UNIVERSITY OF SOUTHAMPTON

A Voting-Based Agent System to Support Personalised e-Learning in a Course Selection Scenario

By

Ali M. Aseere

A thesis submitted for the degree of Doctor of Philosophy

in the

Faculty of Physical and Applied Science

School of Electronics and Computer Science

July 2012

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL AND APPLIED SCINCES

SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

A Voting-Based Agent System to Support Personalised

e-Learning in a Course Selection Scenario

by Ali M. Aseere

Agent technologies are a promising approach to solving a number of problems concerned with personalised learning due to the inherent autonomy and independence they provide for learners. The objective of this thesis is to find out whether a multiagent system could potentially replace a centralised infrastructure, and to explore the impact of agents taking different strategies. More specifically, our aim is to show how intelligent agent systems can not only form a good framework for distributed e-learning systems, but also how they can be applied in contexts where learners are autonomous and independent. The study also aims to investigate fairness issues and propose a simple framework of fairness definitions derived from the relevant literature.

To this end, a university course selection scenario has been chosen, where the university has many courses available, but has only sufficient resources to run the most preferred ones. Instead of a centralised system, we consider a decentralised approach where individuals can make a collective decision about which courses should run by using a multi-agent system based on voting. This voting process consists of multiple rounds, allowing a student agent to accurately represent the student's preferences, and learn from previous rounds. The effectiveness of this research is demonstrated in three experiments. The first experiment explores whether voting procedures and multiagent technology could potentially replace a centralised infrastructure. It also explores the impact of agents using different strategies on overall student satisfaction. The second experiment demonstrates the potential for using multiagent systems and voting in settings where students have more complex preferences. The last experiment investigates how intelligent agent-based e-learning systems can ensure fairness between individuals using different strategies.

This work shows that agent technology could provide levels of decentralisation and personalisation that could be extended to various types of personal and informal learning. It also highlights the importance of the issue of fairness in intelligent and personalised e-learning systems. In this context, it may be said that there is only one potential view of fairness that is practical for these systems, which is the social welfare view that looks to the overall outcome.

Contents

Chapter 1. Introduction **1**

1.1 Research Hypotheses 3

1.2 Contributions 5

1.3 Document Structure 6

Chapter 2. Agent Technology **9**

2.1 Agent based systems 9

2.2 Why Agent Based Systems? 10

2.3 Agent Frameworks 12

2.4 Social choice theory 17

2.4.1 Voting systems 18

2.4.2 Combinatorial preferences 19

2.5 Fairness 20

2.6 Summary 22

Chapter 3. Technology and Education **25**

3.1 E-learning 26

3.1.1 The Emergence of E-Learning 2.0 28

3.2 Intelligent Systems for E-Learning 28

3.2.1 Intelligent tutoring systems 28

3.2.2 Adaptive hypermedia 30

3.2.3 Existing Adaptive E-Learning Systems 31

3.3 Learning theory 33

3.4 Personalised Learning 34

3.4.1 What is personalised learning? 34

3.4.2 The need for personalised learning 35

3.4.3 Technologies and approaches 36

3.5 Personal Learning Environments 36

3.5.1 From formal to informal learning 38

3.6 Multiagent Systems in E-Learning 39

3.6.1 Examples of E-Learning Multiagent Systems 39

3.7 Motivational Scenarios 43

3.7.1 Scenario One: Course Selection 44

3.7.2 Scenario Two: Group Formation 45

3.7.3 Scenario Three: Personalised Learning 46

3.8 Summary 47

Chapter 4. Experimental System Design 49

4.1 Course selection scenario 50

4.2 Architecture 52

4.3 Voting Protocol 53

4.4 Simple Preferences 55

4.5 Voting Strategies 56

4.6 Summary 59

Chapter 5. Agent Simulation for Decentralised E-learning Scenario 61

5.1 (PART 1) Identical Voting Strategies: 62

5.1.1 Methodology 62

5.1.2 Result and Analysis 64

5.2 (PART 2) Combination of Voting Strategies: 67

5.2.1 Methodology 67

5.2.2 Result and Analysis 69

5.3 Discussion 72

5.4 Summary 73

Chapter 6. Agent Simulation for E-Learning System Involving Complex Preferences 75

6.1 Complex Student Preferences 76

6.2 Methodology 78

6.3 Applying Rules to Voting Strategies 80

6.4 Result and Analysis 81

6.5 Discussion 86

6.6 Summary 86

Chapter 7. Fairness in Intelligent E-Learning Systems Using Agent Simulation 89

7.1 A Framework of Fairness in Intelligent E-Learning Systems 90

7.2 (Part1) Fairness of the protocol 92

7.2.1 Methodology 92

7.2.2 Results and Analysis 97

7.2.2.1 Exponent 97

7.2.2.2 Weight 103

7.3 (Part2) Fairness of Strategy Distribution 108

7.3.1 Results and Analysis 109

7.4 Discussion 112

7.5 Summary 113

Chapter 8. Conclusion and Future Work 115

8.1 Research Summary 115

8.2 Research findings 117

8.3 Limitations 121

8.4 Future Work 121

8.4.1 From simulation to reality 122

8.4.2 Exploring other scenarios 122

8.5 Conclusion 123

References 125

Appendix A 139

A.1 Identical Voting Strategies 139

A.2 Combination of Voting Strategies 140

Appendix B 143

Appendix C 145

List of Figures

Figure 3-1: The components of an intelligent tutoring system(Phobun and Vicheanpanya 2010) 29

Figure 3-2: Taxonomy of Adaptive Hypermedia Technologies (Brusilovsky, 2001) 31

Figure 3-3: The architecture (Sun, Joy et al. 2005) 40

Figure 3-4: The ILMDA system and environment (Soh 2006) 41

Figure 3-5: The IDEAL framework (Shang, Shi et al. 2001) 42

Figure 3-6: PVLE architecture (Xu and Wang 2006) 43

Figure 4-1: System architecture 53

Figure 4-2: the voting protocol 55

Figure 4-3: Calculating student's satisfaction with simple 56

Figure 5-1: Sequence for conducting a case with different settings 63

Figure 5-2: Comparison the three single strategies to the optimal solution for case 1 65

Figure 5-3: Comparison the three single strategies to the optimal solution for case 2 66

Figure 5-4: Comparison the three single strategies to the optimal solution for case 3 66

Figure 5-5: Sequence for conducting a case with different settings 69

Figure 5-6: Case 1: Proportional vs. Equal Share 69

Figure 5-7: Case 1: Intelligent vs. Equal Share 70

Figure 5-8: Case 1: Intelligent vs. Proportional 71

Figure 5-9: Case 2: Intelligent vs. Proportional 71

Figure 5-10: Case 3: Intelligent vs. Proportional 72

Figure 6-1: Calculate student's satisfaction with complex preferences 78

Figure 6-2: Generation of the course model 79

Figure 6-3: Case 2: Proportional vs. Equal Share 82

Figure 6-4: Case 2: Intelligent vs. Equal Share 83

Figure 6-5: Case 1: Proportional vs. Intelligent 84

Figure 6-6: Case 2: Intelligent vs. Proportional 85

Figure 6-7: Case 3: Proportional vs. Intelligent 85

Figure 7-1: Flow chart showing how the exponent mechanism works 94

Figure 7-2: Flow chart showing how the weight mechanism works...................95

Figure 7-3: Comparison between intelligent, proportional and equal-share strategies in means over exponents..98

Figure 7-4: Comparison between intelligent and equal-share strategies for differences between means over exponents..99

Figure 7-5: Comparison between All Intelligent, All Proportional and All equal-share in means over exponents 100

Figure 7-6: Utilitarian measure. The mean of a cohort consisting of groups of 7P, 7E and 7I 101

Figure 7-7: Equalitarian measure. The range of a cohort consisting of groups of 7P, 7E and 7I 102

Figure 7-8: Egalitarian measure. The Median of a cohort consisting of groups of 7P, 7E and 7I 102

Figure 7-9: Comparison between intelligent and equal-share strategies in means over weights...104

Figure 7-10: Comparison between intelligent and equal-share strategies for differences between means over weights 105

Figure 7-11: Comparison between All Intelligent, All Proportional and All equal-share in means over weights...106

Figure 7-12: Utilitarian measure. The mean of a cohort consisting of groups of 7P, 7E and 7I 107

Figure 7-13: Equalitarian measure. The range of a cohort consisting of groups of 7P, 7E and 7I 107

Figure 7-14: Egalitarian measure. The Median of a cohort consisting of groups of 7P, 7E and 7I 108

Figure 7-15: Case 3: mean when Equal-Share and Intelligent 110

Figure 7-16: Case 3: median when Equal-Share and Intelligent 110

Figure 7-17: Case 3: range when Equal-Share and Intelligent 111

Figure 7-18: Case 3: the range for the distribution of intelligent and equal-share strategies 112

List of Tables

Table 3-1: The changing focus of educational technology over the past 30 years (Keengwe and Kidd 2010). 27

Table 4-1: Different settings for the cases 51

Table 5-1: Statistical values of comparison between intelligent strategy and optimal 65

Table 5-2: Statistical values of comparison between proportional strategy and optimal 65

Table 5-3: Statistical values of comparison between equal-share strategy and optimal 65

Table 6-1: Different settings for the cases 80

Table 6-2: Statistical values of comparing intelligent and proportional strategies in case 1 84

Table 6-3: Statistical values of comparing intelligent and proportional strategies in case 2 85

Table 6-4: Statistical values of comparing intelligent and proportional strategies in case 3 86

Table 7-1: Aspects of Fairness and their Metrics 91

Table 7-2: The effect of the exponent 94

Table 7-3: The effect of weight 96

Table 7-4: Case 3 settings 96

Table 8-1: Statistical values of comparison between intelligent strategy and optimal 139

Table 8-2: Statistical values of comparison between proportional strategy and optimal 139

Table 8-3: Statistical values of comparison between equal-share strategy and optimal 139

Table 8-4: Statistical values of comparison between intelligent strategy and optimal 139

Table 8-5: Statistical values of comparison between proportional strategy and optimal 139

Table 8-6: Statistical values of comparison between equal-share strategy and optimal 140

Declaration of Authorship

I, Ali Aseere declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

- Aseere, Ali, Gerding, Enrico and Millard, David (2010) A Voting-Based Agent System for Course Selection in E-Learning. At IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Toronto, Canada, 31 Aug - 03 Sep 2010. , 303-310.
- Aseere, Ali, Millard, David and Gerding, Enrico (2010) Ultra-Personalization and Decentralization: The Potential of Multi-Agent Systems in Personal and Informal Learning. In, 5th European Conference on Technology Enhanced Learning, Barcelona, Spain, Springer Lecture Notes in Computer Science, 30-45.
- Aseere, Ali, Millard, David and Gerding, Enrico (2011) An Agent Based Voting System for E-Learning Course Selection Involving Complex Preferences. In, *the 2011 IEEE/WIC/ACM International Conference on Intelligent Agent Technology,* Lyon, France*,* 22 - 27 Aug 2011. , 386-393.

- Aseere, Ali, Millard, David E. and Gerding, Enrico (2012) Evaluation of Fairness Characteristics in an Intelligent System For Academic Course Selection. In, *IADIS Intelligent Systems and Agents 2012 (ISA 2012) Conference,* Lisbon, Portugal

Acknowledgements

I'd like to thank my supervisors Dr. David Millard and Dr. Enrico Gerding from the School of Electronics and Computer Science at the University of Southampton for their wonderful supervision and for always being helpful while I worked on my research. I have been privileged to work with both of them and am grateful to them for helping me from the first day of my study to the last. I am deeply indebted to them for giving me the time and attention needed to shape my mini and final thesis and the research papers I have produced for various conferences. It would not have been possible to overcome the many problems I encountered during the course of my research and to publicise my work at conferences without their help and guidance.

In particular, I would like to express my sincere gratitude to Dr. David Millard for his encouragement, ideas, valuable direction, guidance and constructive criticisms at different stages of my research. He has helped me in many ways and has developed my problem solving skills and academic ability, which will be very helpful in my future work and research. His extensive knowledge and writing experience helped to make my thesis much better. I also sincerely thank Dr. Enrico Gerding for his guidance and, insightful advice and suggestions that refined my research direction and enhanced my outlook, as well as the ways of perceiving the various aspects involved in any topic or issue.

I would like to thank my parents for supporting and encouraging me during my studies.

I also wish to thank my loving family, with special thanks to my wife Norah for her ongoing love and support through all the stages of my research and also for traveling with me far away from home. My warm wishes go to my son Abdulsalam and my daughter Mayar, who helped me keep in touch with the outside world, with humour, fun, and the comfort they bring to my life.

Last, but most certainly not least, I would like to express my gratefulness to my friends and colleagues who helped me to overcome many troubles and supported me during the research.

Finally, I acknowledge and thank the government of Saudi Arabia, which funded my study and my living expenses. When I return home, I plan to repay my society by employing the knowledge that I gained overseas.

Definitions and Abbreviations Used

MAS	Multiagent System
VLE	Virtual Learning Environment
PLE	Personal Learning Environment
SS	Student Satisfaction
JADE	Java Agent Development Framework
FIPA	Foundation for Intelligent Physical Agents
SoFAR	Southampton Framework for Agent Research
JATLite	Java Agent Template, Lite
JACK	JACK Intelligent Agents
BDI	Belief-Desire-Intention
STV	Single transferable vote
ITS	Intelligent Tutoring Systems
AH	Adaptive Hypermedia
CAL	Computer-assisted learning
AEH	Adaptive Educational Hypermedia
SM	Student Model
TEL	Technology enhanced learning
SA	Student Agent
UA	University Agent
H	Hypothesis
Q	Question

Chapter 1. Introduction

Agents are special software components that work together in an agent framework to achieve some end. Their main features include autonomy, reactiveness, proactiveness and social ability (Sampson, Karagiannidis et al. 2002). Multi-agent systems, where several such agents interact, are being used in a wide variety of applications, ranging from comparatively small systems for personal assistance, to open, complex, systems for industrial applications (Bellifemine, Caire et al. 2007). In e-learning, they can provide new models of learning and applications, such as personal assistants, user guides and alternative help systems, which are useful for both students and teachers (Kommers and Aroyo 1999). It has also been argued that using multi-agent systems to design educational systems leads to more versatile, faster and lower cost systems (Silveira and Vicari 2002).

This thesis argues that the major potential in multi-agent systems has yet to be fully explored, and relates to the ability of agent systems to support personalized and informal learning. In the e-learning domain we are increasingly seeing a move from a world of VLEs (Virtual Learning Environments) into a space where students are taking more control of their learning in the form of PLEs (Personal Learning Environments), either as monolithic applications to help students manage their resources and time, or as a collection of online tools (such as Google calendar to manage time, 43 Things to manage goals, etc). In this personalized learning context, agent technology becomes even more appropriate because agents are good at representing the requirements of users, and negotiating a personalized experience. There is also a lot of potential to support informal learning, because in a decentralized agent system there is no need for a central authority (such as a tutor or academic institution) to orchestrate collaborations and learning activities.

Agent technologies could allow us to take this personalization to new levels. For example, consider an online university that has open enrolment for adult learners to work towards a qualification (or a given set of skills needed for a particular job). Adults seek courses to match their own requirements, but the university can only run courses that attract sufficiently high interest. The situation can be more complicated by existing restrictions on which courses might run, due to the overheads of running too many courses, or adding dependencies between courses. In the context of personalized learning, where students take control of their learning, we would like the decision about which courses are run to be made collectively by the students, while taking into account their individual preferences. A voting procedure combined with an agent framework is a powerful technology for tackling this complexity and producing fexible course selection systems. They enable students and the university to collectively decide which courses will run.

This kind of intelligent and personalized e-learning system means that different students can have different experiences since tasks and resources are intelligently modified according to their preferences (Shute and Paotica 1994). This means intelligent e-learning systems must face the issue of fairness, where fairness is broadly defined as ensuring for any given process participants are treated equally and there is an appropriate balance of satisfaction with the outcome.

In this thesis we are interested in determining whether or not agent technology and voting procedures can be used to solve this e-learning scenario, but also what the consequences are for fairness to the student involved. We present a multi-agent simulation for this e-learning scenario, based on voting theory (where the number of candidates corresponds to the number of courses available that a student can vote for), where an autonomous software agent votes on a student's behalf according to the student's preferences. In this context, the agent votes on a student's behalf to automate the voting process and to assist the student in making voting decisions. In particular, the reasons for using agents are because there will be multiple rounds of voting, and in each round one of the courses is cancelled. In such a setting, it would be very inconvenient for a student to participate by voting every time when course is cancelled. More importantly, through these rounds, an agent can learn from the information provided in previous rounds and predict what could happen to make a decision.

We first explore a simple model of student preferences where the student specifies a rating for each course, and then extend the voting system to deal with more complex preferences. In particular, we discuss the situation that takes into account interdependencies between the available courses. In this context the need for a decentralised, agent-based solution is even more pronounced since it is difficult to collect and process such preferences in a centralised way. This is because, first of all, each agent can have different constraints and interdependencies between courses, which can be difficult to express in a uniform manner. Second, computing an optimal centralised solution when there are such constraints and interdependencies is a computationally difficult task. Both of these problems are addressed by using a voting system, where the task of expressing preferences and casting votes is transferred to the individual agent, thereby enabling a higher degree of flexibility. In particular, each agent can use different preference representations and voting strategies, depending on the needs of the individual user, without requiring the voting protocol to change. Another advantage of this approach compared to a centralised approach, is that the users do not need to reveal their preferences, but these are kept private and are only known by the individual agent representing the user.

Having a decentralised system where each agent can choose its own strategy raises the issue of fairness, since agents with very intelligent strategies could potentially have an advantage over simpler or naïve strategies. As a result, in this thesis we consider various notions of fairness, and see through agent-based simulations whether our system meets these fairness criteria. Furthermore, we consider modifications to the voting protocol and the effect on fairness.

1.1 Research Hypotheses

Given the above scenario, the hypotheses of this thesis are stated as follows:

H1: *In an eLearning scenario such as* ***academic course selection****, a decentralised agent system using a* ***voting protocol*** *can achieve a comparable level level of overall* ***student satisfaction*** *as an optimal centralised approach while maintaining levels of privacy and choice.*

- **Academic course selection.** In many educational organizations there exist restrictions on which courses might run, due to the overheads of running too many courses. They must somehow determine which set of courses to run. This is a common scenario with Higher Education degree courses, where students are often offered a number of courses and for economic reasons only the most popular courses are run. In the context of personalized learning, students take more control of their learning, and would like to collectively make the decision about which courses are run, whilst taking into account their individual preferences.

- **Voting protocol.** A set of rules that governs how votes are cast in an election, how they are aggregated, and how winners are determined.
- **Student satisfaction.** This measures how well the courses that are chosen to run match the preferences of an individual student.

H2: *Within system with complex preferences, an individual student that uses* ***an intelligent predictive strategy*** *will on average achieve a higher satisfaction than those taking* ***a naïve or random strategy****.*

- **Intelligent predictive strategy.** This strategy is included as an example of what can be achieved with a more sophisticated strategy that learns as the voting procedure progresses from one round to the next. Its effectiveness can be gauged by comparing it to others strategies (called Proportional strategy and Equal-Share strategy). The main idea behind this strategy is that, in each round, the agent tries to predict the probability that a course will be cancelled based on the number of points currently awarded to each course from previous rounds. Then, from this probability, it can calculate its *expected satisfaction* for a given allocation of points, and it will allocate the points such that the expected satisfaction is maximized.
- **Naïve strategy.** A strategy that is simple but sensible. It behaves simply when dealing with tasks. Consequently, it provides a good benchmark that can be used to compare the performance of more sophisticated strategies. In this research, the naïve strategy is given name called Proportional strategy. The main idea behind a proportional strategy is that the student agent distributes its points proportionally to the student's preferences for each course.
- **Random strategy.** A very simple and ineffective strategy. This strategy behaves without any particular complicated method when dealing with points. It provides a good lower bound on the performance of system. An Equal-Share strategy is based on the principle that give all courses an equal number of votes, regardless of the student's preference.

H3: *As the proportion of individual agents utilising differently performing strategies in the system increases, the overall* ***fairness of the result*** *(as defined by* ***equity theory****) decreases.*

- **Fairness of the result.** In this context, fairness is broadly defined as ensuring that, for any given process, participants are treated equally and there is an appropriate balance of satisfaction with the outcome.
- **Equity theory.** This is the idea that fairness is based on the allocation of resources (in a society) based on the contribution of individuals; so, for example, in a fair society input efforts and reward outputs are balanced (Adams 1966).

H4: *When having a* ***uniform mixture of different strategies*** *we can* ***adjust the protocol*** *by exponent and weight to make the protocol fairer.*

- **Uniformly mixed strategy cohort.** A population of agents using different voting strategies, and where each strategy is equally represented.
- **Adjusting the protocol.** The system is adjusted in two ways: exponent and weight. **Exponent** is to give an individual student agent that rates courses highly more impact than a student agent that rates the courses lower. In other words, students who have stronger preferences are more important to the system than the others. As a result, the system responds to those with extreme views and takes their preferences into account when dealing with courses. **Weight** is to give a boost to the student who has not achieved their desire, and makes them have a more successful impact on the voting process. In other words, the points of a less happy student will become greater by multiplying his returned points by the weight.

1.2 Contributions

This thesis investigates the use of agent technology as an approach to solving a number of problems concerned with personalized learning. With the methodology and findings of this thesis, the key contributions are as follows:

- We implement an e-learning scenario using multi-agent systems where agents communicate to achieve autonomous goals.
- We introduce an agent voting protocol for individuals to vote for a limited set of resources, consisting of multiple rounds that allow the

student agent to accurately represent the student's preferences and learn from previous rounds.

- We present three new e-learning scenarios that are not possible with today's technology to show how common problem solving techniques in the agent world (voting systems, coalition formation and auction systems) could map to problems in e-learning in terms of decentralization and personalization.
- We investigate whether voting procedures in particular, and multi-agent technology in general, could potentially replace a centralised infrastructure where the selection of courses is determined directly by the central authority, and explore the impact of agents using different strategies.
- We demonstrate the potential of using multi-agent systems and voting in settings where students have complex preferences (substitutable and complementary).
- We introduce a simple framework for measuring fairness by three different notions of fairness (Utilitarianism, Equalitarianism and Egalitarianism).
- We use this framework to explore the issue of fairness in intelligent e-learning systems, where different students can have different experiences as tasks and resources are intelligently modified according to their preferences.

1.3 Document Structure

This thesis is divided into eight chapters. This section provides a summary of the content of each one.

Chapter 2 introduces agent technology, with particular focus on approaches that can be used in the area of multi-agent systems. To this end, agent characteristic and framework are reviewed. This chapter also describes the voting approaches and their relevance to agent technology.

Chapter 3 gives an overview of the literature relating to this work in the area of technology and education. Specifically, it introduces intelligent approaches for e-learning and gives details about their definitions and architecture. It then discusses personalised learning, focusing on its need and technology that can support it. It then goes on to show different systems which are supported by agent technology. . The chapter ends by presenting three scenarios that show how solutions from the agent domain might be used to solve problems from the e-learning domain in a novel way.

Chapter 4 describes the architecture of the multi-agent system simulation that is used to conduct this research. It also describes the voting protocol that is being used by the system, and how the preferences of students are modelled. Finally, it presents three different voting strategies.

Chapter 5 describes the first set of experiments, which investigates whether voting procedures used with multi-agent technology could potentially replace a centralized infrastructure, where the decision is made directly by the central authority. To this end, it compares the output obtained by the different strategies with the optimal solution. It then shows the impact of the different agent strategies on the overall student satisfaction.

Chapter 6 describes the second set of experiments, which demonstrates the potential of using multi-agent systems and voting in settings where students have complex preferences, such as complementary and substitutable preferences between courses. As in Chapter 5, it shows the impact of using a range of voting strategies on overall student satisfaction.

Chapter 7 describes the third set of experiments, which investigates the issue of fairness, where individuals use different strategies in intelligent agent-based e-learning systems that provide personalisation. It first introduces a simple framework for measuring the fairness of a result derived from the literature, and sets fairness definitions with formal metrics. It then shows how to use this framework to examine the results of the agent simulation and look at the fairness of different student strategies as measured by the metrics. It also discusses the fairness of the protocol by including factors that make the protocol fairer by rewarding people very highly or rewarding people who do not succeed at first

Chapter 8 concludes and summarises the work contained within the thesis and links them to the findings achieved. It ends with interesting directions for future work.

Chapter 2. Agent Technology

Agent technology is considered an important and promising approach to developing industrial distributed systems and enterprise collaboration (Jennings, Corera et al. 1995). This chapter explains agent technology. It starts by providing background, with a particular focus on the definitions of agent and multi-agent systems, discussing some of their characteristics and describing a number of key agent frameworks. It then goes on to explain how multiagent systems can make a collective decision using social choice theory. It also presents a related area, voting systems, which are sets of rules that determine how decisions are reached, summarising some common voting systems. The chapter then discusses the notion of fairness, with a focus on different type of fairness, also discussing the importance of fairness when constructing a decentralised system. Finally, it concludes by summarising the information presented.

2.1 Agent based systems

The term agent has been widely used in a number of technologies, for example, in artificial intelligence, databases, operating systems and the marketplace. Although there is no unique definition of agent technology, researchers in the field agree that an agent has autonomy (Bellifemine, Caire et al. 2007). Wooldridge and Jennings (1995) provide one of the most common definitions. They define an agent as "a computer system situated in some environment, that is capable of autonomous action in this environment in order to meet its design objectives" (Wooldridge and Jennings 1995). In this context, autonomy means the agent should be able to "act without the intervention of humans or other agents, and should have control over its own actions and internal state".

An agent is considered to be intelligent if an agent is capable of flexible autonomous action in order to meet its design objectives. Flexibility means the agent is expected to have the following capabilities (Wooldridge 2009):

Reactivity: Intelligent agents should perceive their environment and respond in a timely fashion to changes that occur in it in order to satisfy their design objectives.

Proactivity: Intelligent agents should be able to take appropriate initiatives rather than simply taking action in response to the environment in order to satisfy their design objectives.

Sociability: Intelligent agents should be able to interact, when they deem appropriate, with other agents and humans in order to satisfy their design objectives.

A multiagent system (MAS) is a collection of autonomous agents situated within the same environment that carry out their activities in order to solve given problems beyond the capabilities of individual agents (Jennings, Sycara et al. 1998). Agents have their own goals. Therefore, agents interact with each other within the multi-agent system to achieve an end objective or to improve their performance. A key pattern of interaction in multi-agent systems involves a form of coordination, both in cooperative and competitive agents (Weiss 1999). In the case of cooperative multi-agent systems, several decentralised agents try to combine their efforts to collectively accomplish the goals that the individual agents cannot (that is, each agent tries to maximise the utility of the whole system). In the case of competitive multi-agent systems, agents have their own goals and preferences (that is, agents try to maximise their own utility, regardless of the effects on the other agents' utilities). Agents in this design are also called "self-interested". In some cases, a self-interested agent may effectively act in a coordinated manner by applying rules, such as voting rules, to reach collective goals (see section 2.4.1).

2.2 Why Agent Based Systems?

Agent technology is considered an important and promising approach to developing industrial distributed systems and enterprise collaboration (Jennings, Corera et al. 1995). Agent technologies can reduce the complexity of systems that handle complex problems (Shen, Hao et al. 2006). Milgrom et al. (2001)

have illustrated instances where the usage of software agent technology can be useful (Milgrom, Chainho et al. 2001):

Need for Distributed Control

Some problems are too complex for a single monolithic solution, but most designers and implementers find it difficult to design and build distributed solutions. Decentralising control in a distributed system can solve responsibility, privacy and physical constraints. Here, a multiagent system acting independently can be used to control each part of the distributed system. Brennan et al. (2002) developed their distributed intelligent control approach that enables control over widely distributed devices in an environment that is prone to disruptions. With this model, control is achieved by the many simple, autonomous, and cooperative entities that are based on the principles of agent-based systems (Brennan, Fletcher et al. 2002).

Need to concurrently achieve multiple, possibly conflicting, goals

A number of systems face situations where it is not possible to specify their behaviour on a case-by-case basis because it is not possible to foresee at the design stage all of the situations that are going to happen during the planned system stage. In these cases, it is much easier to specify a series of goals to be achieved (Riemsdijk, Dastani et al. 2008). Agent systems are good approaches for this type of system, since agent technology solves this problem by defining how to decide what to do instead of mapping inputs to outputs by defining what to do. This is useful when there are many alternative ways to achieve a goal. The approach leads to more flexible systems that can adapt their behaviour to changing circumstances, so as to satisfy the goals for which they are responsible (Milgrom, Chainho et al. 2001). For example, Adams et al. (2008) developed a system for disaster management that is comprised of autonomous agents that can sense and act in order to achieve collective goals. It is a decentralised control system and as such is more flexible and can respond more quickly to new information related to disasters (Adams, Field et al. 2008).

Need for Autonomous behaviour

Systems acting on behalf of a user may be designed to take action only when prompted by an explicit request. However, it is more efficient if they attempt to satisfy their assigned goals without the need for explicit requests. Autonomous behaviour can help manage a substantial workload, as well as having the ability to take on tasks that are too complex for a human (Gillies and Ballin 2004).

Agent-based systems can behave autonomously more easily than other systems built using traditional techniques. Agent-based systems implement applications as networks of autonomous agents.

Need for Flexibility and Adaptability

In systems that have a purpose that is expected to evolve, the system may need to be significantly expanded or modified during operation. They are more dealing with environments that are unpredictable and changeable like robots (Bernon, Gleizes et al. 2003). In other cases, the system's knowledge of its environment is expected to increase, and subsequently the system is expected to adapt its behaviour. Software agents are considered an efficient approach due to their intrinsic modularity. Therefore, agents can be easily added or replaced, reducing maintenance costs in relation to the overall system.

Need for Interoperability

A software agent-oriented approach can be considered if the system needs to interact with software that is unknown during the design process. Multiagent systems can provide services beyond their individual capabilities and have the capability to exchange and interact with information, as well as provide a flexible and dynamically cooperative model in a cooperative intelligent design environment (Zhao, Deng et al. 2001). Moreover, they can improve the interoperability among applications and build up high-level autonomous cooperation (Suguri, Kodama et al. 2003).

2.3 Agent Frameworks

An Agent framework is an architecture providing the environment in which agents can actively exist, operate and communicate with each other using appropriate protocols to achieve their goals (Leszczyna 2004). The agent framework supports three features for agent developers creation: the ability to create and run agents within particular environment. Communication: supports agent-to-agent communication using speech acts. Discovery: allows agents to find new agents using a service based discovery mechanism (Bailey 2002).

There are a number of different multiagent frameworks available that have been used by researchers. Some frameworks will be explained in this section.

JADE

Java Agent Development Framework (JADE) is Java-based agent framework. It is open source and provides middleware-layer functionalities. It was one of the early systems to support the Foundation for Intelligent Physical Agents (FIPA) specifications (Bailey 2002; Bellifemine, Caire et al. 2007).

JADE was implemented to simplify the development of agent applications. It provides programmers with the following functionalities (Bellifemine, Caire et al. 2007):

- A fully distributed system: each agent runs as a separate thread on different remote mechanisms, i.e. the underlying communication infrastructure is abstracted by a unique location-independent API.
- Full compliance with FIPA: the framework efficiently participates in all FIPA interoperability.
- Efficient transport of asynchronous messages via a location-transparent API: the platform can select the best available way to communicate.
- Support for agent mobility: the agent can migrate between processes and machines. That migration allows the communicating agent to continue to interact.
- Support for ontologies and content languages: Ontology is preformed automatically by the platform to be used by the programmers directly. Also programmers can implement a new ontology to satisfy application requirements.

Within the JADE framework, there is a platform consisting of agent containers that can be distributed over the network. Containers run within a single JVM platform, providing the JADE run-time as well as all services needed for executing each JADE agent. There is a special container called the main container, which is the first container to be launched. All other containers must join the main container by registering with it (Bellifemine, Caire et al. 2007).

SoFAR

The Southampton Framework for Agent Research (SoFAR) is an early agent framework developed at the University of Southampton. It was designed to provide an environment for a distributed information management agent communicating over communications infrastructure. In SoFAR, agents run within a platform, which is a single JVM environment. Each platform contains registry agents that maintain a list of services that each agent provides within the platform. Agents connect with the registry when they want to advertise a set of services they offer. Consequently, if an agent requires any services from another within the platform, it must first query the registry to acquire the services.

The communication model in SoFAR is based on an existing agent language, such as FIPA. With these services, SoFAR provides the necessary infrastructure needed to rapidly develop autonomous, proactive and socially aware agents (Moreau, Gibbins et al. 2000) (Bailey 2002).

JATLite

JATLite (Java Agent Template, Lite) was created by the Computer Science Department at Stanford University. It is a collection of Java packages designed to allow the user to create software agents that communicate across the internet. JATLite is based on the Typed Messages Agents mechanism, where an agent is defined as a part of communities that perform a distributed computation using typed messages. JATLite provides basic communication tools that exchange KQML messages through TCP/IP (Jeon, Petrie et al. 2000).

JATLite provides a basic infrastructure in which agents register with an Agent Message Router. An Agent Message Router is a specialised application which receives a message from the registered agents and routes it to the correct receiver. It enables an agent to connect or disconnect to the Internet, transfer files with the FTP protocol, and exchange information with other agents running on different computers (Ugurlu and Erdogan 2000). In particular, all agents send and receive messages via the Router, which acts like an email server and forwards messages onto the receiver. If an agent disconnects, or accidentally crashes, the message router will buffer the message until the receiving agent resumes its contact with the router. In this, agents are allowed to disconnect, migrate to a new location and then reconnect to the Router to receive their messages (Bailey 2002).

JACK

JACK Intelligent Agents (JACK) were developed by Agent Oriented Software Pty. Ltd. (AOS), a commercial Java-based environment for developing and running multi-agent applications (AOS 2011). The JACK framework can provide a high level of performance and incorporates the Belief-Desire-Intention (BDI) reasoning model. JACK can be easily extended to support different agent models or specific application requirements. It consists of architecture-independent facilities, plus a set of plug-in components that address the requirements of specific agent architectures (Busetta, Rönnquist et al. 1999). Each agent is defined in terms of its goals, knowledge and social capability, and is then left to perform its function autonomously within the environment it was designed to function in (AOS 2011).

Jason

Jason is the interpreter for our extended version of AgentSpeak, which allows the running of a multi-agent system distributed over a net. Jason was very much directed towards using AgentSpeak as the basis and providing various extensions that are required for the practical development of multi-agent systems. It is implemented in Java and is open source (Bordini and Hübner 2006). AgentSpeak is logic programming for the BDI agent architecture and provides an elegant abstract framework for programming BDI agents (Alechina, Bordini et al. 2006).

Bordini et al. identfied some of the features available in Jason (Bordini, Hübner et al. 2007):

- Speech-act based inter-agent communication and annotation of beliefs with information sources;
- Annotations on plan labels, which can be used by elaborate selection functions;
- The possibility to run a multi-agent system distributed over a network using SACI;
- Fully customisable (in Java) selection functions, trust functions, and overall agent architecture (perception, belief-revision, inter-agent communication, and acting);
- Straightforward extensibility (and use of) by means of user-defined; and
- Clear notion of multi-agent environments, which can be implemented in Java (this can be a simulation of a real environment).

dMARS

The distributed Multi-Agent Reasoning System (dMARS) is the best-known implementation of the PRS architecture using the theoretical foundations of the BDI. The dMARS agent consists of a set of beliefs, desires, plan and intentions (d'Inverno, Kinny et al. 1998). It stores its beliefs in a relational database, along with all its information about the world (Jain, Ichalkaranje et al. 2002). In dMARS agents, the BDI model is operationalised by plans and each agent has a plan library.

D'Inverno et al. summarised the operation of the dMARS agent, which executes the following cycle (D'Inverno, Luck et al. 2004):

- Observe the agent's internal state and the world and, based on that, update the event queue to reflect the events that have been observed;
- Find plans whose condition matches an event; this generates possible desires;
- From this set of matching plans, select one for execution (an intended means);
- Push the intended means onto an existing or new intention stack, according to the event whether or not it is a sub-goal;
- Take the topmost plan (intended means) from the intention stack, and execute the next step of this current plan;
- If the step is an action, perform it;
- Otherwise, if it is a sub-goal, post this sub-goal to the event queue.

