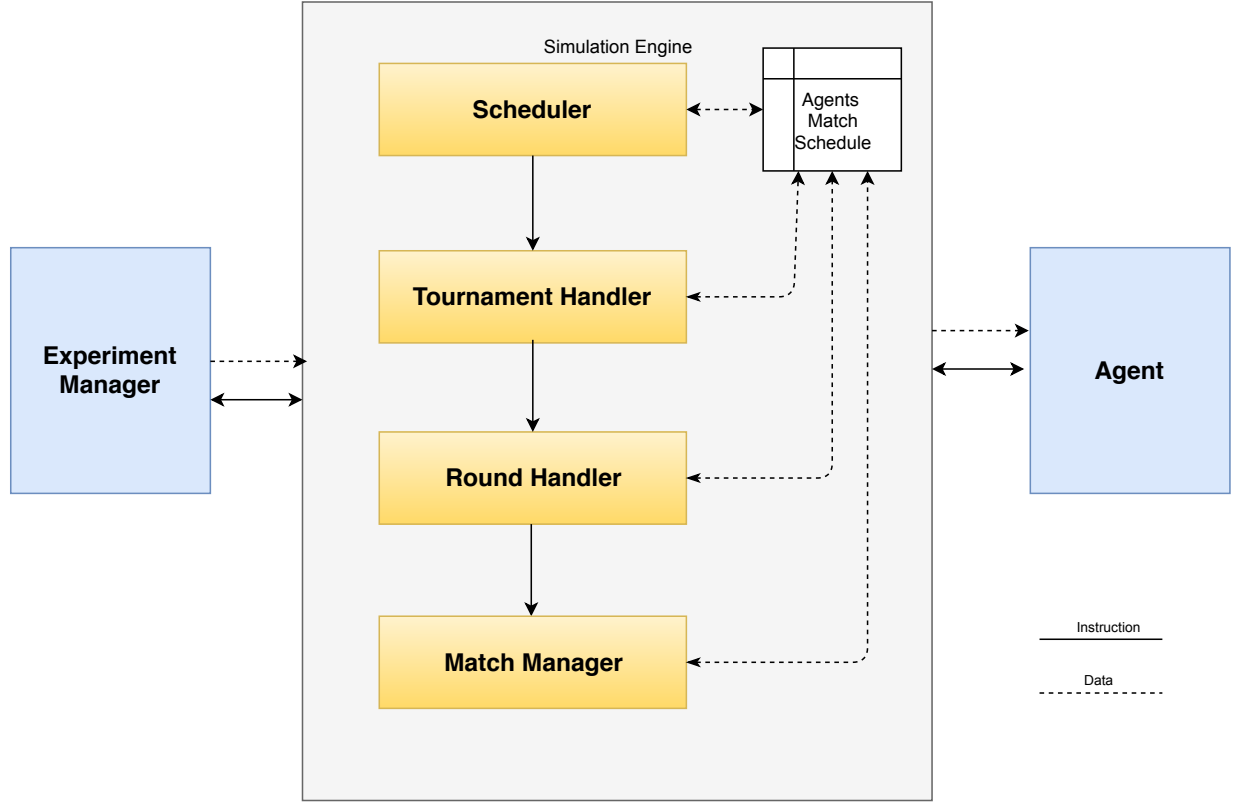


Figure 4.5: Structure of the Simulation Engine

## Historical Information Manager (HIM)

At the end of every encounter, the *Historical Information Manager* (HIM) receives the scores, and actions of the paired agents from Simulation Engine in order to update the Historical Information Repository. Furthermore, this active component of the simulator serves the function of responding to all queries by agents pertaining to information on current or future partners.

## Historical Information Repository (HIR)

The Historical Information Repository (HIR) is basically a storage unit for all strategies, abilities, scores, and past actions of agents in the simulation experiment. This passive component is updated at the end of every round by the Historical Information Manager. Information stored in the Historical Information Repository can only be accessed by agents after submitting requests to the Historical Information Manager.

## Agent

An *agent* in this model describes a goal-oriented, autonomous, and intelligent component of the simulator that competes by submitting actions to the Simulation Engine in every scheduled encounter. During every encounter against a partner, the agent perceives, reasons, and performs an action upon request from the Simulation Engine.

In order to take an optimal action as to whether to cooperate or defect against a scheduled partner, the agent may request information about the partner, or other agents in the tournament by submitting a request to the *Historical Information Manager*. After this, the agent refers to its strategy algorithm and acts appropriately in order to achieve its desired goals. Figure 4.6 shows the interrelated components of the agent component.

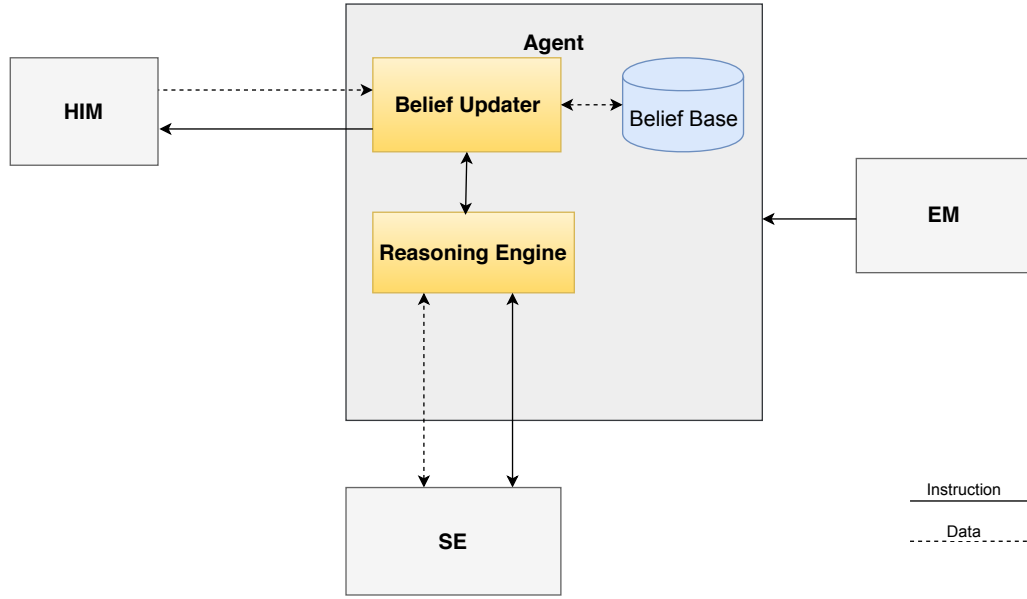


Figure 4.6: Agent Reasoning Architecture

### 4.2.5 Behaviour of the Simulator

This section of the study discusses the behaviour of various essential components of the simulator. The behaviour of elements such as Simulation Engine (SE), User Interface (UI), and Agent will be analyzed in the context of their functions, control operators, inputs and outputs.

#### Simulation Engine

The design of the simulator follows steps of discrete events decomposed into a number of encounters that are executed in each round of a tournament by the Simulation Engine. The number of encounters scheduled per round and the number of rounds played in each tournament typically depend on the number of agents in an experi-



ment run. At the beginning of the experiment, agents selected by the experimenter are assigned their initial strategies broadly categorized into *Naive Cooperators*, *Naive Defectors*, *Selective Cooperators*, and *Limited Cooperators (Fixed and Ratio)*. Apart from the assignment of strategies, agents are also assigned individual ability levels. These agents are then scheduled in a round-robin tournament that require them to submit actions in every repeated round until the end of the experiment. In experiments requiring multiple rounds, the simulator supports the ability of some agents to request and retain summaries of partners' information. A general control structure for the Simulation Engine is as shown in Figure 4.7;

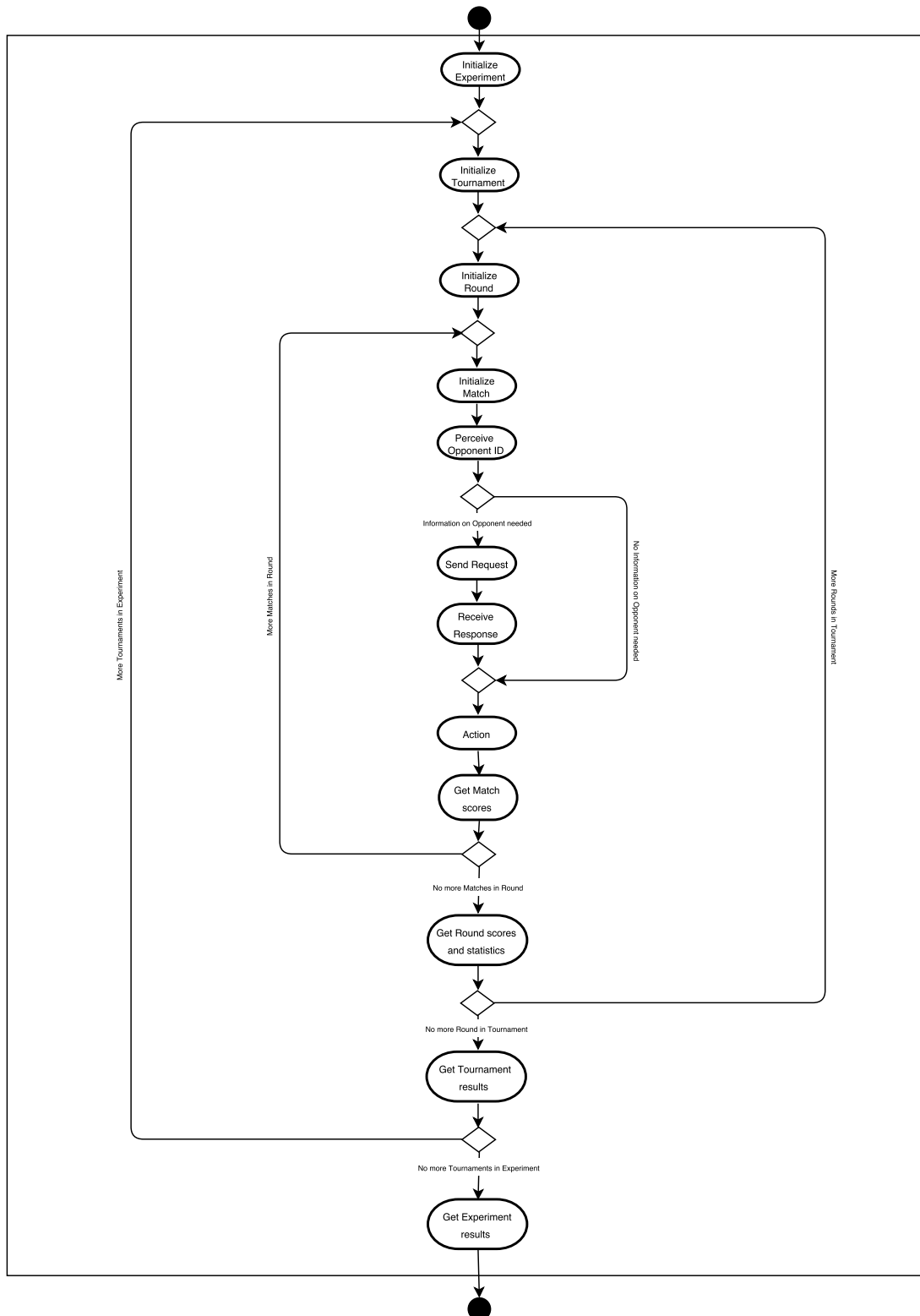


Figure 4.7: *Nested Simulation Control Structure*

In a specific simulation experiment, the Simulation Engine begins its function by activating the *Scheduler* which initializes current experiment configuration values received from the Experiment Manager (EM). After this, the Scheduler assigns to every agent an *AgentId* by which it is identified throughout the experiment. At the end of the experiment, the Scheduler calculates agents' total scores, arranges their performance in a leaderboard based on abilities, and signals the Experiment Manager to store the results in the result repository while at the same requesting Experiment Manager to submit configuration values for the next experiment. The Scheduler's control structure is as shown in Figure 4.8

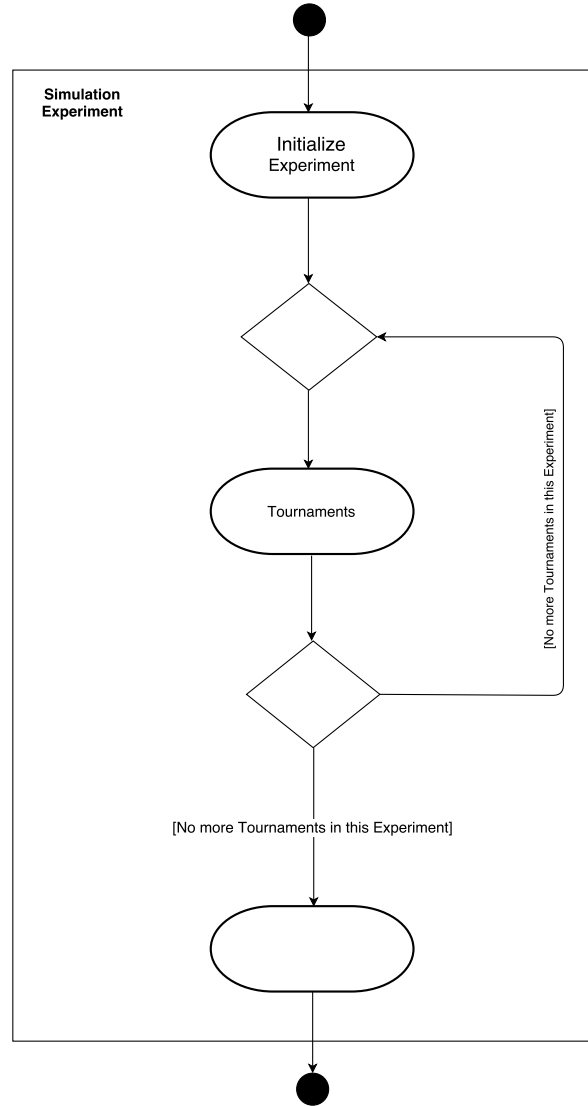


Figure 4.8: *Control Structure of Simulation Experiment*

After setting up configuration values and scheduling agents, the Scheduler invokes the *Tournament Handler* to initiate, coordinate, and control the activities of agents for every tournament specified in the current experiment. In an experiment with  $n$  players, the Tournament Handler ensures that there will be  $(\frac{n}{2}(n-1))$  number of games for every tournament and each player participates in  $(n-1)$  number of games where  $n$  is always an even number value. At the end of a tournament, the Tournament

Handler reports a summary statistics of the maximum score, average score, minimum score, and players' progress. In multiple tournaments, the Tournament Handler ensures that all agents participate in equal number of games before initiating the next tournament as shown in Figure 4.9.