Repast

Repast is a software framework for agent simulation. It is a free open-source toolkit developed at the University of Chicago. It is an integrated library of classes for running and displaying date the created based on agent simulation (Collier 2003). It works as a collection of agents that controls the agent's behaviours according to the schedule. This schedule controls the actions within the agent model, such as updating and recording data (Tatara, Çınar et al. 2007).

North and Macal reviewed the main features of Repast. It is fully object-oriented and can be developed in many languages, including Lava, C# and Visual Basic.Net. It also supports sequential and parallel event operation, whereby a concurrent discrete event scheduler is used. In addition, Repast offers built-in simulation result logging and graphing tools that allow users to access and modify agent properties such as agent behaviour at run time. It includes libraries for genetic algorithms, neural networks and specialised mathematics (North and Macal 2005).

2.4 Social choice theory

Social choice theory is a way of analysing collective decision making. It considers the opinions or values of the members of a given community or society as a preference (given that the preferences of people may conflict), and attempts to make a collective choice or a judgement based on those preferences (Craven 1992; Gaertner 2006). It is available in pure Java or Microsoft.Net forms.

In technologies requiring a mechanism for collective decision making, such as artificial intelligence and computer science more broadly, there is a need to study computational social choice mechanisms such as fair division algorithms or voting procedures (Chevaleyre, Endriss et al. 2007). Computational social choice theory is concerned with importing concepts from social choice theory and applying them to artificial intelligence and computer science. For example, social choice theory was originally developed to study preference aggregation mechanisms and collective decisions made by humans. Similarly is the case for multi-agent systems in order to manage societies of autonomous software agents and make collective decisions (Chevaleyre, Endriss et al. 2008).

In multiagent systems, social choice theory is an active area of research that enables decentralised decisions. This field has increasingly been an area of investigation for researchers of multiagent systems over the last two decades (Lang 2004; Procaccia 2008). Researchers in the field have considered social choice theory as highlighting fresh issues, given that the computational features of some problems need collective decisions (Chevaleyre, Endriss et al. 2008).

When different agents have different preferences within a multi-agent system, it is important to find a way to aggregate these preferences. Although agents can be competitive, having independent motivations, goals or perspectives, they still need to be reconciled and to come to a consensus (Rosenschein and Procaccia 2006).

Designers of multi-agent systems are concerned with analysing and designing the mechanisms needed for collective decision making, as agents are inherently autonomous and may have conflicting goals. Meanwhile, each agent would like to maximise its utility. Voting systems are one way for multiagent systems to choose one out of a set of possible decisions. Each agent expresses its preferences of the possible decisions, and a voting system aggregates these preferences to determine the collective decision (Rossi, Venable et al. 2011). Therefore, a voting system provides an efficient way to make socially collective decision while taking individual preferences into account (Wooldridge 2009).

2.4.1 Voting systems

Voting systems are appropriate for reaching socially desirable decisions that take individual preferences into account (Wooldridge 2009). A voting system applies a set of rules that govern how votes are cast in an election, how they are aggregated, and how winners are determined (Laruelle and Valenciano 2008). There are many voting procedures:

Plurality voting: Everyone casts a single vote. The candidate with the most votes wins. In this system, known as *first past the post*, each voter has one vote and the single candidate who receives the most votes, irrespective of the percentage of these votes among the total number of votes cast, is declared the winner.

Cumulative voting: Each voter is given k votes, which can be cast in a distributed way. Several votes can be cast to one candidate, and the remaining of the votes can be distributed across other candidates. The candidate with the most votes wins (Shoham and Leyton-Brown 2008). Here, each voter receives a number of points (sometime the number of points is equal to the number of candidates and sometime not), and they are free to choose how many points to allocate to each candidate. The candidates with the highest cumulative points are selected as winners. This allows voters to express their intensities of preference rather than simply to rank candidates (Brams 1991)

Single transferable vote (STV): This rule proceeds through a series of m-1 rounds. In each round, the candidate with the lowest plurality score is eliminated and each of the votes for that candidate transfers to the next remaining candidate in the order given in that vote. The candidates are ranked in reverse order of elimination (Conitzer and Sandholm 2005). In (STV) system, each voter provides a ranked list of candidates according to its preferences. The winner determination process proceeds in several rounds. In each round, all votes for the most preferred candidate are counted, and the candidate who has received the least number of votes is eliminated. Then, anyone who has voted for the eliminated candidate as their first preference now has their second preference allocated as their first preference. The process then repeats until the required number of winners remains. The aim of this procedure is to minimise wasted votes and to promote proportional representation.

Plurality with run-off: This voting procedure is a variation of STV. In the first round all candidates are eliminated except the two with the highest plurality vote. Then votes are transferred to these as in the STV rule. In second round the winner is determined from these two. Candidates are ranked according to Plurality scores with the exception of the top two candidates; the winner is determined according to the runoff (Zuckerman, Faliszewski et al. 2011).

Borda Count: this voting method allows each voter to rank candidates in order on preferences. This ordering contributes points to each candidate based on the position that he ranked by the voter. If there are n candidates, it contributes $\mathrm{n}-1$ points to first place ranked candidate, $n-2$ points to the second, and so on. It contributes no points to the lowest ranked candidate. The winner is the one with the highest points (Shoham and Leyton-Brown 2008).

This work introduces a novel voting procedure that combines features from both STV and cumulative voting. Specifically, we take advantage of the features of cumulative voting to express the preferences by using points while allowing for multiple rounds in order to avoid wastage due to transferring points akin to the STV method. Having multiple rounds also allows a student to learn from previous rounds and to adjust voting behaviour accordingly. The details of our voting procedure are presented in Section 4.3.

2.4.2 Combinatorial preferences

Combinatorial preferences mostly have been studied within the context of combinatorial auctions. However, there are a couple of papers that have considered such preferences in a voting setting. Specifically, combinatorial voting has been studied by Lang (Lang 2004; Lang 2008). In Lang's work, agents are assumed to vote for given *bundles* of candidates. Since enumerating all possible combinations of candidates is typically infeasible, the main problem becomes which of these bundles should be selected by the system and voted on by the agents.

In terms of representing preferences, a number of languages have been introduced in both combinatorial voting and auctions. Some of these languages are tailored to represent cardinal preferences while others represent ordinal preferences. With regard to the nature of the language itself, some of the languages are graphical (such as CP-net or GAI-net) and some others are based on propositional logic (such as bidding languages for combinatorial auctions) (Boutilier and Hoos 2001). Expressing preferences that allows agents to specify preferences concisely and clearly include the OR-language and XOR-language from the combinatorial auctions field (Uckelman and Endriss 2008). In the OR-language, the valuation of a bundle is taken to be the maximal value that can be obtained when computing the sum over disjoint bids for subsets of the bundle, while in the XOR-language, at most one bundle can be counted and the valuation of a bundle is simply the highest value offered for any of its subsets (Chevaleyre, Endriss et al. 2008). Conitzer (Conitzer 2010) shows an example to illustrate the mechanism behind these languages. For OR-language, assume the bid as following ({ a } , 3) OR ({ b,c } , 4) OR ({ c,d } , 5). This bid indicates that if the bidder receives the bundle { a,b,c } , his value is 7 (because he computes the sum over his bids); if he receives { b, c, d } , his utility is 5 (each of the last two bundles in his bid are contained in the bundle he receives, but c can only be counted towards one of them, and the last bundle has the greater valuation). For example, the bid ({ a } , 3) XOR ({ b,c } ,4) XOR ({ c, d } ,5) indicates that the bidder's value for { a,b,c } is 4 (only one of the first two bundles can be counted). In this language, any valuation function can be expressed (if necessary, by XOR-ing all possible bundles together).

This thesis work uses the same principle as the languages discussed above, but limits the number of combinations to only consider interdependencies between two courses. In particular, we model two types of relationships: complementary (AND) and substitute (OR). We describe that in more details in Section 6.1.

2.5 Fairness

It is important to consider fairness concepts when constructing decentralised systems (Wierzbicki 2010). These systems usually consist of independent agents interacting with one another. Subsequently, agents make decisions under a degree of uncertainty and the decision of others' influences the outcomes of individual agents. Furthermore, the lack of central authority makes it difficult to prevent conflict.

Typically, multi-agent systems are designed on the assumption that they consist of rational, self-interested agents. However, some agents need to work collaboratively to reach their common goals. This is similar to social welfare, where all agents are trying to maximise the outcome for the whole society.

In these circumstances, agents working together as a whole system need to take fairness into account, otherwise they may not reach their intended goals. This is because the problems lies in controlling the behaviour of individual agents working in a way that enables the whole system to reach a certain goal (Jong, Tuyls et al. 2008).

In politics, fairness is often applied to social welfare or the political process. *Equity theory* is the idea that fairness is based on the allocation of resources (in a society) based on the contribution of individuals. For example, in a fair society input efforts and reward outputs are balanced (Adams 1966). Equity theory, therefore, concentrates on the outcomes of a system and judges whether they are fair based on the actions of the individuals involved. Nevertheless, equity theory has been criticised for not taking into account the (potentially unbalanced) differing needs of individuals and for not considering the process itself (Leventhal 1976). In mathematics, fairness is also used to describe outcomes. For example, the theory of fair division, states that any allocation of resources is fair if any other division that makes one of the participants better off makes at least one of them worse off, and also if every participant subjectively feels himself to be better off than anyone else (Weller 1985). Rabin (1993) extended this idea to one of fairness equilibria, a balanced state of fairness within a game situation, in which players are helping those that help them and hurting those that hurt them (Rabin 1993).

Fairness has different general types. The two most common types identified by social psychology are distributive fairness and procedural fairness (Folger and Konovsky 1989; Tyler 2000). Distributive fairness focuses on giving all member of the society a fair share from available benefits and resources. However, people have different views on what constitutes a fair distribution. While some believe in equity principles, others believe in equality principles and others on need principles (Wagstaff 1994). Under the principle of equity, people should receive a share proportional to their contributions and efforts. Under the principle of equality, people should receive equal shares regardless of their inputs and actions. Under the principle of need, distribution is based on need, with those needing more receiving more (Deutsch 1975; Maiese 2003).

Procedural fairness is concerned with making and implementing decisions according to fair processes, thereby ensuring fair treatment. Procedural justice focuses on the means used to determine and shape the outcome (Folger and Konovsky 1989). Thus, applying a fair procedure will often generate a fair outcome. Tyler (2000) says people are more willing to accept decisions when they feel this decision was made using procedures they view to be fair. Therefore, procedural fairness is a proper mechanism for resolving social conflicts as it focuses on the rules that produce fair decisions (Tyler 2000).

Fairness is based upon notions of justice that can help codify the main principles in terms of satisfaction values, where an agent's satisfaction value is a numerical representation of how satisfied that agent is with a given outcome.

Utilitarian: is the belief that the best outcome is the one that gives the greatest whole welfare to society, where whole welfare is measured by summing the welfare for all individuals (Myerson 1981). In terms of agent satisfaction values this would mean maximising the mean satisfaction of all participants.

Equalitarian: is the belief that the best outcome is the one that gives the least difference between the individual members of society (Dworkin 1981). In terms of agent satisfaction values this would mean minimising the standard deviation of satisfaction values, and minimising the range of satisfaction values. The problem with equality is that you can achieve perfect equality regardless of overall satisfaction, drawing no difference between a system where no one is satisfied and another where everyone is satisfied.

Egalitarian: is the belief that the best outcome is one that gives the greatest overall welfare subject to the restraint that all individual members should have equal benefits, rights and opportunities from society (Myerson 1981). An Egalitarian principle would therefore be to maximise the welfare of a society's weakest member (Endriss, Maudet et al. 2003). In terms of agent satisfaction values, this would mean maximising the minimum satisfaction. Maximising the median satisfaction is another approach (to ensure that the lowest half of the satisfaction values is as high as possible).

2.6 Summary

This chapter has presented a background to agent technology showing definitions, features and usage of multi agent systems, and discussing some of their characteristics and showing some agent frameworks. It then has showed how social choice theory can use some voting rules to manage numbers of autonomous software agents having different preferences and make collective decisions.

This chapter also has included an overview of voting systems, and summarising some common voting systems. The chapter then has discussed the fairness notion, with a focus on different type of fairness.

Having covered the relevant background and issues as entrance to this this thesis, the next chapter will describe related work of technology and education focusing on personalized and informal learning and the existing use of agent technology in e-learning.

Chapter 3. Technology and Education

Using technology to support learning is increasingly popular. Nowadays, technology is integrated into the learning environment and used as one of important options to support learning (NAEYC 1996). The design of technologies tools such as Web 2.0 have changed learning environments in ways that increase the possibility for more interactions between learners and facilitate access to a variety of learning material. Moreover, the creation and extension of various intelligent technologies, such as agent technology and web services, can support flexibility in e-learning.

In order to investigate the feasibility and impact of a multi-agent approach in e-learning, it is necessary to have a clear understanding of several research areas. This chapter starts by giving an overview of e-learning in general, with a focus on its definition and different stages through its history and a comparison of the characteristics of each period. Then, it introduces intelligent approaches for e-learning, Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia (AH), giving details about their definitions, architecture and role in e-learning. This is followed by a detailed survey and discussion of personalised learning, with attention paid to its definitions, needs, and some technologies that can support it. This chapter shows how agent technology can be applied to tackle challenges in educational environments by presenting different systems which are supported by agent technology. Finally, this chapter goes on to illustrate three different e-learning scenarios showing how agent technologies could be used. Each scenario is composed of a *description* of the scenario and an analysis of the *agent solutions* that make the scenarios possible.

3.1 E-learning

E-learning is nowadays one of the most interesting of the e-domains available through the Internet (Anghel and Salomie 2003) and has the potential to overcome some traditional learning limitations such as fixed times and locations for learning. E-learning offers new ways for both learners and educators to enrich their learning and teaching experiences through their learning environments. It enables learning to take place at any time at any location, and also supports the exploration and application of information, as well as the promotion of new knowledge (Holmes and Gardner 2006).

There may be many definitions within the context of e-learning but they all focus on the same functions and features. The European Commission and the E-learning Action Plan defines e-learning as "the use of new multimedia technologies and the Internet to improve the quality of learning by facilitating access to resources and services as well as remote exchanges and collaboration" (CEC 2001). Another definition by the American Society of Training and eDucations (ASTD) states: "E-learning is the application of digital media by the learner in the learning process, where digital media includes the Internet, corporate networks, computers, satellite broadcasts, audio tapes, videos, interactive television, and CD-ROMs, etc. The scope of e-learning applications includes online learning, computerized learning, virtual classrooms, and digital cooperation" (Chen and Chiu 2005).

In general, educators and trainers use computers at all levels of education, business and training in different ways to enhance and support learning and teaching (Molnar 1997). Consequently, the term e-learning has different meanings in different contexts. For example, in the school sector, e-learning refers to a combination of both online and software-based learning, whereas in the higher education and training sectors it refers to a variety of online practices. Also, it relates to Internet-based flexible delivery of content and programs that support particular communities of practice (Nicholson 2007).

Over the past 30 years, technology-based learning systems have changed dramatically. In Table 3-1 below, Keengwe and Kidd (2010) give a summary of e-learning practice and change in education technology over the past 30 years and compare the characteristics of each period. From the beginning, e-learning has focused on computer-assisted learning, providing tools to solve problems for local users. Then it moved and began to concentrate on computer-based training using older CAL models with interactive multimedia courseware. After the emergence of the Internet, the focus moved to web-based training delivering content through the internet. After that, the concept of e-learning becomes more focused and the term even more popular. We see the delivery of internet-based flexible courseware, increased interactivity, online multimedia courseware and remote user-user interactions. Finally, and most recently, e-learning has moved to the stage of mobile learning and social networking. This period has more practical features, such as interactive distance courseware via a portable device such as a laptop. Moreover, when it comes to learning with portable technologies the focus is on the mobility of the learner (Keengwe and Kidd 2010).

Table 3-1: The changing focus of educational technology over the past 30 years (Keengwe and Kidd 2010).

Ear	***Focus***	***Educational characteristics***
1975-1985	Programming; Drill and practice; Computer-assisted learning – CAL	Behaviourist approaches to learning and instruction Programming to build tools and solve problems Local user-computer interaction.
1983-1990	Computer-Based Training; Multimedia;	Use of older CAL models with interactive multimedia courseware Passive learner models dominant Constructivist influences begin to appear in educational software design and use.
1990-1995	Web-based Training	Internet-based content delivery Active learner models developed Constructivist perspectives common Limited end-user interactions.
1995-2005	E-Learning	Internet-based flexible courseware deliver; increased interactivity Online multimedia courseware Distributed constructivist and cognitivist models common Remote user-user interactions
2005-present	Mobile learning and social networking	Interactive distance courseware Distributed online through learning management systems with social networking components Learning that is facilitated via a wireless device such as a PDA, a smart phone or a laptop Learning with portable technologies where the focus is on the mobility of the learner

3.1.1 The Emergence of E-Learning 2.0

The e-learning domain has been influenced by many factors. The nature of the web, new technologies and internet users have been considered to be the most important factors that captured the attention of numerous e-learning researchers. Internet users prefer concurrent information from video, images and text (multiple sources). Moreover, they demand access to these resources and create their community through continuous communication with friends. On the web and the Internet, the Read Web was transformed into the Read-Write Web (Web 2.0). The web therefore became a platform in which content is created, shared, repurposed, and passed over the Internet, allowing communication through images, video, and multimedia (Downes 2005).

These factors led to the emergence of e-learning 2.0, which sees the benefits of traditional e-learning integrated with Web 2.0 services. Although Web 2.0 services, such as blogs, wikis, and other social software, were not designed specifically for use in education, they help make e-learning far more personal, social, and flexible (MacManus 2007). Teachers are starting to investigate the potential of wikis, blogs, sharing services and other social software to create new learning opportunities (O'Hear 2006).

3.2 Intelligent Systems for E-Learning

When considering intelligent approaches for e-learning, there are two types: Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia (AH). These systems approaches behave intelligently when interacting with students and resources. This section will explain them in more detail.

3.2.1 Intelligent tutoring systems

Intelligent tutoring systems (ITSs) are an outgrowth of traditional systems known as computer-aided instruction. The goal of ITS is to engage students in sustained reasoning activity and to interact with the student based on a deep understanding of the student's behaviour (Corbett, Koedinger et al. 1997). ITSs are computer-based learning systems providing methods of teaching and learning based on one-to-one interaction that enables learners to practice their skills by carrying out tasks within highly interactive learning environments. They assess each learner's actions within these interactive environments and develop a model of their knowledge, skills, and expertise (Phobun and Vicheanpanya 2010). They are classified as intelligent, meaning they must present tutoring capabilities analogous to a human. Consequently, they are expected to be able not only to adjust the content but also to deliver it to students' needs by analysing and anticipating their responses and behaviours (Nkambou 2006).

Syed and McRoy have proposed that ITS system is composed of four components (Figure 3-1): the interface module, the expert module, the student module, and the tutor module (Syed and McRoy 2000). The interface module provides the method for the student to interact with the ITS through its graphical user interface (GUI) and sometimes through a rich simulation of the task domain. The expert module references a domain model. This domain contains a description of the knowledge or behaviours that represent expertise in the subject-matter domain, i.e. the expert system or cognitive model that the ITS is teaching. The student module contains descriptions of the student's knowledge or behaviours, including their misconceptions and knowledge gaps. The tutor module takes corrective action, such as providing feedback or remedial instruction. It needs information about what a human tutor in such situations would do (Urban-Lurain 1996).

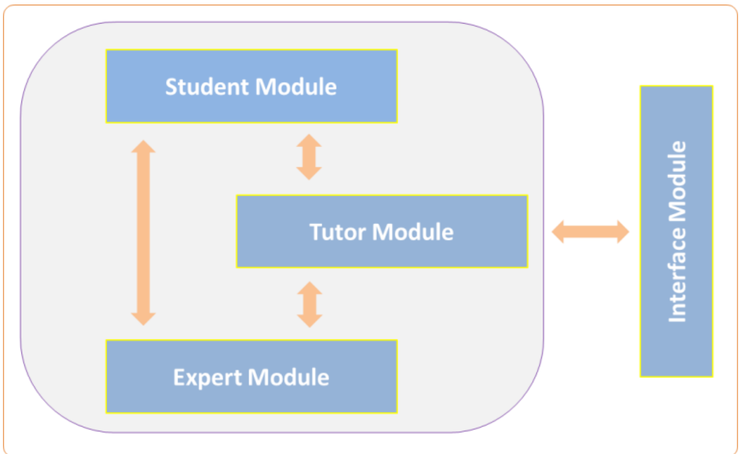


Figure 3-1: The components of an intelligent tutoring system(Phobun and Vicheanpanya 2010)

ITSs provide practice-based instruction to support corporate training and higher education and enable participants to practise their skills by accomplishing tasks in interactive environments. ITS systems assess each learner's actions in these environments and develop a model according to his or her knowledge, skills and expertise (Corbett, Koedinger et al. 1997) (Ong and Ramachandran 2003). Indeed, Ong and Ramachandrann (2003) said that students using ITS-based applications learn faster than in classroom-trained practice. At Carnegie Mellon University, for example, researchers developed an ITS called the LISP Tutor. This system taught computer programming skills to students. In that experiment, students who used the ITS scored 43% higher in the final exam than a control group that received traditional instruction (Ong and Ramachandran 2003).

Both ITS and AH are intrinsically related to education. A number of technologies for AH were developed for adaptive educational hypermedia (AEH). In turn, ITS provided the origins of research on AEH. In early ITS research, the systems provided little or no learning materials. It was thought that the only important job of ITS was to support students in solving the problem they faced. However, after the growth in computer capabilities, it was found that the combination of ITS and learning materials organised as hypermedia was a starting point for the research on AEH (Brusilovsky 2000).

3.2.2 Adaptive hypermedia

AH systems have become common in the last few years as tools for user-driven access to information. AH tries to tackle the fact that users are individuals and it provides them with appropriate solutions for their needs. According to a definition provided by Brusilovsky (2001), adaptive hypermedia systems consider the goals, preferences and knowledge to build a model of the individual user and use this throughout the interaction with the user in order to adapt to the use's needs (Brusilovsky 2001). It can be seen from this definition that AH systems can be useful in e-learning areas where students have different goals, where they can suggest the most relevant links or change the content as required (Brusilovsky, Eklund et al. 1998).

The main issue with AH is adaptation. As we can see in Figure 3-2 Brusilovsky described two methods of providing adaptation in AH (Brusilovsky 1996). They are as follows:

- **Adaptive presentation** is a technique for altering the content of the page to satisfy the needs of a particular user. For instance, a user with more knowledge about a certain area can be provided with detailed information, while a beginner with little knowledge can be given general explanations.

- **Adaptive navigation support** adapts the way links are presented and displayed to the user to help them to find their paths to their goal.

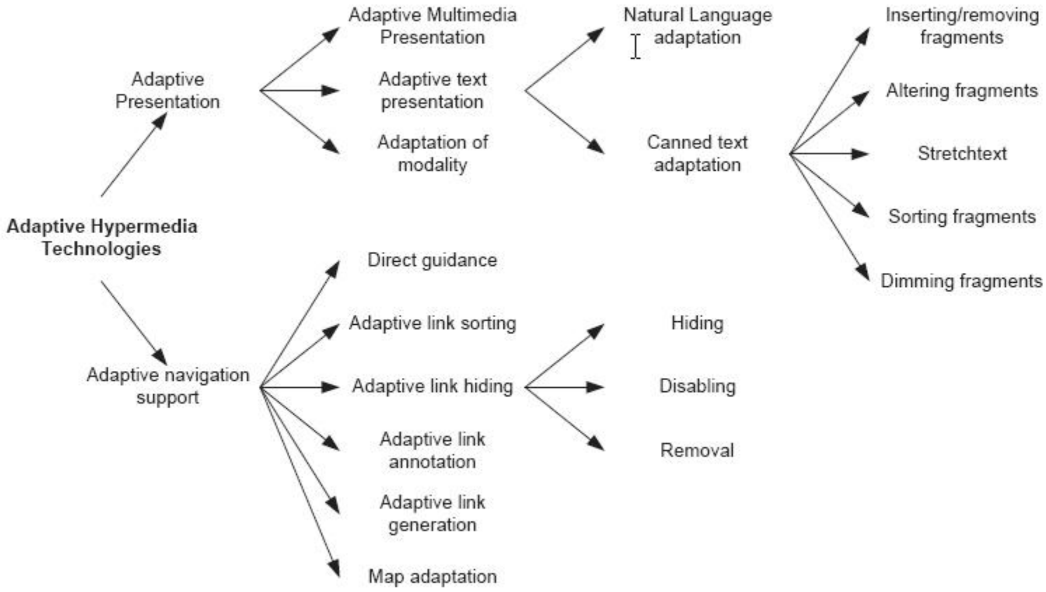


Figure 3-2: Taxonomy of Adaptive Hypermedia Technologies (Brusilovsky, 2001)

The adaptability of AEH is provided by the learner model. Thus, the design of the student model (SM) mainly influences the system adaptation (Papanikolaou, Grigoriadou et al. 2003). In using an SM, optimal teaching actions for given students and supports are selected during the realisation of the action (Brusilovsky 1994). SM is mainly used in order to adapt the AH to each student, which requires creating an individual model for each student that shows the current knowledge of the student. This process adapts the learning material and helps the students when using the system. Gonzalez *et al.* (2006) remarked that, in general, the adaptation process can be described in three stages: getting the information about the student, processing the information to initialise and update an SM, and using the SM to provide the adaptation. The SM can be useful in identifying student plans or solution paths, as well as for evaluating student performance (Gonzalez, Burguillo et al. 2006).

3.2.3 Existing Adaptive E-Learning Systems

This section details two adaptive e-learning systems that have been chosen to demonstrate how adaptivity can be applied in e-learning.

InterBook

InterBook is a tool for authoring and delivering adaptive electronic textbooks on the web. It provides the user with two types of knowledge: the domain model and the SM. The domain model provides a structure for the representation of the student's knowledge of the subject. For each domain model, InterBook stores the individual student's knowledge model, indicating the level of the student's knowledge of this concept and specific type of SM, called an overlay model. The overlay model allows the system to measure the student's knowledge of different topics. Thus, all student actions are tracked in order to set and update the knowledge levels required for that student (Brusilovsky, Eklund et al. 1998).

In InterBook, domain knowledge is indexed using prerequisite information. This helps InterBook to insert different coloured icons next to each page to show their status to the student. Pages visited are marked with a white icon meaning 'nothing-new'; pages with the pre-requisite condition met are labelled green, meaning 'ready-to-be-learned'; and pages that fail the pre-requisite condition are labelled red, indicating 'not-ready-to-be-learned'. This technique allows the student to visually measure a link's condition (Bailey 2002).

ActiveMATH

Melis et al. (Melis, Andrès et al. 2001) developed a generic, web-based, adaptive learning environment system called ActiveMATH. The ActiveMATH system dynamically generates interactive documents and formulae according to the student's content needs and presentation preferences. It was built for use in the field of mathematics and is realised as a client server web architecture that can be accessed using a standard web-browser. Instruction content and pedagogical knowledge can be filled in so as to provide a generic framework. ActiveMATH is configurable with pedagogical strategies and is therefore a tool for experimentation in appropriate learning settings.

ActiveMATH provides student with adaptive content selection and presentation, support of active learning, and adaptive appearance. It consists of the following components: a session manager, the knowledge base, a course generator, a user model, a pedagogical module, and deduction and computation service systems. The components can communicate by a standardised XML protocol. The course generator requests and processes information from the knowledge base, from the user model, and from the pedagogical module, to generate a document adapted to the student's goals, preferences, and knowledge. Information about the student's actions is passed from the session manager to the student model, where it is used for updating (Melis, Andrès et al. 2001; Melis, Budenbender et al. 2002).

3.3 Learning theory

Learning theory underpins the understanding of how students learn in a social context and can extend to the learning organisation, which can improve its learning activities through collective reflection and sharing experiences (Holmes and Gardner 2006). There are many different schools of thought regarding learning theories but this section will focus on the most relevant and applicable to e-learning, including constructivism, behaviourism and cognitivism.

Constructivism is based on the belief that our personal world is constructed in our minds and these personal constructions define our personal realities. Lefoe said that "there are diverse views on what the term constructivism means however they tend to share the tow a belief, (a) learning is an active process of constructing rather than acquiring knowledge and, (b) instruction is a process of supporting that construction rather than communicating knowledge" (Lefoe 1998). Knowledge in constructivism is how the individual creates meaning from his or her experiences, not what is said is true. Thus, the individual is required to examine learning processes by collecting, recording, and analysing data, and to reflect on previous understandings to construct an individual meaning (Jonassen, Davidson et al. 1995).

Behaviourism is connected to the idea that learning is largely unknowable; we cannot possibly understand what goes on inside a person and behaviourism states that only observable behaviours are worthy of research. It was first pioneered by Waston, who looked at how behaviour is affected by the process of learning. It focuses on behaviour and observes it, which is more important than understanding internal activities and takes into account simple elements such as specific stimuli and responses (Siemens 2005).

Cognitivism focuses on the mental activities of the learner and sees the mind as valuable and necessary for understanding how people learn (Limberg and Alexandersson 2010). Cognitivists consider the learning process in internal processes of information as involving thinking, motivation and memory. Cognitive psychology postulates that information is stored in the long-term memory as a node relationship, and maps that to show major concept in a topic. Cognitivism says that an existing knowledge structure must be present in order to compare and process new information for learning. Learning strategies should present the learning material and use the appropriate ways that enable the learner to process that material efficiently (Ally 2004) (McLeod 2003)

3.4 Personalised Learning

3.4.1 What is personalised learning?

Personalisation is a very important issue that is seen in many different E domains, such e-commerce, e-health and e-learning. In an educational setting, it is about working in partnership with the learner to tailor their learning experience and pathways according to their needs and personal objectives. Personalisation is perceived as the task of providing every learner with appropriate learning opportunities to meet individual learning needs supported by relevant resources that promote choice and advance learner autonomy (Bariso 2010). The concept of personalised learning emerged as a result of several developments. Partly, it is a reflection of living and working in a modern society, the development of new technologies and, in particular, how they can enable learners to break down institutional barriers and become a part of a global society.

There is also a growing recognition that current educational provision may be too narrow and restrictive and is not meeting individual learners or society's needs (Conole 2010). Current learners see technology as core to their learning environments, in particular computers and mobile devices. They use the internet to support their learning, to find information and to discuss work with other students and teachers. They are comfortable working with multiple representations, are digitally literate, and happy to turn to internet-based tools to help achieve their learning (Bariso 2010).

Sampson (Sampson 2001) has suggested that e-learning benefits from advanced information processing and internet technologies to provide the following features which could be considered crucial to personalised learning:

- **Personalisation,** where learning material is customised to individual learners, based on an analysis of the learner's objectives, status and learning preferences.
- **Interactivity,** where learners can experience active and situated learning through simulations of real-world events and online collaboration.
- **Media-rich content,** i.e. educational materials presented in different forms and styles.
- **Just-in-time delivery,** i.e. support systems that can facilitate training delivery at the exact time and place that it is needed to complete a certain task.
- **User-centric environments**, where learners take responsibility for their own learning.

Mourlas and Germanakos stated that personalisation is usually applied by three different types of adaptation: content level adaptation, presentation level adaptation, and navigation level adaptation (Mourlas and Germanakos 2009). Content level adaptation means the generating of a lesson from various educational materials depending on the knowledge level of the learner. Thus, novice learners will be provided with more explanations, while the more advanced may receive more detailed and in depth information (Brusilovsky, Eklund et al. 1998; Kosch, Döller et al. 2001; Cantador, Fernández et al. 2008). Presentation level adaptation is typically implemented by removing some information from a piece of text, inserting extra information or changing the layout of the page, instead of the text, such as font type and size and background colour (Specht 2000; Albayrak, Wollny et al. 2005). Lastly, navigation-level adaptation includes direct guidance, hiding, sorting, disabling or removing links, and generating new links (Brusilovsky 2003; Millard, Davis et al. 2003).

3.4.2 The need for personalised learning

There are significant benefits to personalisation. Jarvela summarised the benefits as follows (Järvelä 2006):

- **Increased student interest.** Personalising learning can help to raise the learner's level of interest in learning activities. As a result, learning speed will be increased and learners will achieve better results and understanding.

- **Better learning strategies.** Personalising learning can result in better learning outcomes by involve then in the development. If students learn with an aim at developing, they will learn who to learn new skills and involve social learning activities. This learning method can help to create learning communities with collaborative learning models.
- **Multiple ways to expand learning.** Personalising learning can improve the use of technology in education. Technology is an intelligent tool for providing individual learning, also supporting collaborative learning among different individuals. This can lead to multiple ways to expand the learning potential of every student.

3.4.3 Technologies and approaches

ITS and AH and are technologies and approaches to solving personalised learning. In ITS, the interaction with the student is based on a deep understanding of the student's behaviour. ITS systems assess each learner's actions in his or her learning environment and develop a model according to his or her knowledge, skills and expertise (Corbett, Koedinger et al. 1997). AH systems similarly personalise the learning experience by modifying content and links according to a user model. AH systems can be useful in e-learning areas where students have different goals (Brusilovsky, Eklund et al. 1998). However, these are typically centralised solutions and provide only limited autonomy to users.

Based on this work we believe that agent technology is a good approach to support personalised and informal learning. This is because of the characteristics of intelligent agents, principally their autonomy, social ability, adaptability, and reaction skills. Because of these characteristics, agents are a powerful way of representing learners in a system, adapting content and acting autonomously on their behalf. In addition, they can interact with multiple students and agents at the same time in order to facilitate collaborative and team learning without the need for a formal centralised authority (Sampson, Karagiannidis et al. 2002).

3.5 Personal Learning Environments

Rapid technological development and the move away from single centralised architectures towards shared, distributed architectures (White and Davis 2011) have influenced learning and teaching methods and activities. Traditional learning systems support the management of learning content with a focus on traditional learning environments, in which control comes from the teacher. In contrast, personal learning environments (PLEs) are based on user control through software tools that connect people and facilitate collaboration and communication (Schaffert and Hilzensauer 2008). The emergence of the read-write web through the implementation of web 2.0 and the growth in social web applications are identified as the origin of PLEs (White and Davis 2011). The idea behind this is that learning will take place in different contexts and situations and will no longer be provided by a single learning provider (Attwell 2007). Harmelen (2006) defines a PLE as a single user's e-learning system that provides access to a variety of learning resources, and that may provide access to learners and teachers who use other PLEs and/or VLEs (Harmelen 2006).

PLEs have some advantages that support e-learning. Anderson identifies some advantages of PLEs, which are (Anderson 2006):

- **Identity:** Learners have existences beyond formal school. This can be used to help learners to contextualise their own understanding and others to understand their epistemological legacy.
- **Persistence:** The reflective posting of a blog is a digital record of the learning process. When combined, they can be an integral part of the lifelong learning accomplishment and e-portfolio of the learner. They should not disappear at the end of a course.
- **Ease of use:** PLE environments can be customised and personalised, allowing education to flow into the learners' other net applications. A PLE can be customised by both teachers and learners to set educational environments.
- **Control and responsibility of ownership:** The PLE sets the context of the learning, which is created and sustained by the learner and is not owned by the institution.
- **Social presence:** The PLE supports online culture, allowing learners to project themselves socially and emotionally and to communicate with each other.

Although PLEs offer a new learning paradigm, there are some disadvantages. These are listed by Dron and Bhattacharya (Dron and Bhattacharya 2007) as follows:

- **Technical problems:** many technologies require computing capacity that not everyone has access to.
- **Technophobia:** it is hard for older teachers and learners to use the available technologies.
- **Loss of monitoring and control:** when using PLE, it may be difficult or impossible to police interaction between students.
- **Loss of history:** it is difficult to keep track of all the records needed to evaluate and reflect on the success or failure of the efforts made.
- **Inequalities:** there is a risk that those who use tools inefficiently are disadvantaged compared to those who use them more capably.

3.5.1 From formal to informal learning

There are many definitions of formal and informal learning, but the key distinction is that formal learning is typically described as learning that is managed in some manner by an authority (for example, a school or university), while informal learning is less managed, or may be managed by the learner themselves (Coombs and Ahmed 1974; McGivney 1999). A survey by Cross showed that 70% of adult learning is self-directed (Cross 1992) and informal learning is increasingly recognised as a key domain for TEL. Informal learning includes all that is learned in daily life experience at home, work and leisure. It is all the learning that occurs away from the world of organised and structured learning (Conner 2008) (Mason and Rennie 2007). Formal learning is planned learning in a structured setting. However, according to Conner (2008), most learning does not take place in formal programmes but happens through activities not formally organised by any institute. Indeed, informal learning accounts for over 75% of learning in modern organisations (Conner 2008).

Informal learning can be supported by e-learning concepts and technology, constituting a social web. These factors alter informal learning and it requires little effort for individuals to keep in touch and to stay connected. For many, the social web is a place for networking, community building and sharing collective experience (Mason and Rennie 2007).

3.6 Multiagent Systems in E-Learning

In e-learning, multiagent systems appear to be a promising approach to deal with the challenges in educational environments. They can provide new patterns of learning and applications, such as personal assistants, user guides and alternative help systems, which are helpful for both students and teachers in their computer-aided learning or teaching process (Aroyo and Kommers 1999). Using multiagent systems to design educational systems could lead to more versatile, faster and lower cost systems and also provides a dynamic adaptation of the behaviours of individual learners (Silveira and Vicari 2002) (Bednarik, Joy et al. 2005).

3.6.1 Examples of E-Learning Multiagent Systems

A number of researchers have applied agent technology to e-learning. Below are some different systems which are supported by agent technology and which use various approaches to facilitate e-learning.

ABADAL

Sun, Joy et al. proposed an agent-based sytem that is adaptive, able to learn and dynamic (Sun, Joy et al. 2005). The system is implemented as a BDI-based agent which makes decisions according to its knowledge, and is able to reason about its actions. This system can achieve adaptivity by the use of learning style schemes through a set of agents that use prebuilt knowledge. This knowledge is used to determine the learning styles and learning objects that are appropriate for individual students.

The system is functionally constructed by five agents, as shown in ure 3-3. Each agent is designed to satisfy a certain functional requirement that provides adaptive and dynamic learning materials. The agents are:

Student Agent: the agent which controls the communication with students and provides information through the interface from the user to other agents in the system.

Record Agent: the main data storage centre of the system; most of the data contained in the system is stored within this agent, which is also responsible for answering questions and feedback from the student agent and learning object agent.

Modelling Agent: this models individual students' needs and their knowledge background, based on good data for the model from the information provided by the Record Agent and provides this modelling to the learning object agent.

Learning Object Agent: the agent responsible for managing the learning objects, which are organised beads on the learning style scheme. By communicating with the modelling agent, it can provide students with different learning styles with relevant learning objects.

Evaluation Agent: the agent which presents the learning objects in an individual and adaptive learning path to each student. It uses the student data in the system to decide which learning objects will be sent to each student.

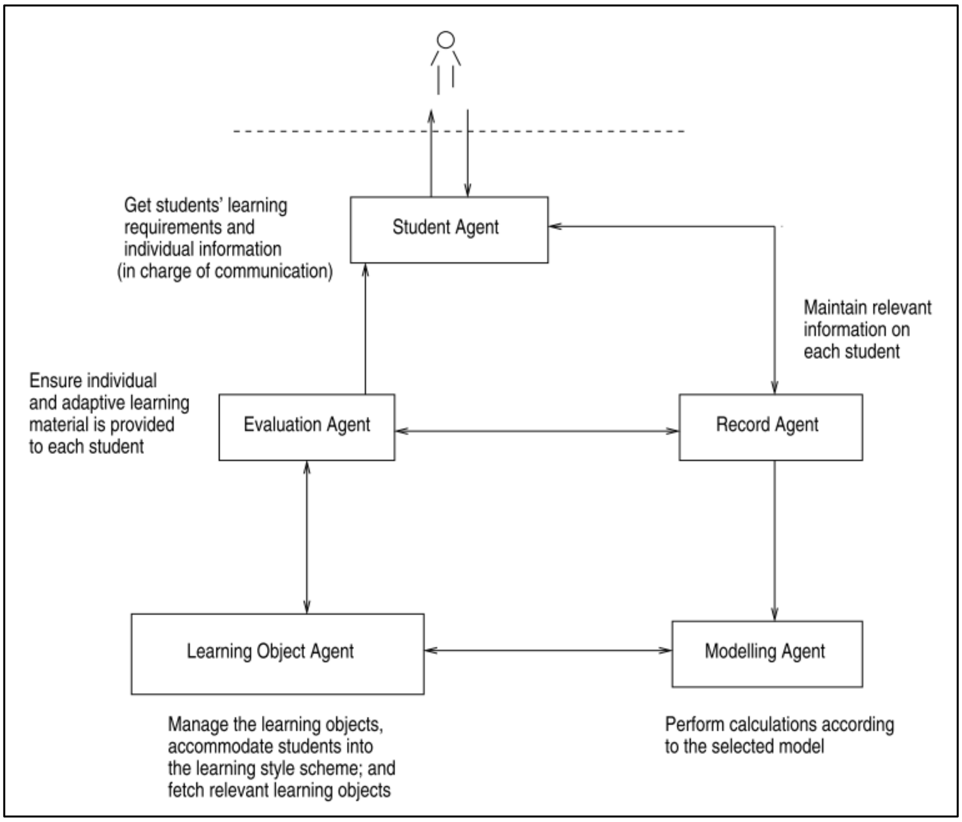


Figure 3-3: The architecture (Sun, Joy et al. 2005)

ILMDA

Soh presented an intelligent agent prototype called ILMDA. It uses the history of learning to delivers to the learner a learning materials based on the student's profile, interaction and the student dynamic activity profile (Soh 2006). These learning materials are composed of a tutorial and exercises to assess the student's understanding. The system chooses the appropriate examples and exercise based on how the student progresses through the learning materials and based on his or her profile.

The system is built on the three-tier approach shown in Figure 3-4. ILMDA includes GUI, a database and reasoning in between. Through the GUI the student can interact with the system and access the learning materials. There is an agent capturing the student's interactions with the GUI and using a reasoning module with a parametric profile of the student. This reasoning module can help a search query to retrieve from the database most appropriate example or problem and adapt to the student and delivers in real time to the user through the interface (Soh 2006).

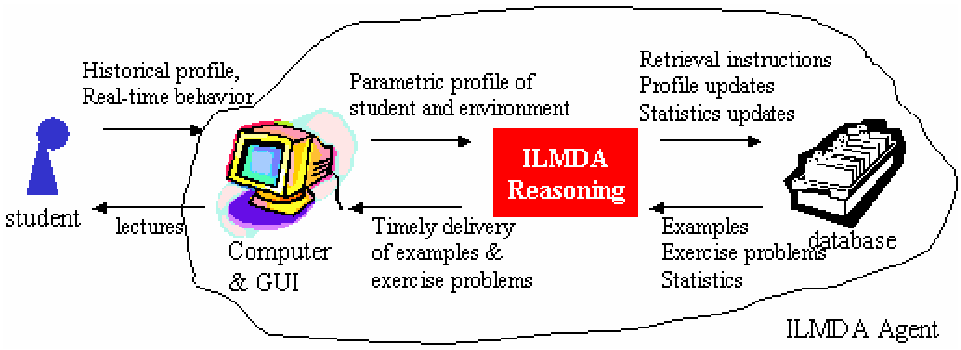


Figure 3-4: The ILMDA system and environment (Soh 2006)

IDEAL

Shang et al. presented an intelligent agent-assisted environment for active learning (Shang, Shi et al. 2001). This web-based multiagent learning system, known as IDEAL, has a three-tier architecture (Figure 3-5). The system provides a rich set of online contents, maximises the interactivity between the intelligent learning system and the students, and customises the learning process to the needs of individual students.

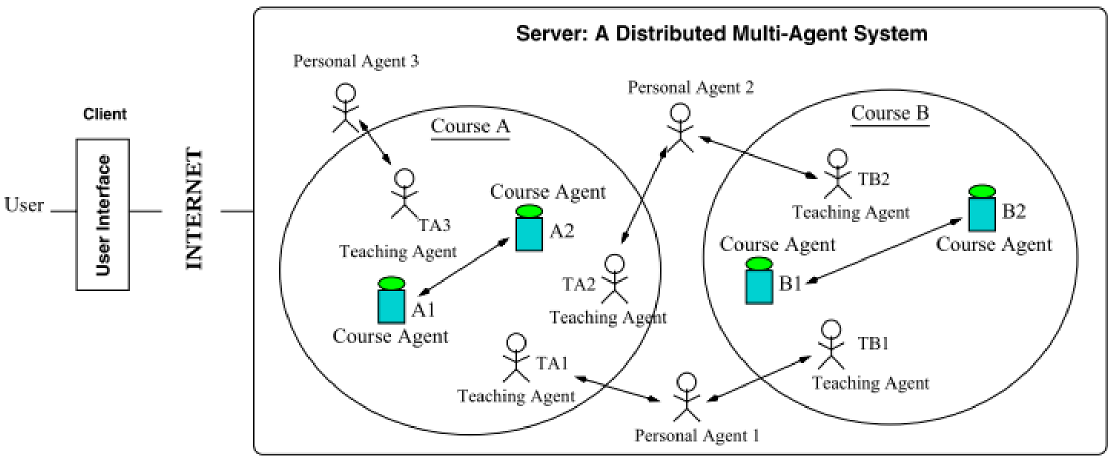


Figure 3-5: The IDEAL framework (Shang, Shi et al. 2001).

The system supports personalised interaction between users and the learning systems, enables adaptive delivery of content, facilitates automatic evaluation of learning outcomes, and provides easy-to-use authoring tools.

In IDEAL, each student is assigned an agent managing learning styles and interests to generate a student profile. This agent will contact other agents in the system via communication channels to deliver the learning material to the student. The system has *course agents* managing course materials and techniques for teaching connected with the course, also acting as mediators for communication among agents. Also, there are *teaching agents* that can talk to any course agent and select one nearby for better performance (Shang, Shi et al. 2001).

The system uses active XML documents to deliver course materials. These materials consist of small components (lecturelets) around the subjects to be learned. Lecturelets contain XML documents and instructions explaining how the documents should be displayed.

PVLE

Xu and Wang developed a personalisation model, from which they proposed intelligent decision-making agents to achieve the personalisation in PVLEs by employing intelligent agents. Subsequently, they designed a multi-agent based PVLE architecture (Xu and Wang 2006) (Figure 3-6). It is based on the personalisation model and is designed in three layers; learner layer, agent layer and repository layer. The learner layer provides an adaptive interface for online learners. The agent layer contains a number of intelligent decision-making agents that support personalisation. These agents are designed based on the personalisation model and fitted into two learning process stages. This layer comprises three agents:

Activity Agent: This agent records learning activities and learning duration on a particular task and documents. All these activities are stored in the learner profile by the Activity Agent.

Modelling Agent: abstracts the learner model, based on the learner profile.

Planning Agent: uses learner model based on the and the content model to analyses the current learning plan of the particular learner.

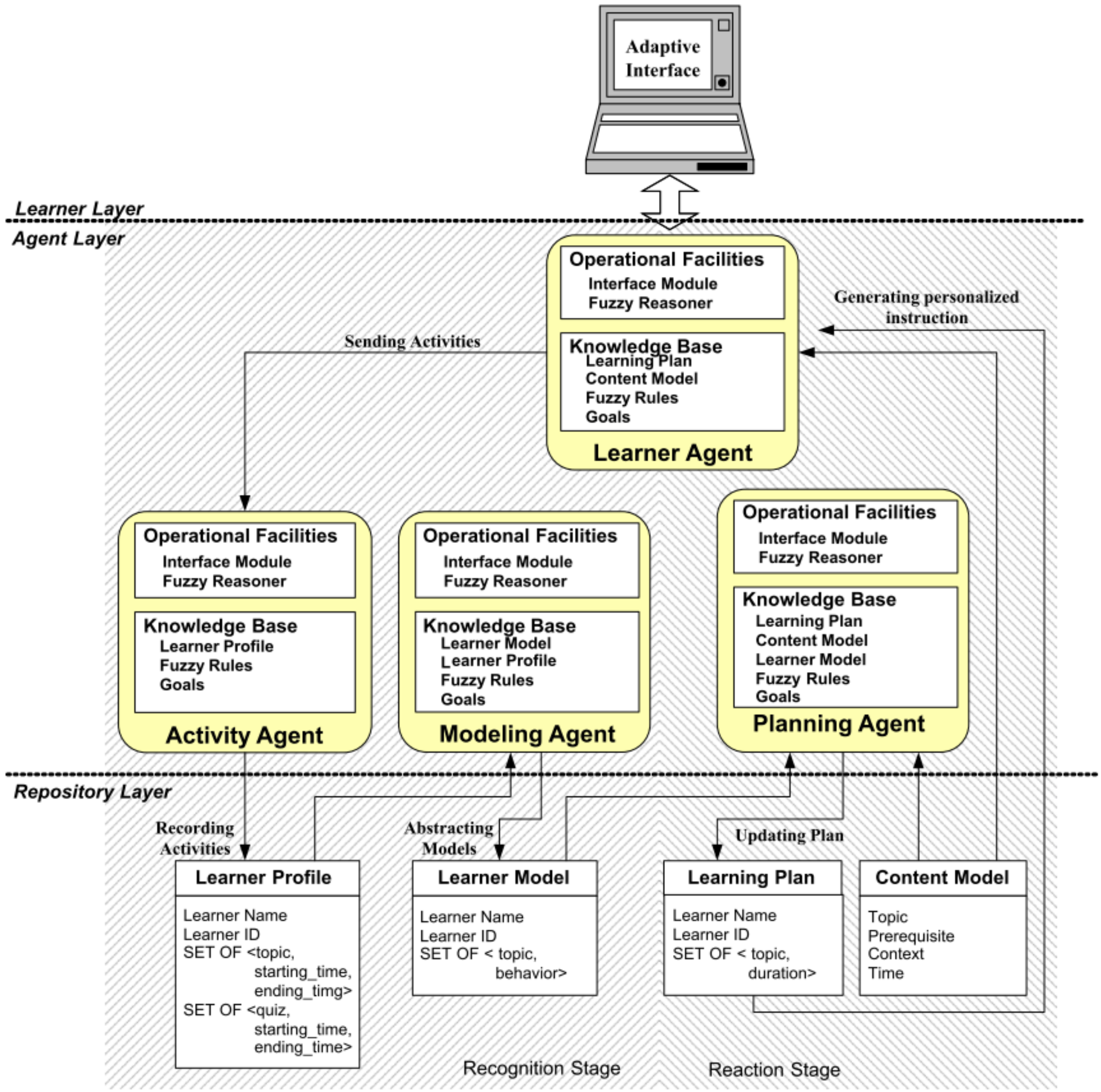


Figure 3-6: PVLE architecture (Xu and Wang 2006)

3.7 Motivational Scenarios

It is believed that agents have the potential to transform Technology Enhanced Learning (TEL) by enabling scenarios that are simply not feasible with today's technology. This is possible because of some of the key features of agent systems, such as distributed control and agent autonomy. In this section, we illustrate this potential through three different TEL scenarios that show how agent technologies could be used in e-learning to take full advantage of the agent's ability to communicate and negotiate. Each case is composed of a *description* of the scenario, an analysis of the *agent solutions* that make the scenarios possible, and more speculative *variations* of the scenario that would share the same features.

Although the system explained in previous section applied agents in e-learning, none of these systems apply any fundamental agent theories (such as mechanism design or social choice theory) to guide their design choices. They treat agent systems as component architectures, without really taking advantage of the distribution or autonomy of agents.

Through the scenarios, we hope to show how certain types of problem in e-learning fit with known agent solutions (voting systems, coalition formation, and auction systems). We also hope to show how agent systems enable a very high level of personalisation and to start a discussion about the implications for education in the future.

3.7.1 Scenario One: Course Selection

Description: This scenario concerns a university that wants to support students who are interested in a wider variety of courses than it is possible for the university to offer. They must be chosen from among alternatives that are related to particular expected outcomes (Babad and Tayeb 2003). The university must therefore somehow choose which subset of courses to run. This is a common scenario with Higher Education degree courses, where often students are offered a number of courses and for economic reasons only the most popular courses are run. However, current solutions are centralised, requiring students to hand over their preferences to a central algorithm controlled by the university. In addition students are unable to respond to cancelled courses by changing their preferences. From a personalised learning point of view this is undesirable, as despite the tension between the goals of the institution and the students (the institution really wants to run as few courses as possible whereas each student wants to get the courses in which he or she has most interest), the student must hand over almost all control to an opaque process managed by the university (Brown, Varley et al. 2009).

Agent Solutions: In agent systems this scenario can be characterised as a voting problem. It occurs whenever agents are required to invest in or vote for a limited number of options within a greater number of more or less attractive possibilities. There are numerous potential solutions to voting problems where the outcome impacts all the agents (sometimes described as problems of social choice) but through transparent protocols they offer fairness, decentralisation and independence, as they allow agents to choose their own voting strategies. This distribution of control fits well with personalised learning.

Variations: This scenario describes students making choices about courses within a single institution, but because agent solutions are decentralised an agent solution could also work in situations where students were choosing courses from multiple institutions (for example, as part of a personalised degree programme across Bologna compliant universities). Bologna is a European higher education process that focuses on cooperation between universities for academic exchange purposes. It aims at enhancing the mobility of students and higher education staff within the European Higher Education Area and at establishing a high-quality advanced knowledge base. A further aim is to make it more attractive for people from non-European countries to study in Europe (Bologna-Secretariat 2010). In a case using the Bologna process in a personalised degree programme, the factors taken into account in an individual agent's voting strategy might include issues such as institutional reputation, distance from home and student facilities.

3.7.2 Scenario Two: Group Formation

Description: Collaborative learning is an effective means of learning. The students benefit from sharing each other's perspectives and learn from each other in addition to learning from the instructor. Students can get help from more advanced students in their group when they struggle, and also the better students can learn by teaching the students who are struggling (Redmond 2001). In addition, it is practical for students to arrange themselves into groups for learning, for example to share equipment, to help with timetabling, or for pedagogical activities such as discussion. Students can group themselves or be grouped by a teacher either randomly or based on some criteria. Group formation is important because although all students need to be allocated to a group, the mix of students might be important. For instance, it may be desirable to have a mix of abilities, so that no one group has an advantage over another in assessment (Christodoulopoulos and Papanikolaou 2007).

Current solutions are normally centralised, meaning that students cannot have different criteria for group selection (for example, some students might wish to be in the most effective groups, while others would rather learn with existing friends) similarly to Scenario One this one-size-fits-all approach is at odds with personalised learning and the requirement to consider the learner's experience (Ounnas, Davis et al. 2007) .

An interesting aspect of this scenario is that sometimes the goals of the teachers are at odds with the goals of the students. The students may wish to be placed in groups with their friends or with students that will help them to achieve good marks, while the teacher may want to arrange students in groups that will help them to learn more material or to learn it more quickly. This means that even non-centralised solutions may need to be mediated by a central authority.

Agent Solutions: In agent systems, an appropriate metaphor for this scenario is coalition formation a process by which agents form, join and switch groups until a stable set of coalitions is made. There are numerous potential protocols for this, for example by having an initial allocation, perhaps based on criteria set by the teacher, and then for the students to negotiate exchanging their places with students in other groups. The agent framework provides the conversational mechanism for this negotiation, but the agents need some self-organisation. For example, each coalition might produce a virtual leader agent to negotiate with the leaders of the other groups. At the same time, each leader agent has to negotiate with the teacher agent because any changes made in group membership still have to conform to the constraints set by the teacher agent.

Variations: This scenario envisages group formation occurring under the supervision of a teacher or lecturer, and therefore implies a more formal educational context. However, distributed group formation enabled by agents could enable informal learners to also benefit from group work, by helping them form coalitions with other (potentially remote) learners who share similar pedagogical goals. Such distributed agent-based group formation systems could be of great help to life-long learners, and could form the basis of informal group work and peer assessment without the need for a mediating teacher or institution.

3.7.3 Scenario Three: Personalised Learning

Description: Different students may have different personal preferences about the way they want to learn or to be assessed (Karagiannidis and Sampson 2002). These preferences may be because of preferred learning styles but could also be for other practical reasons (such as time commitments in their personal lives or different project requirements). An institution has difficulty catering for these preferences, due to the mixed cost of providing different activities (for example, lectures are cheaper than tutorials) and resource restrictions (such as time commitments of staff or access to specialised equipment or information sources) and their own guidelines and regulations about having a mixed set of assessment styles (for example, many universities are cautious about having courses assessed totally by coursework).

It is therefore rare for an institution to allow much flexibility at an individual level. Although there are limited solutions that allow a cohort to make choices about how they will be taught or assessed, these tend to be managed directly by teachers and are therefore of limited complexity (for example, it might be possible for the students to negotiate with their teacher about the methods of learning or assessment that will be used).

Agent Solutions: In this kind of scenario, there are a number of limited resources (tutorial slots, lab equipment, seminar places, etc.) and many individuals competing for them. In agent systems this situation is characterised as an *auction*. The institution associates a cost with each type of activity and wants to minimise the total cost or at least prevent it from rising above an agreed level. This cost need not be purely financial; it could include factors such as value to external assessors or complexity for staff to manage.

There are many different kinds of auction and therefore different solutions to this problem. But as an example we can define a utility function for each agent that calculates a student's satisfaction with the activities they have been allocated. Following an initial allocation, agents could then bargain (negotiate) with their institution, exchanging items according to their cost until their utility function is maximised within the constraints of the institution's cost level.

Variations: Using an economic model allows a university to adjust the wealth (and therefore purchasing power) of certain students according to circumstances. For example, students with learning difficulties such as dyslexia could be given more credit, allowing them to tailor their learning experience within the same economic framework as other students. More controversially, students might actually be allowed to purchase additional credit, in effect buying themselves more expensive tuition through the university fees system.

3.8 Summary

This chapter started with a brief description of e-learning and its systems with a focus on different approaches, such as ITS and AH. It went on to describe how technology can assist education by providing flexible services applications and ways to learn. A number of different scenarios in e-learning where agent technology might be useful were then presented.

It also presented different existing systems which are supported by agent technology. Although existing system apply agents to e-learning, none of them applies any fundamental agent theories, such as mechanism design or social choice theory, to guide their design choices. In contrast, the approach presented in this thesis is to explore how agent systems can be used for decentralization as well as personalization. In our experiment we examine how voting mechanisms can be used in an e-learning scenario where a University agent represents all the courses available, and where student agents can vote in any way he or she prefers. Thus work will be explained in details in the following chapters.

Chapter 4.
Experimental System Design

In Chapters 2 and 3, we discussed agent technology focusing on its definition and the related topics that enable multiagent systems to make collective decisions such as social choice theory and voting protocols. We also discussed related work in technology and education. This focused on personalised and informal learning, the existing use of agent technology, and how these technologies change the way that students use it when learning.

Laying out the foundation of the thesis work, this chapter describes the multiagent system simulation that has been used to investigate how agent systems can form a good framework for distributed e-learning systems. It also considers how they can be applied in personal learning contexts where students are autonomous and independent.

To investigate the research hypotheses which are in section 1.2, we ran three main experiments as follow:

Experiment One: Agent Simulation for Decentralised E-learning Scenario (Chapter 5).

Experiment Two: Agent Simulation for E-Learning System Involving Complex Preferences (Chapter 6).

Experiment Three: Fairness in Intelligent E-Learning Systems Using Agent Simulation (Chapter 7).

This chapter will explain the agent environment designed to run these experimental simulations. It describes the architecture of the system and the protocol and voting procedure that the agents follow. It also sets out how student preferences are modelled and describes three different voting strategies that have been implemented. The changes required for implementing each simulation will be described in its chapter.

4.1 Course selection scenario

In section 3.6 we consider three different e-learning scenarios showing how agent technologies could be used in e-learning to take full advantage of the agent's ability to communicate and negotiate. In this research, the course selection scenario was chosen as the basis for carrying out the investigation. This scenario concerns a higher education institution which offers a set number of courses. Students are required to register for these courses but the institution may choose not to run a course if there is insufficient interest. This scenario was chosen because it is both simple and sensible and it provides a straightforward way of evaluating agent satisfaction. In this scenario there is one agent that represents the university which is in charge of resources and many agents that represent students. However, despite its simplicity, this scenario still features autonomy and negotiation and reflects the potential of using voting based agent technology in e-learning.

In more details, many educational organisations there exist restrictions on which courses might run, due to the overheads of running each course. In the context of personalised learning, where students take more control of their learning, the university would like students collectively to make the decision about which courses are run, while taking into account any individual preferences. The intention is that by using a voting protocol, students can express their preferences over the courses available and choose their own strategies for deciding collectively which courses to run. Multiagent systems are a powerful technology to tackle this complexity and to enable flexible course selection systems through enabling features such as autonomy, responsibility, social ability and intelligence (Jennings and Wooldridge 1998; Wang, Sun et al. 2005).

We considered three different cases. The reason for this is to ensure that the system works well and that the data is representative. These cases differ in terms of the number of students, the number of total courses and the number of running courses. Table 4-1 shows the settings for these cases. The cases have been chosen to reflect the kind of courses typical in UK university departments. The column *# courses (m)* shows how many courses there are for each case. We consider three modules: Case 1 represents a large undergraduate course with 100 students, which is a typical bachelor course; Case 2 represents a medium-sized undergraduate university course with 60 students, which can be considered a typical third-year course; and Case 3, which is a postgraduate university course with 20 MSc students. The column *#running courses (r)* shows the number of courses that will eventually run. The column *#student (n)* shows the number of students in the experiments.

Table 4-1: Different settings for the cases

case	*#courses (m)*	*#running courses (r)*				*#student (n)*
1	51	10	20	30	40	100
2	33	11	18	26		60
3	18	4	9	14		20

There are many factors that influence the behaviour of the agents. In order to evaluate the strategies, the following variables were identified:

Number of courses (m): This is the total number of courses that the university provides and which the student can vote for.

Number of running courses (r): This is the remaining total number of courses, after the ones with the lowest student interest have been cancelled.

Number of students (n): This is the total number of students in the system.

In addition to the above variables, we also have a number of constants:

Initial points (IP): This is the number of points that each student initially receives. Without loss of generality, we set this value to 100 in the experiments.

β: This is used when calculating the probability of a course being cancelled for the intelligent strategy. This will be explained when talking about the systems strategies (see Section 4.5). Throughout our experiments, we set this value to five since it was shown in initial tests to perform well.

Fraction of point to invest(fb): The maximum number of points that would be spent in the current round.

4.2 Architecture

This section describes the architecture of the agent environment that was designed to investigate the potential of using multiagent systems to support the personalised e-learning systems. The reason for using agent to vote is because there will be multiple rounds of voting where it would be very difficult for a student to participate every time when course is cancelled to the voting system to vote. Moreover, through these rounds, an agent can learn from the information provided and predict what could happen to make a decision.

All entities and objects that exist in the environment are shown in ure 4-1. This architecture consists of the two essential components: *student agents* (SAs) and *university agent* (UA).

The student agent has two functions: one is to enable students to expresses their preferences to the student agent and it also chooses an appropriate voting strategy. The second is to interact with the university agent based on these settings. The university agent manages the votes that are cast by the student agents and decides, based on the voting protocol (see section 4.3) and the votes received, which courses will be cancelled. Furthermore, after completing the entire process, the university agent will provide the student agent with a final list of running courses. More detail about how the components interact is explained in the following section.

The agents in this architecture are autonomous in that they have control over their own votes and can decide how to spend them without the intervention of other agents. They also can perceive the environment around them and respond to any changes occurring. They are also able to take appropriate initiatives rather than only taking action in response to the environment. This includes an intelligent strategy wherein they expect a course to be cancelled and thus vote on other.

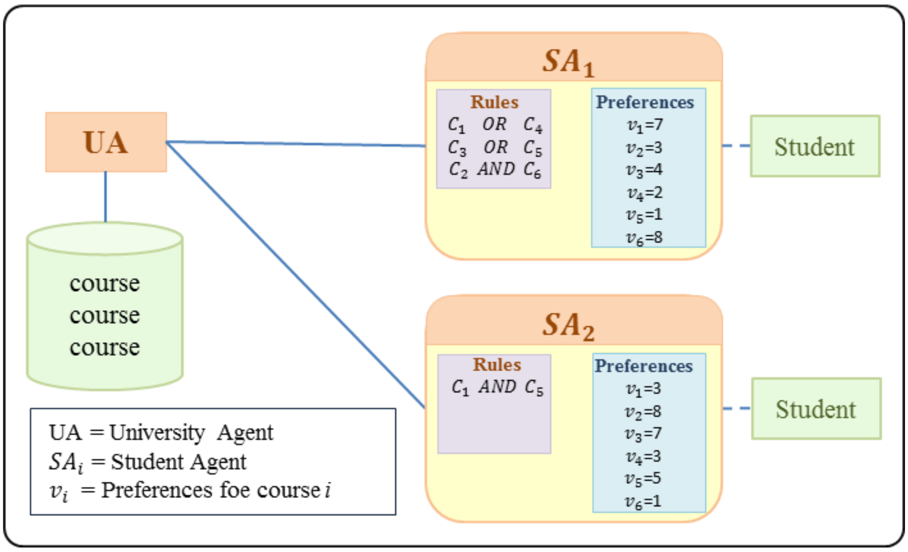


Figure 4-1: System architecture

Because we are intending to test this in a simulation that will run 10,000 times, we need a very fast execution time. For this reason it is an appropriate to use an agent framework such as Jade because of the communication overheads. We implemented the agent using a normal object-oriented JAVA program, where agent communication is achieved through local method calls. This makes simulation faster and enables the cases to run thousands of times, though it would not be appropriate for a deployed agent environment. If this applied with a real student, it would be straightforward to translate to a distributed solution.

4.3 Voting Protocol

In general, a protocol is the set of rules that controls the interactions between agents and determines the beginning and end conditions of a given conversation (Beer, Inverno et al. 1998). Voting procedures describe the way and the roles by which the preferences of individuals are aggregated to produce a collective decision (Brams and Fishburn 2002). In this work, we introduce a novel voting protocol which combined features from both single transferable vote (STV) and cumulative voting (see section 2.4.1). Specifically, it takes advantage of the features of cumulative voting to express preferences using points. At the same time, it allows for multiple rounds to avoid wastage by allowing the transfer of points in a similar way to the transfer of votes in STV. Having multiple rounds also allows a student to learn from previous rounds and to adjust its voting behaviour accordingly.

In more details, the protocol works in several stages. In each stage, the student agents cast their votes over the courses by allocating points to them, taking into account their preferences. The course that receives the lowest number of cumulative points is cancelled and the points that were allocated to the cancelled course are refunded. In the next round, the student agents can use these points (and any points that they did not use in the previous rounds) to vote again. Furthermore, in each round, the students are informed about the total number of points that have been allocated to the remaining courses so far. Note that, once allocated, students cannot retrieve their points, unless the course is cancelled. The advantage of this iterative approach is that votes are not wasted since points allocated to the cancelled course can be reused for the remaining courses. Another very important thing to note is that students can use the information about the current "popularity" (i.e. the current cumulative points) of the courses to guide their voting behaviour (we discuss this in more detail in Section 4.5 where we discuss the voting strategies of students). The protocol proceeds as follows (see Figure 4-2)

1. Each student initially receives an equal and fixed number of points, IP, from the university agent that they can use to cast their votes.
2. Each student has their own preference over the courses and can allocate some or all of their available points to the available courses based on their preference (they do not have to allocate all their points, but cannot allocate more than they have).
3. After receiving votes from all students, the university agent calculates the cumulative points for each course.
4. The university agent cancels the course with the lowest cumulative points.
5. The university agent refunds the points for the cancelled course to any students who voted for it.
6. The university agent informs all the student agents about the cancelled course and the current cumulative points allocated to the remaining courses.
7. Now student agents can vote again using their remaining points (this includes the refunded points as well as any points which were not allocated in the previous rounds) and the process is repeated until the desired number of courses is remaining.

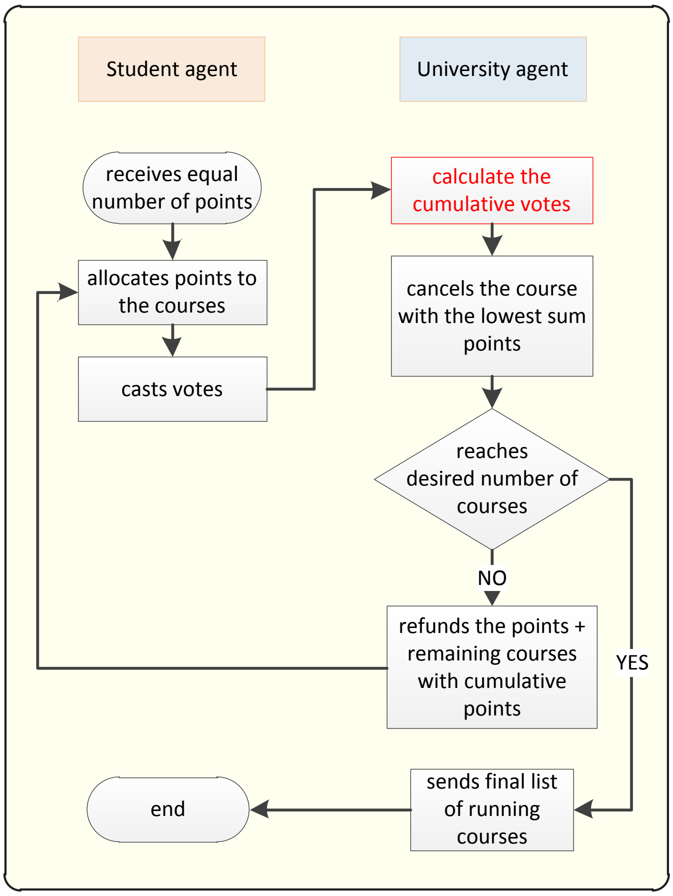


Figure 4-2: the voting protocol

For example, if there are 40 courses available in total, but the university only has sufficient resources (e.g. staff and lecture rooms) to run 30 courses, then the voting will proceed for ten iterations or rounds. At the end of each of these rounds, the course with the least number of cumulative points is cancelled.

4.4 Simple Preferences

A simple student preference is used in the first simulation that investigates agent technology for the decentralised e-learning scenario (see Chapter 5). Here, a student's preference for a certain combination of courses is simply the sum of the preferences for each individual course.

Students are asked to proclaim their preferences over a set of courses. Each student has their own preference for the different courses that the university offers. These preferences are modelled using a simple scoring model that describes a student's preference for each course as a number between 0 and 10, where 0 means that the student has no interest in the course and 10 means maximum interest.