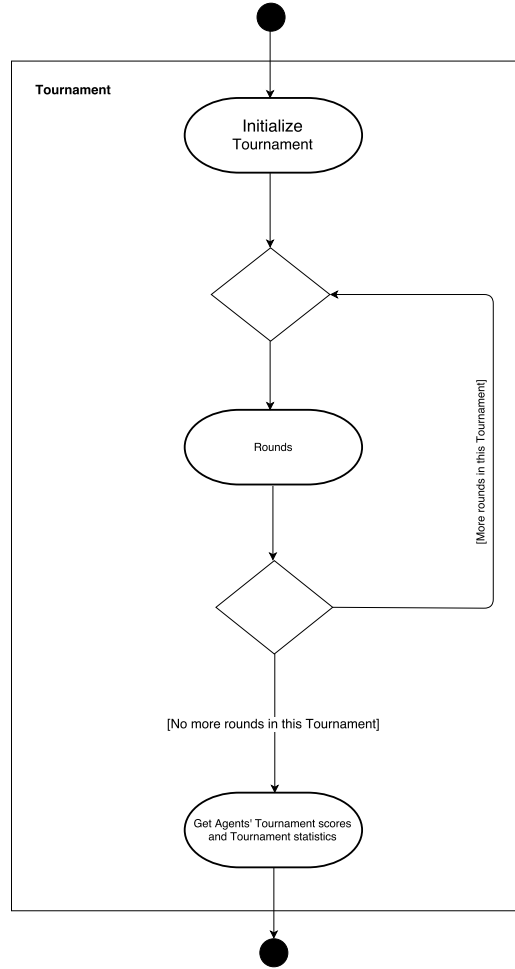


Figure 4.9: *Control Structure of a Tournament*

A single tournament may consist of multiple rounds of encounters depending on the number of agents. Each round is managed by the *Round Handler* which monitors the activities of agents in every round. The Round Handler ensures that every agent participates in a single encounter against a different partner in every round and

submits a single action against its scheduled partner. The action could be either to *cooperate* or *defect*. The control structure of the Round Handler is as shown in Figure 4.10.

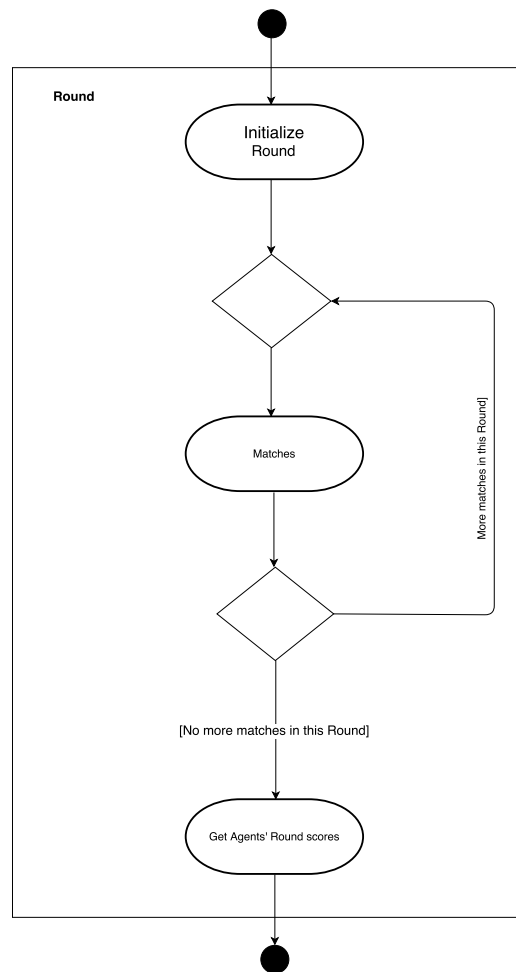


Figure 4.10: *Control Structure of a Round*

Encounters between agents and their partners are controlled by the *Match Handler*. The Match Handler initializes a new encounter, schedules, and requests agents to submit their actions. Based on agents' submitted actions, the *Match Handler* assigns

scores to the respective agents. At the end of every scheduled encounter, the *Match Handler* reports on the agents' strategies, actions, and scores. In addition, the Match Handler returns the calculated scores to both agents. Figure 4.11 describes the flow structure of the Match Handler and agents' actions during an encounter.

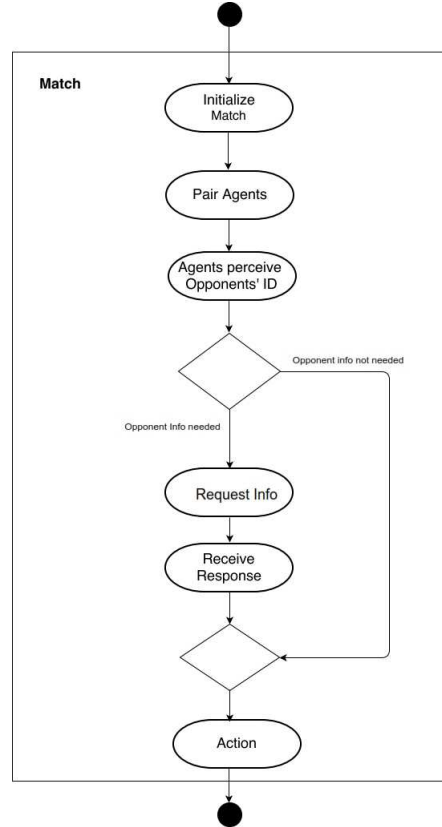


Figure 4.11: *Control Structure of an encounter*

## The UI

The *User Interface* (UI) component of the simulator is responsible for providing interactive access between the experimenter and the simulator. As an important component of the simulator, the *User Interface* has the goal of ensuring an easy, efficient, and user-friendly use of the system. The implementation design of this component

has multiple tabs that perform different functions. These are *About* , *Setup*, *Agent*, and *Display*. The functionalities of these components are briefly explained below;

## About

The About tab provides a brief description of the simulator, a list of agent strategies, experiment's input parameters and other general information as shown in Figure 4.12

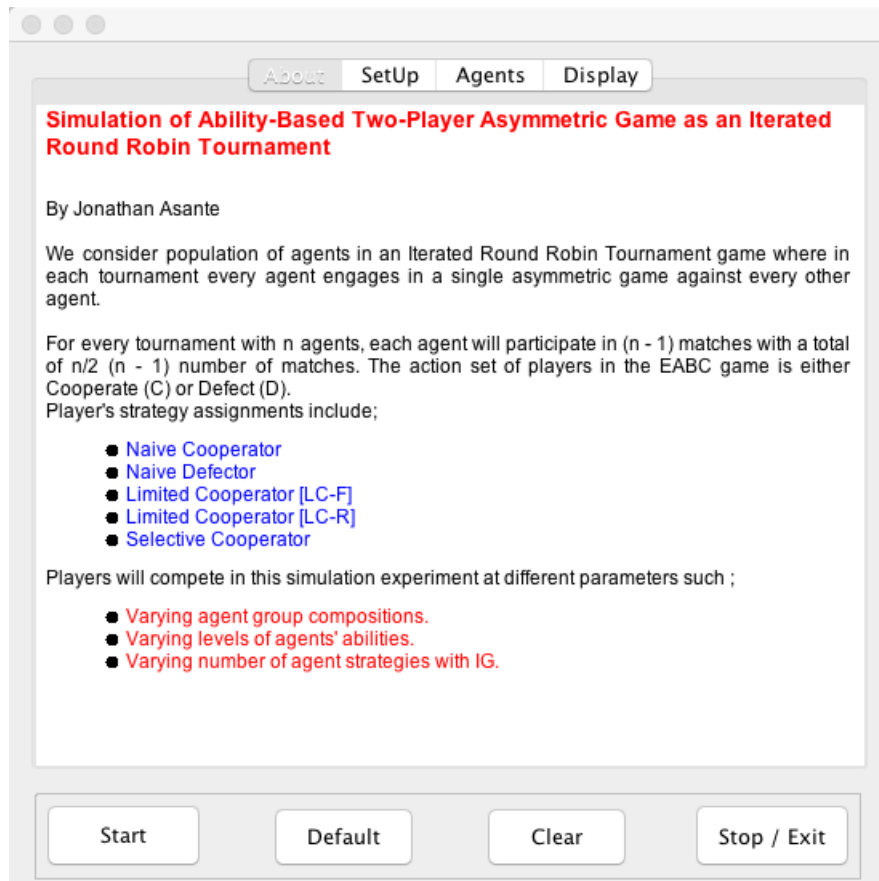


Figure 4.12: *About component of the Simulator*



## Setup

The Setup tab which is mainly controlled by the Experiment Manager provides access for the experimenter to define, validate, and update the settings required for the simulation experiment. Experiment settings specified in the Setup tab includes payoff values (*Temptation to Defect*, *Reward*, *Sucker Punch*, and *Punishment for Defection*), *Number of Tournaments*, *Information Request Limit*, and *Uncertainty Level* that may be varied to study their effects on agents' performances and the evolution of cooperation. Alternatively, the experimenter may decide to upload the experiment's setup using the *Command Line Interface* (CLI). Figure 4.13 shows the Setup tab of the simulator.

☐ **Load Experiment Setup**

Temptation PayOff [D, C]	10
Reward PayOff [C, C]	8
Punishment PayOff [D, D]	2
Sucker's PayOff [C, D]	0

Number of Tournaments	1
Evolutionary Model	No Evolution
Scaling Factor:	0.5

Uncertainty Level: 
1.0

☐ **Information Request Limit**
☐ Selective Coop
 ☐ Limited Coop

☐ Assign
 ☐ Random

Figure 4.13: *SetUp* pane of the Simulator

## Agent pane

This pane provides the experimenter access to specify the total number of agents in the experiment as well as select the different strategies (*Naive Cooperators*, *Naive Defectors*, *Limited Cooperators* (*Fixed* and *Ratio*), and *Selective Cooperators*.) that must be assigned to agents. As shown in Figure 4.14, the agent pane also specifies the minimum and maximum ability intervals. An experimenter may decide to specify

the different ability levels.

Num Of Agents : 18

Select Agent Strategies

☒ Naive Cooperation    ☐ Selective Cooperator

☒ Naive Defection    ☒ LimitedC- F    ☐ LimitedC- R

Agents Abilities Interval

Minimum Level : 2    Maximum Level : 12

Enter Abilities Levels : 12-10-8-6-4-2

Agents' Info. Request Approach

Opponent's Strategy

Start    Default    Clear    Stop / Exit

Figure 4.14: *Agent pane of the Simulator*

#### **4.2.6 Approaches to Experimentation**

The game-theoretic agent-based model described in this research provides the possibility to study through simulation experiments agents' strategic behaviour and measure their success in terms of individual and group performance. We continue to examine these measures of success by considering the impact of certain agents' abilities on the development of cooperation, and individual and group measures of success as described in our ABC model. We also consider the comparable effect of information gathering among agents with abilities in the defined EABC model.

## Chapter 5

# Experimental Results and Evaluation

This chapter presents results and evaluations of different simulation experiments conducted using the Enhanced-ABC model defined in Section 4.1. Our evaluation of agents' individual and group performance profiles highlights the potential impact of agents' abilities and information gathering on individual and collective group performance.

Section 5.1 provides the definitions and parameter settings for different composition of agent groups, game model details, and performance measures. In Section 5.2, we continue to discuss our observations and analysis of different simulation experiments, and then further our discussion with inter-group comparisons in Section 5.3. The chapter concludes with an overall analysis of our observations in Section 5.4.

## 5.1 Definitions and Parameter Settings

### 5.1.1 Composition of Agent Group

We explore 6 different compositions of agent groups representing the number of different simulation experiments conducted in this study. Each group structure consists of  $n$  number of agents determined as a product of 6 different levels of abilities (12, 10, 8, 6, 4, 2) and a varying number of agent strategies. For example, a composition of 6 ability levels and 2 agent strategies will result in a total of 12 agents. We define the different kinds of agent strategies below and Table 5.1 displays the compositions of agent groups;

- **Naive Cooperator:** also known as NaiveC; always cooperates with its partner.
- **Naive Defector :** also called NaiveD in our experiments; defects in all situations against its partner.
- **Selective Cooperator:** has the capacity to determine if the partner is a defector; the Selective Cooperator also called SelectiveC always defects against Defector, but always cooperates otherwise.
- **LimitedC-Fixed:** also referred to as LimitedC-F can determine the partner's ability; always cooperates if partner's ability is equal to or above a *fixed threshold* of 8, otherwise always defects.
- **LimitedC-Ratio:** shares some characteristics with the LimitedC-F strategy and can determine the partner's ability. It always cooperates if  $a_j R / a_i P \geq 1$ ; otherwise, it defects. This strategy is referred to as LimitedC-R in our simulation experiments.

Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
NaiveC NaiveD	NaiveC NaiveD SelectiveC	NaiveC NaiveD LimitedC-F	NaiveC NaiveD LimitedC-R	NaiveC NaiveD SelectiveC LimitedC-F	NaiveC NaiveD SelectiveC LimitedC-R

Table 5.1: Compositions of Agent Groups

### 5.1.2 Game Model

Following the definitions detailed in Section 4.1.3 we adopt the *base* Prisoner’s Dilemma (PD) game in its symmetric form with payoff values  $T = 10$ ,  $R = 8$ ,  $P = 2$ ,  $S = 0$  to discuss the *Ability-Based Cooperation* (ABC). Unlike the classical PD game, the ABC game has the payoff values for a participating player  $i$  against a partner  $j$  as  $T_{ij} = a_j R + a_i P$ ,  $R_{ij} = a_j R$ ,  $P_{ij} = a_i P$ ,  $S_{ij} = 0$ . The *EABC* includes the following parameters;

- *Agent’s score*: is the score agent  $i$  receives after an encounter with agent  $j$ . This is defined as  $s_i^{(j)} = \alpha a_i + (1 - \alpha) p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of agent  $i$  vs.  $j$ .
- *Balancing factor*: is a constant parameter  $\alpha \in [0, 1]$  that regulates the relative impact of the two components of the score, which are the ability component and the payoff. In all simulation experiments,  $\alpha$  was set to 0.9.
- *Round-Robin tournament structure*: provides a schedule of pairwise encounters that give equal advantage to every agent to meet all other partners in the game.

### 5.1.3 Performance Measures

In this study, we explore two measures of success; individual success and group success. These are discussed separately below.

## Measures Individual of Success

Let the multiagent system consist of  $n = k * q$  agents, where  $k$  is the number of strategies and  $q$  is the number of ability levels. Each agent  $i$  has the ability  $a_i$ . Each of the strategies is played by exactly one agent of each ability level.

For an *agent*  $i$  that has an ability  $a_i$  we explore the following measures of individual success:

- *Total score*: An agent  $i$ 's total score for the tournament is calculated as  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
- *Average score*: We define the average score of an agent  $i$  as  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
- *Adjusted total score*: An agent  $i$  with ability  $a_i$  has adjusted total score which is calculated as the total score from all games with partners whose abilities differ from  $a_i$ . Formally,  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
- *Adjusted average score*: This is an approximate performance measure which allows direct comparisons of the individual performance of an agent with given ability and strategy across groups of different sizes. It is the adjusted total score of an agent  $i$  divided by the number of games against partners whose abilities differ from  $a_i$ . Thus,  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .

## Measures Group of Success

For each group composed of agent strategies, we also examine the following measures group of success:



- *Failure threshold*: refers to the experimenter-set limit representing the minimum acceptable level of individual performance. It is uniformly applied in all group compositions in our study. In each experiment, we use the failure threshold to determine the failure percentage.
- *Failure percentage*: is the percentage of agents in a group that falls below the failure threshold. As a measure of group success, we compare the failure percentage between groups. A lower group failure percentage indicates a higher group success and vice versa.
- *Group total score*: calculates the sum of the total scores for all agents in a specific group structure. This measure of success provides a better approach to compare group performance between compositions with an equal number of agents, encounters, and strategies. It provides a better result to compare the influence of two or more different strategies in similar group structures.
- *Group average score*: is the sum of total scores for all agents in the group structure divided by the number of agents. Used in similar conditions as the group total score to compare the overall average performance among groups.
- *Group adjusted total score*: calculates the sum of the adjusted total scores for all agents in a specific group structure. It only serves as an auxiliary measure to derive group adjusted average score since it cannot be used as a performance measure to compare group compositions of different size.
- *Group adjusted average score*: refers to the sum of the total adjusted scores for all agents in the group structure divided by the number of agents. This measure shows the overall average success of the group members in each composition.

## 5.2 Observation and Analysis of Results

We present the simulation results and analyze our observations for different compositions of group structures as follows.

### 5.2.1 Group Composition 1

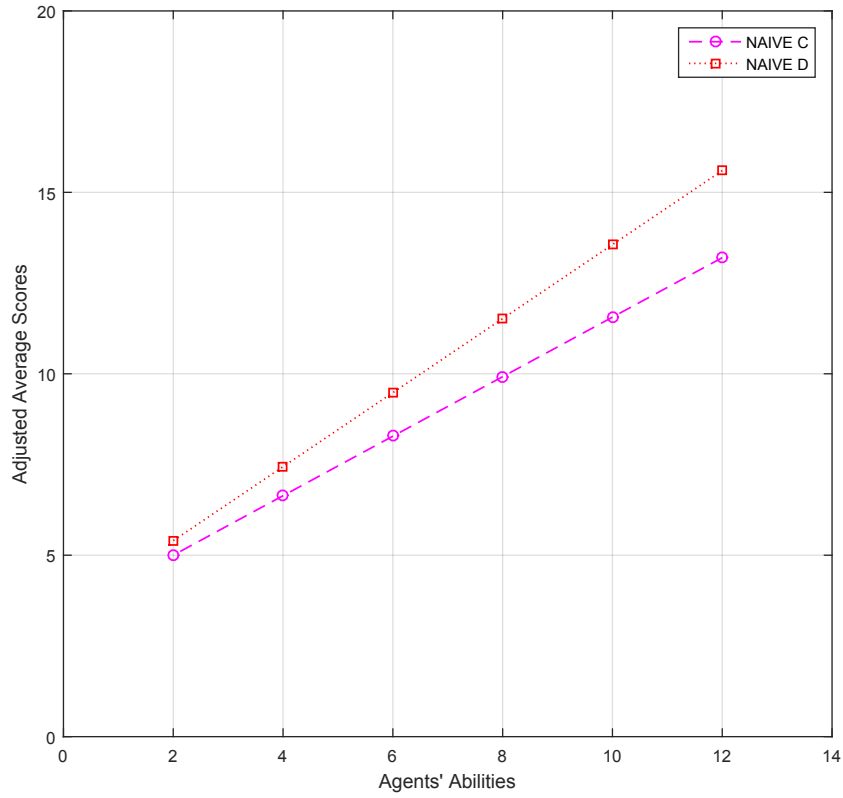


Figure 5.1: Agents' performance for different ability levels in group composition 1

## Individual Success

- Increasing the abilities of agents also increases their individual performance. Thus, agents with higher abilities, in general, perform better than those with lower abilities across the same strategy.
- Individually, Naive Cooperators at every ability level perform worse than Naive Defectors. This follows from the easily verified fact that defection is the dominant strategy in the ABC game (regardless of whether the game happens to be an asymmetric PD).
- The gap between Naive Defectors and Naive Cooperators increases as abilities grow. This is because  $i$ 's defection reward  $a_i P$  increases with  $i$ 's own ability, while  $i$ 's cooperation reward  $a_j R$  is independent of  $i$ 's own ability.

## Group Success

- At a general failing threshold of 5, we observe that none of the 12 agents fall below the failing threshold. This is because lower ability Naive Defectors gain more from exploiting Naive Cooperators while lower ability Naive Cooperators gain from interaction reward with higher ability cooperators.
- The group adjusted average score for this composition is 98 representing a very low group performance.
- Naive Cooperators do not fare well in this group because interaction reward for cooperators is almost independent of player's own ability.

### 5.2.2 Group Composition 2

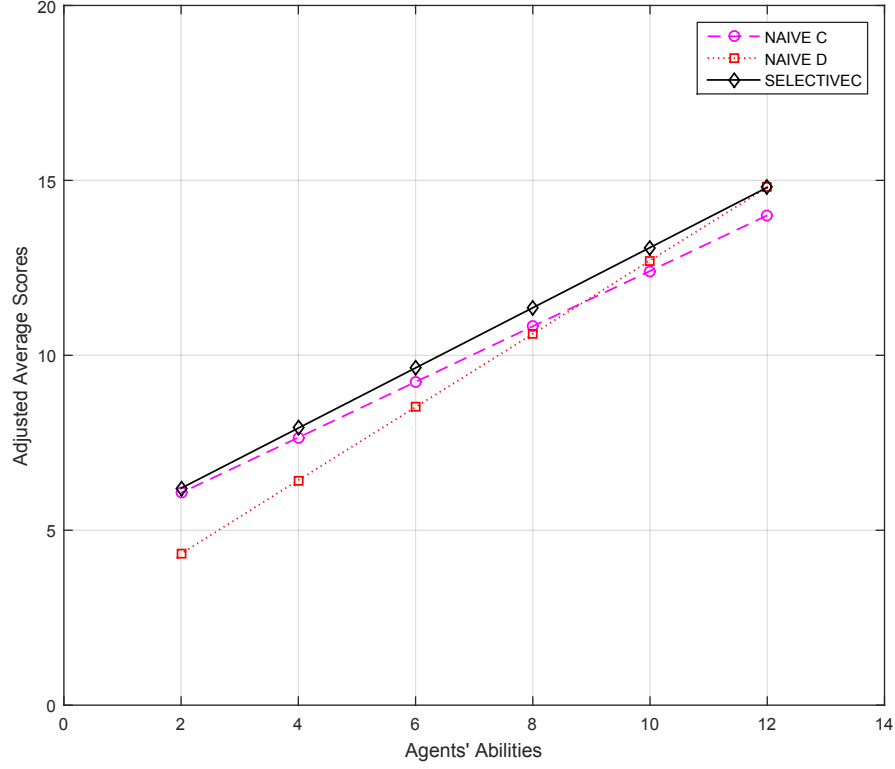


Figure 5.2: Agents' performance for different ability levels in group composition 2

#### Individual Success

- Naive Defectors with lower abilities ( $\leq 6$ ) perform worse than all other agents at the same levels. This is because Naive Defectors in this group structure face defections from Selective Cooperators at all ability levels.
- The presence of Selective Cooperators increases the individual performance of Naive Cooperators by doubling the number of their cooperating partners and decreases the individual performance of Naive Defectors by doubling the number of their defecting partners.
- We observe an increasing gap between Selective Cooperators and Naive Coop-

erators because each Selective Cooperator defects against  $1/3$  of the population with interaction gain proportional to its own ability.

### **Group Success**

- The impact of Selective Cooperators reduces the overall group failure percentage.
- The group adjusted average score which is similar to social welfare in game theory and represents the overall success of group members for this composition of agents is 150.5.
- The increased group success is because Selective Cooperators cooperates with  $2/3$  of population irrespective of their ability levels.
- The most supportive environment for individual Naive Cooperators is when there are Selective Cooperators.

### 5.2.3 Group Composition 3

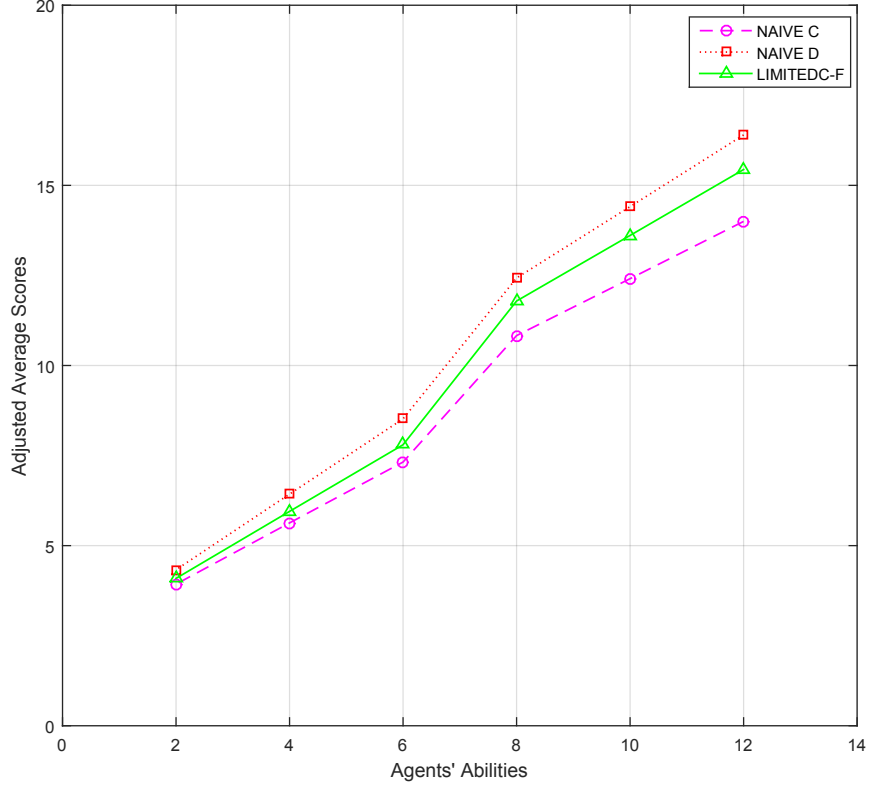


Figure 5.3: Agents' performance for different ability levels in group composition 3

#### Individual Success

- Higher ability Naive Defectors perform better in this group than group 1 and 2 due to gains from interaction with both Naive Cooperators and LimitedC-Fixed strategies at all levels.
- LimitedC-Fixed strategy reduces interaction gains of all lower ability agents and causes them to record low individual performance.
- Naive Cooperators with abilities  $\geq 8$  gain from twice the number of cooperators in this group than in group 1.

- LimitedC-Fixed doesn't seem to be the best strategy in this composition because of its approach to cooperate or defect based on a fixed ability threshold. However, it is also not the worst strategy in this group structure.

### **Group Success**

- Failure percentage is higher in this group than group 1 and 2 because LimitedC-Fixed strategy defects against 1/2 of the population based on abilities.
- The group adjusted average score representing overall success of group members in this composition is 146.1. This is slightly below the group performance of composition 2 but higher than composition 1.
- The presence of LimitedC-Fixed strategy significantly reduces the contribution of low ability agents toward group success.

### 5.2.4 Group Composition 4

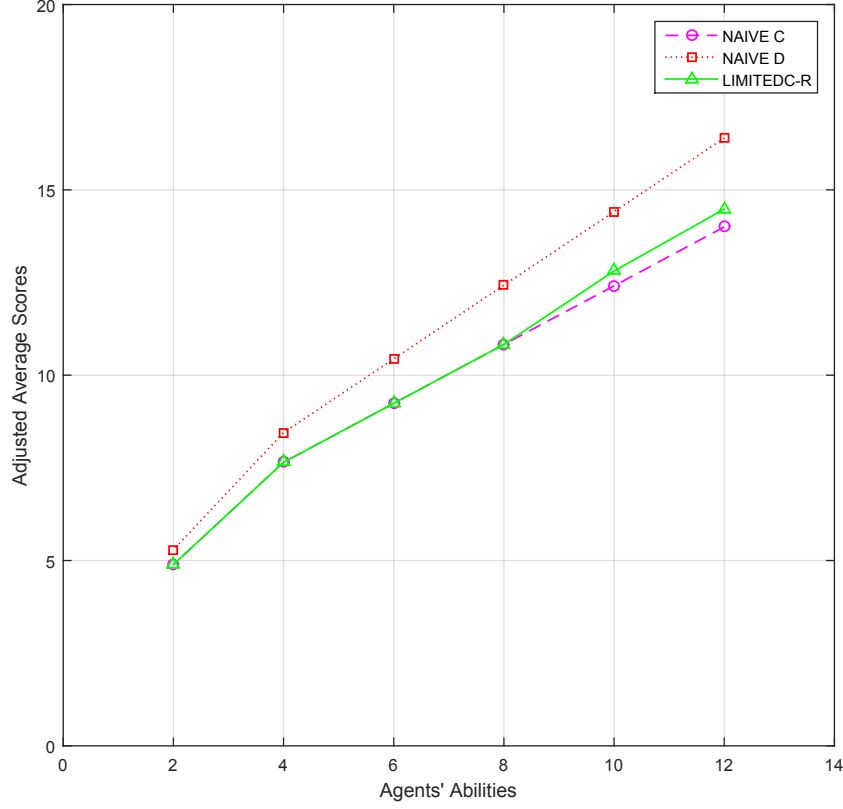


Figure 5.4: Agents' performance for different ability levels in group composition 4

#### Individual Success

- At  $a = 2$ , the Naive Defector performs slightly better due to the benefit of individual progress from non-interaction and cooperative behaviour from other non-defectors with abilities  $\leq 8$ .
- We observe a slight increasing gap between LimitedC-Ratio agents and Naive Cooperators because LimitedC-Ratio agents with ability ( $a \geq 10$ ) lack the incentive to cooperate with lower ability agents when they can benefit from both partners' cooperation and individual reward for non-engagement.



- Unlike LimitedC-Fixed agents in group 3, LimitedC-Ratio agents improve group performance but record low individual scores.

### **Group Success**

- Compared to group 2 the individually rational approach of LimitedC-Ratio strategy toward cooperation and defection significantly reduces the overall group failure percentage.
- The group adjusted average score for this composition is 155.3. This is higher than group 3 because the individually rational approach of LimitedC-Ratio strategy increases overall group performance.
- Compared to group 3, this group records a higher contribution from low ability agents toward group success due to the significant impact of LimitedC-Ratio strategy.