After the voting process has completed, we can use these preferences to calculate a given *student's satisfaction* for the running courses. This is calculated by summing the preferences for courses that are running, as a fraction of the total preferences. For example, there are 7 courses available in total, and the preferences of a student are as follows: $v_1 = 7, v_2 = 3, v_3 = 1, v_4 = 10, v_5 = 4, v_6 = 8, v_7 = 3$, where v_i is the preference for the i^{th} course. The university decides, based on the voting process, only to run courses 2, 4 and 6. The student satisfaction is then calculated as follows: the sum of the preferences for the running course (21) is divided by the sum of the preferences for all courses (36). So the student satisfaction, S, is:

$$S = \frac{21}{36} \cdot 100 = 58.33$$

Figure 4-3 illustrates the process as follows:

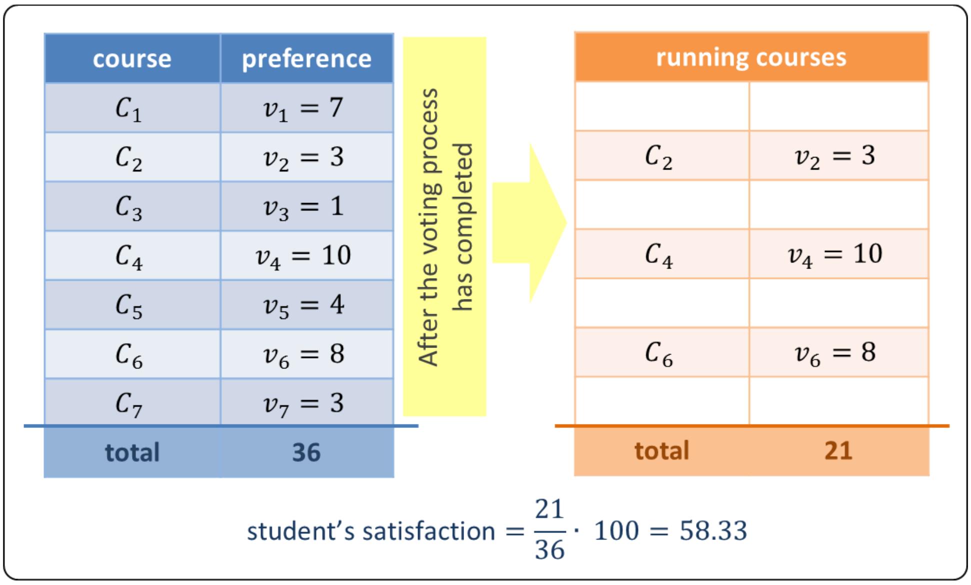


Figure 4-3: Calculating student's satisfaction with simple

4.5 Voting Strategies

Abstractly, a strategy determines the agent's plan of action to achieve a particular goal. It specifies the way in which an agent behaves in a given environment (Wooldridge 2009). In our scenario, the strategy determines the number of points to allocate to the courses in each voting round, given the preferences of the agent and the information received by the UA about the outcomes of the previous voting round (i.e. the points allocated so far, and the course cancelled). In this section we introduce three different strategies for the SAs to allocate the points that are used across all of the experimental simulations, namely: proportional, equal share and intelligent so that we can see different level of intelligence and also to explore the effect that different proportion of agents taking different strategies has on the outcome. In what follows we describe each of these strategies in detail.

Proportional: The proportional strategy is included as an example of a simple but sensible strategy. Consequently, it provides a good benchmark that we can use to compare the performance of more sophisticated strategies. This strategy is simple in that it does not consider the information received by the UA about the current number of points allocated to the courses. The main idea behind a proportional strategy is that, in each round of voting, the student agent distributes its points proportionally to the student's preferences for each course.

In more detail, the number of points allocated to course j is calculated as follows. Let *RP* denote the total number of points remaining (in the first round *IP*=*RP*), m is the total number of courses available, and the vector $\vec{v} = \{v_1, v_{2,}, \dots, v_m\}$ denotes the student preferences. Then, the total number of points to be allocated to course j, b_j is:

$$b_j = \frac{RP}{\sum_{i=1}^{m} v_i} \cdot v_j$$

Equal-share: The equal share strategy is included as an example of a very simple and ineffective strategy which provides a good lower bound on the performance of the system. An equal share strategy is based on the principle that the SA gives all courses an equal number of votes, regardless of the student's preference. The following formula was used to calculate voting points each course:

$$b_j = \frac{RP}{m}$$

Intelligent: The intelligent strategy is considered an advanced measure of what can be achieved. It is included as an example of what can be achieved with a more sophisticated strategy that learns as the voting procedure progresses from one round to the next. Its effectiveness can be gauged by comparing it to the proportional strategy and the lower bound given by the equal share strategy. The main idea behind this strategy is that, in each round, the SA tries to predict the probability that a course will be cancelled based on the number of points currently allocated to each course from previous rounds. (this information is provided to all the SAs by the UA at the end of each round). When allocating points, this strategy does not spend all the points in the first round, in order to take advantage of the information that is received in subsequent rounds. Otherwise, it would have no more points to use in these rounds (unless a course for which votes were cast is cancelled, in which case the points are returned). In the last voting round, it allocates all remaining points. Furthermore, in the first round, because the strategy does not have any information about courses, it distributes half of the points using the proportional strategy as explained above.

In more details, the intelligent strategy consists of three parts. First, it tries to estimate the probabilities of a course being cancelled. Then, given these probabilities, it tries to estimate the expected utility (we use utility here instead of satisfaction) for a given distribution of points. Finally, it uses a search algorithm to find the point distribution which maximises expected satisfaction. In what follows, we discuss these components in turn.

Then, based on this probability for which a course will be cancelled, it can calculate its expected satisfaction for a given allocation of points, and it will allocate the points such that the expected satisfaction is maximised. The probability of a course being cancelled is estimated using a softmax function, which is commonly used in discrete choice theory to make decisions in the case of incomplete information (Hensher, Rose et al. 2005). The probability that a course i is going to be cancelled in the future is given as:

$$P_{CANCEL}(C_i|\vec{b}) = \frac{e^{\frac{-(cp_i+b_i)}{\sum_{j=1}^{m}(cp_j+b_j)}\cdot\beta}}{\sum_{k=1}^{m} e^{\frac{-(cp_k+b_k)}{\sum_{j=1}^{m}(cp_j+b_j)}\cdot\beta}}$$

Where cp_i is the cumulative number of points which have so far been allocated to course i, and b_i is the number of points that the student agent is planning to allocate to course i in the current voting round and $\vec{b}$ is the vector of points to be allocated. Furthermore, β is constant which enables a range of different strategies. For example, if $\beta = 0$, then each course is equally likely to be cancelled, irrespective of the cumulative number of points currently allocated. At the other extreme, as $\beta \to \infty$, the course with the lowest total number of points will be cancelled with probability 1 and all other courses will be cancelled with probability 0. All other cases fall somewhere in between. In our experiments we tune the parameter β such that it performs well in practice. Also we set the parentage of points fb representing the maximum number of points that would be spent in the current round as a fraction of total point.

We now show how we can use this probability to calculate the *expected satisfaction*, *ES*, of the student, and how to find the allocation which maximises this expected utility. The expected satisfaction is given as:

$$ES(\vec{b}) = \sum_{i=1}^{m} (1 - ProbCancel_i(\vec{b})) \cdot v_i$$

Note that the expected utility depends on $\vec{b}$, i.e. the number of points SA is going to allocate to each course in the next round. The next step is then to find the allocation that maximises this expected utility. We estimate this using a search algorithm based on *random sampling*, which proceeds as follows:

1. We randomly generate an allocation vector $\vec{\mathrm{b}}$ subject to the constraint that the total number of points is equal to (fb) the maximum number of points that we would like to spend in the current round.
2. The student agent calculates the expected satisfaction.
3. If the current solution has a higher expected satisfaction than any previous solution, then keep the solution. Otherwise, discard it.
4. This process is repeated 1,000 times and the solution with the highest expected utility is kept.

Finally, the SA provides the UA with the point allocation obtained from the above algorithm and preceding rounds until reaching the list contain the running course.

4.6 Summary

This chapter presented the multiagent system that has been used to investigate how agent systems can form a good framework for distributed e-learning systems. It began by explaining the agent environment, then describing the architecture of the system and the protocol and voting procedure the agents follow.

It also described how student preferences are modelled. It then presented three different voting strategies.

In the coming chapters, we will cover the remaining contributions of this thesis by using the multiagent system simulation described in this chapter and conducting various experiments to test the hypotheses of this thesis.

Chapter 5. Agent Simulation for Decentralised E-learning Scenario

Having discussed in previous chapters the foundation of the thesis by signposting the concepts and technologies that can be utilised to build a decentralised agent-based e-learning system, this chapter and the following ones will cover the contributions of this thesis by discussing the various experiments conducted to justify the hypotheses.

The main objective of this chapter is to use the multiagent system environment described in chapter 4 to investigate whether voting procedures in particular and multiagent technology in general could potentially replace a centralised infrastructure. We will also explore the impact of agents using different strategies on overall student satisfaction, which is done by testing ***H1:***

> *In e-learning scenarios such as academic course selection, a decentralised agent system using a voting protocol can achieve a comparable level of overall student satisfaction to an optimal centralised approach, while maintaining levels of privacy and choice.*

In order to see the potential for an agent system replacing the centralised approach, we ran an experiment in two parts. In the first part called the Identical Voting Strategy, we ran a number of simulations across the three cases but with all the students using the same voting strategy. We then compare the results to optimal social welfare i.e. the social welfare when the overall most preferred courses are selected assuming that the university knows all the student preferences and selects the courses which maximise the social welfare

In the second part called Combination of Voting Strategies, we repeated those simulations but this time the student could take a mixture of strategies.

We compare a case in which a proportion of the students use one strategy and the remainder of the students use another strategy.

The main questions this experiment tries to answer are:

Q1: Can voting procedures in particular and multiagent technology in general potentially replace a centralised infrastructure? This is answered by Part 1.

Q2: What is the impact of the three strategies described in the previous chapter on overall student satisfaction in terms of personalisation? This is answered by Part 2.

This chapter starts by explaining in detail how to conduct these two parts of the experiment, including methodology, analyses and results. A discussion and concluding remarks end the chapter.

5.1 (PART 1) Identical Voting Strategies:

This experiment has the aim of demonstrating decentralisation. It measures the performance of each strategy separately by allocating all students to a single strategy and then compares the outcome with the optimal solution. Here, "optimal solution" means that the central authority (i.e. the university) knows all the student preferences and selects the courses that maximise social welfare.

5.1.1 Methodology

In order to evaluate the experiment and analyse the outcome, the student preferences need first to be set up in a way that matches the objective of the experiment. Consequently, the experiment starts with the setting of the student agent preferences. In section 4.4, we explained in detail how to model the student preferences in general. In this section, however, we explain how generate these preferences to run the experiments.

To this end we use an approach that randomly initialise the preferences is through settings. Every student agent has individual preferences. For each student agent and each course, we generate preferences from a uniform distribution between 0 and 10. These preferences were randomly generated so it was unlikely that any two agents have the same preferences.

Having the student agent preferences in place, we considered the three different cases mentioned in section 4.1 to reflect the reality of different types of courses in traditional UK universities. A number of different cases were selected to ensure that it is not just the setup of a particular set of variables that enable the experiment to work and to show that the system works in general and not only for a specific case.

Here, a case means a situation where there are fixed *number of courses* and *number of students* while the *number of running course* is changeable. Because the aim of this experiment is to compare a single strategy with the optimal, all students take a single strategy. This means that there is one simulation for each value of running courses in each case (a total of ten simulations). Each simulation runs 50 times with different student preferences and the results shown are the average results over these runs. Figure 5-1 shows the simplified sequence for conducting the experiment with different settings.

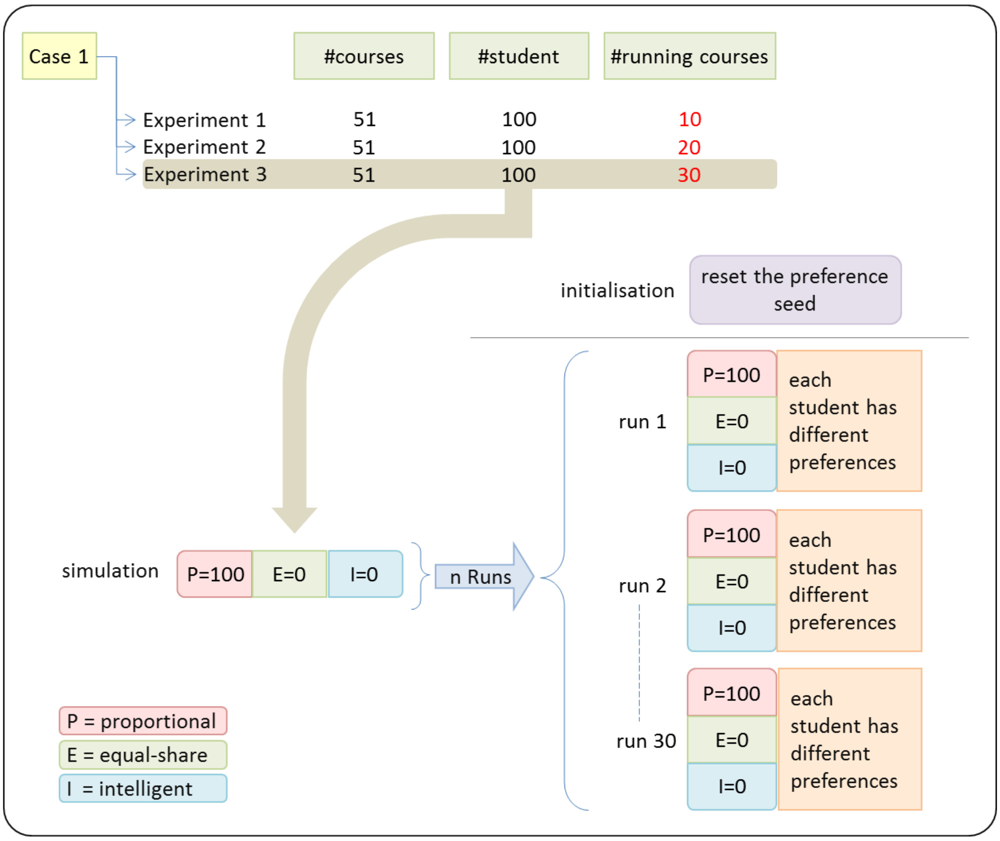


Figure 5-1: Sequence for conducting a case with different settings

In order to run the experiment, *initial points (IP)* is set to 100. The *fraction of point to invest* (fb) variable set to 50% of the total point. We set β value to 5 since it was shown in initial tests to perform well. This explained when talking about the systems strategies (see Section 4.5).

5.1.2 Result and Analysis

We now proceed to discussing the results. Figure 5-2,Figure 5-3 and Figure 5-4 show the results for cases 1, 2 and 3 respectively. Here, the y-axis shows the percentage of student satisfaction. This is calculated by the total satisfaction of the running courses, as a percentage of the total satisfaction if all the courses were running. Furthermore, on the x-axis we vary the total number of running courses while keeping the other parameters in the cases fixed. The graphs show the differences in the satisfaction of the agents using different strategies (when all agents use the strategy) and also comparing this with the satisfaction level seen in the optimal solution. These graphs do not include error bars because the confidence interval s is very small, making it difficult to put the error bars on them.

These results show that the outcome of the proportional strategy is almost identical to the optimal strategy although there is a small difference, t is not statistically significant. The intelligent strategy does slightly less well but is still very close to optimal.

Table 5-1and Table 5-2 display a statistical data to test whether there is a statistically significant difference in mean between intelligent and proportional strategies and the optimal. Statistically, at 95% level of confidence we fail to reject the null hypothesis that there is no difference in mean between both intelligent strategy and also the mean of proportional strategy and optimal because p value > 0.05, and also according to the t-test , it is less than the 1.98 for both of them. We consider case 1 with 20 running courses. Others are the same so to avoid repetition they are put in appendix A).

On the other hand, we see that the equal share strategy does worse because it distributes points among the courses equally regardless of the student preferences. Statistically, according to data in Table 5-3 we reject the null hypothesis, which there is no difference in mean between equal-share strategy and optimal because p value > 0.05, and we accept alternative hypothesis that there is a significant difference at 95% level of confidence.

This suggests that a decentralised solution using voting results in high-quality solutions that are close to optimal as long as the student agents take a sensible strategy. Linking this to the first question, it could be said that a voting-based agent system can replace the centralised authority by providing an outcome that is close to the optimal strategy.

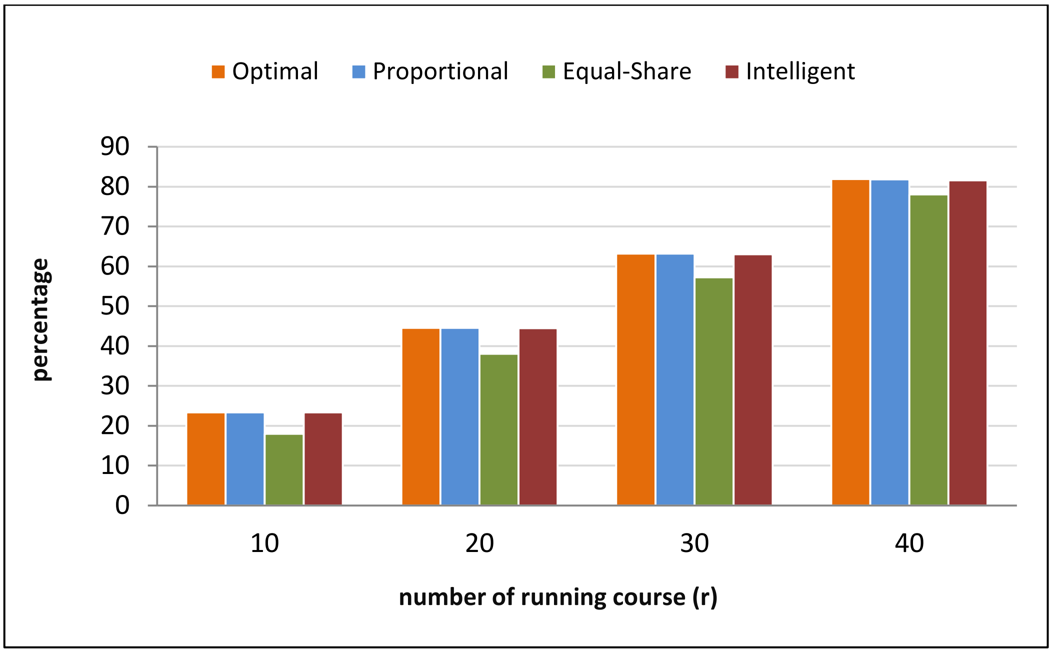


Figure 5-2: Comparison the three single strategies to the optimal solution for case 1

Table 5-1: Statistical values of comparison between intelligent strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	44.582	0.263	0.067
Intelligent	44.489	0.260	0.068

t = 1.778 df =98 p =0784

Table 5-2: Statistical values of comparison between proportional strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	44.582	0.263	0.069
Proportional	44.574	0.0.264	0.070

t = 0.152 df = 98 p = 0.879

Table 5-3: Statistical values of comparison between equal-share strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	44.582	0.263	0.069
Equal-share	38.054	0.0.86	0.149

t = 98.826 df = 98 p = 5.85×10^{-100}

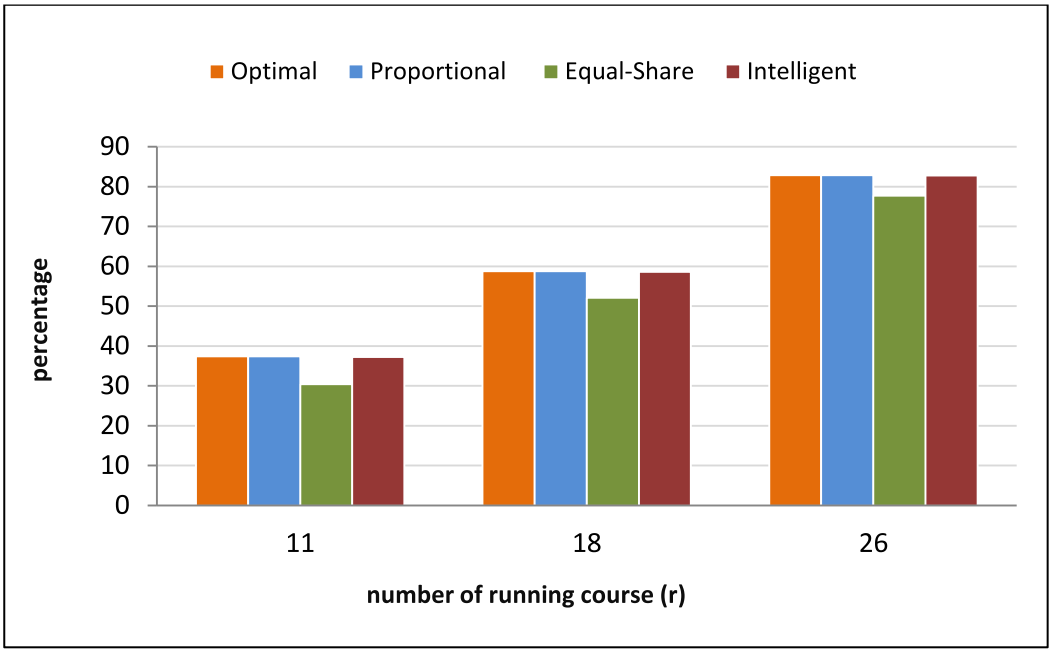


Figure 5-3: Comparison the three single strategies to the optimal solution for case 2

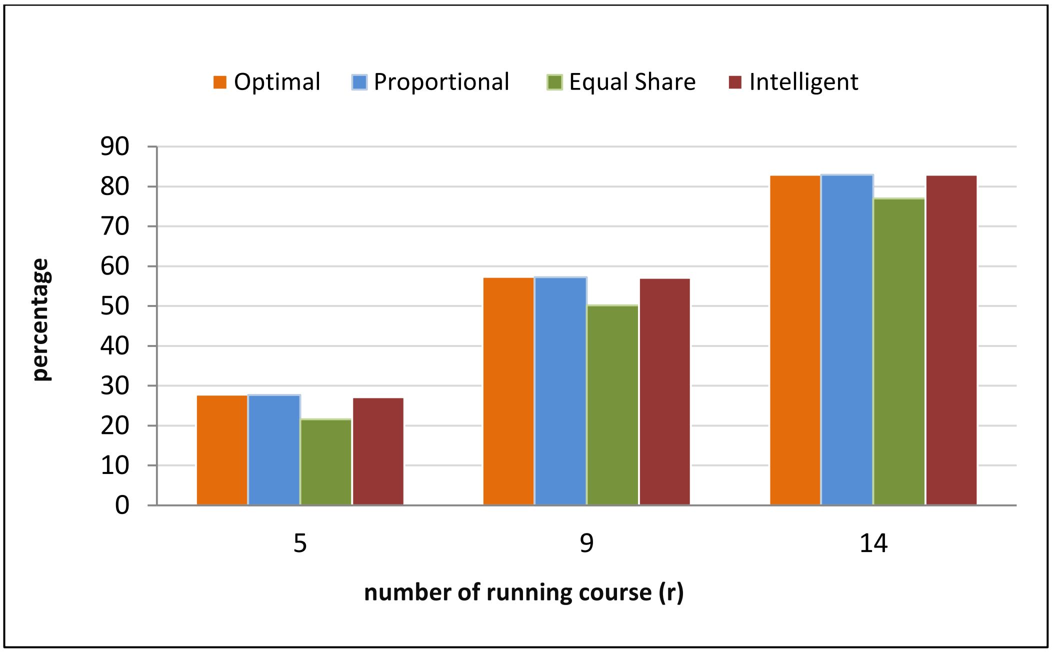


Figure 5-4: Comparison the three single strategies to the optimal solution for case 3

Interestingly, in this setting the intelligent strategy performs slightly worse than the simpler proportional strategy. This can be expected because proportional strategy is similar to voting for "true" preferences, i.e. it does not try to outsmart the system. However, it may be possible for a group of students to outsmart the simple proportional strategy and increase their satisfaction at the expense of those students using the proportional strategy. To analyse this situation and also to investigate the performance of a different strategies that student agent can use to allocate point to courses, in the next section we consider a setting where students use different strategies.

5.2 (PART 2) Combination of Voting Strategies:

In this set of experiments a situation was compared where a number of student agents use one strategy and the remainder use another strategy. The aim here was inferred from the previous experiment, which is to explore how a student agent taking a different strategy would perform and what the impact on the outcome is where different strategies run together.

5.2.1 Methodology

In this set of experiments, we consider a case where a group of students are allocated to one strategy, while the rest are allocated to another strategy. For example, if the total number is 60 student agents, 40 student agents could use the proportional strategy and the remaining 20 use the intelligent strategy. This combination of strategies can reflect the potential of voting approach when integrated with autonomous agent, it allows different student agent to use different strategy. It also allows them to express their preferences and vote for courses to reach a decision that satisfies their goals.

In order to conduct this experiment, the setting of student agent preferences should be considered first. From the individual preferences explained in the previous experiment, we found that, when the student population is large and when the preferences are initialised completely randomly, voting has little effect because it does not really matter which courses are selected: for any subset of courses, there are many students who have a high satisfaction. Consequently, even the equal share strategy performed at a level close to optimal. In addition, in practice, preferences are not independent but there are groups of students with similar interests. To address these issues, a bias was introduced in the preferences. The agents allocated to each strategy were likely to have the same preferences. This means that the intelligent agents, the proportional agents and the equal share agents each fought the other two sets of agents

To create the bias, for each student and each course we start by randomly generating preferences from a uniform distribution between 0 and 10 as explained in section 5.1.1. Then, for a subset of students we multiply the preferences of the subset of courses by a factor of 2. Then, we limit all the preferences to being no more than 10. The result is that, because of the limit, this subset of students all favours a particular subset of courses. At the same time, their preferences are not the same.

After modelling the student preferences, we repeated the same process in terms of the three cases mentioned in the previous experiment. The change here is that there are many simulations and the number of simulations is determined according to how the students are split between strategies. Accordingly, each case will run with fixed parameters, including number of courses, number of running courses, and number of total students, while the variable here is the number of student agents employing each strategy. Regarding the number of running course, we chose one setting for each case in the middle to represent the other. However, the results are very similar for the other settings in Table 4.1, and there are no qualitative differences. The number of running courses was set to: 30 for case 1, 18 for case 2 and 9 for case 3. To discuss the result, we showed the result for case 1 and the results for other cases are very similar and are not shown in order to avoid repetition (see Appendix A).

Each simulation was run 50 times with different student preferences. Thus, the results shown are the average results over these runs. Figure 5-5 shows the sequence for running the experiment with different numbers of simulations.

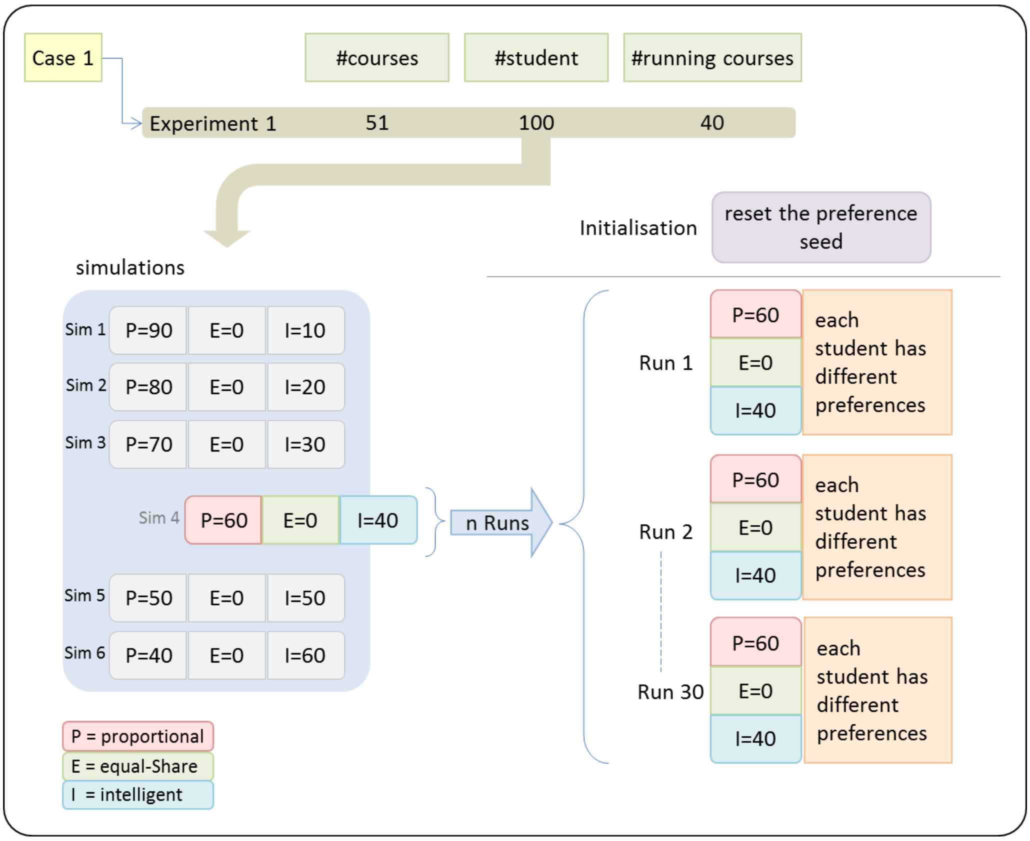


Figure 5-5: Sequence for conducting a case with different settings

5.2.2 *Result and Analysis*

We now proceed to discussing the results. In the results that follow, the y-axis shows the percentage of satisfaction for each group of agents using a particular strategy. Furthermore, on the x-axis we vary the proportion of students using a particular strategy. For example, in Figure 5-6, 90-10 means that 90 students use the proportional strategy, and 10 students use the equal share strategy.

The results in Figure 5-6 Figure 5-7 show that the intelligent and proportional strategies are both significantly more effective than the equal share, irrespective of the proportion of students that use this strategy. On average, the improvement is around 9% compared to the equal share strategy. (see Appendix A)

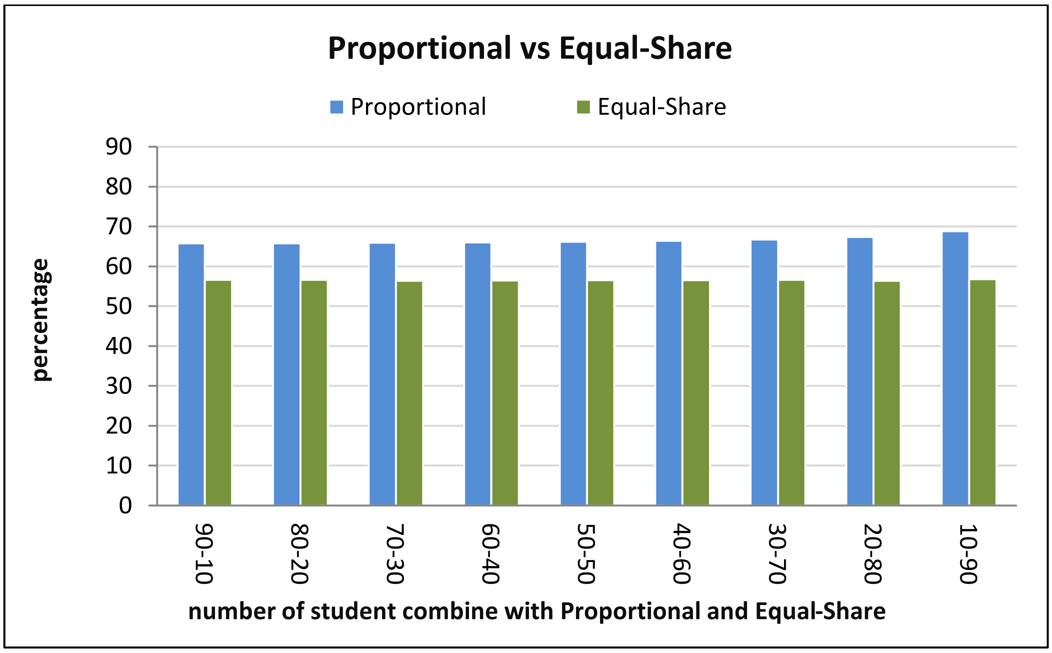


Figure 5-6: Case 1: Proportional vs. Equal Share

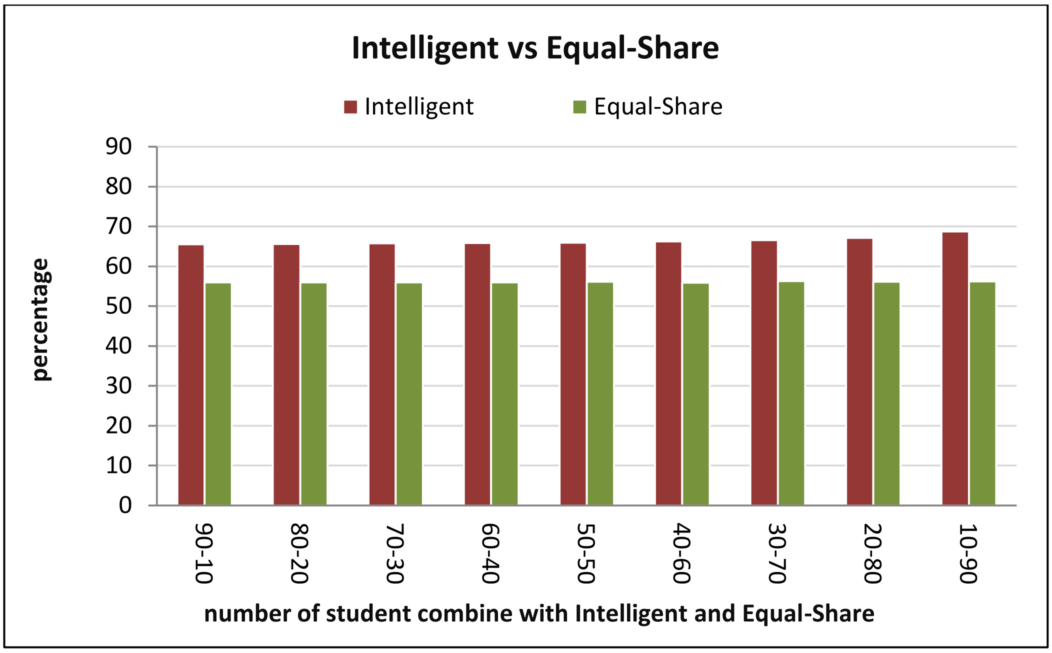


Figure 5-7: Case 1: Intelligent vs. Equal Share

Figure 5-8, Figure 5-9 and Figure 5-10 show the results with both the intelligent strategy and the proportional strategy for the three different cases. The results show that, as the number of students allocated to a particular strategy increases, the student satisfaction for these students also increases. However, this is mainly because of the bias that has been introduced; since students with the same strategy have similar preferences, when more students have these preferences they have greater voting power since they act as a group.

Comparing the intelligent and proportional strategies it can be seen that there is not much difference between them. Although in some cases, as in Figure 5-10, the intelligent strategy slightly outperforms the proportional strategy (given that the same number of students are using that strategy), in the other two cases the proportional strategy outperforms the intelligent strategy. This suggests that the system cannot be easily exploited by an intelligent strategy. We have also tried to vary the parameters of the intelligent strategy (such as the beta parameter), but again the results do not change significantly.