### 5.2.5 Group Composition 5

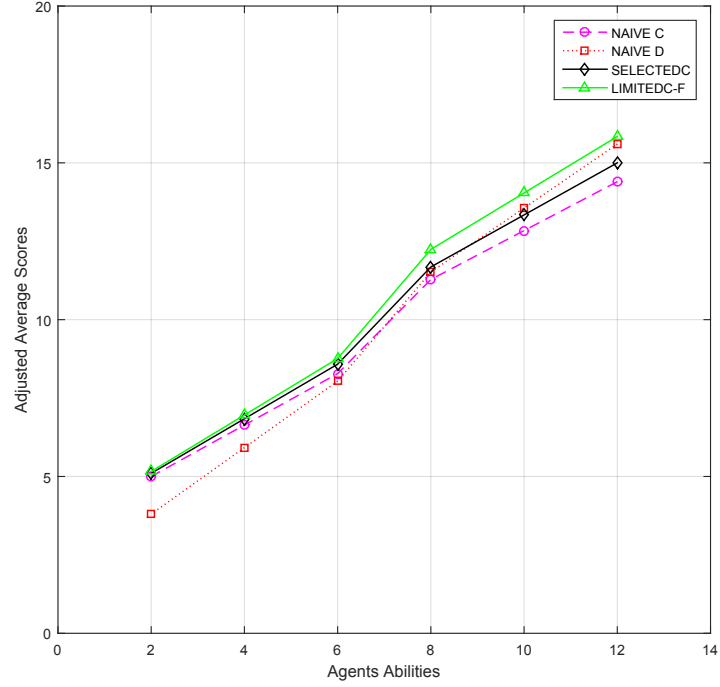


Figure 5.5: Agents' performance for different ability levels in group composition 5

#### Individual Success

- Higher ability LimitedC-Fixed strategy is the best in this group because of gains from cooperating with high ability cooperators and defecting gains against low ability agents.
- The lowest ability Naive Defector records the overall lowest score in this study due to defections from 3/4 of the group population.
- We observe an increasing gap between Selective Cooperators and Naive Cooperators because:
  - Selective Cooperators' interaction reward for defection increases with players' own abilities.

- Naive Cooperators' benefit with Selective Cooperators and Naive Cooperators is almost independent of own abilities.
- Naive Defectors may not benefit significantly from interaction with Selective Cooperators and LimitedC-Fixed strategies, higher abilities defectors perform well due to higher individual activity.

### **Group Success**

- A low ability Naive Defector that faces mutual non-engagement against 75% of the group population falls below the failure threshold.
- The group adjusted average score is 200.35. This is basically because Selective Cooperators promote cooperation among Native Cooperators and LimitedC-Fixed strategy supports all agents with higher abilities.
- Only agents with higher level abilities contribute significantly to group success with very minimal support from lower ability agents.

### 5.2.6 Group Composition 6

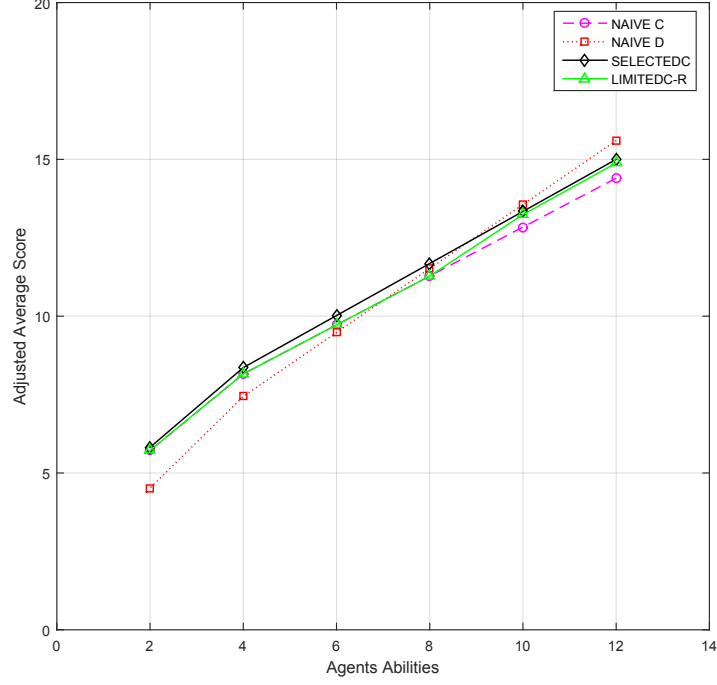


Figure 5.6: Agents' performance for different ability levels in group composition 6

#### Individual Success

- Individual performance of LimitedC-Ratio strategy with higher abilities are lower than LimitedC-Fixed strategies in group 5 because:
  - LimitedC-Ratio agents only defect when reward for mutual cooperation is lower than mutual defection.
  - LimitedC-Fixed agents defects against all agents with abilities  $\leq 6$ .
- LimitedC-Ratio strategy does not dominate in individual performance but significantly increases group performance.
- Naive Cooperators perform better in this group than all other groups.

## Group Success

- We observe an equal failure percentage as group 5. However, the performance of the agent below the failure threshold in this group is better than group 5.
- Consistent with our expectation, the group adjusted average score for this composition is 209.55 representing the highest in our study so far. This is because Selective Cooperators promote cooperation with all agents except Naive Defectors and LimitedC-Ratio agents use an ability-based individually rational technique to promote group success.
- The individual rational approach of LimitedC-Ratio strategy may reduce their own individual performance but significantly increase group performance.

## 5.3 Inter-group Comparisons

We now compare the performance of each of the six group compositions examined in this study. The purpose of inter-group comparison is to determine and analyze the overall nature of the different group compositions and also compare the performance levels of two or more groups.

### 5.3.1 Failure Percentage

For all group compositions, we measure the percentage of agents that scored an adjusted average value below the failure threshold. We assume a failure threshold of 5 and analyze the failure percentage for all 6 group structures as shown in Figure 5.7.

We observe the following:

- The more generalized approach by LimitedC-Fixed strategy to defect against all agents with abilities ( $a < 8$ ) reduces the performance of lower ability agents and therefore increases the failure percentage to 16.67%.
- Selective Cooperators' collective rationality to support cooperation reduces the failure percentage in group 2. This is lower than the higher failure percentage observed in group 3.
- Both groups 5 and 6 record only 1 agent (low ability Naive Defector) below the failure threshold because of the presence of advanced strategies. However, with regards to individual performance, group 6 is safer for the low ability Naive Defector because it performs better than in group 5.

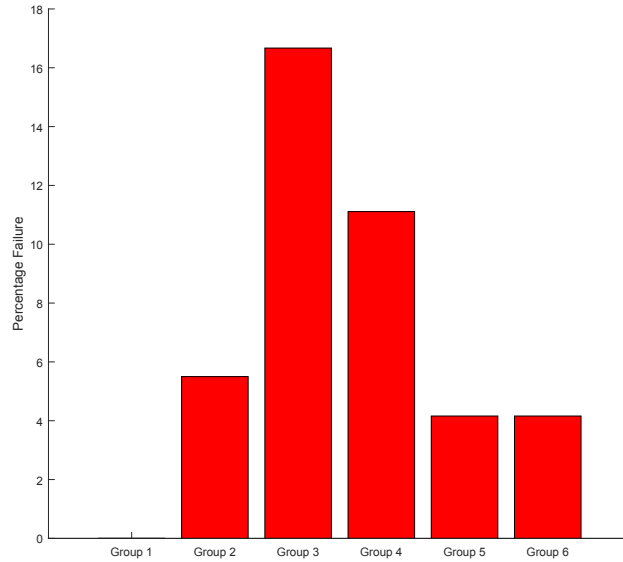


Figure 5.7: Failure Percentage distribution for different group composition

### 5.3.2 Group Total Score

We compare the group total score between agents in group composition 3 and 5 that has LimitedC-Fixed strategy with group 4 and 6 that has LimitedC-Ratio strategy due to shared similarities in the number of agents and other kinds of strategies. Our observation as shown in Figure 5.8 indicates that the overall group total score in group 4 (3173.4) is slightly higher than group 3 (2993.4). Similarly, a comparison between group 5 and 6 as shown in Figure 5.9 indicates that group structure 6 reports a group total score of (5785.2) whilst group 5 only had (5542.8). The higher group total score in composition 4 and 6 comparable to 3 and 5 respectively is because the LimitedC-Ratio strategy which is implemented in group 4 and 6 cooperates with some lower-ability defectors against which LimitedC-Fixed strategy defects.

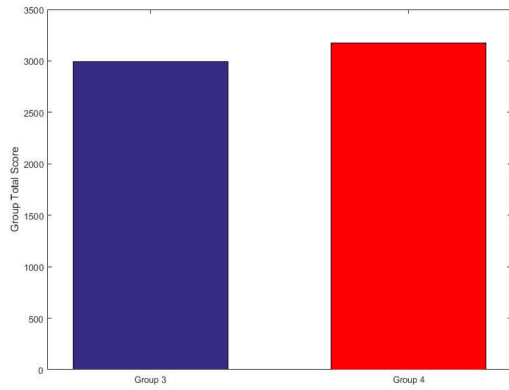


Figure 5.8: Group Total Score: 3 vs. 4

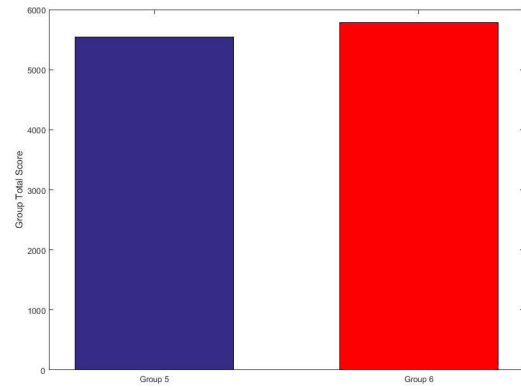


Figure 5.9: Group Total Score: 5 vs. 6

### 5.3.3 Group Average Score

The group average score provide an exact method to compare group success between group 3 vs. 4 and 5 vs. 6. We observe a group average score of 176.3 and 241.05 in group 4 and 6 respectively. This is slightly higher than the comparable average group score of 166.3 and 230.95 reported in group 3 and 5 respectively. Figures 5.10 and 5.11 display our results.

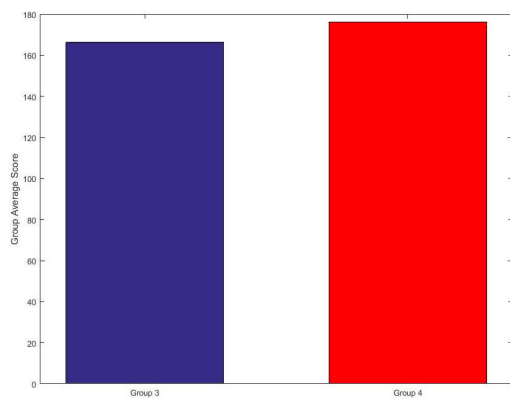


Figure 5.10: Group Avg Score: 3 vs. 4

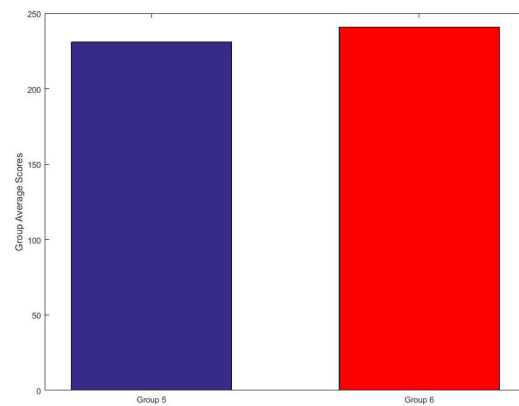


Figure 5.11: Group Avg Score: 5 vs. 6



### 5.3.4 Group Adjusted Average Scores

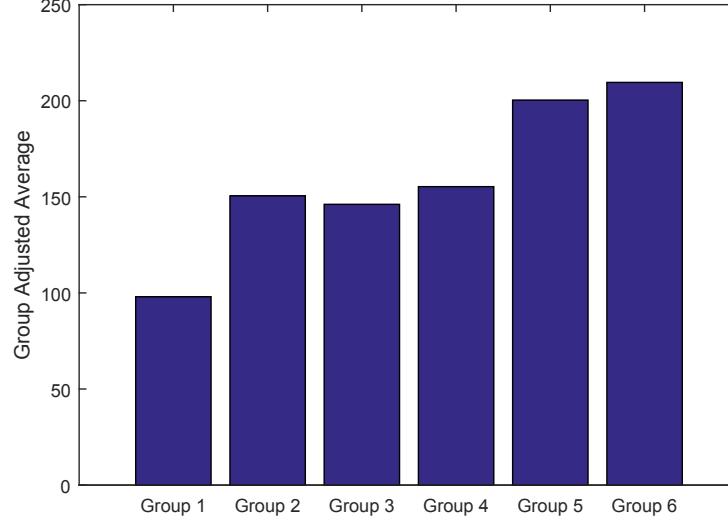


Figure 5.12: Group adjusted average scores

Across all 6 group structures, we measure and compare the performance of each group based on their group adjusted average scores. We observe that;

- The combination of Selective Cooperators and LimitedC-Ratio strategies in group 6 promotes cooperation and results in the highest group adjusted average score of 209.55. This is higher than the adjusted average score observed in group 5 (200.35).
- The individually rational approach of LimitedC-Ratio strategy gives group 4 a higher adjusted average score than group 3.
- The adjusted average score of group 2 (150.5) is higher than group 3 (146.1) but slightly lower than group 4 (155.3).
- The lowest group adjusted average score (98) was recorded in group 1 because it consist of only naive agents that make decisions without any form of information gathering mechanism.

## 5.4 Overall Observation and Analysis

- In all group compositions, we observe that increasing ability levels positively affect the individual scores of agents due to high impact of individual activity.
- In all instances, we have observed that the presence of Selective Cooperators positively favour the performance of Naive Cooperators but reduce the individual performance of Naive Defectors.
- The use of acquired information assist advanced agents to increase their individual performance and also support the development of cooperation. This was observed in group compositions 2, 3, 4, 5 and 6.
- A generalized approach to cooperation and defection based on ability levels increases group failure percentage. Group success increases when agents that cooperate or defect based on abilities decide individually.
- Naively cooperating or defecting does not promote group success even though it may support the performance of some individuals.
- A comparison of the group adjusted average performance between group 2 and group 4 leads us to conclude that group success is not only about naively helping cooperators but more about using an advanced approach to cooperate by also considering ability levels.
- Furthermore, we can infer from the adjusted average score of group composition 6 that an appropriate proportion of strategies that cooperate based on gathered information on partner's strategy and ability levels increases the overall group success.

## Chapter 6

# Conclusions and Future Work

In this thesis, we have introduced, implemented, and investigated a novel modelling and simulation framework for the study of cooperation among self-interested players in multiagent systems (MAS). As the basis for our game-theoretic modelling, we adopt the Ability Based Cooperation (ABC) game and the Extended ABC (EABC) game defined within our research group at UNBC. The framework allows for the modelling of several common aspects of natural and artificial systems that, to the best of our knowledge, have not been adequately addressed in the existing literature.