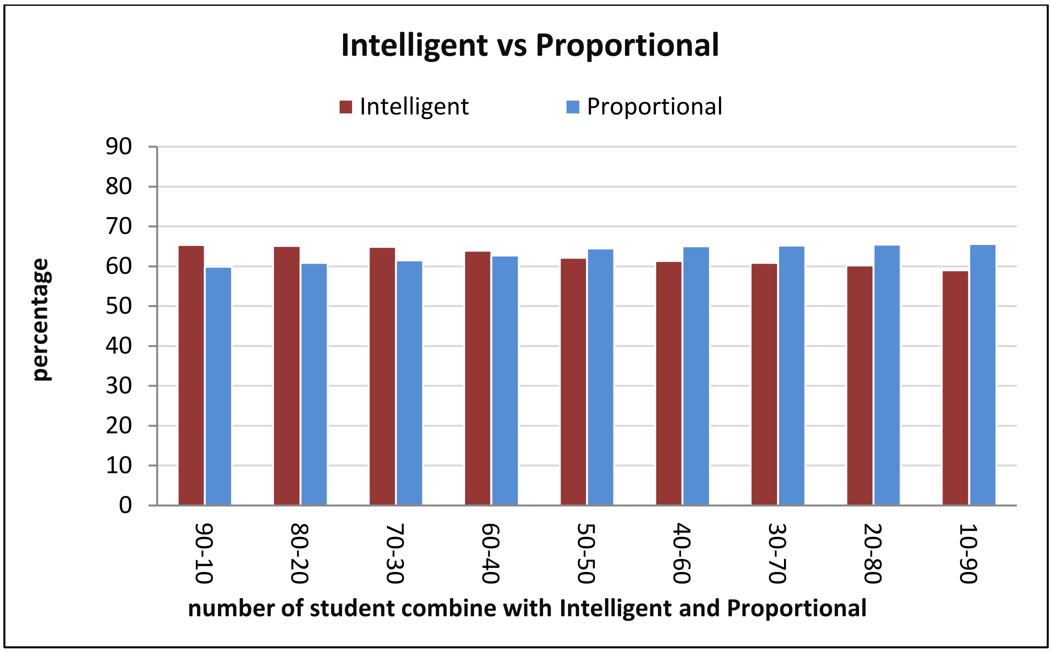


Figure 5-8: Case 1: Intelligent vs. Proportional

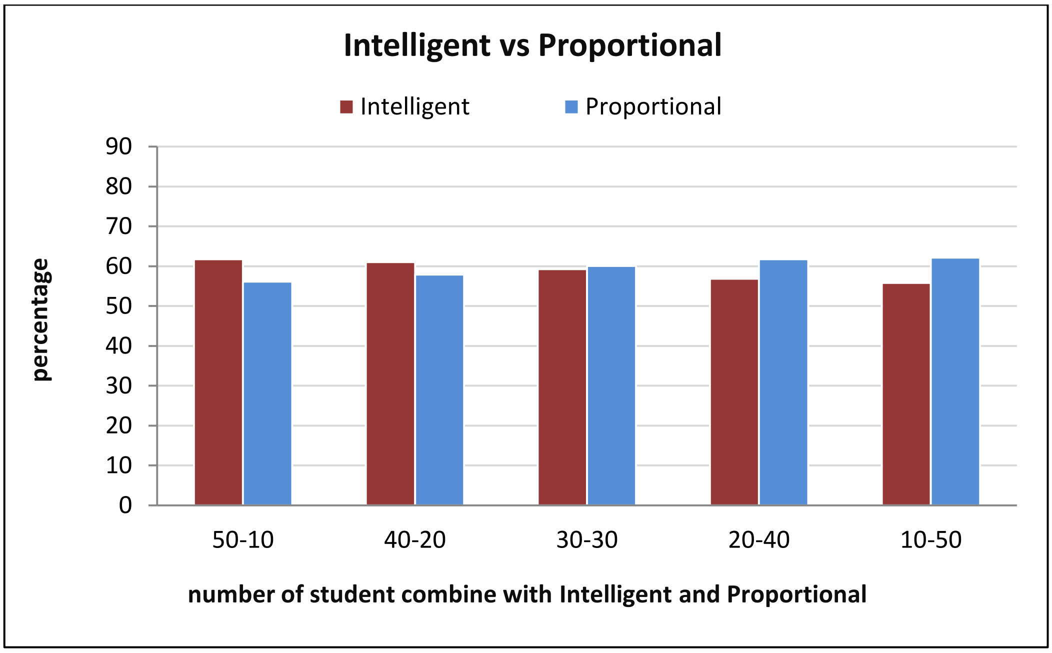


Figure 5-9: Case 2: Intelligent vs. Proportional

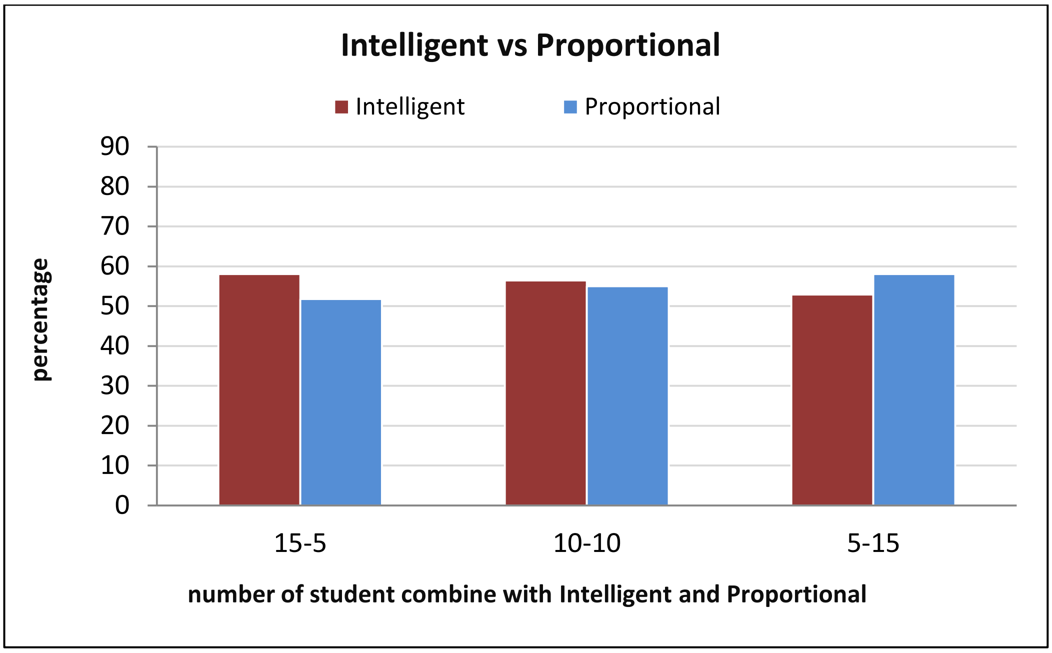


Figure 5-10: Case 3: Intelligent vs. Proportional

5.3 Discussion

We have explored how multiagent systems could be used for e-learning and can solve problems in a decentralised way where an autonomous software agent votes on a student's behalf according to the student's preferences. We have focused on an agent's ability to act autonomously in a system and to communicate with one another to reach a collective decision whilst trying to maximise the outcome.

After completing the two experiments, they show that when agents are used in this way, three points can be highlighted:

- **Decentralisation**

Helping learners build the decentralised navigation way that fulfils their learning goals can release them from unnecessary traditional way using learning activities and help them to maintain their privacy. It also supports collaborative learning among different individuals, which can lead to multiple ways to expand the learning potential of every student. We argued that agent systems could provide decentralised solutions to a number of key e-learning problems. This experiment shows that not only is this possible, but that if students choose sensible strategies the results tend towards an optimal solution, which is calculated as the result of a centralised approach. This answers question one.

- **Personalisation**

In a multiagent system, autonomous agents allow for unprecedented levels of personalisation. From an educational perspective, not only can students have preferences about any given learning scenario, but by selecting different strategies they can change the way in which their agents negotiate. For example, in the course selection, scenario students have different preferences and in the experiment can choose different strategies for voting based on those preferences. However, it would also be possible to introduce other student agents that had a completely different basis for choosing courses (for example, based on the choices of their friends or on the requirements of some qualification framework).

Multiagent systems provide the necessary level of abstraction for the tailoring of every aspect of the process, including the basis for making choices (e.g. preferences or some other criteria), the individual's personal data (e.g. the preferences themselves), and the algorithm that uses that data to negotiate (e.g. how to vote according to those preferences). This fulfils the outcome sought in question two.

However, we also believe that our work highlights potential concerns:

- **Fairness**

In situations of personalisation and decentralisation it is very difficult to guarantee that all students will have the same potential for satisfaction. This is because, although the agents are handled equally, the system relies on the agents themselves making sensible choices and selections. Power and responsibility are transferred from a central authority (the institution) to individual agents (the students). If an agent makes irrational choices, or chooses a bad strategy, then their student will be disadvantaged when compared to others. We demonstrated this by showing how a foolish equal share strategy penalised students who acted in that way. However, we also showed how a well-designed protocol makes it difficult for a more intelligent (or intentionally subversive) strategy to gain advantage over a sensible strategy. Because of the importance of this issue, we discuss it in more detail in a separate chapter (chapter 7).

5.4 Summary

This chapter has presented a multiagent simulation that uses a suitable voting protocol to support course selection. Using this simulation, the results show that a decentralised agent approach not only works but when using reasonable agent strategies it is very close to an optimal centralised solution. Furthermore it can be seen that how different agent strategies compare to one another, showing that with this particular protocol intelligent strategy was unable to exploit other, more naïve voters. This is encouraging for the e-learning domain, where institutions are often required to be fair.

The next chapter considers complex preferences. Whereas the student agents in this chapter have simple additive preferences, in practice the preferences are often interdependent. That is, the preferences in relation to one course depend on whether or not another course is running. In such setting, decentralised approaches even more important since such interdependence are difficult to handle in centralised way.

CHAPTER 6.

AGENT SIMULATION FOR E-LEARNING SYSTEM INVOLVING COMPLEX PREFERENCES

In the previous chapter, student preferences were modelled using a simple model where the preferences for each course are independent from others (meaning that if a course was cancelled, this had no effect on the remaining preferences). However, it is not realistic to assume that all courses are independent. In practice, voters typically like to express dependencies between courses, such as "I would like to study Course A only if I can also study Course B" (complementary preferences) or "I would like to study Course C or Course D, but not both" (substitutable preferences).

In this chapter we extend the voting system from chapter four to these more complex preferences. In particular, the strategies were developed to take into account the interdependencies that exist when casting votes for the available courses. The objective in doing so is to demonstrate the potential of using multiagent systems and voting in settings where students have complex preferences and use a range of voting strategies. In this context, the complexity is increased so such voting protocol is needed because it became difficult to calculate the central solution. We investigate how students using an intelligent strategy try to predict the courses which will be running, using this to estimate the expected student satisfaction and thus achieve a higher average satisfaction. This is done by testing ***H2:***

Within system with complex preferences, an individual student that uses an intelligent predictive strategy will on average achieve a higher satisfaction than those taking a naïve or random strategy.

The main questions this experiment tries to answer are:

Q1: If the student agent has complex preferences and uses an intelligent strategy, will he or she achieve a higher level of satisfaction?

Q2: What is the impact of using the other voting strategies on the overall student satisfaction when students' preferences become sophisticated by including AND and OR rules?

This chapter starts by explaining in detail how we conducted the experiment, including the methodology, analyses and results. A discussion and concluding remarks end the chapter.

6.1 Complex Student Preferences

In the previous chapter, student preferences were modelled using a simple model where the preferences for each course are independent from others (meaning that if a course was cancelled it had no effect on the remaining preferences). However, in reality it is not realistic to assume that all courses are independent. In practice, a student typically would like to express dependencies between courses such as "I would like to study course A only if I can study course B" (so-called complementary preferences) or "I would like to study course C or course D, but not both" (so-called substitutable preferences).

The modelling here is built on the previous model but adds relationships between courses and makes the case more complex. Specifically, in this model the utility of a course may depend of which other courses are running. To reflect this, we modelled two types of relationship: *complementary* (AND) and *substitute* (OR). Two courses, A and B, are complementary when the student is only interested in choosing course A if course B is running, and vice versa. Conversely, courses A and B are substitutes if a student is interested in course A or course B, but not both. We assume that each student has a (possible) different set of *rules*, where each represents either an AND or an OR relationship between a pair of courses.

Given this, the utility function of a student is modelled as follows. Let $C = \{C_1, C_2, \dots, C_m\}$ denote the set of courses, and $\vec{v} = \{v_1, v_{2,}, \dots., v_m\}$ denote the individual utilities (in case there are no rules) for these courses, where m is the total number of available courses to choose from. Furthermore, let R_{OR} denote the set of OR rules, which specifies a set of pairs of courses(C_i, C_j), and similarly R_{AND} denotes the set of AND rules. To avoid conflicts, it is assumed that each course is only part of one rule. Therefore, the same course cannot appear both in an OR rule and an AND rule. Then, the utility for a set of running courses is calculated as follows:

- For any running course, C_i, that does not appear in a rule, the utility is simply the sum of the individual utilities v_i of those courses.
- For any pair of courses $(C_i, C_j) \in R_{OR}$, if both courses are running, the utility for the pair is $\max(v_i, v_j)$. If only one of them is running, then the utility is equal to the individual utility for that course.
- For any pair of courses $(C_i, C_j) \in R_{AND}$, if both courses are running, the utility for the pair is $v_i + v_j$. Otherwise, the utility is zero.

The total utility is then the sum of the individual courses without rules, and the pairs with rules. Using this utility, we then calculate the *student satisfaction* by taking the utility as a percentage of the utility the student would achieve when all the courses would be running.

To clarify, consider this example. Here, there are six courses, and two student agents. 2 students will make it more clear how to calculate the student satisfaction with complex preferences. The preferences of SA_1 are as follows: $\vec{v} = \{7,3,4,2,1,8\}$ and the rules are: C_1 OR C_4, C_3 OR C_5, C_2 AND C_6. The preferences of a SA_2 are: $\vec{v} = \{3,8,7,3,5,1\}$ the rules are as follows: C_1 AND C_5. Let's consider SA_1 first. The total utility if all courses are running is calculated as follows. The relationship between C_1 and C_4 is OR, so we take the higher value which is $v_1 = 7$. The relationship between C_3 and C_5 is also OR so we will take the higher one which is $v_3 = 4$. Finally, and v_6 has an AND relationship with v_2 so we will take the sum of them 3+ 8 = 11. The total utility is therefore 7+4+11=22. Now, the university decides to cancel C_1. In this case, the actual utility will be 2+4+11=17. The *student satisfaction* is then given as 17/22*100% ≈ 77%. (Figure 6-1 illustrates the process.)

With regards to SA_2, the total utility, if all courses are running, is calculated as follows. Where there is only a relationship between C_1 and C_5 which is AND, so we take the sum of them 3+5 = 8. And we will calculate the rest of the preferences as normal summation bases 8+7+3+1 = 19. The total utility is therefore 8+19 = 27. As mentioned earlier, the university decides to cancel C_1. In this case, the actual utility will be 8+7+3+1 = 19. C_5 is eliminated as it has an AND relationship with C_1. The student satisfaction for this student is then given as 19/27*100% ≈ 70%.

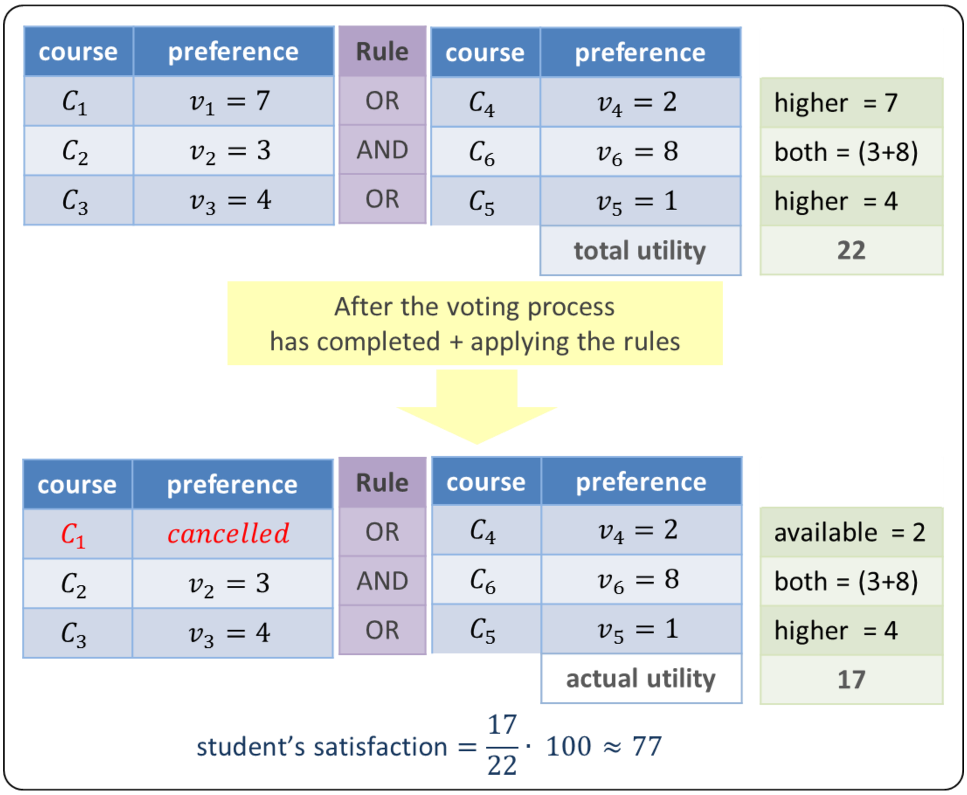


Figure 6-1: Calculate student's satisfaction with complex preferences

6.2 Methodology

This experiment is aimed to at demonstrating if the voting approach might prove to be an advantage for those students who use more intelligent agents, where student preferences are complex. This means using AND and OR rules. In doing so, the setting of student agent preferences should be modified in such a way as to fulfil the objective of the experiments. For each student and each course, we start by randomly generating preferences from a uniform distribution as described before in section 5.2.1. Subsequently, for every time the simulation is run, AND and OR relationships are generated. This is done as follows. First, we generate all possible AND and OR relationships. Then, we select a subset of these rules from which the students can select. We do this to increase the likelihood that groups of students identify similar relationships between courses. This is true in practice, since often students do have the same relationships between courses, even if they value individual courses differently. For example when there are two similar programming courses students will often share the desire to take one of them but not both. Now, the degree of similarity can be varied by a parameter which sets the size of the subset as a percentage of the total number of possible rules.

Figure 6-2 shows an example of this generation, for m = 6 courses available in total. In this case, the number of relationships that can be generated here is $n*(n\text{-}1) = 6*5 \quad = 30$ relationships. According to the percentage given for this experiment (50%), we have 15 rules left.

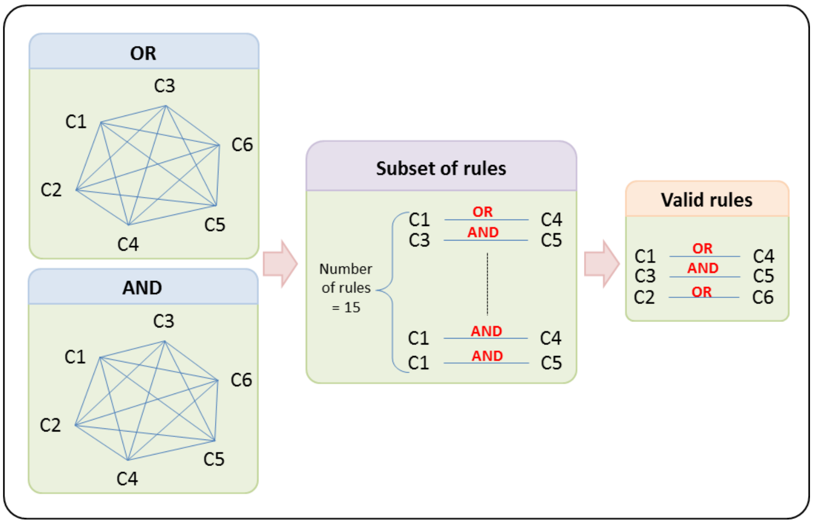


Figure 6-2: Generation of the course model

Following this, a number of simple constrains are applied to the generation of the student preferences: no more than one relationship involving any given course to avoid cycles and only one type of relationship (OR or AND) between the same courses to avoid conflicts. In this case, we end up with three rules after applying the constraints: (C_1 OR C_4), (C_3 AND C_5), (C_2 OR C_6).

After that, we consider the same three cases explained in section 4.1 and also used in chapter 5. The number of running courses is confined to 40, 11, and 9 for cases 1, 2, and 3 respectively (Table 6-1). We repeat the same procedure that used to conduct the previous experiment in terms of splitting the student between strategies and also in terms of the way experiments and simulations ware conducted including the variables[1].

Table 6-1: Different settings for the cases

case	*#courses (m)*	*#running courses (r)*	*#student (n)*
1	51	30	100
2	33	11	60
3	18	9	20

6.3 Applying Rules to Voting Strategies

One important issue in this experiment is how the three different strategies work at the stage when casting the votes over the courses in the presence of rules between courses. The voting strategies have been described in section 4.5, but they need to be adapted to deal with complex preferences, and taking into account the rules.

Proportional: This will take the rules between courses into account when distributing the points and distribute them proportionally to the student's preferences for each course. The strategy takes the rules into account in the following way in the event of complementary preferences between courses (i.e. AND rules). Suppose a cancelled course has an AND relationship with another course (e.g. C_1 AND C_2 , and C_1 is cancelled). Then, the SA will exclude the dependent course (C_2) from the points distribution process and try to benefit from these point by spending them elsewhere. The strategy does not take into account any OR relationships, since it cannot know which courses will eventually be running.

Equal share: This does not take rules into account, providing the lower bound on the performance of the system. It enables SA to gives all courses an equal number of votes, regardless of the student's preference.

[1] They are: *initial points (IP)*, *fraction of point to invest*(fb) and β.

Intelligent: This strategy differs from the one in previous experiments in that it will consider the rules between courses when calculating probability. Specifically , for each pair of courses the probability will be calculated differently for an OR relationship and an AND relationship. Probability is used to calculate the expected utility. This works as follows. Suppose we have an AND relationship between C_1 and C_2. In order to calculate the expected utility, we need to know the probability that both of these courses will run, i.e.

$$P_{RUN}(C_1|\vec{b}) \cap P_{RUN}(C_2|\vec{b}),$$

where

$$P_{RUN}(C_i|\vec{b}) = 1 - P_{CANCEL}(C_i|\vec{b}).$$

By assuming there is independence in regard to the probabilities for different courses, we can do this by simply taking the product. Now, in the case of the OR relationship between C_1 and C_2, we need to consider also the possibility that only one of them will run, meaning the other is cancelled. This gives the following expected utility for individual or pairs of courses, depending on the rules between courses:

No rule: $EU = P_{RUN}\left(C_i|\vec{b}\right) * v_i$

C_i AND C_j: $EU = P_{RUN}\left(C_i|\vec{b}\right) * P_{RUN}\left(C_j|\vec{b}\right) * (v_i + v_j)$

C_i OR C_j: $EU = P_{RUN}\left(C_i|\vec{b}\right) * P_{RUN}\left(C_j|\vec{b}\right) * MAX(v_i, v_j) +$
$P_{RUN}\left(C_i|\vec{b}\right) * P_{CANCEL}\left(C_j|\vec{b}\right) * v_i + P_{CANCEL}\left(C_i|\vec{b}\right) * P_{RUN}\left(C_{j|\vec{b}}\right) * v_j$

The total expected utility is therefore the sum of the expected utility for all courses without rules, and the expected utility of all course pairs with rules. The rules continue to work as described in order to ascertain the allocation that maximises this expected utility (see section 4.5 for more details).

6.4 Result and Analysis

In the results that follow, the y-axis shows the student satisfaction for each group of agents using a particular strategy, as well as the overall average of student satisfaction. Furthermore, on the x-axis we vary the proportion of students using a particular strategy for a different percentage of the applied rules.

This is repeated for three different setting: NO rules, 50% rules and 100% rules. The error bars show the 95% confidence intervals.

The results in Figure 6-3 and Figure 6-4 show that the intelligent and proportional strategies are both clearly more effective than the equal share strategy. Specifically, it can be seen that, as the number of rules increases, the better the intelligent and proportional strategies perform relative to equal share. On average, the improvement is around 4%, 6%, and 9%, for NO rules, 50% rules, and 100% rules respectively. Furthermore, the average satisfaction of all students also increases, which means that the allocation is more efficient when students use a more intelligent voting approach. The results for other cases are very similar and are not shown to avoid repetition (for more details, see Appendix B).

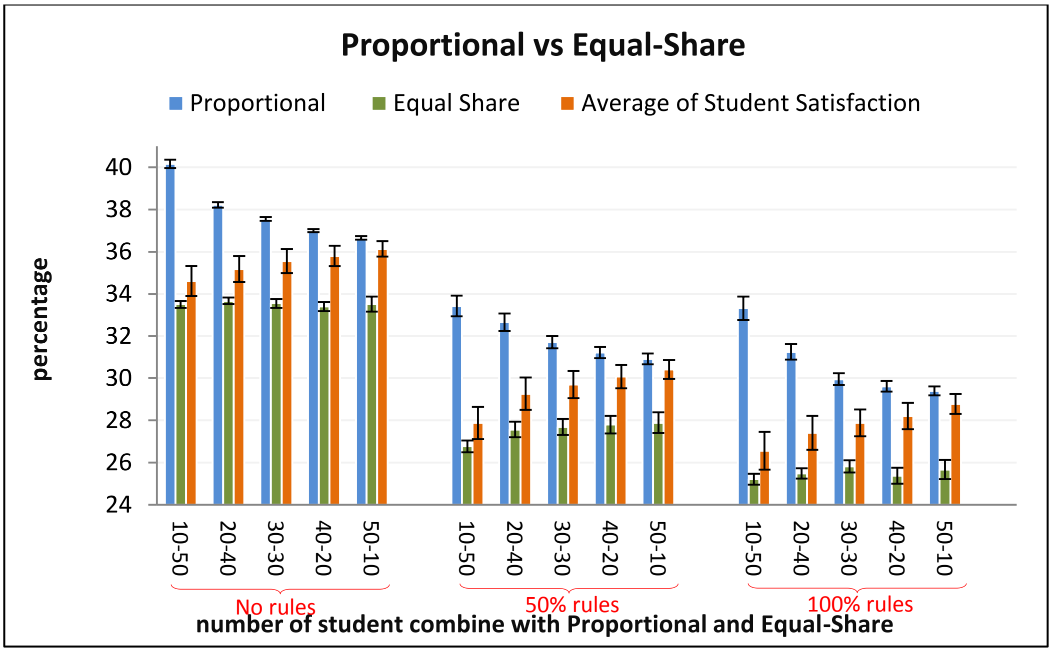


Figure 6-3: Case 2: Proportional vs. Equal Share

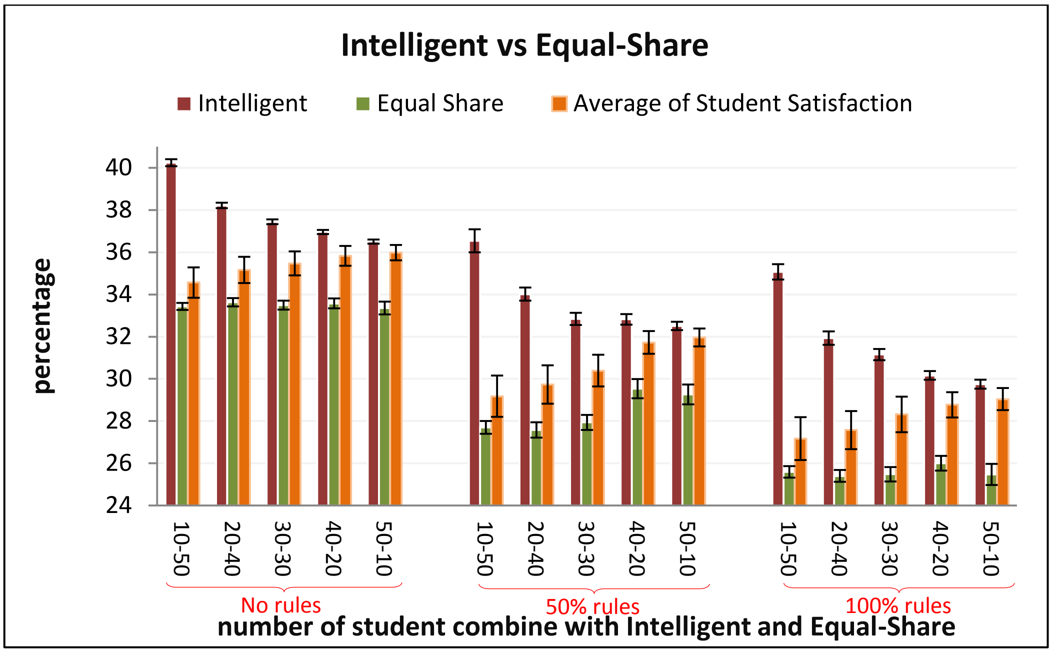


Figure 6-4: Case 2: Intelligent vs. Equal Share

Figure 6-5, Figure 6-6 and Figure 6-7 compare the results using the intelligent strategy and the proportional strategy for the three different cases. Figure 6-5 shows that, at first glance, the performance of the intelligent strategy increases compared to the proportional strategy, as the students apply more rules. However, in most cases this result is not statistically significant. We choose the combination of 30 students for each strategy with 50% rules case and conduct some statistical test. According to the statistics shown in Table 6-2, the hypothesis that there is no difference in mean intelligent strategy and proportional strategy fail to be rejected because p value > 0.05 and t-test value < 1.98 at a 95% level of confidence.

The reasons the results of the two strategies are similar are as follows. First, the number of courses that the students vote over is large. This means that the range of student choice is wide and students have a wide range of preferences. Second, the number of students voting is also large. This means that each individual student has very little voting power. To analyse this, we will consider a setting where the number of students and courses are small.

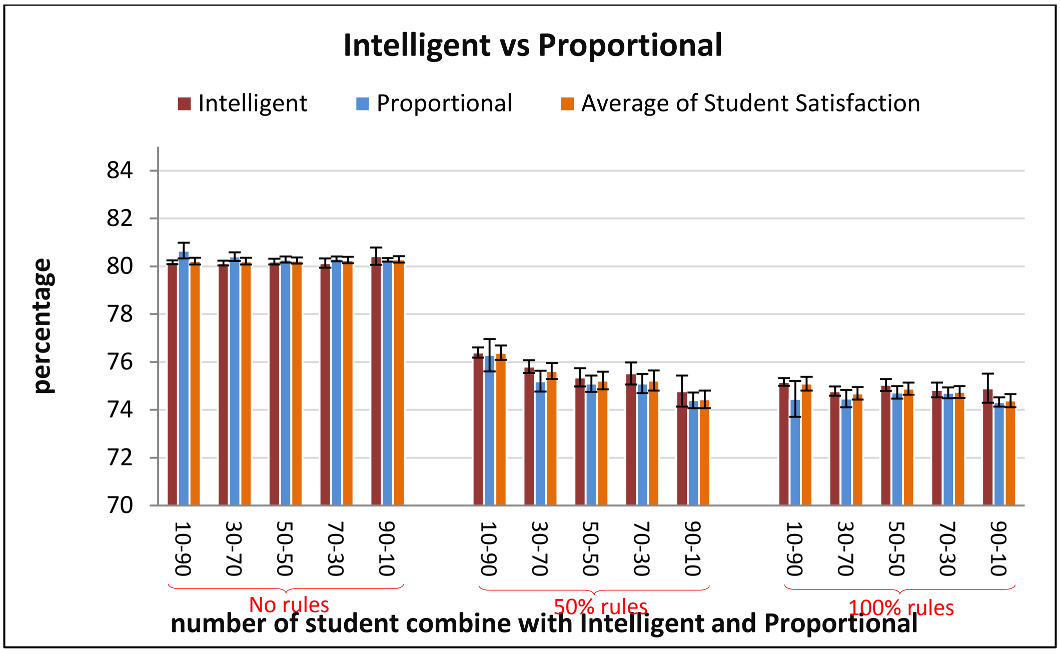


Figure 6-5: Case 1: Proportional vs. Intelligent

Table 6-2: Statistical values of comparing intelligent and proportional strategies in case 1

strategy	*mean*	*SD*	*variance*
Intelligent	75.356	1.378	1.899
proportional	75.092	1.224	1.498

t = 1.013 df =98 p =0.3136

Figure 6-6 and Figure 6-7 show that, when there are relatively few courses and students, there are clear differences in the performance of the intelligent and proportional strategies. This difference between them becomes bigger as long as the number of student and course is smaller, as seen in Figure 6-7. The intelligent strategy significantly outperforms the proportional strategy when a student applies the rules and this superiority is increased as more rules are applied. Statistically, according to data displayed in Table 6-3 and Table 6-4 we reject the null hypothesis and we accept alternative one at 95% level of confidence. Therefore this is a significant difference in means between intelligent and proportional strategies because p value < 0.05 also the value of t-test > 1.98.

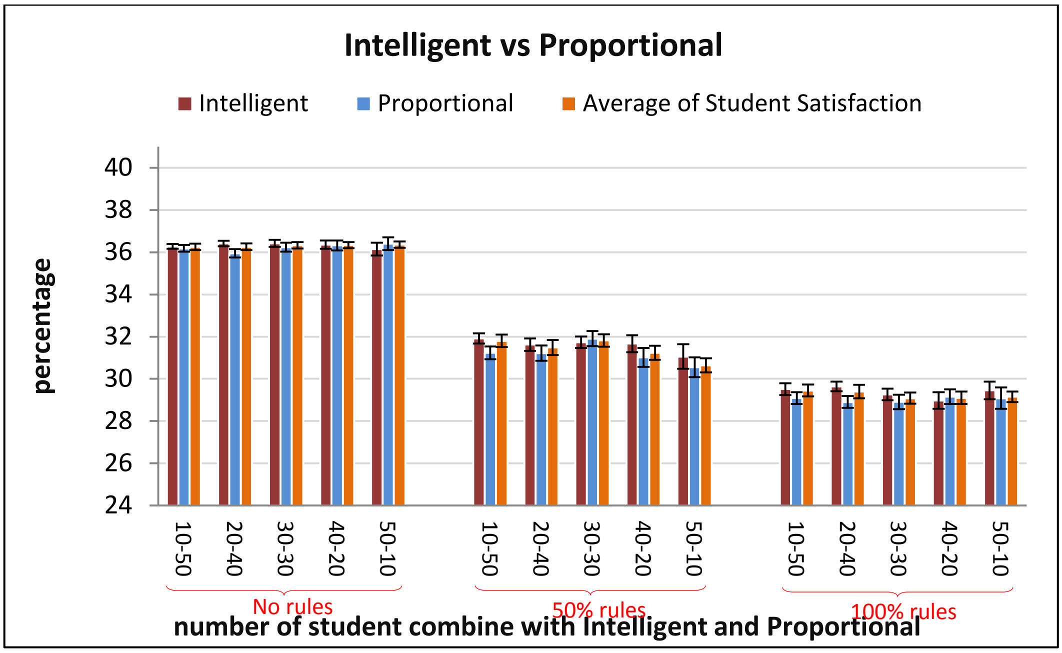


Figure 6-6: Case 2: Intelligent vs. Proportional

Table 6-3: Statistical values of comparing intelligent and proportional strategies in case 2

strategy	*mean*	*SD*	*variance*
Intelligent	31.991	0.918	0.843
Proportional	31.601	0.990	0.980

t = 2.043 df =98 p =0.0431

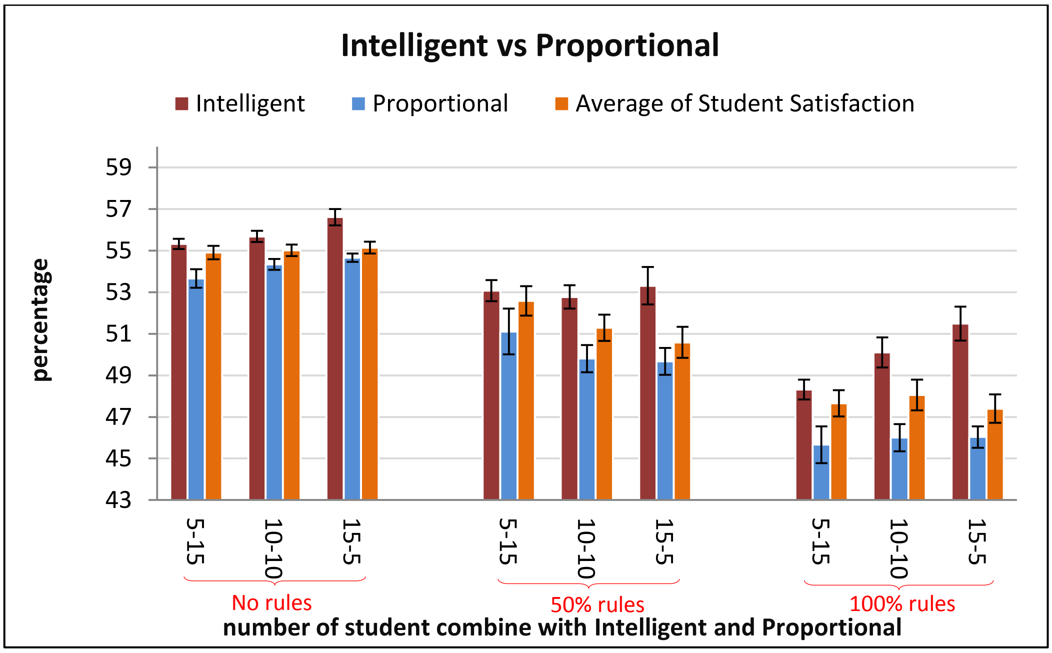


Figure 6-7: Case 3: Proportional vs. Intelligent

Table 6-4: Statistical values of comparing intelligent and proportional strategies in case 3

strategy	*mean*	*SD*	*variance*
Intelligent	52.771	2.034	4.137
Proportional	49.803	2.352	5.532
t = 6.749	df =58	p = 1.05×10^{-8}	

6.5 Discussion

After completing the experiments, we can see that when students have complex preferences various points can be highlighted. The hypothesis that the intelligent strategy's improvement is greater than the improvement in the proportional strategy is true in almost all cases where rules were applied, according to statistical data this suggests that when students have fewer choices and therefore there is less differentiation between the students, and when the number of students is not too large, the intelligent strategy performs better than the proportional one. Note also that, as the proportion of students using the intelligent strategy increases, the student satisfaction of all students either stays the same or increases. Therefore, using a more intelligent approach does not harm the system as a whole. This answers question one

In practice, there are interdependencies between courses for students. They want to study some courses together at the same time, or they may want to study one course but not another. In this context, this experiment shows that, when more dependencies between courses are identified by applying more rules, the better the intelligent and proportional strategies perform. On average, the improvement is around 4%, 6%, and 9%, for NO rules, 50% rules, and 100% rules respectively. This also shows that the performances of intelligent and proportional strategies are better than the equal share in all cases where rules are applied. This answers question two.

6.6 Summary

This chapter takes the same multiagent system simulation but complicates the situation by incorporating the fact that students have complex preferences and use a range of voting strategies. It has shown how the autonomous software agent votes on a student's behalf according to the kind of complementary and substitutable preferences between courses. We found that, when students have complex preferences and the number of students is not too large (such that each individual student can affect the voting outcome), the intelligent strategy performs significantly better than the proportional one. Moreover, it can be seen that, as the number of rules increases, the intelligent and proportional strategies also perform better relative to equal share.

Chapter 7. Fairness in Intelligent E-Learning Systems Using Agent Simulation

As e-learning systems become more advanced, they enable a more heterogeneous student experience. Personal choice in regard to e-learning tools is a core part of personal learning and is becoming a defining feature of informal learning (Wilson, Liber et al. 2007). Moreover, intelligent and personalised e-learning systems mean that different students can have different experiences as tasks, and resources being intelligently modified according to their preferences (Shute and Paotica 1994). This means that intelligent e-learning systems must face the issue of fairness, where fairness is broadly defined as ensuring for any given process that participants are treated equally and there is an appropriate balance of satisfaction with the outcome.

The previous two experiments appear to show that a distributed agent system may produce unfair results, because an agent applying an intelligent strategy is likely to perform better than those who do not. The main objective of this chapter is to investigate how intelligent agent-based e-learning systems can ensure fairness between individuals using different strategies in a system that is personalised to them. This is done by testing ***H3*** and ***H4:***

- *As the proportion of individuals agents utilising differently performing strategies in the system increases, the overall fairness of the result (as defined by equity theory) decreases.*

- *When having a uniform mixture of different strategies we can adjust the protocol by exponent and weight to make the protocol fairer.*

In order to explore the issue of fairness, and look at whether different students mixing different strategies and having different preferences will receive equal treatment, the investigation is divided into two parts. The first part discusses the fairness of the protocol itself. This is examined by adding two factors: exponent and weight. The former is used to give individual student agents who rate courses highly more impact than students who rate the courses at the lower end (exaggerating the impact of strong preferences). The latter is intended to give the student who has not achieved their desire and is therefore not successful more impact on the voting process (increasing their ability to catch up).

The second part focuses on the fairness of the strategies' distribution using the current protocol, where the proportion of students using one strategy is increased and the students using another strategy is decreased.

The main questions these experiments try to answer are:

Q1: How can we make sure the protocol is as fair as could be? And does it produce a fair result?

Q2: Does the exponent affect the performance of the strategies and does it affect fairness when we have a mixed set of strategies?

Q3: Does weight decrease the difference between strategies and does it affect the fairness when we have a mixed set of strategies?

Q4: How does fairness change as the distribution of strategies changes?

This chapter first introduces a simple framework for measuring the fairness of a result by setting fairness definitions with formal metrics to describe them. Then, it describes in detail how to conduct the two parts of the experiment, including methodology, analyses and results. A discussion and concluding remarks end the chapter.

7.1 A Framework of Fairness in Intelligent E-Learning Systems

Before investigating fairness in intelligent e-learning systems, from the literature we derive a simple framework of fairness definitions that are useful for the e-learning domain, describe different aspects of fairness, and define the ways of being measured.

These definitions of fairness are used as an assessment of the outcome of a system or process. Within our proposed fairness framework, we have identified three different notions of fairness, each of which can be described statistically given a system where each participant has a satisfaction value with their personal result. These three notions have been described in detail in section 2.5. In brief, they are:

- **Utilitarian:** the best outcome that gives the greatest whole welfare to society. For our satisfaction values, this means maximising the mean satisfaction of all participants (which is equivalent to maximising the sum of the satisfactions).
- **Equalitarian:** the best outcome that gives the least difference between the individual members of society. For our satisfaction values, this means minimising the standard deviation of satisfaction values and minimising the range of satisfaction values.
- **Egalitarian:** the best outcome that gives the greatest overall welfare so that all individual members should have equal benefits. It is intended to maximise the welfare of a society's weakest member. For our satisfaction values, this would mean maximising the minimum satisfaction, although another approach would be to maximise the median satisfaction.

These three aspects of fairness in our framework allow us to assess the outcome of any system as long as we have individual satisfaction values for participants in that system. Table 7-1 shows a summary of the framework and the metrics:

Table 7-1: Aspects of Fairness and their Metrics

Aspect	***Metrics***
Utilitarian	Maximise mean
Equalitarian	Minimise standard deviation
	Minimise range
Egalitarian	Maximise minimum
	Maximise median

Given this simple framework for fairness, it is possible to examine the outcomes of an intelligent e-learning system and evaluate them for fairness. We are unable to ascertain absolute fairness, where a system is judged as fair or unfair, because it is not clear what the criteria would be for any given scenario (for example, at what point does a range become unfair in equalitarian terms?). However, we can compare systems to see which are fairer by our measures. We might expect to see fairer results in systems where participants are treated equally, but more unfair results in systems where participants can use different strategies and take different risks.

7.2 (Part1) Fairness of the protocol

In this experiment, we will test the fairness of the protocol by including factors that make the protocol fairer by rewarding people very highly or rewarding people who do not succeed at first. In particular, we will look at what happens to each group representing one strategy within a cohort of strategies, and what that means for the performance of overall strategies when mixing them together. We attempt to make the protocol fairer by applying two mechanisms: exponent and weight. For the exponent, when it is turned up, the power of the vote is amplified for strong preferences, enabling the agent to have more ability to affect the system. Conversely, if the exponent is turned down, voting power is dampened, making preferences less important overall. For the weight, as it is increased, the students who have been unsuccessful in their voting are given more power, whereas if it is turned down, we give them less power.

7.2.1 Methodology

Exponent

In all previous experiments, when the university agent intends to cancel a course, it looks at all courses, finds out which course has the lowest points, and cancels it. However, here the mechanism is slightly changed. In this experiment, for each course, the university agent applies an exponent to the point voted by each individual student and then sums these adjusted points. It then uses the summed adjusted points to determine which course to cancel. More formally, the adjusted points are given by the following equation:

$$ab_{ij} = {b_{ij}}^{\gamma},$$

where b_{ij} is the points allocated by student agent i to course j, γ is a parameter which sets the value of the exponent, and ab_{ij} is the adjusted points, and then, the following equation is used to sums the adjusted points for one course (TP_j):

$$TP_j = \sum_{i=1}^{n} ab_{ij}$$

The chart in Figure 7-1 shows the process in terms of how the cancellation of courses proceeds when applying the exponent.

The main point of using exponents is that this gives the individual student agent who highly rates courses a disproportionally greater impact than a student who rates courses at a lower scale. In other words, students with a stronger preference are much more important to the system than the others. As a result, the system responds to those with extreme views and more heavily weighs their preferences when deciding which courses to cancel. We used different exponents, with some being smaller than 1 and some larger; 0.0001, 0.1, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, and 6.0

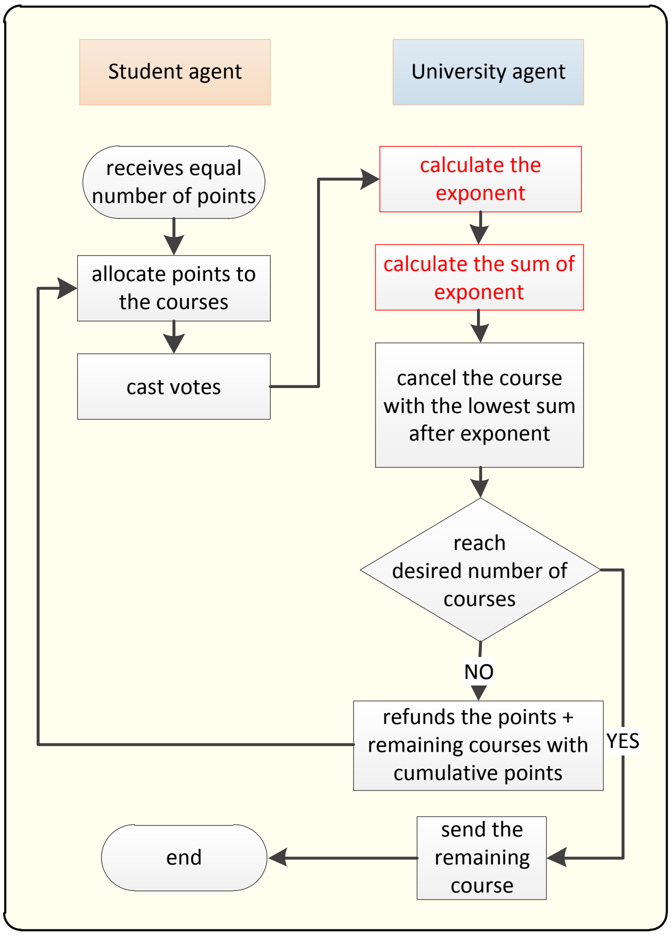


Figure 7-1: Flow chart showing how the exponent mechanism works

The experiment was run with different exponents; some are below 1 and some above. An exponent of 1 is equivalent to running the system as before. The exponent applies on points cast by a student to either dampen or magnify the value. For example, let us say in the normal situation when γ is 1, and b_{1j} is 2 and b_{2j} is 4, there will be no change. However, if γ with a value of 2 is applied then b_{1j} becomes 4 and b_{2j} becomes 16. Thus, b_{1j} now is worth twice as much (i.e. it becomes 4) and b_{2j} is worth four times more (i.e. it becomes 16). On the other hand, if γ set to 0.5 and b_{1j} and b_{2j} are 2 and 4 respectively, their votes will be valued less; 1.41 with a value of about 0.71 times and 2 with a value about 0.50 times respectively (Table 7-2).

The effect of the exponent would be:

- Setting γ equal to 1 is the same as the previous experiment in which votes are counted on a one-for-one basis.
- Setting γ greater than 1, where the difference between votes is magnified.
- Setting γ lower than 1, the difference between votes is dampened.

Table 7-2: The effect of the exponent

	Vote				
Student A	**2**	**Exponent**	0.5	1	2
		Value	1.41	2.00	4.00
		Worth	0.71 time	1 time	2 times
Student B	**4**	**Exponent**	0.5	1	2
		Value	2.00	4.00	16.00
		Worth	0.50 time	1 time	4 times

Weight

The main point from doing weighting is to give more power to the student who has not been successful. When a course has been cancelled which was highly desirable by a student, a weight mechanism lets them have more impact on future rounds of the voting process. This is achieved by multiplying the number of points returned by a constant weight. We run the experiment with different weights, with some being smaller than 1 and some larger; 0.0001, 0.1, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, and 6.0. A weight of 1 is equivalent to running the experiment as before. If the weight is above 1, a student will receive relatively more points if they had voted highly on a course which was cancelled, whereas if the weight is below 1 the returned points have less value. Note that having a value close to zero means that effectively no points are returned. We include this setting in our experiments to see the effect of returning votes in the protocol. The following formula was used to calculate the returning points.

$$wb_{ij} = b_{ij} \times \delta,$$

where b_{ij} is the points returned to student agent i after cancelling course j, δ is a parameter which sets the value of the weight, and wb_{ij} is the weighted points.

The flow chart in Figure 7-2 shows how to apply weights on the returned points.

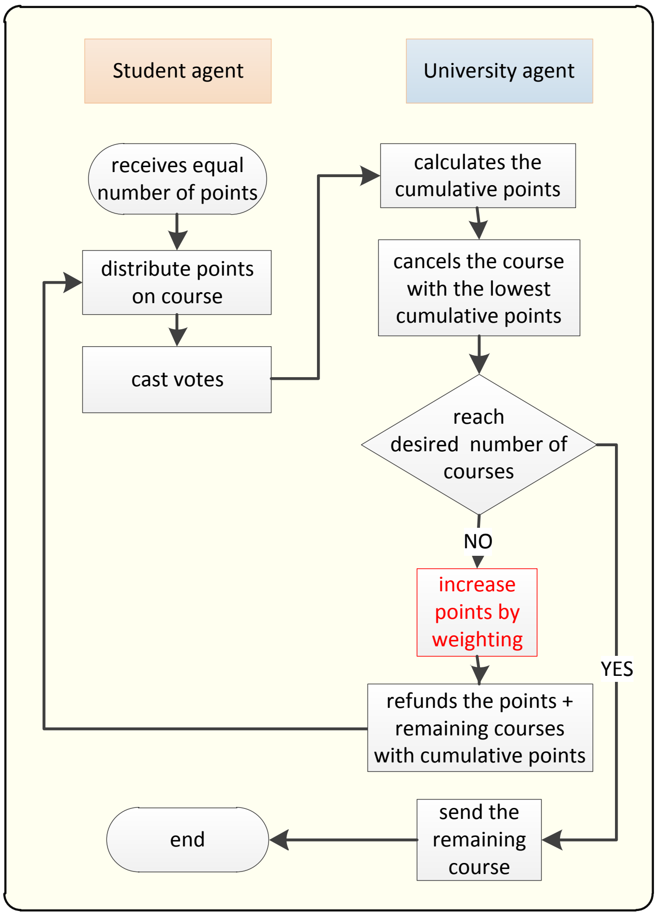


Figure 7-2: Flow chart showing how the weight mechanism works

In the following, an example shows how to calculate the weight. In the normal situation, when δ is 1, b_{1j} is 2 points and b_{2j} is 4 points and the university agent cancels $course_j$, there will be no change. However, if δ is set to 2 and applied on the points, then b_{1j} will be 4 points with an increase in worth of about double and b_{2j} will be 8 points with an increase in worth also about double. On the other hand, if a weight of 0.5 is applied, then b_{1j} and b_{2j} will 0.50 times for both (Table 7-3).

The effect of the weight would be:

- Setting δ equal to 1 is the same as the previous experiment, where points are counted on a one-for-one basis.
- Setting δ greater than 1, where the number of points increases.
- Setting δ lower than 1, where the number of points decreases.

Table 7-3: The effect of weight

	Vote				
Student A	**2**	***Weight***	0.5	1	2
		Value	1.00	2.00	4.00
		Worth	0.50 time	1 time	2 times
Student B	**4**	***Weight***	0.5	1	2
		Value	2.00	4.00	8.00
		Worth	0.50 time	1 time	2 times

Having the student agent preferences in place, we consider case number 3 used in the previous experiment and also described in section 5.1.1, where the number of students, courses and running courses is 21, 18 and 9 respectively. We define the set of all students as a cohort and a subset of students with a particular strategy as a strategy group. Thus, one cohort contains an equal proportion of student groups using different strategies. For example, if the total number of students is 21, they will be distributed equally: 7 students for the intelligent strategy, 7 students for the proportional strategy, and 7 students for the equal-share strategy. Table 7-4 shows the setting for this case:

Table 7-4: Case 3 settings

Scenario	*#courses (m)*	*#running courses (r)*	*#students (n)*
Case 3	18	9	21

Furthermore, for both experiments, we use the complex preference described in section 6.1. Each strategy was run with rules set at 50%. That means that each student valued only 50% of the available rules. This value was chosen because, in test runs, 50% was found to demonstrate the fairness aspect more clearly. Finally, each experiment was performed 500 times with different randomly generated student preferences to obtain statistically significant results.

7.2.2 Results and Analysis

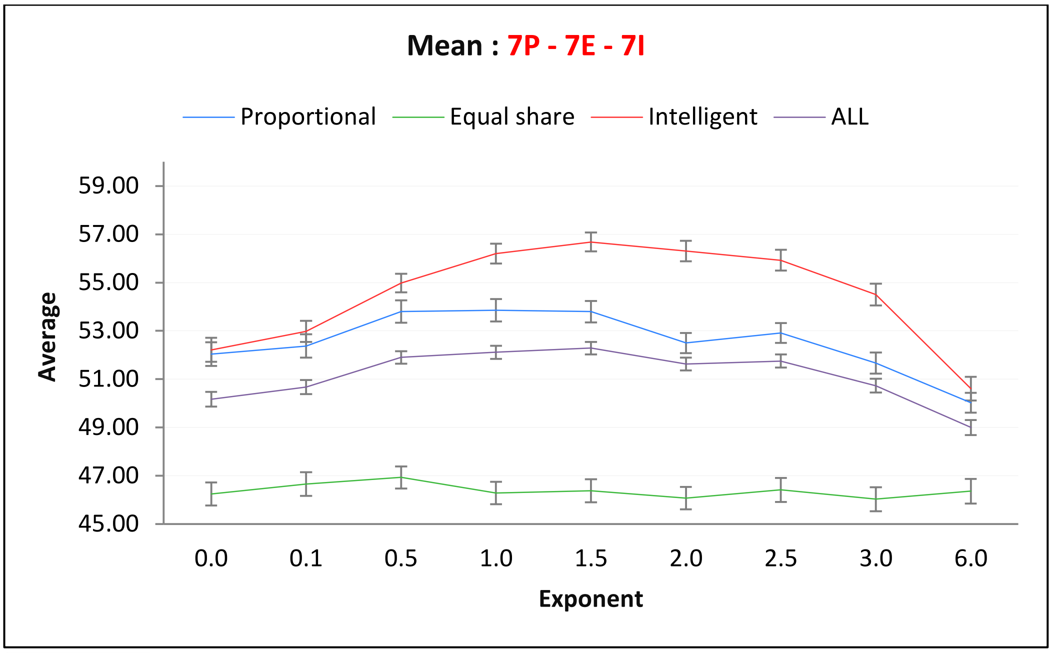
7.2.2.1 Exponent

This section will first present the performance of the strategies, and then will look at the effect on the system in terms of the fairness measures.

Performance of strategies

Figure 7-3, Figure 7-4 and Figure 7-5 show the results for comparison of the performance of strategies in the system for selected exponents. In Figure 7-3 and Figure 7-5, the y-axis shows the average of the mean, while in Figure 7-4 the y-axis shows the differences in mean between strategies. The x-axis in these figures shows different exponents between 0.0001 and 6.

Figure 7-3 shows the actual mean for each strategy. It shows in terms of the mean that the intelligent and proportional group has a peak at around 1 and 1.5 respectively, and then decrease especially when the exponent is larger. However, the equal-share group stays about the same in all cases for different exponents. This is because the equal-share strategy distributes the points equally regardless of student preference, meaning it gets the same even applying the exponent.

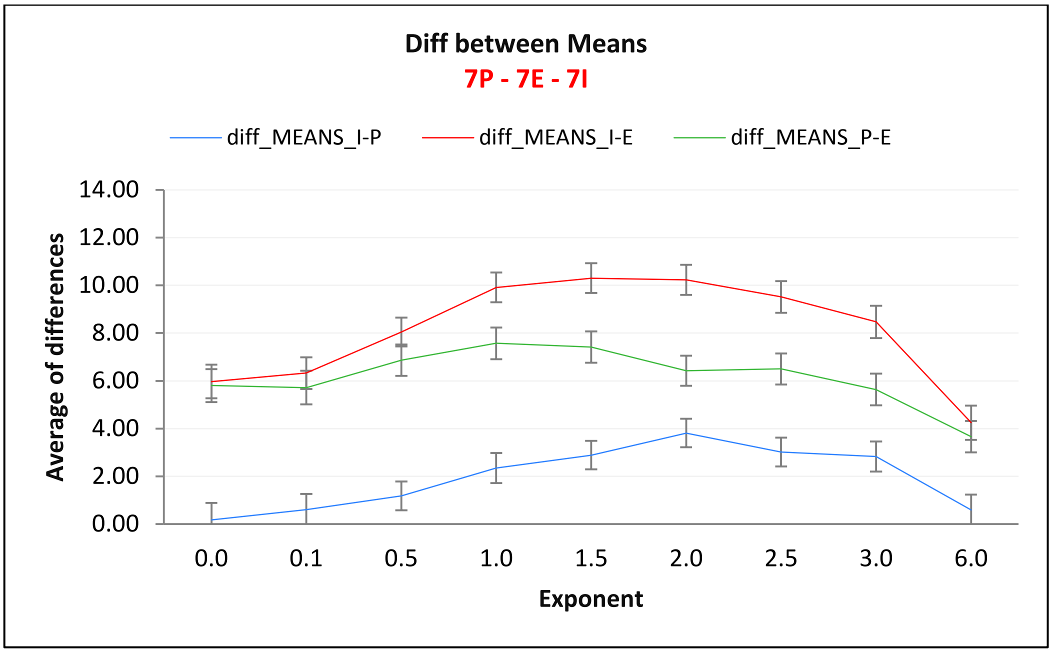


	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
Proportional	52.04	52.37	53.80	53.85	53.79	52.50	52.91	51.67	50.02
Equal Share	46.24	46.66	46.93	46.28	46.38	46.08	46.41	46.03	46.36
Intelligent	52.22	52.98	54.98	56.20	56.68	56.31	55.93	54.50	50.61
All	50.17	50.67	51.90	52.11	52.29	51.63	51.75	50.73	49.00

Figure 7-3: Comparison between intelligent, proportional and equal-share strategies in means over exponents

The result in Figure 7-4 displays the differences between strategies. It shows that, in most cases, the intelligent and proportional strategies are both better than the equal-share strategy whatever the exponent applied. When looking at the case comparing the difference between the intelligent and equal-share strategies, the trend reaches the highest level, which is 10.30 at exponent 1.5, while the highest value for proportional and equal share is 7.57 at exponent 1.0. This shows that the intelligent strategy benefits from the exponent and outperforms the other strategies in performance, although that difference begins to decrease as the exponent becomes higher than 2.0. This is because high exponents can produce very large numbers, which causes the intelligent strategy to lose its ability to influence the outcome.

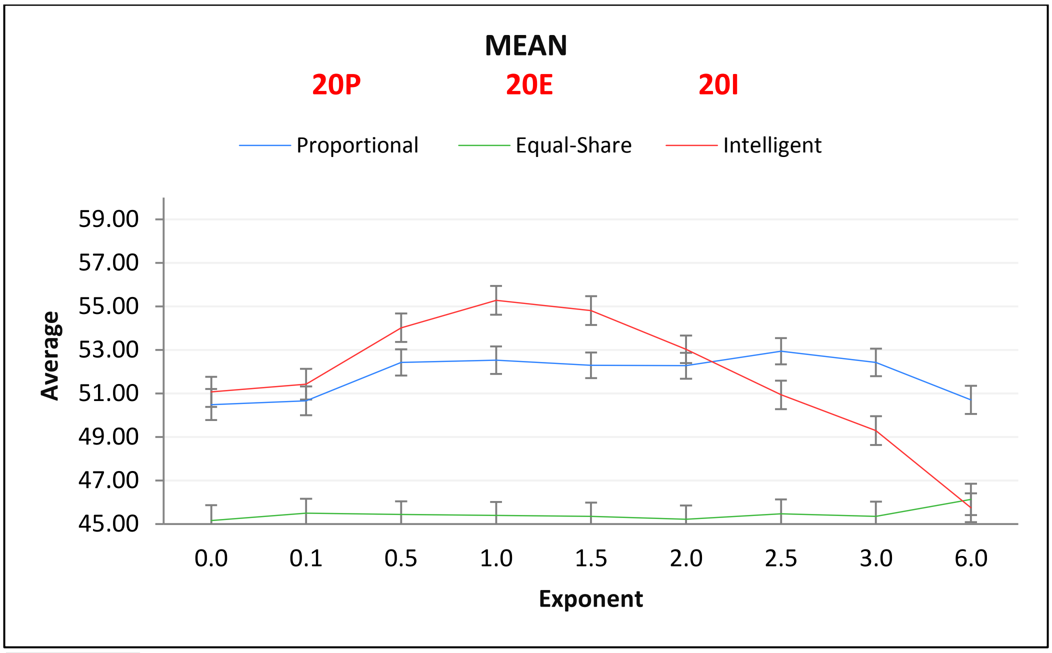
Looking at the difference in performance between strategies, Figure 7-4 shows that the biggest difference between intelligent and proportional is at 2. It shows that intelligent strategy is capable of taking advantage of an exponent, as long as that exponent is not too high (i.e. between 1.5 and 3.0)



	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
diff_ I-P	0.17	0.61	1.18	2.34	2.89	3.81	3.01	2.83	0.59
diff_ I-E	5.97	6.32	8.05	9.91	10.30	10.23	9.51	8.47	4.25
diff_ P-E	5.80	5.72	6.87	7.57	7.41	6.42	6.50	5.64	3.66

Figure 7-4: Comparison between intelligent and equal-share strategies for differences between means over exponents

To clarify this, Figure 7-5 displays the performance of different strategies, where we have All intelligent, All Proportional, and All equal share. It shows that proportional strategy and equal share have relatively stable performance whatever the strategy, but that intelligent benefits from a slightly higher exponent especially before 1, but drops significantly once the exponent becomes high (over 2.0).



	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
Proportional	50.49	50.67	52.43	52.53	52.30	52.28	52.95	52.43	50.71
Equal Share	45.16	45.49	45.45	45.39	45.36	45.23	45.47	45.36	46.14
Intelligent	51.08	51.43	54.02	55.29	54.81	53.03	50.94	49.30	45.75

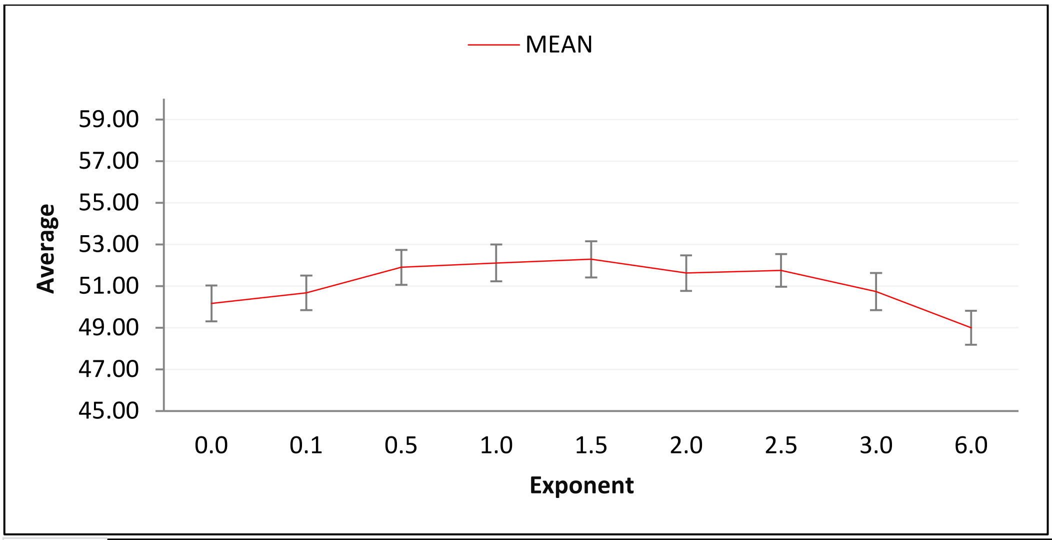
Figure 7-5: Comparison between All Intelligent, All Proportional and All equal-share in means over exponents

Fairness

This section will give an overview of the effects on the system in terms of fairness aspects, including egalitarian, utilitarian and equalitarian aspects when applying different exponents, and shows whether the fairness of the system alters as the exponent changes.

A) Utilitarian

To see how the exponent affected the utilitarian aspects when applying a mixed set of strategies, Figure 7-6 the results in terms of the utilitarian fairness measure for a setting with uniformly mixed strategies (7 intelligent, 7 proportional, 7 equal-share). This figure shows that the mean peaks at an exponent value of 1.5. However, the changes from a 0.5 exponent to a 2.5 exponent are not significant and give almost the same mean as the 1.0 exponent, which can be considered as standard. Therefore, it could be said that the system is not significantly affected by the exponent with regards to making it a more utilitarian system. Also, we can clearly see from the graph that, when the exponent gets very large, the mean satisfaction goes down. This is because votes become bigger and strategies lose the power to make a difference. So, it could be said that a mid-range exponent makes no difference in terms of the utilitarian aspect, but has a negative effect when very small (less than 0.5) or very large (greater than 2.5).



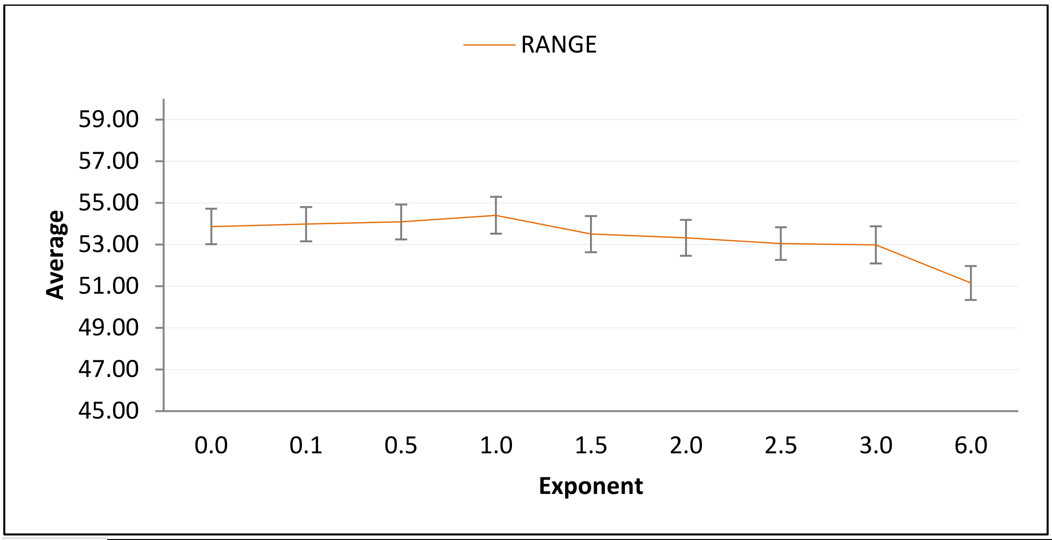
exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
MEAN	50.17	50.67	51.90	52.11	52.29	51.63	51.75	50.73	49.00

Figure 7-6: Utilitarian measure. The mean of a cohort consisting of groups of 7P, 7E and 7I

B) Equalitarian

Equalitarian aspects can be measured by the size of the difference between the maximum and minimum satisfaction, which we call the range. It can be said that a system is fair from the equalitarian perspective if it makes this range as small as possible.

Figure 7-7 illustrates the range of a cohort consisting of the three strategies: intelligent, proportional, and equal-share. Again, these are uniformly mixed. Considering the errorbars, the range is not significantly different for the different cases. In terms of fairness, this means that the exponent has no significant effect on the system from an equalitarian point of view. This is probably because the group of proportional and equal-share strategies perform quite consistently, reducing the impact of the improvements of the intelligent strategy on the range.



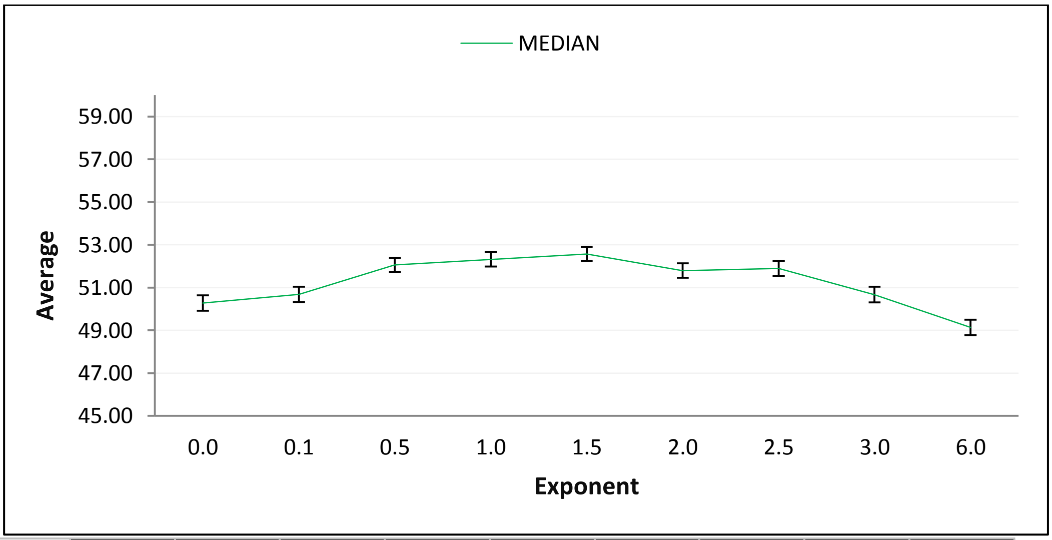
exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
RANGE	53.87	53.98	54.09	54.41	53.51	53.32	53.05	52.99	51.16

Figure 7-7: Equalitarian measure. The range of a cohort consisting of groups of 7P, 7E and 7I

C) Egalitarian

Egalitarian in terms of fairness here means to maximise the worst individual outcome and/or the median. If the median of the agent satisfaction increases, that means the system is fairer from an egalitarian point of view.

Figure 7-8 shows us a curved trend with a peak at an exponent of 1.5 (52.57). However, these changes are not significant enough in the mid-range (0.5 to 2.5) for us to say that the exponent can make a difference in egalitarian terms.



exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
MEDIAN	50.28	50.68	52.06	52.32	52.57	51.80	51.90	50.67	49.14

Figure 7-8: Egalitarian measure. The Median of a cohort consisting of groups of 7P, 7E and 7I

In conclusion, having an exponent does improve the performance of the intelligent strategy slightly, but this makes no significant difference to fairness. If we look across the board, we could say in general that, in the mid-range from 1 to 3, there is almost no difference. This demonstrates that the protocol is fair and, moreover, that it is in fact very difficult to make it fairer. Below 1, where the effective power of the vote decreases, or above 3, where the power of the vote similarly declines, we found degradation in fairness. However, in the middle rage where the protocol is functioning it is difficult to improve the fairness. This leads us to our next experiment where we increase the weight for returned points, which will give more power to people who have not been successful and should theoretically create a fairer system.