The specific contributions of the framework include:

- Balancing individual activity and interaction. A player in our study can make progress through individual activity or by interaction with others.
- The abilities of players are included in the modelling. The scores from both individual activities and interactions depend on the individual abilities of the

players. This means that the payoff a player receives from an encounter is based on the ability level of the player and/or its partner and the base payoff.

- We define several advanced agent strategies that use knowledge about partners' abilities or strategies in deciding whether or not to cooperate. Our interest is in the strategic effect of acquired information and not in the mechanism for information acquisition.
- There is a balance between individual and group success. The players' abilities, strategies, and the group composition influence both individual scores and the success of the group as a whole. Our framework provides the possibility to measure both kinds of success. Individual strategies can be designed with the intent to achieve specific impact with respect to individual and group performance.
- We evaluate individual strategies as to how much they contribute to the individual success of the players and the group success. This is done to observe the influence of information gathering on individual achievement and overall group performance.

We have demonstrated the properties of this framework with a comparative experimental study of six group compositions, each with a different strategy combination and a full range of individual abilities. For each strategy, the study allows us to analyze its performance in different group compositions, its impact upon other strategies, and its impact upon group performance.

The results lead us to conclude that the framework is a suitable tool for such studies, and expect that it could be successfully applied to more complex strategies and group compositions.

In future studies, we can:

- explore higher levels of modelling where different mechanisms of information gathering can be examined and their impact on individual and group success analyzed;
- introduce and observe increasingly complex groups of agent strategies such as an advanced strategy that seeks to promote ability development among group members;
- investigate the influence of other experimenter-controlled limitations such as pay-off matrix on individual and group performance.

# Bibliography

Michael Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, June 2009.

Robert M. Axelrod. *The Evolution of Cooperation*. Basic Books Inc., New York, December 2006.

Maurice Grinberg, Evgenia Hristova, and Emilian Lalev. Models for cooperative decisions in prisoner’s dilemma. In S. Nefti and J. Gray, editors, *Advances in Cognitive Systems*, chapter 7, pages 169–210. Institute of Engineering and Technology: London, 2010.

Tobias Kretz. A round-robin tournament of the iterated prisoner’s dilemma with complete memory-size-three strategies. *Complex Systems*, 19(4):363–389, January 2011.

Satoru Takahashi. Community enforcement when players observe partners’ past play. *Journal of Economic Theory*, 145(1):42–62, January 2010.

Jernej Polajnar and Desa Polajnar. An asymmetric two-player cooperation game based on player abilities. Modelling suggestions for Jonathan Asante Nyantakyi, unpublished short paper, UNBC, June 2018.

- Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2009.
- Jacques Ferber and Gerhard Weiss. *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*, volume 1. Addison-Wesley Reading, 1999.
- Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, third edition, 2010.
- Michael Wooldridge and Nicholas R. Jennings. The cooperative problem-solving process. *Journal of Logic and Computation*, 9(4):563–592, 1999.
- Simon Parsons and Michael Wooldridge. Game theory and decision theory in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 5(3):243–254, September 2002.
- Robert Axelrod and William D. Hamilton. The evolution of cooperation. *Science Journal*, 211(27):1390–1396, March 1981.
- Daniel Friedman. On economic applications of evolutionary game theory. *Journal of Evolutionary Economics*, 8(1):15–43, 1998.
- Jörgen W. Weibull. *Evolutionary Game Theory*. MIT press, 1997.
- John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behaviour*. Princeton University Press, 1944.
- Robert J. Aumann. Game theory. In John Eatwell, Murray Milgate, and Peter Newman, editors, *The New Palgrave Game Theory*, volume 4 of *The New Palgrave*, chapter 1, pages 1–53. Palgrave Macmillan UK, 1 edition, 1989.
- Guillermo Owen. Applications of game theory to economics. *International Game Theory Review*, 15(03):8, September 2013.

- Zach Ernst. Game theory in evolutionary biology. *Philosophy after Darwin: Classic and Contemporary Readings*, pages 464–76, 2009.
- Parag C. Pendharkar. Game theoretical applications for multi-agent systems. *Expert Systems with Applications*, 39(1):273–279, January 2012.
- Martin Beckenkamp, Heike Hennig-Schmidt, and Frank P. Maier-Rigaud. Cooperation in symmetric and asymmetric prisoner’s dilemma games. *Max Planck Institute for Research on Collective Goods*, 25:40, March 2007.
- Yuval Heller and Erik Mohlin. Observations on cooperation. *Social Science Research Network*, pages 1–62, April 2016.
- S. Chong, J. Humble, G. Kendall, J. Li, and X. Yao. Iterated prisoner’s dilemma and evolutionary game theory. In Graham Kendall, Xin Yao, and Yew Chong, Siang, editors, *The Iterated Prisoners Dilemma: 20 Years On*, volume 20 of *Advances in Neural Computations*, chapter 2, pages 23–62. Singapore: World Scientific, 2007.
- M. P. Kim, W. Suksompong, and V. V. Williams. Who can win a single-elimination tournament? *Society for Industrial and Applied Mathematics Journal on Discrete Mathematics*, 31(3):1751–1764, 2017.
- W. A. Glenn. A comparison of the effectiveness of tournaments. *Biometrika*, 47(3/4): 253–262, 1960.
- Rasmus V. Rasmussen and Michael A Trick. Round robin scheduling—a survey. *European Journal of Operational Research*, 188(3):617–636, June 2008.
- F. C. Santos, M. D. Santos, and J. M. Pacheco. Social diversity promotes the emergence of cooperation in public goods games. *Nature*, 454(7201):213, 2008.



- M. O. Souza, J. M. Pacheco, and F. C. Santos. Evolution of cooperation under n-person snowdrift games. *Journal of Theoretical Biology*, 260(4):581–588, 2009.
- Garret Ridinger and Michael McBride. Theory of mind ability and cooperation in the prisoner’s dilemma. Technical report, Working paper, 2016.
- Toh-Kyeong Ahn, Myungsuk Lee, Lore Ruttan, and James Walker. Asymmetric payoffs in simultaneous and sequential prisoner’s dilemma games. *Public Choice*, 132(3-4):353–366, 2007.
- J. P. Sheposh and Philip S. Gallo Jr. Asymmetry of payoff structure and cooperative behaviour in the prisoner’s dilemma game. *Journal of Conflict Resolution*, 17(2):321–333, 1973.
- Keith J. Murnighan. Cooperating when you know your outcomes will differ. *Simulation & Gaming*, 22(4):463–475, 1991.
- Ian Sommerville. *Software Engineering*. International Computer Science. Pearson, ninth edition, 2011. ISBN 0321210263.

# Appendix A

## A.1 Graphs and Tables of Results

### A.1.1 Group Composition 1

#### Parameter Settings

- *Agent strategies* ( $k = 2$ ): NaiveC = 6, NaiveD = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 12
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9

- *Number of tournament iterations:* 1

## Formulas

- Individual Scores of agent  $i$ :

– *Score vs. agent  $j$ :*  $s_i^{(j)} = \alpha a_i + (1 - \alpha) p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .

– *Total score:*  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$

– *Average score:*  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$

– *Adjusted total score:*  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$

– *Adjusted average score:*  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .

- Group Scores:

– *Group total score:*  $s^{tot} = \sum_1^n s_i^{tot}$

– *Group average score:*  $s^{avg} = \frac{s^{tot}}{n}$

– *Group adjusted total score:*  $s^{atot} = \sum_1^n s_i^{atot}$

– *Group adjusted average score:*  $s^{aavg} = \frac{s^{atot}}{n}$

## Experimental Results

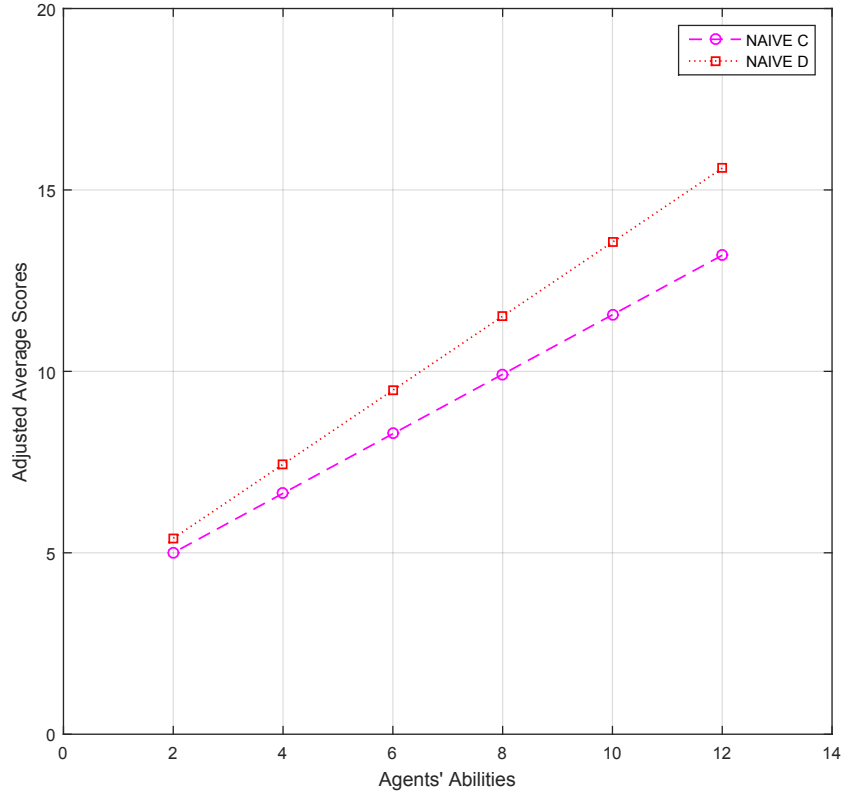


Figure A.1: Agents' performance for different ability levels in composition group 1

ID	Strategy	Ability	Total Score	Average Scores	Adjusted Total Scores	Adjusted Average Scores
1	NaiveC	12	142.80	12.98	132.00	13.20
2	NaiveD	12	178.80	16.25	156.00	15.60
3	NaiveC	10	124.60	11.33	115.60	11.56
4	NaiveD	10	154.60	14.05	135.60	13.56
5	NaiveC	8	106.40	9.67	99.20	9.92
6	NaiveD	8	130.40	11.85	115.20	11.52
7	NaiveC	6	88.20	8.02	82.80	8.28
8	NaiveD	6	106.20	9.65	94.80	9.48
9	NaiveC	4	70.00	6.36	66.40	6.64
10	NaiveD	4	82.00	7.45	74.40	7.44
11	NaiveC	2	51.80	4.71	50.00	5.00
12	NaiveD	2	57.80	5.25	54.00	5.40

Table A.1: Agents' average and adjusted average scores for Group Composition 1

		1	2	3	4	5	6	7	8	9	10	11	12
		NC	ND	NC	ND	NC	ND	NC	ND	NC	ND	NC	ND
1	NC		10.80	18.80	10.80	17.20	10.80	15.60	10.80	14.00	10.80	12.40	10.80
2	ND	22.80		21.20	13.20	19.60	13.20	18.00	13.20	16.40	13.20	14.80	13.20
3	NC	18.60	9.00		9.00	15.40	9.00	13.80	9.00	12.20	9.00	10.60	9.00
4	ND	20.60	11.00	19.00		17.40	11.00	15.80	11.00	14.20	11.00	12.60	11.00
5	NC	16.80	7.20	15.20	7.20		7.20	12.00	7.20	10.40	7.20	8.80	7.20
6	ND	18.40	8.80	16.80	8.80	15.20		13.60	8.80	12.00	8.80	10.40	8.80
7	NC	15.00	5.40	13.40	5.40	11.80	5.40		5.40	8.60	5.40	7.00	5.40
8	ND	16.20	6.60	14.60	6.60	13.00	6.60	11.40		9.80	6.60	8.20	6.60
9	NC	13.20	3.60	11.60	3.60	10.00	3.60	8.40	3.60		3.60	5.20	3.60
10	ND	14.00	4.40	12.40	4.40	10.80	4.40	9.20	4.40	7.60		6.00	4.40
11	NC	11.40	1.80	9.80	1.80	8.20	1.80	6.60	1.80	5.00	1.80		1.80
12	ND	11.80	2.20	10.20	2.20	8.60	2.20	7.00	2.20	5.40	2.20	3.80	

Table A.2: Agents' scores Group Composition 1

## A.1.2 Group Composition 2

### Parameter Settings

- *Agent strategies* ( $k = 3$ ): NaiveC = 6, NaiveD = 6, SelectiveC = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 18
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9
- *Number of tournament iterations*: 1

### Formulas

- Individual Scores of agent i:
  - *Score vs. agent j*:  $s_i^{(j)} = \alpha a_i + (1 - \alpha)p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .
  - *Total score*:  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
  - *Average score*:  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
  - *Adjusted total score*:  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
  - *Adjusted average score*:  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .
- Group Scores:
  - *Group total score*:  $s^{tot} = \sum_1^n s_i^{tot}$
  - *Group average score*:  $s^{avg} = \frac{s^{tot}}{n}$

- Group adjusted total score:  $s^{atot} = \sum_1^n s_i^{atot}$
- Group adjusted average score:  $s^{avg} = \frac{s^{atot}}{n}$

## Experimental Results

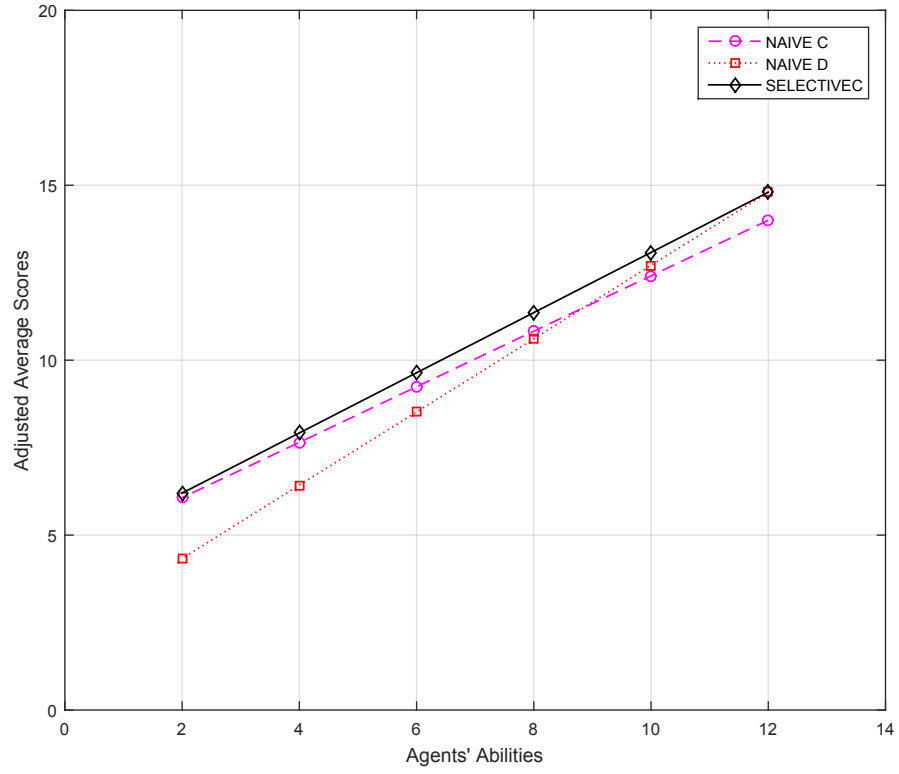


Figure A.2: Agents' performance for different ability levels in composition group 2