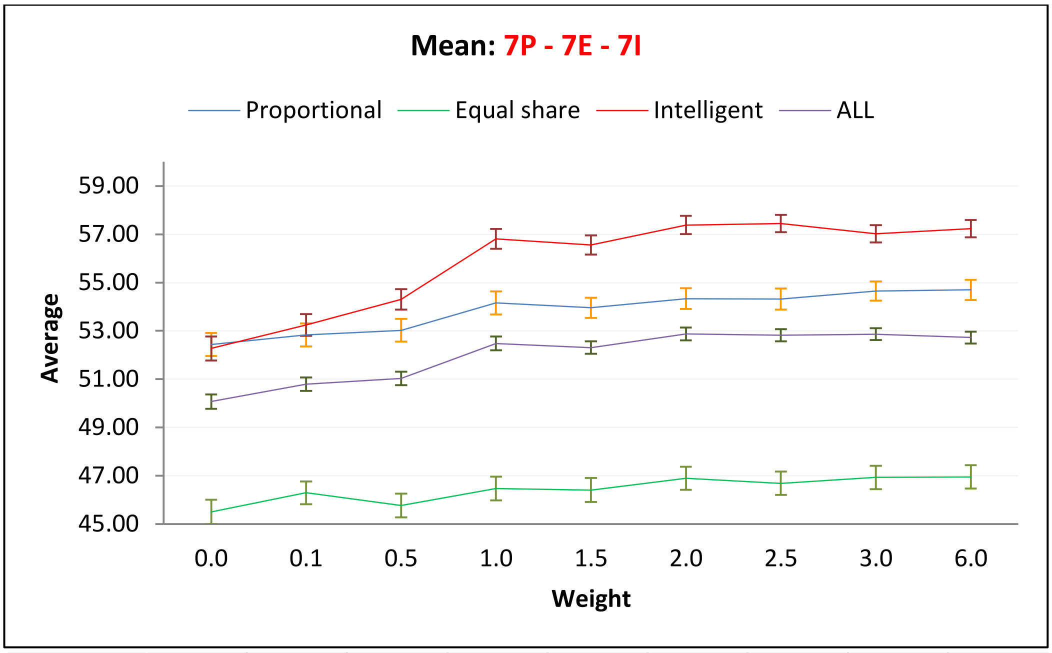
7.2.2.2 Weight

This experiment tries to produce a fair system by giving the students who have been unsuccessful in their voting more power. If an agent is doing badly, it gets compensated; therefore, in terms of strategies we would expect to see the difference between the strategies decrease. If we look at the performance of the intelligent versus proportional versus the equal-share strategies, we would expect them to equalise somewhat. In other words, we expect to see the average for each strategy group to become closer.

Performance of the strategies

Figure 7-9, Figure 7-10 and Figure 7-11 show a comparison between the performances of strategies in the system for different weights. In Figure 7-9, and Figure 7-11, the y-axis shows the average of the actual mean, while in Figure 7-10, the y-axis shows the differences between actual mean for strategies showed in Figure 7-9. The x-axis in these figures shows different weights between 0.0001 and 6.

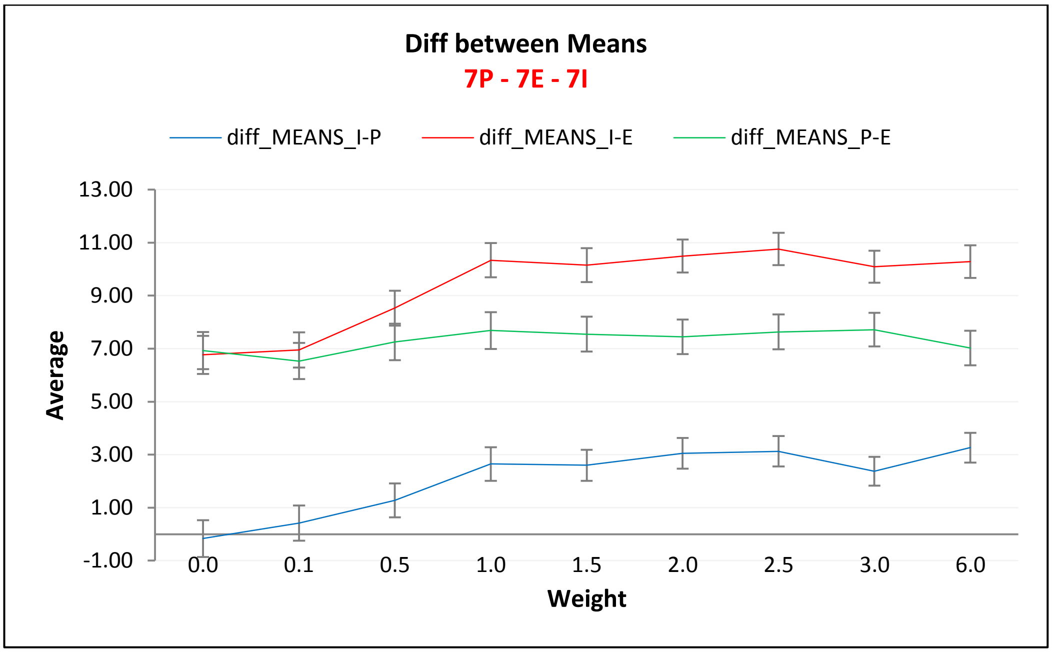
It can be seen from Figure 7-9 (presenting the actual mean of each strategy in the cohort) that the group of intelligent and proportional strategies have in all cases a better performance in comparison with the equal-share strategy. A clear change in performance is seen before weight 1 and then it stays stable as a flat trend (within the errorbars).



	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
Proportional	52.43	52.83	53.02	54.16	53.96	54.34	54.32	54.65	54.70
Equal Share	45.51	46.29	45.77	46.47	46.41	46.89	46.69	46.93	46.95
Intelligent	52.27	53.24	54.30	56.81	56.56	57.39	57.45	57.02	57.24
All	50.07	50.79	51.03	52.48	52.31	52.87	52.82	52.87	52.72

Figure 7-9: Comparison between intelligent and equal-share strategies in means over weights

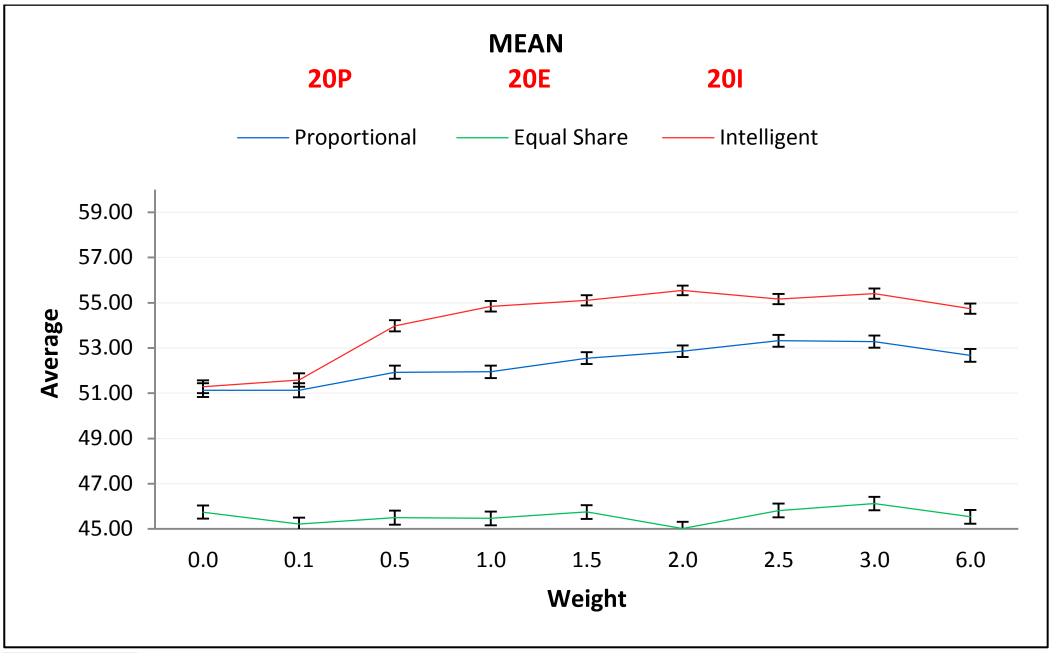
Figure 7-10 shows the differences in mean satisfaction between different strategies. It can be seen that in all cases the intelligent and proportional strategies are both significantly better than the equal-share strategy. The trend gradually increases as long as the weight increases especially before 1 weight. This is because these two strategies can better take advantage of the increase in points returned. Looking at the differences in the means for the intelligent and proportional strategies for different weights, according to the error bars there is no difference between them when the weight is greater than 1. However, when the weight drops below 1, the difference decreases.



	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
diff_ I-P	-0.16	0.42	1.28	2.65	2.60	3.05	3.13	2.37	3.26
diff_ I-E	6.77	6.95	8.53	10.33	10.15	10.49	10.76	10.09	10.28
diff_ P-E	6.93	6.53	7.25	7.69	7.55	7.45	7.63	7.72	7.02

Figure 7-10: Comparison between intelligent and equal-share strategies for differences between means over weights

To clearly this, Figure 7-11 shows the performance of different strategies where we have All intelligent, All Proportional, and All equal share. It clearly demonstrates the finding that the intelligent and proportional strategies produce increases in the performance which are better than the equal-share strategy.



	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
Proportional	51.14	51.13	51.92	51.95	52.55	52.85	53.32	53.28	52.67
Equal Share	45.74	45.22	45.50	45.47	45.75	45.02	45.82	46.12	45.54
Intelligent	51.28	51.58	53.98	54.84	55.11	55.55	55.16	55.40	54.73

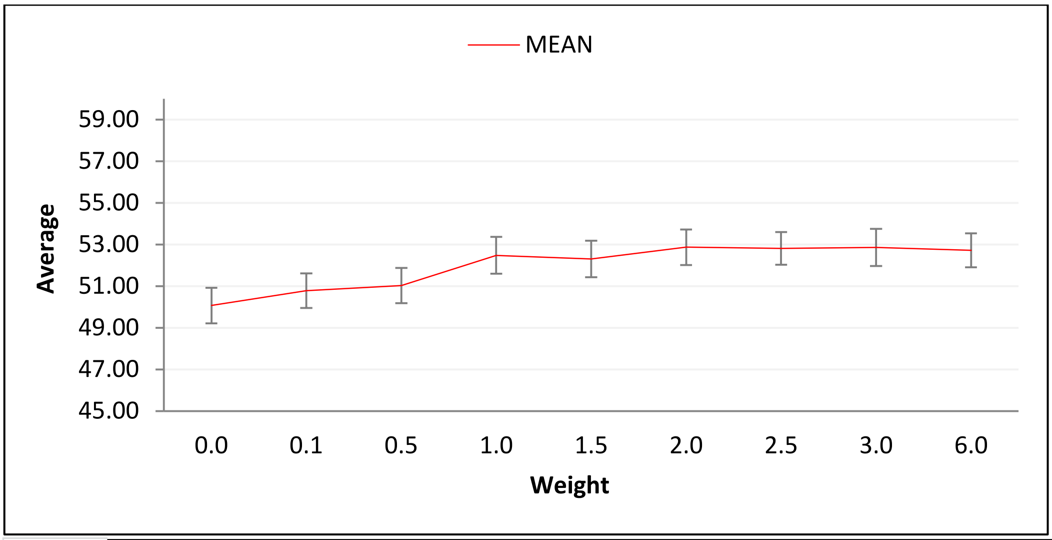
Figure 7-11: Comparison between All Intelligent, All Proportional and All equal-share in means over weights

Fairness

This section will discuss whether or not the system is affected in terms of the fairness aspect, including the egalitarian, utilitarian and equalitarian aspects when applying different weights and showing if the fairness of the system changes as the weight does.

A) Utilitarian

We will look at the mean to see how increasing the returned point affects the utilitarian aspects when applying a mixed set of strategies. Figure 7-12 shows that the average started with 44.44 at a weight of 0.0001 and climbs gradually to just over 47.80 at a weight of 6. With an overview of the graph, we can say before 1.0 there is a significant decrease in the performance of the cohort overall, but above 1.0 there is no real difference

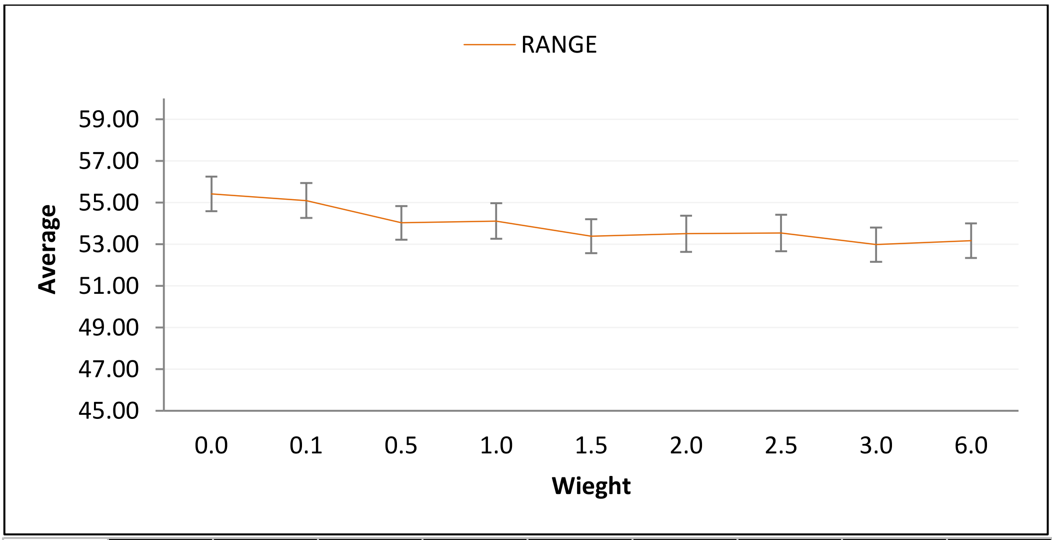


exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
MEAN	50.07	50.79	51.03	52.48	52.31	52.87	52.82	52.87	52.72

Figure 7-12: Utilitarian measure. The mean of a cohort consisting of groups of 7P, 7E and 7I

B) Equalitarian

Figure 7-13 shows that there is a gradual drop in the range from a peak of 55.42 to a low of 53.17 between the weight of 0.0000001 and 6.0. However, this is not a significant decrease and especially after the weight is set at 1 it is flat. This result does not match what we expected in terms of the equalitarian aspect, because the group of different strategies within the entire cohort produces a flat difference in the range.

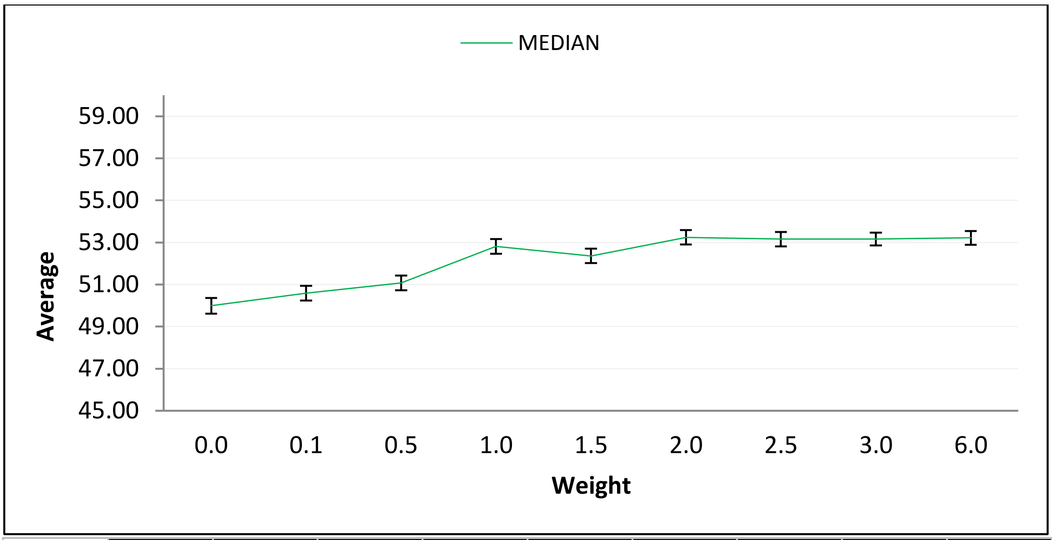


exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
RANGE	55.42	55.10	54.02	54.12	53.38	53.50	53.54	52.98	53.17

Figure 7-13: Equalitarian measure. The range of a cohort consisting of groups of 7P, 7E and 7I

C) Egalitarian

Figure 7-14 shows that there is a gradual rise in the trend from 49.99 at weight 0.0001 to 53.22 at weight 6.0 in terms of the median value. The increase is clear between 0.0001 and 1, but after that the trend stays about the same. In general, the system is affected by the weight of the returned value only if the weight is less than 1 and flat after. As a result, increasing the weight does not have an effect on the egalitarian aspect.



exp	0.0	0.1	0.5	1.0	1.5	2.0	2.5	3.0	6.0
MEDIAN	49.99	50.59	51.08	52.81	52.36	53.24	53.16	53.16	53.22

Figure 7-14: Egalitarian measure. The Median of a cohort consisting of groups of 7P, 7E and 7I

In conclusion, a weighting greater than 1 makes no significant difference from the utilitarian, egalitarian and equalitarian points of view. However, a weighting below 1 has a detrimental effect on all of our metrics.

To sum up from the exponent and weight experiments, in both cases increasing beyond 1 had no significant effect on the fairness of the system, but that below one all fairness measures declined. This is because the protocol is already very fair. Thus, the framework works as it shows the protocol is less fair when the exponent is set below 1. However, a potential problem with the fairness measures used is that they are likely to be correlated. Therefore, if the utilitarian value is increased there will also be an increase in the egalitarian and probably the equalitarian values too.

7.3 (Part2) Fairness of Strategy Distribution

The aim of this experiment is to ascertain the fairness of outcome with different strategy distributions, by applying the fairness measures described above. We will look at how fairness changes as long as the number of students allocated to the two strategies changes. The expectation of this experiment is that we would see less fairness when there is a split of student agents with an equal number utilising two strategies in one cohort.

In order to investigate this assumption, we consider case number 3, which was described in section 5.1.1, where the number of students, courses and running courses is 20, 18 and 9 respectively. Students are split between two strategies so that a group of students are allocated to one strategy increases, while the rest (allocated to another strategy) decreases. For example, if there are 20 students in total the first experiment will have 19 students for the equal-share strategy and one for the intelligent strategy. Then, for each following experiment, the student number will decrease for the equal-share and increase for intelligent strategies, until there is student using the equal-share, and 19 using the intelligent. This way we can see how the proportion of different strategies affects the fairness of the system.

7.3.1 Results and Analysis

Figure 7-15, Figure 7-16 and Figure 7-17 show the mean, median and range for utilitarian, egalitarian and equalitarian respectively as measures for fairness when two groups of strategies are combined in one cohort. The y-axis shows the overall average of each measure when there are two different groups of strategies in one cohort. On the x-axis, the total number of students using the two different strategies in one cohort is shown.

Figure 7-15 and Figure 7-16 show the mean and median for fairness measures as the number of students using the intelligent strategy increases and the equal-share strategy share decreases. When there are two groups of students within one cohort, we expect to see a peak in the middle at which point the different groups have the same number of students. However, we found a steady increase. This is because the intelligent strategy gets better results, and so the more students adopt the intelligent strategy, the better the overall cohort gets.

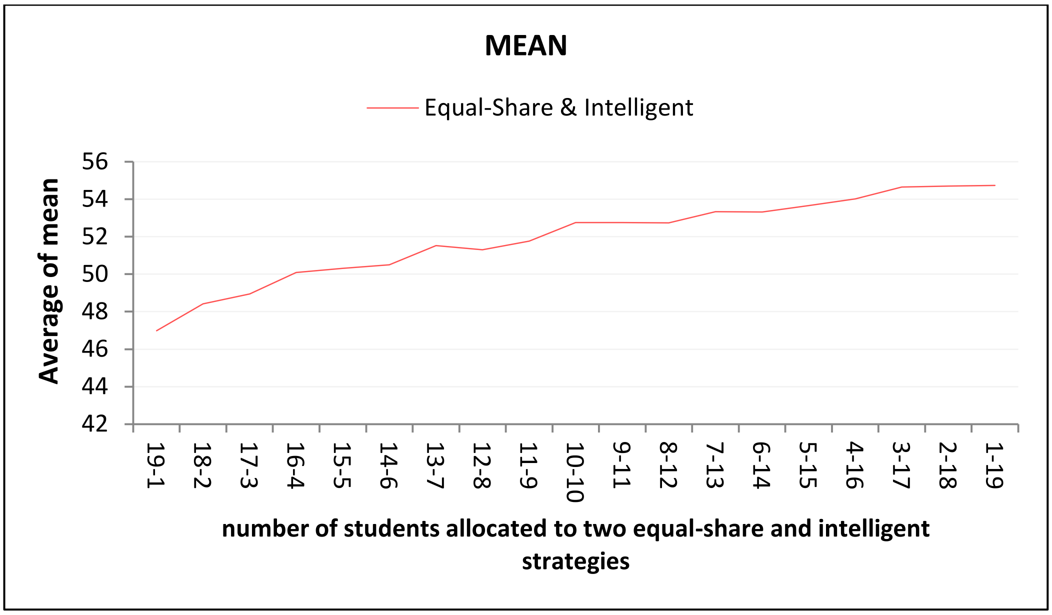


Figure 7-15: Case 3: mean when Equal-Share and Intelligent

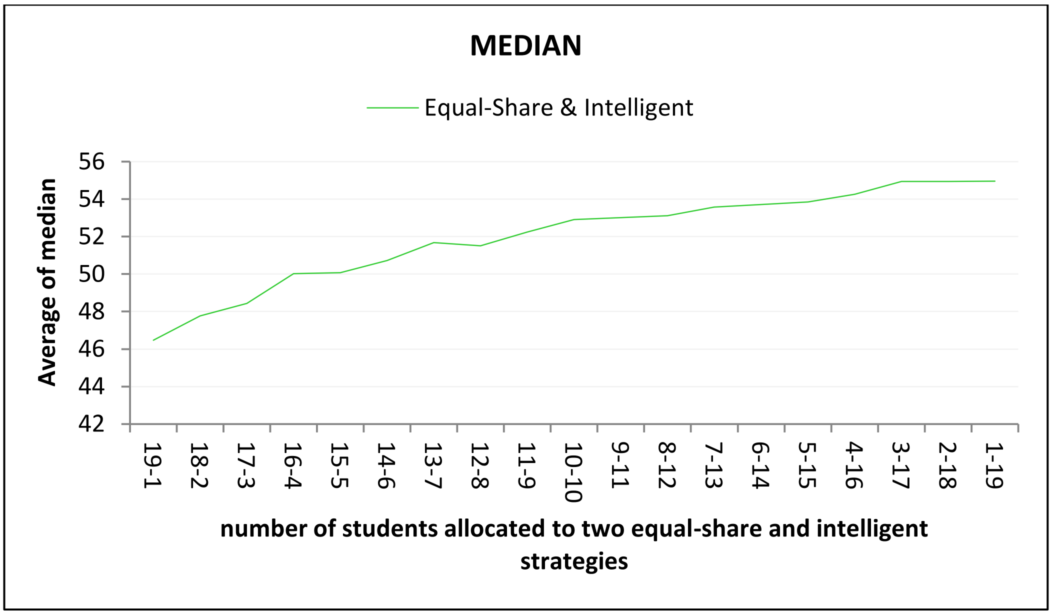


Figure 7-16: Case 3: median when Equal-Share and Intelligent

In Figure 7-17 we can see what happened to the range when applying the same setting where the number of students using the intelligent strategy increases and the equal-share strategy share decreases. We expect it to decrease until the number of students using each strategy is the same, and then increase again. However, in fact Figure 7-17 shows a steadily decreasing range. As mentioned above, the reason for this is that, at every step, one student who does not do well is removed and a student who does do well is inserted. Therefore, because the intelligent strategy gets better results, the more students adopt the intelligent strategy the better the overall cohort gets.

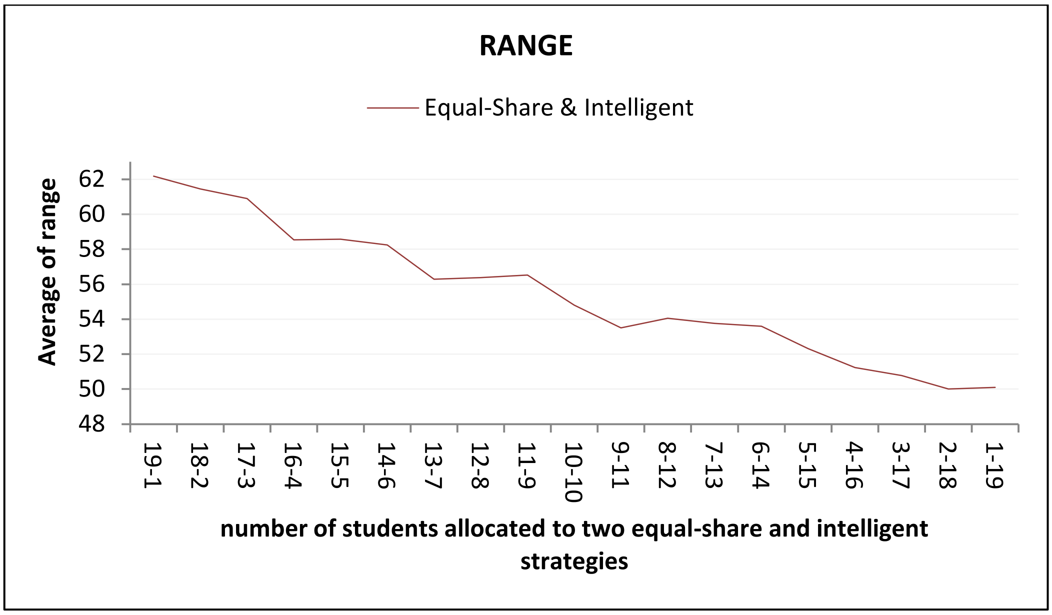


Figure 7-17: Case 3: range when Equal-Share and Intelligent

To see what happens to the range, it was taken to show the distribution of students when allocated to the two strategies during the run. Figure 7-18 shows the result of this. The y-axis shows the name of the two strategies within the cohort. The x-axis shows the overall range for the entire cohort as well as the individual ranges for the groups within the cohort taking each strategy.

What exactly happens is counteractive, every time the number student using intelligent strategy is increased the number students using equal-share is reduced. Therefore, there is a counter process resulting in the system appearing to be fairer as more students take a good strategy.

To avoid repetition, we restrict the explanation to covering what happens between intelligent and equal-share strategies. The results for other cases are very similar (for more details, see Appendix C).

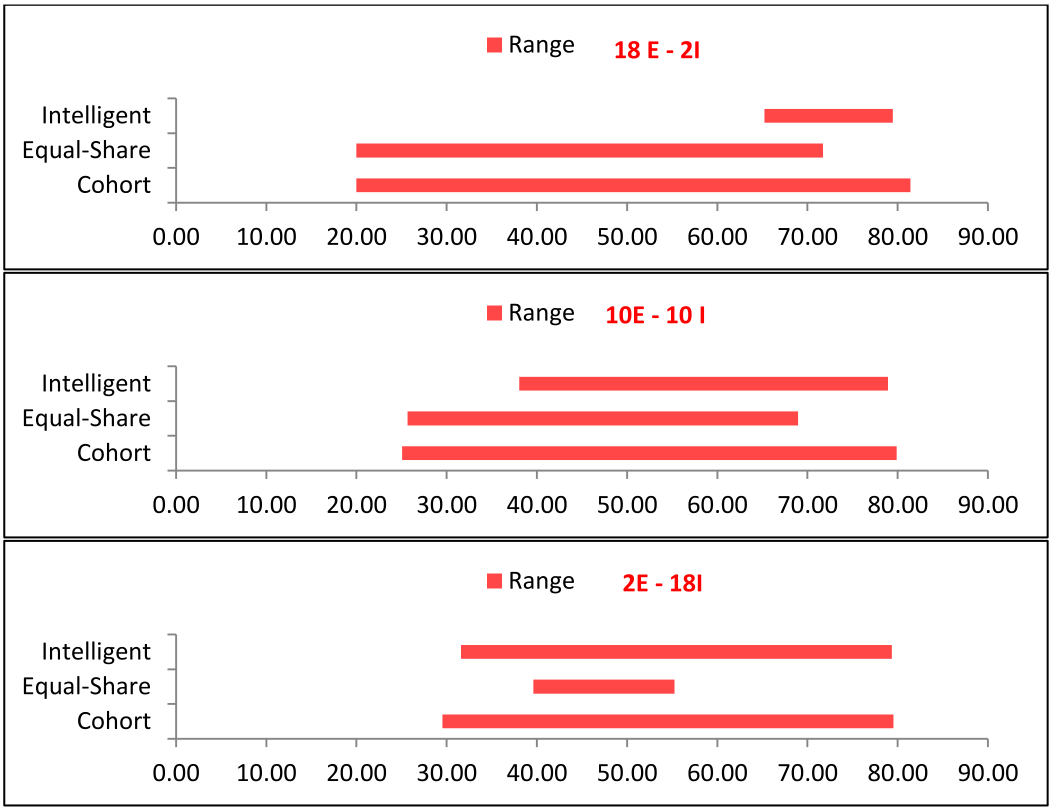


Figure 7-18: Case 3: the range for the distribution of intelligent and equal-share strategies

7.4 Discussion

In these experiments we have looked at the fairness of an e-learning distributed system. In e-learning we are not only interested in the utilitarian value but also in other measures, including egalitarian and equalitarian measures. We want to see whether the framework is functioning and able to distinguish between different situations by comparing strategies and then seeing if the framework could be used to measure changes in the fairness. After completing these experiments, some points can be highlighted.

- **Fairness framework**

From these experiments, we could say that the framework does reveal a certain type of fairness outcome, which is the overall fairness of the system rather than fairness in relation to any individual entering the system. However, the metrics used to measure fairness are linked; they are not independent. When a student using a bad strategy is removed and more students use a good strategy, the system as a whole gets fairer for all aspects of fairness.

- **Fairness as Social welfare**

This interdependence suggests that it might be more useful to look at fairness from the individual's point of view, rather than by assessing the outcome for the whole cohort. However, in order to assess this we need to have more knowledge about the agent such as the strategy he will use, which is not realistic in a decentralised mechanism. Whereas if we consider the whole as a social welfare system, we only look at the entire outcome, which is practical in a real situation.

- **Fairness of the protocol**

If we accept this pragmatic argument then the result shows we have a very fair protocol. We test the fairness of the protocol by adding two adjustments: exponent and weight. We have learned that, if we turn these measures up, although we expect this to make a positive difference to fairness, this actually does not seem to have much effect. Nevertheless, we know the fairness measures are working because, as they are turned down to zero, the protocol becomes less fair and the level of fairness drops. This shows that, although a system where a student can take a different strategy can produce unfair results, if you have the correct protocol you can produce a result that is fair as a whole system.

7.5 Summary

This chapter has presented an argument that, as e-learning systems become more sophisticated, they will introduce problems of fairness, where some students are disadvantaged because their e-learning systems use inappropriate strategies to negotiate or collaborate with others. We have presented a framework for fairness that looks at the outcomes in order to measure fairness in terms of the satisfaction students have with the outcome. The framework identifies utilitarian, egalitarian and equalitarian metrics that can be used to assess fairness.

We have then used this framework to look at fairness in our own agent system, where students use a voting protocol to collectively agree on the number of courses to be run by a university.

The experiments have shown that the framework works but measures a certain type of fairness, which is the overall fairness of the system rather than the individual level. We have also discovered that the protocol we have developed is already very fair and it is difficult to make this protocol much fairer using the two measures demonstrated (exponent and weight).

If we take an e-learning perspective that accepts a social welfare view of fairness, then we can say we have a distributed system and a voting protocol that gives results that are fair. However, our framework metrics are linked so that, as a strategy performs better, the more overall utilitarian fairness there is and thus the fairer the system is likely to be in equalitarian and egalitarian terms. There is another view of fairness which focuses on the advantages of the individual rather than the whole (for example, calculating the probability that a given students will achieve their goals and then comparing that probability with others). If this kind of fairness view is considered, it could be said that it is simply impossible in a distributed system to calculate whether it is fair or not because we do not know the strategy an individual will take or their preferences. Therefore, we could say there is only one potential view of fairness for a distributed e-learning system, which is the pragmatic view of fairness based on social welfare taken here.

Chapter 8. Conclusion and Future Work

This chapter provides a conclusion to the thesis and a summary of the work done on simulating a voting-based agent system to support an e-learning scenario. Section 8.1 gives an overview of the actions undertaken by summarising the main points from each of the previous chapters. Section 8.2 revisits the hypothesis and gives a summary of the core findings of the research. Finally in Section 8.3 suggestions are made for future work that may arise as a result of the research.

8.1 Research Summary

This thesis presented an agent based voting system for personalised e-learning scenario where autonomous agents vote on behalf of the students according to the student's preferences. We introduced three different e-learning scenarios showing how agent technologies could be used in e-learning to take full advantage of the agent's ability to map to problems in e-learning. We chose scenario one (course selection) as basis for carrying the investigation in this thesis. We then introduced a novel voting protocol for this scenario in which agents allocate points to different courses and voting occurs in several rounds. This way the agents are able to freely express their preferences and at the same time use the information provided from previous rounds to vote intelligently and strategically. We then developed three different voting strategies, with their performance evaluated through simulation in a range of settings

The key objective of this thesis is to show how agent systems can not only form a good framework for distributed e-learning systems, but also how they can be applied in personal learning contexts where the learners are autonomous and independent. To show this, we investigated three main points in separate chapters.

Specifically, these points were:

Agent system for Decentralised E-learning Scenario

In Chapter 5 the investigation centred on decentralised approaches, where student agents control their learning through collective decision-making. Such an approach marked a shift from more centralised approaches, where decisions are made by central authorities. Investigations into the decentralised approaches used the multiagent systems described in Chapter 4 to run two experiments. The first one compared a situation in which all students were allocated one strategy and compared it with an optimal centralised situation. The second involved a comparison of a number of student agents using one particular strategy and the rest using a different strategy. A decentralised approach was shown to work well; the independent voting strategy provides outcomes which are close to the optimal centralised solution. It was also shown that with this particular protocol more naïve voters were not exploited. The output was encouraging for the e-learning domain, and indicates that student agents can behave in a decentralised way and obtain their desired outcomes.

Agent system for E-Learning System Involving Complex Preferences

In chapter 5, we assumed that the preferences for different courses were independent; however this is not always realistic. In chapter 6, we aimed to demonstrate the potential of using multiagent systems and voting in settings where students have complex preferences and use a range of voting strategies. We extended the work by considering sophisticated preferences. In doing so, we investigated how the software agent votes on a student's behalf according to different degrees of complementary and substitutable preferences between courses. We adjusted the three strategies to work in the presence of rules between courses. We showed that, in the case where students have complex preferences, as the number of rules increases, the student using intelligent and proportional strategies perform better relative to equal-share. Also, we showed that, when the number of students is small, it enables each individual student to more significantly affect the voting outcome, and the intelligent strategy performs significantly better than proportional.

Fairness in intelligent system for E-Learning

The aim of Chapter 7 was to investigate the issue of fairness in intelligent agent-based e-learning systems and ensure fairness between individuals using different strategies in a system personalised to them. A simple framework was first introduced for measuring the fairness of results. This was derived from literature containing definitions of formal metrics to describe the fairness. Three different notions of fairness were identified based on the theory of social welfare: utilitarianism, equalitarianism and egalitarianism. We then used this framework to look at the fairness of different student agent strategies as measured by certain metrics. Two sets of experiments were carried out.

In the first one, we modified the protocol to try and make it fairer by applying certain factors: exponent and weight. The exponent factor refers to the situation in which a student is given more power to express voting in the system, whilst weight is given to the student who was not successful in impacting the system. The experiments showed that adding an exponent and weight does not make the system fairer, but if the exponent or the weight is lower than 1 what happens reduces the impact of the protocol and therefore the protocol becomes less fair. Drawing from that, we could say that the proposed fairness framework works because it shows that the protocol is less fair when the factors are set below 1.

The second set of experiments examined how the fairness of the system changed when two strategies worked in one cohort so that the number of students for one strategy increased, whilst for the other it decreased. This type of distribution of the strategies showed that rather than getting maximum fairness when there was only a single strategy we find it increases as adding students with a better strategy. This is because all of our fairness measures are linked. Looking at fairness from an individual agent's point of view might reveal more, but when looking at a distributed system it is impossible to do this because the objectives of the individual agents are unknowable. So, in these distributed autonomous systems, there is only one type of fairness that can be measured - in the form of social welfare - and by this measure our protocol is fair.