<b>ID</b>	<b>Strategy</b>	<b>Ability</b>	<b>Total Score</b>	<b>Average Scores</b>	<b>Total Adjusted Scores</b>	<b>Average Adjusted Scores</b>
1	NaiveC	12	241.20	14.19	210.00	14.00
2	NaiveD	12	258.00	15.18	222.00	14.80
3	SelectiveC	12	255.60	15.04	222.00	14.80
4	NaiveC	10	212.20	12.48	186.20	12.41
5	NaiveD	10	220.60	12.98	190.60	12.71
6	SelectiveC	10	224.20	13.19	196.20	13.08
7	NaiveC	8	183.20	10.78	162.40	10.83
8	NaiveD	8	183.20	10.78	159.20	10.61
9	SelectiveC	8	192.80	11.34	170.40	11.36
10	NaiveC	6	154.20	9.07	138.60	9.24
11	NaiveD	6	145.80	8.58	127.80	8.52
12	SelectiveC	6	161.40	9.49	144.60	9.64
13	NaiveC	4	125.20	7.36	114.80	7.65
14	NaiveD	4	108.40	6.38	96.40	6.43
15	SelectiveC	4	130.00	7.65	118.80	7.92
16	NaiveC	2	96.20	5.66	91.00	6.07
17	NaiveD	2	71.00	4.18	65.00	4.33
18	SelectiveC	2	98.60	5.80	93.00	6.20

Table A.3: Agents’ average and adjusted average scores for Group Composition 2



$\infty$ 

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
		NC	ND	SC	NC	ND	SC	NC	ND	SC	NC	ND	SC	NC	ND	SC	NC	ND	SC
1	NC		10.80	20.40	18.80	10.80	18.80	17.20	10.80	17.20	15.60	10.80	15.60	14.00	10.80	14.00	12.40	10.80	12.40
2	ND	22.80		13.20	21.20	13.20	13.20	19.60	13.20	13.20	18.00	13.20	13.20	16.40	13.20	13.20	14.80	13.20	13.20
3	SC	20.40	13.20		18.80	13.20	18.80	17.20	13.20	17.20	15.60	13.20	15.60	14.00	13.20	14.00	12.40	13.20	12.40
4	NC	18.60	9.00	18.60		9.00	17.00	15.40	9.00	15.40	13.80	9.00	13.80	12.20	9.00	12.20	10.60	9.00	10.60
5	ND	20.60	11.00	11.00	19.00		11.00	17.40	11.00	11.00	15.80	11.00	11.00	14.20	11.00	11.00	12.60	11.00	11.00
6	SC	18.60	11.00	18.60	17.00	11.00		15.40	11.00	15.40	13.80	11.00	13.80	12.20	11.00	12.20	10.60	11.00	10.60
7	NC	16.80	7.20	16.80	15.20	7.20	15.20		7.20	13.60	12.00	7.20	12.00	10.40	7.20	10.40	8.80	7.20	8.80
8	ND	18.40	8.80	8.80	16.80	8.80	8.80	15.20		8.80	13.60	8.80	8.80	12.00	8.80	8.80	10.40	8.80	8.80
9	SC	16.80	8.80	16.80	15.20	8.80	15.20	13.60	8.80		12.00	8.80	12.00	10.40	8.80	10.40	8.80	8.80	8.80
10	NC	15.00	5.40	15.00	13.40	5.40	13.40	11.80	5.40	11.80		5.40	10.20	8.60	5.40	8.60	7.00	5.40	7.00
11	ND	16.20	6.60	6.60	14.60	6.60	6.60	13.00	6.60	6.60	11.40		6.60	9.80	6.60	6.60	8.20	6.60	6.60
12	SC	15.00	6.60	15.00	13.40	6.60	13.40	11.80	6.60	11.80	10.20	6.60		8.60	6.60	8.60	7.00	6.60	7.00
13	NC	13.20	3.60	13.20	11.60	3.60	11.60	10.00	3.60	10.00	8.40	3.60	8.40		3.60	6.80	5.20	3.60	5.20
14	ND	14.00	4.40	4.40	12.40	4.40	4.40	10.80	4.40	4.40	9.20	4.40	4.40	7.60		4.40	6.00	4.40	4.40
15	SC	13.20	4.40	13.20	11.60	4.40	11.60	10.00	4.40	10.00	8.40	4.40	8.40	6.80	4.40		5.20	4.40	5.20
16	NC	11.40	1.80	11.40	9.80	1.80	9.80	8.20	1.80	8.20	6.60	1.80	6.60	5.00	1.80	5.00		1.80	3.40
17	ND	11.80	2.20	2.20	10.20	2.20	2.20	8.60	2.20	2.20	7.00	2.20	2.20	5.40	2.20	2.20	3.80		2.20
18	SC	11.40	2.20	11.40	9.80	2.20	9.80	8.20	2.20	8.20	6.60	2.20	6.60	5.00	2.20	5.00	3.40	2.20	

Table A.4: Agents' scores Group Composition 2

### A.1.3 Group Composition 3

#### Parameter Settings

- *Agent strategies* ( $k = 3$ ): NaiveC = 6, NaiveD = 6, LimitedC-Fixed = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 18
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9
- *Number of tournament iterations*: 1

#### Formulas

- Individual Scores of agent i:
  - *Score vs. agent j*:  $s_i^{(j)} = \alpha a_i + (1 - \alpha)p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .
  - *Total score*:  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
  - *Average score*:  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
  - *Adjusted total score*:  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
  - *Adjusted average score*:  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .
- Group Scores:
  - *Group total score*:  $s^{tot} = \sum_1^n s_i^{tot}$
  - *Group average score*:  $s^{avg} = \frac{s^{tot}}{n}$

- Group adjusted total score:  $s^{atot} = \sum_1^n s_i^{atot}$
- Group adjusted average score:  $s^{avg} = \frac{s^{atot}}{n}$

## Experimental Results

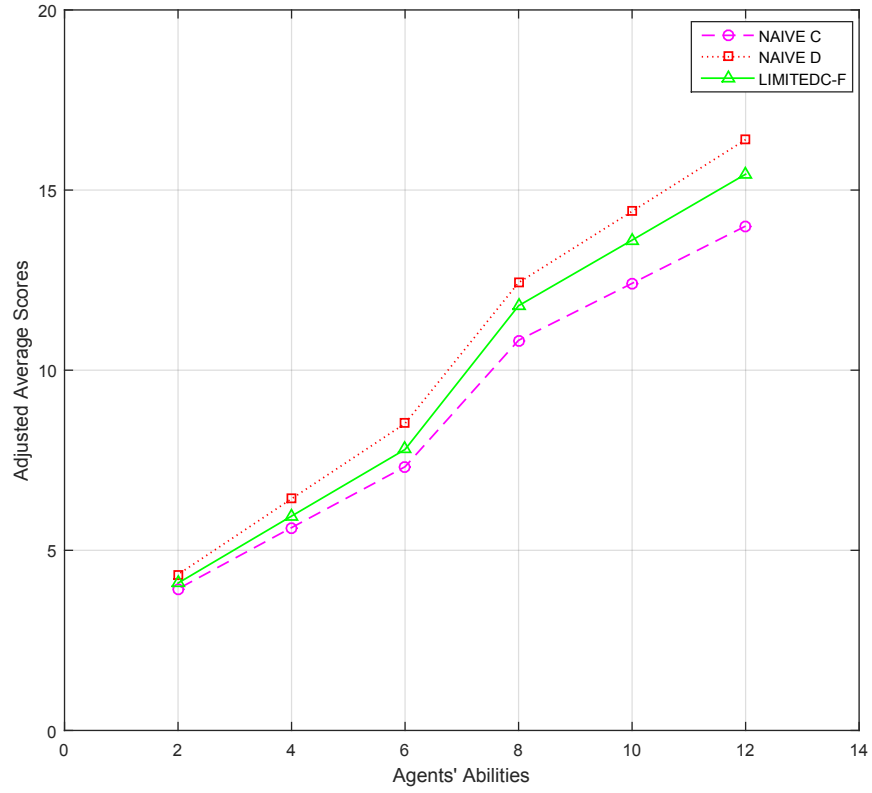


Figure A.3: Agents' performance for different ability levels in group composition 3

<b>ID</b>	<b>Strategy</b>	<b>Ability</b>	<b>Total Score</b>	<b>Average Scores</b>	<b>Total Adjusted Scores</b>	<b>Average Adjusted Scores</b>
1	NaiveC	12	241.20	14.19	210.00	14.00
2	NaiveD	12	291.60	17.15	246.00	16.40
3	LC-F	12	262.80	15.46	231.60	15.44
4	NaiveC	10	212.20	12.48	186.20	12.41
5	NaiveD	10	254.20	14.95	216.20	14.41
6	LC-F	10	230.20	13.54	204.20	13.61
7	NaiveC	8	183.20	10.78	162.40	10.83
8	NaiveD	8	216.80	12.75	186.40	12.43
9	LC-F	8	197.60	11.62	176.80	11.79
10	NaiveC	6	120.60	7.09	109.80	7.32
11	NaiveD	6	145.80	8.58	127.80	8.52
12	LC-F	6	135.00	7.94	117.00	7.80
13	NaiveC	4	91.60	5.39	84.40	5.63
14	NaiveD	4	108.40	6.38	96.40	6.43
15	LC-F	4	101.20	5.95	89.20	5.95
16	NaiveC	2	62.60	3.68	59.00	3.93
17	NaiveD	2	71.00	4.18	65.00	4.33
18	LC-F	2	67.40	3.96	61.40	4.09

Table A.5: Agents' average and adjusted average scores for Group Composition 3

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
		NC	ND	LC-F	NC	ND	LC-F	NC	ND	LC-F	NC	ND	LC-F	NC	ND	LC-F	NC	ND	LC-F
1	NC		10.80	20.40	18.80	10.80	18.80	17.20	10.80	17.20	15.60	10.80	15.60	14.00	10.80	14.00	12.40	10.8	12.40
2	ND	22.8		22.8	21.2	13.2	21.2	19.6	13.2	19.6	18	13.2	18	16.4	13.2	16.4	14.8	13.2	14.8
3	LC-F	20.40	10.80		18.80	10.80	18.80	17.20	10.80	17.20	18.00	13.20	18.00	16.40	13.20	16.40	14.80	13.20	14.80
4	NC	18.60	9.00	18.60		9.00	17.00	15.40	9.00	15.40	13.80	9.00	13.80	12.20	9.00	12.20	10.60	9.00	10.60
5	ND	20.60	11.00	20.60	19.00		19.00	17.40	11.00	17.40	15.80	11.00	15.80	14.20	11.00	14.20	12.60	11.00	12.60
6	LC-F	18.60	9.00	18.60	17.00	9.00		15.40	9.00	15.40	15.80	11.00	15.80	14.20	11.00	14.20	12.60	11.00	12.60
7	NC	16.80	7.20	16.80	15.20	7.20	15.20		7.20	13.60	12.00	7.20	12.00	10.40	7.20	10.40	8.80	7.20	8.80
8	ND	18.40	8.80	18.40	16.80	8.80	16.80	15.20		15.20	13.60	8.80	13.60	12.00	8.80	12.00	10.40	8.80	10.40
9	LC-F	16.80	7.20	16.80	15.20	7.20	15.20	13.60	7.20		13.60	8.80	13.60	12.00	8.80	12.00	10.40	8.80	10.40
10	NC	15.00	5.40	5.40	13.40	5.40	5.40	11.80	5.40	5.40		5.40	5.40	8.60	5.40	5.40	7.00	5.40	5.40
11	ND	16.20	6.60	6.60	14.60	6.60	6.60	13.00	6.60	6.60	11.40		6.60	9.80	6.60	6.60	8.20	6.60	6.60
12	LC-F	15.00	5.40	5.40	13.40	5.40	5.40	11.80	5.40	5.40	11.40	6.60		9.80	6.60	6.60	8.20	6.60	6.60
13	NC	13.20	3.60	3.60	11.60	3.60	3.60	10.00	3.60	3.60	8.40	3.60	3.60		3.60	3.60	5.20	3.60	3.60
14	ND	14.00	4.40	4.40	12.40	4.40	4.40	10.80	4.40	4.40	9.20	4.40	4.40	7.60		4.40	6.00	4.40	4.40
15	LC-F	13.20	3.60	3.60	11.60	3.60	3.60	10.00	3.60	3.60	9.20	4.40	4.40	7.60	4.40		6.00	4.40	4.40
16	NC	11.40	1.80	1.80	9.80	1.80	1.80	8.20	1.80	1.80	6.60	1.80	1.80	5.00	1.80	1.80		1.80	1.80
17	ND	11.80	2.20	2.20	10.20	2.20	2.20	8.60	2.20	2.20	7.00	2.20	2.20	5.40	2.20	2.20	3.80		2.20
18	LC-F	11.40	1.80	1.80	9.80	1.80	1.80	8.20	1.80	1.80	7.00	2.20	2.20	5.40	2.20	2.20	3.80	2.20	

Table A.6: Agents' match scores for Group Composition 3

## A.1.4 Group Composition 4

### Parameter Settings

- *Agent strategies* ( $k = 3$ ): NaiveC = 6, NaiveD = 6, LimitedC-Ratio = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 18
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9
- *Number of tournament iterations*: 1

### Formulas

- Individual Scores of agent i:
  - *Score vs. agent j*:  $s_i^{(j)} = \alpha a_i + (1 - \alpha)p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .
  - *Total score*:  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
  - *Average score*:  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
  - *Adjusted total score*:  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
  - *Adjusted average score*:  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .
- Group Scores:
  - *Group total score*:  $s^{tot} = \sum_1^n s_i^{tot}$
  - *Group average score*:  $s^{avg} = \frac{s^{tot}}{n}$

- Group adjusted total score:  $s^{tot} = \sum_1^n s_i^{tot}$
- Group adjusted average score:  $s^{avg} = \frac{s^{tot}}{n}$

## Experimental Results

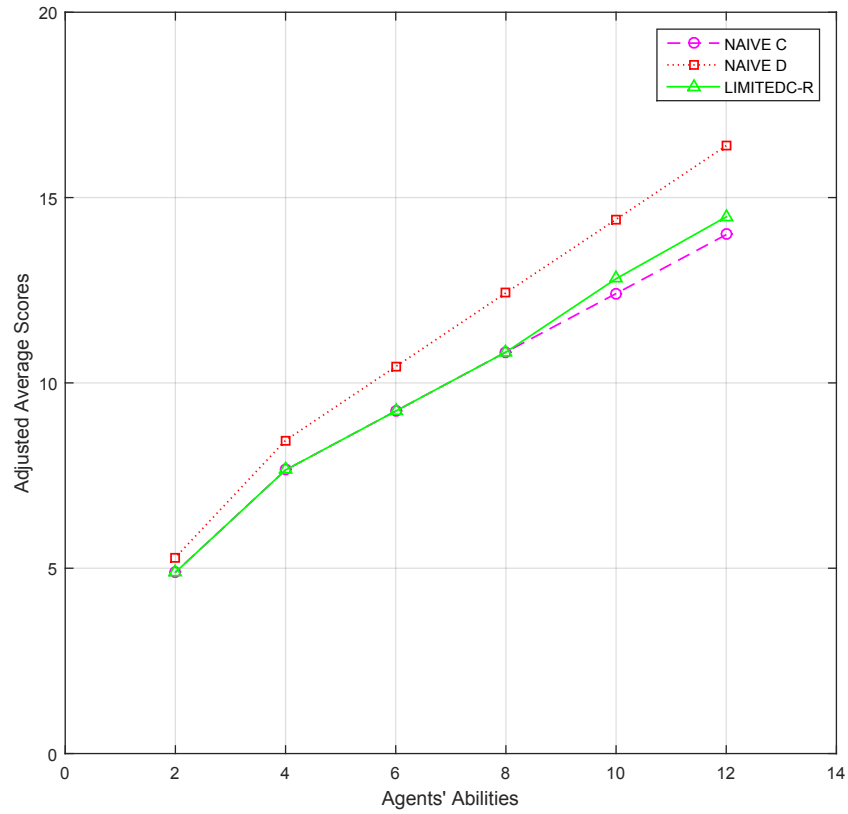


Figure A.4: Agents' performance for different ability levels in group composition 4

<b>ID</b>	<b>Strategy</b>	<b>Ability</b>	<b>Total Score</b>	<b>Average Scores</b>	<b>Total Adjusted Scores</b>	<b>Average Adjusted Scores</b>
1	NaiveC	12	241.20	14.19	210.00	14.00
2	NaiveD	12	291.60	17.15	246.00	16.40
3	LC-R	12	248.40	14.61	217.20	14.48
4	NaiveC	10	212.20	12.48	186.20	12.41
5	NaiveD	10	254.20	14.95	216.20	14.41
6	LC-R	10	218.20	12.84	192.20	12.81
7	NaiveC	8	183.20	10.78	162.40	10.83
8	NaiveD	8	216.80	12.75	186.40	12.43
9	LC-R	8	183.20	10.78	162.40	10.83
10	NaiveC	6	154.20	9.07	138.60	9.24
11	NaiveD	6	179.40	10.55	156.60	10.44
12	LC-R	6	154.20	9.07	138.60	9.24
13	NaiveC	4	125.20	7.36	114.80	7.65
14	NaiveD	4	142.00	8.35	126.80	8.45
15	LC-R	4	125.20	7.36	114.80	7.65
16	NaiveC	2	78.60	4.62	73.40	4.89
17	NaiveD	2	87.00	5.12	79.40	5.29
18	LC-R	2	78.60	4.62	73.40	4.89

Table A.7: Agents' average and adjusted average scores for Group Composition 4



		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
		NC	ND	LC-R	NC	ND	LC-R	NC	ND	LC-R	NC	ND	LC-R	NC	ND	LC-R	NC	ND	LC-R
1	NC		10.80	20.40	18.80	10.80	18.80	17.20	10.80	17.20	15.60	10.80	15.60	14.00	10.80	14.00	12.40	10.80	12.40
2	ND	22.80		22.80	21.20	13.20	21.20	19.60	13.20	19.60	18.00	13.20	18.00	16.40	13.20	16.40	14.80	13.20	14.80
3	LC-R	20.40	10.80		18.80	10.80	18.80	17.20	10.80	17.20	15.60	10.80	15.60	14.00	10.80	14.00	14.80	13.20	14.80
4	NC	18.60	9.00	18.60		9.00	17.00	15.40	9.00	15.40	13.80	9.00	13.80	12.20	9.00	12.20	10.60	9.00	10.60
5	ND	20.60	11.00	20.60	19.00		19.00	17.40	11.00	17.40	15.80	11.00	15.80	14.20	11.00	14.20	12.60	11.00	12.60
6	LC-R	18.60	9.00	18.60	17.00	9.00		15.40	9.00	15.40	13.80	9.00	13.80	12.20	9.00	12.20	12.60	11.00	12.60
7	NC	16.80	7.20	16.80	15.20	7.20	15.20		7.20	13.60	12.00	7.20	12.00	10.40	7.20	10.40	8.80	7.20	8.80
8	ND	18.40	8.80	18.40	16.80	8.80	16.80	15.20		15.20	13.60	8.80	13.60	12.00	8.80	12.00	10.40	8.80	10.40
9	LC-R	16.80	7.20	16.80	15.20	7.20	15.20	13.60	7.20		12.00	7.20	12.00	10.40	7.20	10.40	8.80	7.20	8.80
10	NC	15.00	5.40	15.00	13.40	5.40	13.40	11.80	5.40	11.80		5.40	10.20	8.60	5.40	8.60	7.00	5.40	7.00
11	ND	16.20	6.60	16.20	14.60	6.60	14.60	13.00	6.60	13.00	11.40		11.40	9.80	6.60	9.80	8.20	6.60	8.20
12	LC-R	15.00	5.40	15.00	13.40	5.40	13.40	11.80	5.40	11.80	10.20	5.40		8.60	5.40	8.60	7.00	5.40	7.00
13	NC	13.20	3.60	13.20	11.60	3.60	11.60	10.00	3.60	10.00	8.40	3.60	8.40		3.60	6.80	5.20	3.60	5.20
14	ND	14.00	4.40	14.00	12.40	4.40	12.40	10.80	4.40	10.80	9.20	4.40	9.20	7.60		7.60	6.00	4.40	6.00
15	LC-R	13.20	3.60	13.20	11.60	3.60	11.60	10.00	3.60	10.00	8.40	3.60	8.40	6.80	3.60		5.20	3.60	5.20
16	NC	11.40	1.80	1.80	9.80	1.80	1.80	8.20	1.80	8.20	6.60	1.80	6.60	5.00	1.80	5.00		1.80	3.40
17	ND	11.80	2.20	2.20	10.20	2.20	2.20	8.60	2.20	8.60	7.00	2.20	7.00	5.40	2.20	5.40	3.80		3.80
18	LC-R	11.40	1.80	1.80	9.80	1.80	1.80	8.20	1.80	8.20	6.60	1.80	6.60	5.00	1.80	5.00	3.40	1.80	

Table A.8: Agents' match scores for Group Composition 4

## A.1.5 Group Composition 5

### Parameter Settings

- *Agent strategies* ( $k = 4$ ): NaiveC = 6, NaiveD = 6, SelectiveC = 6, LimitedC-Fixed = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 24
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9
- *Number of tournament iterations*: 1

### Formulas

- Individual Scores of agent i:
  - *Score vs. agent j*:  $s_i^{(j)} = \alpha a_i + (1 - \alpha) p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .
  - *Total score*:  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
  - *Average score*:  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
  - *Adjusted total score*:  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
  - *Adjusted average score*:  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .
- Group Scores:
  - *Group total score*:  $s^{tot} = \sum_1^n s_i^{tot}$

- Group average score:  $s^{avg} = \frac{s^{tot}}{n}$
- Group adjusted total score:  $s^{atot} = \sum_1^n s_i^{atot}$
- Group adjusted average score:  $s^{aavg} = \frac{s^{atot}}{n}$

## Experimental Results

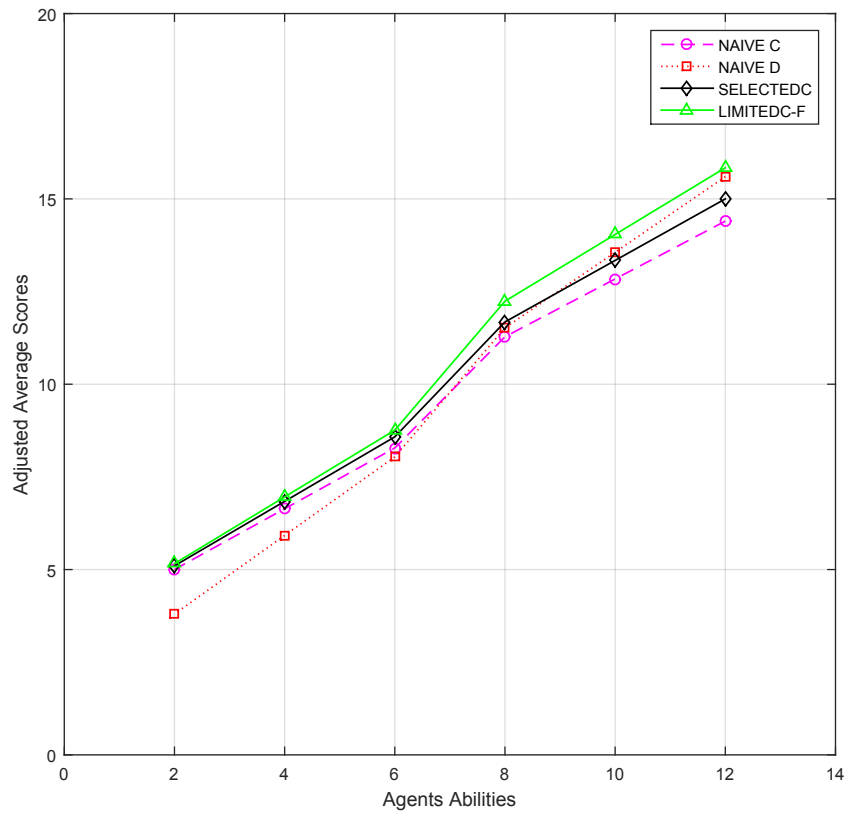


Figure A.5: Agents' performance for different ability levels in group composition 5

<b>ID</b>	<b>Strategy</b>	<b>Abilities</b>	<b>Total Score</b>	<b>Average Scores</b>	<b>Adjusted Total Score</b>	<b>Adjusted Average Score</b>
1	NaiveC	12	339.60	14.77	288.00	14.40
2	NaiveD	12	370.80	16.12	312.00	15.60
3	SelectiveC	12	354.00	15.39	300.00	15.00
4	LC-F	12	368.40	16.02	316.80	15.84
5	NaiveC	10	299.80	13.03	256.80	12.84
6	NaiveD	10	320.20	13.92	271.20	13.56
7	SelectiveC	10	311.80	13.56	266.80	13.34
8	LC-F	10	323.80	14.08	280.80	14.04
9	NaiveC	8	260.00	11.30	225.60	11.28
10	NaiveD	8	269.60	11.72	230.40	11.52
11	SelectiveC	8	269.60	11.72	233.60	11.68
12	LC-F	8	279.20	12.14	244.80	12.24
13	NaiveC	6	186.60	8.11	165.60	8.28
14	NaiveD	6	185.40	8.06	160.80	8.04
15	SelectiveC	6	193.80	8.43	171.60	8.58
16	LC-F	6	204.60	8.90	175.20	8.76
17	NaiveC	4	146.80	6.38	132.80	6.64
18	NaiveD	4	134.80	5.86	118.40	5.92
19	SelectiveC	4	151.60	6.59	136.80	6.84
20	LC-F	4	158.80	6.90	139.20	6.96
21	NaiveC	2	107.00	4.65	100.00	5.00
22	NaiveD	2	84.20	3.66	76.00	3.80
23	SelectiveC	2	109.40	4.76	102.00	5.10
24	LC-F	2	113.00	4.91	103.20	5.16

Table A.9: Agents' average and adjusted average scores for Group Composition 5

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
		NC	ND	SC	LC-F	NC	ND	SC	LC-F	NC	ND	SC	LC-F	NC	ND	SC	LC-F	NC	ND	SC	LC-F	NC	ND	SC	LC-F
1	NC		10.80	20.40	20.40	18.80	10.80	18.80	18.80	17.20	10.80	17.20	17.20	15.60	10.80	15.60	15.60	14.00	10.80	14.00	14.00	12.40	10.80	12.40	12.40
2	ND	22.80		13.20	22.80	21.20	13.20	13.20	21.20	19.60	13.20	13.20	19.60	18.00	13.20	13.20	18.00	16.40	13.20	13.20	16.40	14.80	13.20	13.20	14.80
3	SC	20.40	13.20		20.40	18.80	13.20	18.80	18.80	17.20	13.20	17.20	17.20	15.60	13.20	15.60	15.60	14.00	13.20	14.00	14.00	12.40	13.20	12.40	12.40
4	LC-F	20.40	10.80	20.40		18.80	10.80	18.80	18.80	17.20	10.80	17.20	17.20	18.00	13.20	18.00	18.00	16.40	13.20	16.40	16.40	14.80	13.20	14.80	14.80
5	NC	18.60	9.00	18.60	18.60		9.00	17.00	17.00	15.40	9.00	15.40	15.40	13.80	9.00	13.80	13.80	12.20	9.00	12.20	12.20	10.60	9.00	10.60	10.60
6	ND	20.60	11.00	11.00	20.60	19.00		11.00	19.00	17.40	11.00	11.00	17.40	15.80	11.00	11.00	15.80	14.20	11.00	11.00	14.20	12.60	11.00	11.00	12.60
7	SC	18.60	11.00	18.60	18.60	17.00	11.00		17.00	15.40	11.00	15.40	15.40	13.80	11.00	13.80	13.80	12.20	11.00	12.20	12.20	10.60	11.00	10.60	10.60
8	LC-F	18.60	9.00	18.60	18.60	17.00	9.00	17.00		15.40	9.00	15.40	15.40	15.80	11.00	15.80	15.80	14.20	11.00	14.20	14.20	12.60	11.00	12.60	12.60
9	NC	16.80	7.20	16.80	16.80	15.20	7.20	15.20	15.20		7.20	13.60	13.60	12.00	7.20	12.00	12.00	10.40	7.20	10.40	10.40	8.80	7.20	8.80	8.80
10	ND	18.40	8.80	8.80	18.40	16.80	8.80	8.80	16.80	15.20		8.80	15.20	13.60	8.80	8.80	13.60	12.00	8.80	8.80	12.00	10.40	8.80	8.80	10.40
11	SC	16.80	8.80	16.80	16.80	15.20	8.80	15.20	15.20	13.60	8.80		13.60	12.00	8.80	12.00	12.00	10.40	8.80	10.40	10.40	8.80	8.80	8.80	8.80
12	LC-F	16.80	7.20	16.80	16.80	15.20	7.20	15.20	15.20	13.60	7.20	13.60		13.60	8.80	13.60	13.60	12.00	8.80	12.00	12.00	10.40	8.80	10.40	10.40
13	NC	15.00	5.40	15.00	5.40	13.40	5.40	13.40	5.40	11.80	5.40	11.80	5.40		5.40	10.20	5.40	8.60	5.40	8.60	5.40	7.00	5.40	7.00	5.40
14	ND	16.20	6.60	6.60	6.60	14.60	6.60	6.60	6.60	13.00	6.60	6.60	6.60	11.40		6.60	6.60	9.80	6.60	6.60	6.60	8.20	6.60	6.60	6.60
15	SC	15.00	6.60	15.00	5.40	13.40	6.60	13.40	5.40	11.80	6.60	11.80	5.40	10.20	6.60		5.40	8.60	6.60	8.60	5.40	7.00	6.60	7.00	5.40
16	LC-F	15.00	5.40	15.00	5.40	13.40	5.40	13.40	5.40	11.80	5.40	11.80	5.40	11.40	6.60	11.40		9.80	6.60	9.80	6.60	8.20	6.60	8.20	6.60
17	NC	13.20	3.60	13.20	3.60	11.60	3.60	11.60	3.60	10.00	3.60	10.00	3.60	8.40	3.60	8.40	3.60		3.60	6.80	3.60	5.20	3.60	5.20	3.60
18	ND	14.00	4.40	4.40	4.40	12.40	4.40	4.40	4.40	10.80	4.40	4.40	4.40	9.20	4.40	4.40	4.40	7.60		4.40	4.40	6.00	4.40	4.40	4.40
19	SC	13.20	4.40	13.20	3.60	11.60	4.40	11.60	3.60	10.00	4.40	10.00	3.60	8.40	4.40	8.40	3.60	6.80	4.40		3.60	5.20	4.40	5.20	3.60
20	LC-F	13.20	3.60	13.20	3.60	11.60	3.60	11.60	3.60	10.00	3.60	10.00	3.60	9.20	4.40	9.20	4.40	7.60	4.40	7.60		6.00	4.40	6.00	4.40
21	NC	11.40	1.80	11.40	1.80	9.80	1.80	9.80	1.80	8.20	1.80	8.20	1.80	6.60	1.80	6.60	1.80	5.00	1.80	5.00	1.80		1.80	3.40	1.80
22	ND	11.80	2.20	2.20	2.20	10.20	2.20	2.20	2.20	8.60	2.20	2.20	2.20	7.00	2.20	2.20	2.20	5.40	2.20	2.20	2.20	3.80		2.20	2.20
23	SC	11.40	2.20	11.40	1.80	9.80	2.20	9.80	1.80	8.20	2.20	8.20	1.80	6.60	2.20	6.60	1.80	5.00	2.20	5.00	1.80	3.40	2.20		1.80
24	LC-F	11.40	1.80	11.40	1.80	9.80	1.80	9.80	1.80	8.20	1.80	8.20	1.80	7.00	2.20	7.00	2.20	5.40	2.20	5.40	2.20	3.80	2.20	3.80	

Table A.10: Agents' match scores for Group Composition 5

## A.1.6 Group Composition 6

### Parameter Settings

- *Agent strategies* ( $k = 4$ ): NaiveC = 6, NaiveD = 6, SelectiveC = 6, LimitedC-Ratio = 6
- *Ability levels* ( $q = 6$ ): 12, 10, 8, 6, 4, 2
- *Number of agents* ( $n = k*q$ ): 24
- *Base payoffs*: T = 10, R = 8, P = 2, S = 0
- *Balancing factor* ( $\alpha$ ): = 0.9
- *Number of tournament iterations*: 1

### Formulas

- Individual Scores of agent i:
  - *Score vs. agent j*:  $s_i^{(j)} = \alpha a_i + (1 - \alpha) p_i^{(j)}$  where  $p_i^{(j)}$  is the payoff of  $i$  vs.  $j$ .
  - *Total score*:  $s_i^{tot} = \sum_{\substack{j=1 \\ j \neq i}}^n s_i^{(j)}$
  - *Average score*:  $s_i^{avg} = \frac{s_i^{tot}}{n-1}$
  - *Adjusted total score*:  $s_i^{atot} = \sum_{\substack{j=1 \\ a_j \neq a_i}}^n s_i^{(j)}$
  - *Adjusted average score*:  $s_i^{aavg} = \frac{s_i^{atot}}{k(q-1)}$ .
- Group Scores:
  - *Group total score*:  $s^{tot} = \sum_1^n s_i^{tot}$

- Group average score:  $s^{avg} = \frac{s^{tot}}{n}$
- Group adjusted total score:  $s^{atot} = \sum_1^n s_i^{atot}$
- Group adjusted average score:  $s^{aavg} = \frac{s^{atot}}{n}$

## Experimental Results

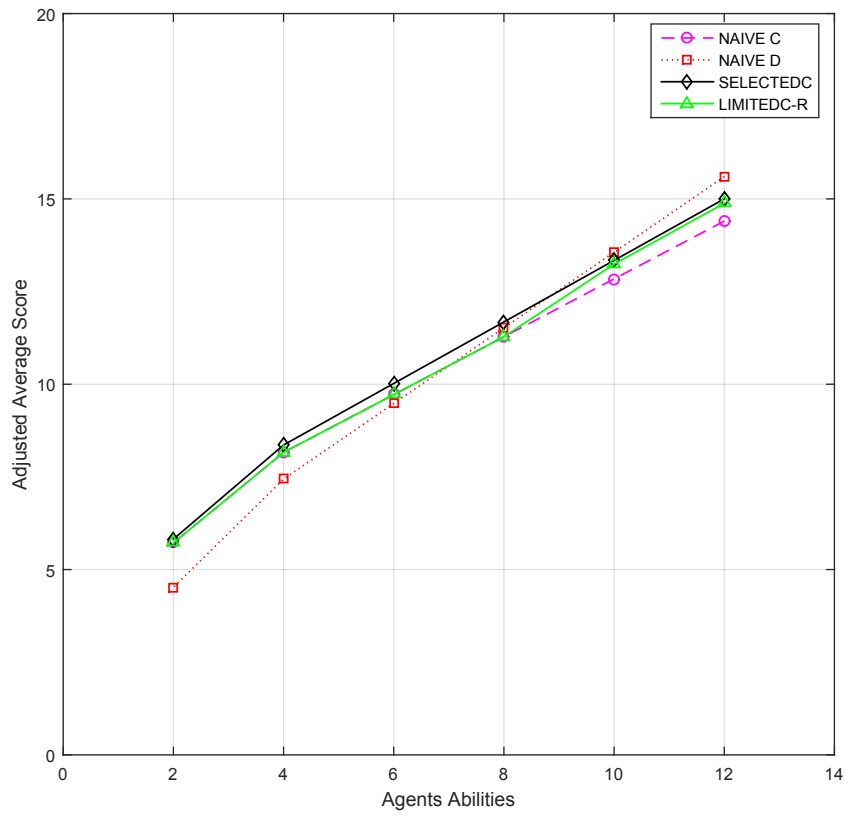


Figure A.6: Agents' performance for different ability levels in group composition 6

<b>ID</b>	<b>Strategy</b>	<b>Abilities</b>	<b>Total Score</b>	<b>Average Scores</b>	<b>Adjusted Total Score</b>	<b>Adjusted Average Score</b>
1	NaiveC	12	339.60	14.77	288.00	14.40
2	NaiveD	12	370.80	16.12	312.00	15.60
3	SelectiveC	12	354.00	15.39	300.00	15.00
4	LC-R	12	349.20	15.18	297.60	14.88
5	NaiveC	10	299.80	13.03	256.80	12.84
6	NaiveD	10	320.20	13.92	271.20	13.56
7	SelectiveC	10	311.80	13.56	266.80	13.34
8	LC-R	10	307.80	13.38	264.80	13.24
9	NaiveC	8	260.00	11.30	225.60	11.28
10	NaiveD	8	269.60	11.72	230.40	11.52
11	SelectiveC	8	269.60	11.72	233.60	11.68
12	LC-R	8	260.00	11.30	225.60	11.28
13	NaiveC	6	220.20	9.57	194.40	9.72
14	NaiveD	6	219.00	9.52	189.60	9.48
15	SelectiveC	6	227.40	9.89	200.40	10.02
16	LC-R	6	220.20	9.57	194.40	9.72
17	NaiveC	4	180.40	7.84	163.20	8.16
18	NaiveD	4	168.40	7.32	148.80	7.44
19	SelectiveC	4	185.20	8.05	167.20	8.36
20	LC-R	4	180.40	7.84	163.20	8.16
21	NaiveC	2	123.00	5.35	114.40	5.72
22	NaiveD	2	100.20	4.36	90.40	4.52
23	SelectiveC	2	125.40	5.45	116.40	5.82
24	LC-R	2	123.00	5.35	114.40	5.72

Table A.11: Agents' average and adjusted average scores for Group Composition 6



		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
		NC	ND	SC	LC-R	NC	ND	SC	LC-R	NC	ND	SC	LC-R	NC	ND	SC	LC-R	NC	ND	SC	LC-R	NC	ND	SC	LC-R
1	NC		10.80	20.40	20.40	18.80	10.80	18.80	18.80	17.20	10.80	17.20	17.20	15.60	10.80	15.60	15.60	14.00	10.80	14.00	14.00	12.40	10.80	12.40	12.40
2	ND	22.80		13.20	22.80	21.20	13.20	13.20	21.20	19.60	13.20	13.20	19.60	18.00	13.20	13.20	18.00	16.40	13.20	13.20	16.40	14.80	13.20	13.20	14.80
3	SC	20.40	13.20		20.40	18.80	13.20	18.80	18.80	17.20	13.20	17.20	17.20	15.60	13.20	15.60	15.60	14.00	13.20	14.00	14.00	12.40	13.20	12.40	12.40
4	LC-R	20.40	10.80	20.40		18.80	10.80	18.80	18.80	17.20	10.80	17.20	17.20	15.60	10.80	15.60	15.60	14.00	10.80	14.00	14.00	14.80	13.20	14.80	14.80
5	NC	18.60	9.00	18.60	18.60		9.00	17.00	17.00	15.40	9.00	15.40	15.40	13.80	9.00	13.80	13.80	12.20	9.00	12.20	12.20	10.60	9.00	10.60	10.60
6	ND	20.60	11.00	11.00	20.60	19.00		11.00	19.00	17.40	11.00	11.00	17.40	15.80	11.00	11.00	15.80	14.20	11.00	11.00	14.20	12.60	11.00	11.00	12.60
7	SC	18.60	11.00	18.60	18.60	17.00	11.00		17.00	15.40	11.00	15.40	15.40	13.80	11.00	13.80	13.80	12.20	11.00	12.20	12.20	10.60	11.00	10.60	10.60
8	LC-R	18.60	9.00	18.60	18.60	17.00	9.00	17.00		15.40	9.00	15.40	15.40	13.80	9.00	13.80	13.80	12.20	9.00	12.20	12.20	12.60	11.00	12.60	12.60
9	NC	16.80	7.20	16.80	16.80	15.20	7.20	15.20	15.20		7.20	13.60	13.60	12.00	7.20	12.00	12.00	10.40	7.20	10.40	10.40	8.80	7.20	8.80	8.80
10	ND	18.40	8.80	8.80	18.40	16.80	8.80	8.80	16.80	15.20		8.80	15.20	13.60	8.80	8.80	13.60	12.00	8.80	8.80	12.00	10.40	8.80	8.80	10.40
11	SC	16.80	8.80	16.80	16.80	15.20	8.80	15.20	15.20	13.60	8.80		13.60	12.00	8.80	12.00	12.00	10.40	8.80	10.40	10.40	8.80	8.80	8.80	8.80
12	LC-R	16.80	7.20	16.80	16.80	15.20	7.20	15.20	15.20	13.60	7.20	13.60		12.00	7.20	12.00	12.00	10.40	7.20	10.40	10.40	8.80	7.20	8.80	8.80
13	NC	15.00	5.40	15.00	15.00	13.40	5.40	13.40	13.40	11.80	5.40	11.80	11.80		5.40	10.20	10.20	8.60	5.40	8.60	8.60	7.00	5.40	7.00	7.00
14	ND	16.20	6.60	6.60	16.20	14.60	6.60	6.60	14.60	13.00	6.60	6.60	13.00	11.40		6.60	11.40	9.80	6.60	6.60	9.80	8.20	6.60	6.60	8.20
15	SC	15.00	6.60	15.00	15.00	13.40	6.60	13.40	13.40	11.80	6.60	11.80	11.80	10.20	6.60		10.20	8.60	6.60	8.60	8.60	7.00	6.60	7.00	7.00
16	LC-R	15.00	5.40	15.00	15.00	13.40	5.40	13.40	13.40	11.80	5.40	11.80	11.80	10.20	5.40	10.20		8.60	5.40	8.60	8.60	7.00	5.40	7.00	7.00
17	NC	13.20	3.60	13.20	13.20	11.60	3.60	11.60	11.60	10.00	3.60	10.00	10.00	8.40	3.60	8.40	8.40		3.60	6.80	6.80	5.20	3.60	5.20	5.20
18	ND	14.00	4.40	4.40	14.00	12.40	4.40	4.40	12.40	10.80	4.40	4.40	10.80	9.20	4.40	4.40	9.20	7.60		4.40	7.60	6.00	4.40	4.40	6.00
19	SC	13.20	4.40	13.20	13.20	11.60	4.40	11.60	11.60	10.00	4.40	10.00	10.00	8.40	4.40	8.40	8.40	6.80	4.40		6.80	5.20	4.40	5.20	5.20
20	LC-R	13.20	3.60	13.20	13.20	11.60	3.60	11.60	11.60	10.00	3.60	10.00	10.00	8.40	3.60	8.40	8.40	6.80	3.60	6.80		5.20	3.60	5.20	5.20
21	NC	11.40	1.80	11.40	1.80	9.80	1.80	9.80	1.80	8.20	1.80	8.20	8.20	6.60	1.80	6.60	6.60	5.00	1.80	5.00	5.00		1.80	3.40	3.40
22	ND	11.80	2.20	2.20	2.20	10.20	2.20	2.20	2.20	8.60	2.20	2.20	8.60	7.00	2.20	2.20	7.00	5.40	2.20	2.20	5.40	3.80		2.20	3.80
23	SC	11.40	2.20	11.40	1.80	9.80	2.20	9.80	1.80	8.20	2.20	8.20	8.20	6.60	2.20	6.60	6.60	5.00	2.20	5.00	5.00	3.40	2.20		3.40
24	LC-R	11.40	1.80	11.40	1.80	9.80	1.80	9.80	1.80	8.20	1.80	8.20	8.20	6.60	1.80	6.60	6.60	5.00	1.80	5.00	5.00	3.40	1.80	3.40	

Table A.12: Agents' match scores for Group Composition 6