8.2 Research findings

In this section, the research findings are summarised. The following findings have been found in the thesis:

H1: *In e-learning scenario such as academic course selection a decentralised agent system using voting protocol, can achieve almost identical level of overall student satisfaction as an optimal centralised approach while maintaining levels of privacy and choice.*

This hypothesis was tested in Chapter 5 and two main results were found:

1. Voting procedures in particular and multiagent technology in general potentially can replace a centralised infrastructure. The most interesting findings were that:
    a. There were no differences in mean satisfaction between students taking intelligent and proportional strategies and the optimal according to the statistic results (for intelligent and optimal: t = 1.778, dt = 98, p = 0.784 and for proportional and optimal: t = 0.152, df = 98, p = 0.87).
    b. Because equal-share strategy is a naïve strategy in which voting points are allocated on an equal basis for each course, it has a poor performance. The results led us to reject the null hypothesis and accept the alternative that there is a difference in mean between equal-share strategy and optimal (equal-share and optimal produced the following results: t = 98.826, dt = 98, p = 5.85×10^{-100}).
2. Agent systems, where students use different strategies, can provide a way of providing personalisation of the learning process in a decentralised matter. This can be seen by the impact of the three strategies on overall student satisfaction. The findings were that:
    a. There was not much difference between the intelligent and proportional strategies when they were compared. In some cases where the same numbers of students were using different strategies, the intelligent strategy slightly outperformed the proportional strategy; whereas in other cases the proportional strategy outperformed the intelligent strategy. These results show that the system (voting protocol) cannot be easily exploited by an intelligent strategy.
    b. Intelligent and proportional strategies were both significantly more effective than the equal-share, irrespective of the proportion of students that used this strategy.

H2: *Within system with complex preferences, an individual student that uses an intelligent predictive strategy will on average achieve a higher satisfaction than those taking a naïve or random strategy*

In Chapter 6 this hypothesis was tested with two main results found:

1. Student agents who have complex preferences and use an intelligent strategy can achieve a higher level of satisfaction. The findings were that:
    a. As the number of rules increased, the better the intelligent strategy performed relative to the equal share. On average, the improvement was around 4%, 6%, and 8% for NO rules, 50% rules, and 100% rules respectively.
    b. There were clear differences in the performance of the intelligent and proportional strategies where there were relatively fewer courses and students. The intelligent strategy significantly outperformed the proportional strategy when a student applied the rules, and this superiority increased as more rules were applied (intelligent is better than proportional: $t = 6.769$, $dt = 98$, $p = 1.05\times 10^{-8}$).
2. Even when students with sophisticated preferences used a range of voting strategies the overall students' satisfaction was not affected.
    a. As the proportion of students using the intelligent strategy increased, the satisfaction of all students either stayed the same or increased. Therefore, using a more intelligent approach did not harm the system as a whole.

H3: *As the proportion of individual agents utilising differently performing strategies in the system increases, the overall fairness of the result (as defined by equity theory) decreases.*

In Chapter 7 this hypothesis was tested, with two main results found:

1. Fairness can be measured by the framework discussed in the thesis. This framework showed that the suggested protocol is sufficiently fair. The findings were that:
    a. Counteractively, as the number of intelligent student is increased and the number of equal-share is reduced we get fairer results. This is because more and more students are doing well, making the system appears fairer overall.

H4: *When having a uniform mixture of different strategies we can adjust the protocol by exponent and weight to make the protocol fairer.*

In Chapter 7 this hypothesis was tested, with two main results found:

1. The exponent affects the performance of the strategies but does not affect the fairness when we have mixed set of strategies. we found that:
   a. The intelligent and proportional strategies are both better than the equal-share strategy. The equal-share group stays about the same in all cases for different exponents. Moreover, in the mid-range from 1 to 3, the intelligent strategy benefits from the exponent and outperforms the other strategies in performance.
   b. From the utilitarian point of view, it could be said that a mid-range exponent makes no difference in terms of the utilitarian aspect, but has a negative effect when small (less than 0.5) or large (greater than 2.5). The system is not affected by the exponent with regard to making it a more utilitarian system.
   c. In terms of equalitarian, we found exponent has no effect on the system and produce same trend for range over different exponent. The reason for that is proportional and equal-share strategies do not produce a decrease in their performance.
   d. Connected with egalitarian, the changes made by the exponent are not significant enough in the mid-range for us to say the exponent can make a difference in egalitarian terms.
2. The weight increases the difference between strategies and it affects the utilitarian egalitarian fairness in some cases, but it does not affect for equalitarian. In particular, we found that:
   a. Over different weights, that the group of intelligent and proportional strategies have in all cases a better performance in comparison with the equal-share strategy. The clear performance is seen before weight 1 and then it stays stable as a flat trend. This is because the student is compensated for the cancellation of his or her preferred course which increases the points and gives them more value in the vote. The improvement gradually increases as long as the weight increases. Comparing intelligent and proportional, we note the intelligent strategy works better.

b. In terms of utilitarian fairness, before 1.0 there is a significant decrease in the performance of the cohort overall, but above 1.0 there is no real difference.

c. From equalitarian point of view, because the group of different strategies within the entire cohort produces a flat difference in the range epically above 1 , the system was not affected by weight and does not provide a differences in equalitarian.

d. For the egalitarian, weight can affect the egalitarian of the system only if the weight is less than 1 and flat after. But weight does not have an effect on the egalitarian aspect.

8.3 Limitations

The main limitation of the results presented in this thesis is the usage of simulation. We haven't used a real situation that maybe kind of human's factor causes a problems. For example, in real life people my fail to express their preferences correctly, something that cannot happen in the simulation. A simulation is often set up based on assumptions. Sometimes the success of the simulation depends on these assumptions but when the experiment is put in front of real people in a real situation, differences can appear.

Also we looked particularly one kind of fairness which is social welfare the fairness outcome model which does not consider the potential of individual. But it seems to capture some views of the outcome in terms of an e-learning perspective that accepts a social welfare view of fairness.

A further limitation is that we only look at one scenario and one protocol. Thus, if we were to take different scenarios and different protocols into account, the framework would be tested more carefully and evaluated more deeply. This would then provide more evidence about the functionality of the framework and how efficiently it works.

8.4 Future Work

There are a number of ways in which this work could be furthered, both in terms of a practical application, and also in exploring simulations of other scenarios.

8.4.1 From simulation to reality

Throughout this thesis an investigation was undertaken into how agent systems using voting procedures can be applied in the e-learning domain. With important features of an agent system, such as autonomy and sociability, it is important to consider how students can be provided with a personalised course. It was demonstrated that this approach was effective and, in terms of overall social welfare, fair and therefore it could be applied to a real situation to find out more about the human aspects of such as system. If a course selection agent based system could be constructed whereby the course is distributed and the student is free to choose a certain strategy when making a selection, from an educational perspective, three questions could be considered: firstly, how to make such an infrastructure work; secondly, how fair it is perceived to be by real students; and thirdly whether there would be a demand amongst students for such a system.

8.4.2 Exploring other scenarios

The agent system was applied to the particular case of course selection, but there were a number of other cases identified in this thesis. Conducting a simulation for these cases would explore whether or not fairness exists in these cases as well, and whether or not the agent solution can provide more personalisation to students in areas other than course selection. We are particularly interested in applying agent technology to the cases mentioned in section 3.6 which involve group formation and personalised learning.

There are other important questions raised when considering the case of group formation and the use of agent technology. Firstly, how can a desirable mix of abilities be reached, so that no one set of student has advantage over another? Secondly, how can students be placed in the groups they want, whilst allowing the teacher the ability to arrange students in groups that will help them learn more material or more quickly? Thirdly, how fair is such a system?

Regarding the case where there are a number of limited resources, such as tutorial slots, laboratory equipment, seminar places, the questions should be addressed thus: firstly, how agents could bargain with the institution until their utility function is maximised within the constraints of the institution's cost level and; secondly, would such a scenario provide fair results to the student.

8.5 Conclusion

In this thesis, the research has focused on how agent technology and voting procedures can be used in the e-learning domain, especially for personalised learning. We have drawn extensively on the literature on agent systems and technology and education, including personalised and informal learning, and the existing use of agent technology in e-learning.

We believe that agent systems have a great deal of potential for e-learning as their widespread use could allow genuine decentralisation and personalisation, allowing some scenarios to be extended to include types of personal and informal learning that are difficult to support with today's systems. We also researched the issue of fairness in intelligent and personalised e-learning systems, where students can have different experiences as tasks. In this context, we could say there is only one potential view of fairness for a distributed e-learning system, which is the social welfare view that looks at the overall outcome. From an e-learning perspective that accepts a social welfare view of fairness, we can say we have a distributed system and a voting protocol that gives results that are fair.

The results and the research in this thesis can be used as a foundation for more research to advance both voting-based agent systems and personalised education systems by the researchers in these fields.

REFERENCES

Adams, J. S. (1966). "Inequity in Social Exchange." Advances in Experimental Social Psychology **2**: 267-299.

Adams, N., M. Field, E. Gelenbe, D. Hand, N. Jennings, D. Leslie, D. Nicholson, S. Ramchurn and A. Rogers (2008). Intelligent agents for disaster management. Proceedings of the IARP/EURON Workshop on Robotics for Risky Interventions and Environmental Surveillance (RISE).

Albayrak, S., S. Wollny, N. Varone, A. Lommatzsch and D. Milosevic (2005). Agent Technology for Personalized Information Filtering: the PIA-system. Proceedings of the 2005 ACM symposium on Applied computing. Santa Fe, New Mexico, ACM**:** 54-59.

Alechina, N., R. H. Bordini, J. F. Hübner, M. Jago and B. Logan (2006). Automating Belief Revision for AgentSpeak. Declarative Agent Languages and Technologies IV. M. Baldoni and U. Endriss, Springer Berlin Heidelberg. **4327:** 61-77.

Ally, M. (2004). "Foundations of educational theory for online learning." Theory and practice of online learning: 3-31.

Anderson, T. (2006). "PLEs versus LMS: Are PLEs ready for Prime time?" Retrieved 20/03, 2012, from http://www.downes.ca/cgi-bin/page.cgi?post=33197.

Anghel, C. and I. Salomie (2003). "JADE Based Solutions for Knowledge Assessment in eLearning Environments." TILAB & University of Limerick.

AOS, A. O. S. (2011). "JACK An Agent Infrastructure for Providing the Decision-Making Capability Required for Autonomous Systems."

Retrieved 11/05, 2012, from http://www.aosgrp.com/downloads/JACK_WhitePaper_US.pdf.

AOS, A. O. S. (2011). JACK Intelligent Agents Agent Practicals. Victoria, Australia, Agent Oriented Software Pty. Ltd.**:** 88.

Aroyo, L. and P. Kommers (1999). "Special Issue Preface: Intelligent Agents for Educational Computer-Aided Systemsb." Journal of Interactive Learning Research **10**: 235-242.

Attwell, G. (2007). "Personal Learning Environments: The future of eLearning?" Retrieved 12/03, 2012, from http://www.elearningeuropa.info/files/media/media11561.pdf.

Babad, E. and A. Tayeb (2003). "Experimental analysis of students' course selection." British Journal of Educational Psychology **73**(3): 373-393.

Bailey, C. P. (2002). An Agent-Based Framework to Support Adaptive Hypermedia. PhD, University of Southampton.

Bariso, E. U. (2010). Personalised eLearning in Further Education. Technology-Supported Environments for Personalized Learning: Methods and Case Studies. J. O'Donoghue. London,UK, Information science reference**:** 138-156.

Bednarik, R., M. Joy, A. Moreno, N. Myller, S. Sun and E. Sutinen (2005). Multi-Agent Educational System for Program Visualization. Proceedings of 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05), Vienna, Austria.

Beer, M., M. Inverno, M. Luck, N. R. Jennings, C. Preist and M. Schroeder (1998). "Negotiation in Multi-Agent Systems." The Knowledge Engineering Review **14**(3): 285-289.

Bellifemine, F. L., G. Caire and D. Greenwood (2007). Developing Multi-Agent Systems with JADE. Chichester,UK, John Wiley.

Bernon, C., M.-P. Gleizes, S. Peyruqueou and G. Picard (2003). ADELFE: A Methodology for Adaptive Multi-agent Systems Engineering

Engineering Societies in the Agents World III. P. Petta, R. Tolksdorf and F. Zambonelli, Springer-Verlag Berlin Heidelberg

2577: 70-81.

Bologna-Secretariat. (2010). "About the Bologna Process." Retrieved 08/10, 1212.

Bordini, R. H. and J. F. Hübner (2006). BDI Agent Programming in AgentSpeak Using Jason. Computational Logic in Multi-Agent Systems. F. Toni and P. Torroni, Springer Berlin Heidelberg. **3900:** 143-164.

Bordini, R. H., J. F. Hübner and M. J. Wooldridge (2007). Programming multi-agent systems in agentSpeak using Jason. Chichester, John Wiley.

Boutilier, C. and H. H. Hoos (2001). Bidding Languages for Combinatorial Auctions. Proceedings of the 17th international joint conference on Artificial intelligence. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. **2:** 1211-1217.

Brams, S. J. (1991). "Alternative Voting Systems." Political Parties and Elections in the United States: An Encyclopedia **1**: 23-31.

Brams, S. J. and P. C. Fishburn (2002). Voting Procedures. Handbook of Social Choice and Welfare. A. K. S. Kenneth J. Arrow and S. Kotaro, Elsevier. **1:** 173-236.

Brennan, R. W., M. Fletcher and D. H. Norrie (2002). "An Agent-Based Approach to Reconfiguration of Real-Time Distributed Control Systems." IEEE Transactions on Robotics and Automation **18**(4): 444-451.

Brown, C., P. Varley and J. W. Pal (2009). "University course selection and services marketing." Marketing intelligence & planning **27**(3): 310-325.

Brusilovsky, P. (1994). "The Construction and Application of Student Models in Intelligent Tutoring Systems." Journal of Computer and Systems Sciences International **32**(1): 70-89.

Brusilovsky, P. (1996). "Methods and Techniques of Adaptive Hypermedia." User Modeling and User Adapted Interaction **6**: 87-129.

Brusilovsky, P. (2000). Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-Based Education Intelligent Tutoring Systems. Proceedings of the 5th International Conference on Intelligent Tutoring Systems. London, UK, Springer-Verlag Berlin Heidelberg**:** 1-7.

Brusilovsky, P. (2001). "Adaptive Hypermedia." User Modeling and User-Adapted Interaction **11**(1-2): 87-110.

Brusilovsky, P. (2003). "Adaptive Navigation Support in Educational Hypermedia: The Role of Student Knowledge Level and the Case for Meta-Adaptation." British Journal of Educational Technology **34**(4): 487-497.

Brusilovsky, P., J. Eklund and E. Schwarz (1998). "Web-Based Education for all: A Tool for Development Adaptive Courseware." Computer Networks and ISDN Systems **30**(1–7): 291-300.

Busetta, P., R. Rönnquist, A. Hodgson and A. Lucas (1999). JACK Intelligent Agents-Components for Intelligent Agents in Java. AgentLink News Letter. Melbourne,Australia, Europe's ESPRIT-Funded Network of Excellence for Agent-Based Computing. **2:** 2-5.

Cantador, I., M. Fernández, D. Vallet, P. Castells, J. Picault and M. Ribière (2008). "A Multi-Purpose Ontology-Based Approach for Personalised Content Filtering and Retrieval." Advances in Semantic Media Adaptation and Personalization **93**: 25-51.

CEC (2001). The eLearning Action Plan: Designing Tomorrow's Education. Brussels, Commission of the European Communities.

Chen, S. Y. and M. L. Chiu (2005). "Building an Agent-Based System for e-Learning in Digital Design." Computer-Aided Design and Applications **2**(4): 469-476.

Chevaleyre, Y., U. Endriss, J. Lang and N. Maudet (2007). A Short Introduction to Computational Social Choice. Proceedings of the 33rd conference on Current Trends in Theory and Practice of Computer Science, Harrachov, Czech Republic, Springer-Verlag Berlin Heidelberg.

Chevaleyre, Y., U. Endriss, J. Lang and N. Maudet (2008). "Preference Handling in Combinatorial Domains: From AI to Social Choice." AI Magazine **29**(4): 37-46.

Christodoulopoulos, C. E. and K. A. Papanikolaou (2007). A group formation tool in an e-learning context. Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence, IEEE.

Collier, N. (2003). "Repast: An extensible framework for agent simulation." The University of Chicago's Social Science Research **36**.

Conitzer, V. (2010). "Comparing Multiagent Systems Research in Combinatorial Auctions and Voting." Annals of Mathematics and Artificial Intelligence **58**(3-4): 239-259.

Conitzer, V. and T. W. Sandholm (2005). Common Voting Rules as Maximum Likelihood Estimators. the 26th National Conference on Artificial Intelligence (AAAI-12). Toronto, Canada**:** 145-152.

Conner, M. (2008). "Informal Learning." Retrieved 04/12, 2011, from http://marciaconner.com/resources/informal-learning/.

Conole, G. (2010). Personalisation Through Technology-Enhanced Learning. Technology-Supported Environments for Personalized Learning: Methods and Case Studies. J. O'Donoghue. London, Information science reference**:** 30-44.

Coombs, P. H. and M. Ahmed (1974). Attacking Rural Poverty : How Nonformal Education Can Help. Baltimore, Johns Hopkins University Press.

Corbett, A. T., K. R. Koedinger and J. R. Anderson (1997). Intelligent Tutoring Systems. Handbook of Human-Computer Interaction. Amsterdam.

Craven, J. (1992). Social Choice : A Framework for Collective Decisions and Individual Judgements. Cambridge, Cambridge University Press.

Cross, K. P. (1992). Adults as Learners : Increasing Participation and Facilitating Learning. San Francisco, John Wiley & Sons.

d'Inverno, M., D. Kinny, M. Luck and M. Wooldridge (1998). A Formal Specification of dMARS. Proceedings of the 4th International Workshop on Intelligent Agents IV, Agent Theories, Architectures, and Languages, Springer-Verlag**:** 155-176.

D'Inverno, M., M. Luck, M. Georgeff, D. Kinny and M. Wooldridge (2004). "The dMARS Architecture: A Specification of the Distributed Multi-Agent Reasoning System." Autonomous Agents and Multi-Agent Systems **9**(1-2): 5-53.

Deutsch, M. (1975). "Equity, Equality, and Need: What Determines Which Value Will Be Used as the Basis of Distributive Justice?" Journal of Social Issues **31**(3): 137-149.

Downes, S. (2005). "E-Learning 2.0." eLearn Magazine.

Dron, J. and M. Bhattacharya (2007). Lost in the Web 2.0 jungle. Proceedings o of the 7the IEEE International Conference on Advanced Learning Technologies. Niigata, Japan.

Dworkin, R. (1981). "What is equality? Part 2: Equality of Resources." Philosophy and Public Affairs **10**(4): 283-345.

Endriss, U., N. Maudet, F. Sadri and F. Toni (2003). "Resource Allocation in Egalitarian Agent Societies." Secondes Journées Francophones sur les Modeles Formels d'Interaction: 101–110.

Folger, R. and M. A. Konovsky (1989). "Effects of Procedural and Distributive Justice on Reactions to Pay Raise Decisions." The Academy of Management Journal **32**(1): 115-130.

Gaertner, W. (2006). A Primer in Ssocial Choice Theory. Oxford, Oxford University Press.

Gillies, M. and D. Ballin (2004). Integrating Autonomous Behavior and User Control for Believable Agents. Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems. New York, USA, IEEE Computer Society. **1:** 336-343.

Gonzalez, C., J. C. Burguillo and M. Llamas (2006). A Qualitative Comparison of Techniques for Student Modeling in Intelligent Tutoring Systems. Proceedings of the 36th ASEE/IEEE Frontiers in Education Conference. San Diego, CA**:** 13-18.

Harmelen, M. v. (2006). Personal Learning Environments. Proceedings of the 6th International Conference on Advanced Learning Technologies (ICALT'06). Kerkrade, The Netherlands**:** 815-817.

Hensher, D. A., J. M. Rose and W. H. Greene (2005). Applied Choice Analysis: a Primer. Cambridge, Cambridge University Press.

Holmes, B. and J. Gardner (2006). E-learning: Concepts and Practice. London, SAGE Publications Ltd.

Jain, L. C., N. S. Ichalkaranje and G. Tonfoni (2002). Advances in Intelligent Systems for Defence. USA, World Scientific Publishing Co., Inc.

Järvelä, S. (2006). Personalised learning? New Insights into Fostering Learning Capacity. Paris, Farnce, OECD.

Jennings, N., J. Corera and I. Laresgoiti (1995). Developing Industrial Multi-Agent Systems. Proceedings of 1stIinternational Conferences on Multi-Agent Systems (ICMAS '95). San Francisco, USA**:** 423-430.

Jennings, N. R., K. Sycara and M. Wooldridge (1998). "A Roadmap of Agent Research and Development." Autonomous Agents and Multi-Agent Systems **1**(1): 7-38.

Jennings, N. R. and M. Wooldridge (1998). Applications of intelligent agents. Agent technology: Foundations, Applications, and Markets. New York, USA, Springer-Verlag Berlin Heidelberg**:** 3-28.

Jeon, H., C. Petrie and M. R. Cutkosky (2000). "JATLite: A Java Agent Infrastructure with Message Routing." IEEE Internet Computing **4**(2): 87-96.

Jonassen, D., M. Davidson, M. Collins, J. Campbell and B. B. Haag (1995). "Constructivism and computer-mediated communication in distance education." American Journal of Distance Education **9**(2): 7-26.

Jong, S., K. Tuyls and K. Verbeeck (2008). "Fairness in multi-agent systems." The Knowledge Engineering Review **23**(02): 153-180.

Karagiannidis, C. and D. Sampson (2002). Accommodating Learning Styles in Adaptation Logics for Personalised Learning Systems. World Conference on Educational Multimedia, Hypermedia and Telecommunications 2002. P. Barker and S. Rebelsky. Denver, Colorado, USA, AACE**:** 1715-1726.

Keengwe, J. and T. T. Kidd (2010). "Towards Best Practices in Online Learning and Teaching in Higher Education." MERLOT Journal of Online Learning and Teaching **6**(2): 533-541.

Kommers, P. and L. Aroyo (1999). "Special Issue Preface - Intelligent Agents for Educational Computer-Aided Systems." Journal of Interactive Learning Research **10**(3): 235-242.

Kosch, H., M. Döller and L. Böszörményi (2001). Content-Based Indexing and Retrieval Supported by Mobile Agent Technology. Proceedings of 2th International Workshop on Multimedia Databases and Image Communication. M. Tucci. Amalfi, Italy, Springer-Verlag Berlin Heidelberg. **2184:** 152-166.

Lang, J. (2004). "Logical Preference Representation and Combinatorial Vote." Annals of Mathematics and Artificial Intelligence **42**(1-3): 37-71.

Lang, J. (2008). Voting in Combinatorial Domains: What Logic and AI Have to Say. Logics in Artificial Intelligence. S. Hölldobler, C. Lutz and H. Wansing, Springer Berlin - Heidelberg. **5293:** 5-7.

Laruelle, A. and F. Valenciano (2008). Voting and Collective Decision-Making: Bargaining and Power, Cambridge University Press.

Lefoe, G. (1998). Creating constructivist learning environments on the web: The challenge in higher education. Proceedings of Australasian Society for Computers in Learning in Tertiary Education Conference., Australia.

Leszczyna, R. (2004). Evaluation of agent platforms. Ispra, Italy, European Commission, Joint Research Centre, Institute for the Protection and security of the Citizen.

Leventhal, G. S. (1976). "What Should Be Done with Equity Theory? New Approaches to the Study of Fairness in Social Relationships." Social Exchanges: Advances in theory and research: 27-55.

Limberg, L. and M. Alexandersson (2010). Encyclopedia of Library and Information Sciences. London, CRC Press.

MacManus, R. (2007). "e-learning 2.0: All You Need To Know." Retrieved 04, 2009 from www.readwriteweb.com/archives/e-learning_20_all_you_need_to_know.php.

Maiese, M. (2003). Types of Justice. Beyond Intractability. Boulder, USA, Guy Burgess and Heidi Burgess. Conflict Information Consortium, University of Colorado.

Mason, R. and F. Rennie (2007). "Using Web 2.0 for learning in the community." The Internet and Higher Education **10**(3): 196-203.

McGivney, V. (1999). Informal Learning in the Community: A Trigger for Change and Development. Leicester, UK, National Institute of Adult Continuing Education.

McLeod, G. (2003). "Learning theory and instructional design." Learning Matters **2**: 35-43.

Melis, E., E. Andrès, J. Büdenbender, A. Frischauf, P. L. George Goguadze, M. Pollet and C. Ullrich (2001). "ActiveMath: A Generic and Adaptive Web-Based Learning Environment " International Journal of Artificial Intelligence in Education **12**: 385-407.

Melis, E., J. Budenbender, G. Goguadze, P. Libbrecht and C. Ullrich (2002). "Knowledge Representation and Management in ActiveMATH." Annals of Mathematics and Artificial Intelligence **38**(1): 47-64.

Milgrom, E., P. Chainho, Y. Deville, R. Evans, P. Kearney and P. Massonet (2001). Final Guideline for the Identification of Relevant Problem Areas where Agent Technology is Appropriate. Methodology for Engineering Systems of Software Agents, EURESCOM.

Millard, D., H. Davis, M. Weal, K. Aben and P. D. Bra (2003). AHA! meets Auld Linky: Integrating Designed and Free-form Hypertext Systems. Proceedings of the 14th ACM conference on Hypertext and hypermedia. Nottingham, UK, ACM**:** 161-169.

Molnar, A. (1997). "Computers in Education: A Brief History." THE Journal (Technological Horizons In Education) **24**(11): 63-68.

Moreau, L., N. Gibbins, D. DeRoure, S. El-Beltagy, W. Hall, G. Hughes, D. Joyce, S. Kim, D. Michaelides, D. Millard, S. Reich, R. Tansley and M. Weal (2000). SoFAR with DIM Agents: An Agent Framework for Distributed Information Management. Proceedings of the 5th International Conference and Exhibition on The Practical Application of Intelligent Agents and Multi-Agent Technology Manchester, UK.

Mourlas, C. and P. Germanakos (2009). Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies, Information Science Publishing.

Myerson, R. B. (1981). "Utilitarianism, Egalitarianism, and the Timing Effect in Social Choice Problems." Econometrica **49**(4): 883-897.

NAEYC (1996). Technology and young children – Ages 3 through 8: Position statement. Washington, DC, National Association for the Education of Young Children.

Nicholson, P. (2007). A History of E-Learning Echoes of the pioneers. Computers and Education: E-Learning, From Theory to Practice. Dordrechr, The Netherlands, Springer**:** 1-11.

Nkambou, R. (2006). Towards Affective Intelligent Tutoring System. Workshop on Motivational and Affective Issues in ITS. In: the 8th International Conference on ITS**:** 5-12.

North, M. J. and C. M. Macal (2005). "Escaping the Accidents of History: An Overview of Aartificial Life Modeling with Repast." Artificial Life Models in Software: 115-142.

O'Hear, S. (2006). "E-learning 2.0- howWeb technologies are shaping education." ReadWriteWeb **Access Date 09/09/2009, from http://www.readwriteweb.com/archives/e-learning_20.php**

Ong, J. and S. Ramachandran (2003). Intelligent Tutoring Systems: Using AI to Improve Training Performance and ROI. San Mateo, USA, Stottler henke Associates.

Ounnas, A., H. C. Davis and D. E. Millard (2007). Semantic modeling for group formation. Workshop on Personalisation in E-Learning Environments at Individual and Group Level (PING) at the 11th International Conference on User Modeling UM2007. Corfu, Greece,.

Papanikolaou, K. A., M. Grigoriadou, H. Kornilakis and G. D. Magoulas (2003). "Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE." User Modeling and User-Adapted Interaction **13**(3): 213-267.

Phobun, P. and J. Vicheanpanya (2010). "Adaptive Intelligent Tutoring Systems for e-learning Systems." Procedia - Social and Behavioral Sciences **2**(2): 4064-4069.

Procaccia, A. D. (2008). Computational Voting Theory: Of the Agents, By the Agents, For the Agents, PhD thesis, The Hebrew University of Jerusalem.

Rabin, M. (1993). "Incorporating Fairness Into Game Theory and Economics." The American Economic Review **83**(7): 1281-1302.

Redmond, M. A. (2001). A computer program to aid assignment of student project groups. Proceedings of the thirty second SIGCSE technical symposium, USA, ACM.

Riemsdijk, M. B. v., M. Dastani and M. Winikoff (2008). Goals in Agent Systems: a Unifying Framework. Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems. Estoril, Portugal, International Foundation for Autonomous Agents and Multiagent Systems. **2:** 713-720.

Rosenschein, J. and A. Procaccia (2006). Voting in Cooperative Information Agent Scenarios: Use and Abuse. Proceedings of the 10th International Conference on Cooperative Information Agents. Edinburgh, UK, Springer-Verlag Berlin Heidelberg**:** 33-50.

Rossi, F., K. B. Venable and T. Walsh (2011). A Short Introduction To Preferences: Between Artificial Intelligence and Social Choice, Morgan & Claypool Publishers.

Sampson, D. (2001). Current and Future Research and Technology Developments in e-Learning. Proceedings of the 2nd International Conference on New Horizons in Industry and Education. Milos Island, Greece, Technological Educational Institute of Crete.

Sampson, D., C. Karagiannidis and Kinshuk (2002). "Personalised Learning: Education, Technological and Standardisation Perspective." Interactive Education Multimedia **4**: 24-39.

Schaffert, S. and W. Hilzensauer (2008). "On the way towards Personal Learning Environments: Seven Crucial Aspects." eLearning Papers.

Shang, Y., H. Shi and S.-S. Chen (2001). "An intelligent Distributed Environment for Active Learning." Journal on Educational Resources in Computing (JERIC) **1**(2): 308 - 315.

Shen, W., Q. Hao, H. J. Yoon and D. H. Norrie (2006). "Applications of Agent-Based Systems in Intelligent Manufacturing: An Updated Review." Advanced Engineering Informatics **20**(4): 415-431.

Shoham, Y. and K. Leyton-Brown (2008). Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations, Cambridge University Press.

Shute, V. J. and J. Paotica (1994). Intelligent Tutoring Systems: Past, Present, and Future, D. Jonassen (Ed). Handbook on Research on Educational Communications and Technology.

Siemens, G. (2005). "Connectivism: A learning theory for the digital age." International Journal of Instructional Technology and Distance Learning **2**(1): 3-10.

Silveira, R. and R. Vicari (2002). Developing Distributed Intelligent Learning Environment with JADE — Java Agents for Distance Education Framework. Proceedings of the 6th International Conference in Intelligent Tutoring System. Biarritz, France and San Sebastian, Spain, Springer Berlin / Heidelberg. **2363:** 105-118.

Silveira, R. A. and R. M. Vicari (2002). "Developing Distributed Intellient Learning Environment with JADE - Java Agents for Distance Education Framework." Springer: 105-118.

Soh, L.-K. (2006). Incorporating an Intelligent Tutoring System into CS1. Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '06). New York, USA, ACM **38:** 486-490.

Specht, M. (2000). ACE-Adaptive Courseware Environment. Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems. Trento, Italy, Springer Berlin / Heidelberg. **1892:** 380-383.

Suguri, H., E. Kodama and M. Miyazaki (2003). Assuring Interoperability in Heterogeneous, Autonomous and Decentralized Multi-Agent Systems. Proceedings of the The 6th International Symposium on Autonomous Decentralized Systems (ISADS'03). Washington, DC, USA, IEEE Computer Society**:** 17.

Sun, S., M. Joy and N. Griffiths (2005). An Agent-Based Approach to Dynamic Adaptive Learning. Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS). Utrecht, Netherlands: 12-29.

Syed, A. and S. McRoy (2000). What is an Intelligent Tutoring System? Intelligence. York, NY, USA. **11:** 15-17.

Tatara, E., A. Çınar and F. Teymour (2007). "Control of complex distributed systems with distributed intelligent agents." Journal of Process Control **17**(5): 415-427.

Tyler, T. R. (2000). "Social justice: Outcome and Procedure." International Journal of Psychology **35**(2): 117-125.

Uckelman, J. and U. Endriss (2008). Winner Determination in Combinatorial Auctions with Logic-Based Bidding Languages. Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems. Estoril, Portugal. **3:** 1617-1620.

Ugurlu, S. and N. Erdogan (2000). An Agent-Based Information Retrieval System. Advances in Information Systems. T. Yakhno. Turkey, Springer Berlin / Heidelberg. **1909:** 429-436.

Urban-Lurain, M. (1996). "Intelligent Tutoring Systems: An Historic Review in the Context of the Development of Artificial Intelligence and Educational Psychology." Retrieved 09/04, 2012, from http://www.cse.msu.edu/rgroups/cse101/ITS/its.htm.

Wagstaff, G. (1994). "Equity, Equality, and Need: Three Principles of Justice or One? An Analysis of "Equity as Desert"." Current Psychology **13**(2): 138-152.

Wang, J., Y.-H. Sun, Z.-P. Fan and Y. Liu (2005). A Collaborative E-learning System Based on Multi-agent. Internet and Network Economics. X. Deng and Y. Ye, Springer Berlin / Heidelberg. **3828:** 455-463.

Weiss, G. (1999). Multiagent Systems : A Modern Approach to Distributed Artificial Intelligence. Cambridge, MA, USA, MIT Press.

Weller, D. (1985). "Fair Division of a Measurable Space." Journal of Mathematical Economics **14**(1): 5-17.

White, S. and H. Davis (2011). "Making it rich and personal: crafting an institutional personal learning environment." International Journal of Virtual and Personal Learning Environments **2**(3).

White, S. and H. Davis (2011). Rich and personal revisited: translating ambitions for an institutional personal learning environment into a reality. Proceedings of the The PLE Conference 2011. Southampton, UK**:** 1-12.

Wierzbicki, A. (2010). Trust and Fairness in Open, Distributed Systems. New York, Springer-Verlag Berlin Heidelberg.

Wilson, S., O. Liber, M. Johnson, P. Beauvoir, P. Sharples and C. Milligan (2007). "Personal Learning Environments: Challenging the Dominant Design of Educational Systems." Journal of e-Learning and Knowledge Society **2**(3): 173-182.

Wooldridge, M. and N. R. Jennings (1995). "Intelligent Agents: Theory and Practice." Knowledge Engineering Review **10**(2): 115-152.

Wooldridge, M. J. (2009). An introduction to multiagent systems. Chichester. UK, John Wiley & Sons.

Xu, D. and H. Wang (2006). "Intelligent Agent Supported Personalization for Virtual Learning Environments." Decision Support Systems **42**(2): 825-843.

Zhao, G., J. Deng and W. Shen (2001). "CLOVER: An Agent-Based Approach to Systems Interoperability in Cooperative Design Systems." Computers in Industry **45**(3): 261-276.

Zuckerman, M., P. Faliszewski, V. Conitzer and J. Rosenschein (2011). An NTU Cooperative Game Theoretic View of Manipulating Elections Internet and Network Economics. N. Chen, E. Elkind and E. Koutsoupias, Springer. **7090:** 363-374.

Appendix A

A.1 Identical Voting Strategies

Table A-1, Table A-2 and Table A-3 display the statistical result for case 2 with 18 running courses.

Table A-1: Statistical values of comparison between intelligent strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	58.770	0.417	0.173
Intelligent	58.626	0.431	0.068

t = 1.697 df =98 p =0.092

Table A-2: Statistical values of comparison between proportional strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	58.770	0.417	00.173
Proportional	58.759	0.419	0.175

t = 0.131 df = 98 p = 0.895

Table A-3: Statistical values of comparison between equal-share strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	58.770	0.417	0.173
Equal-share	52.137	0.798	0.636

t = 52.091 df = 98 p = $\mathbf{3.052 \times 10^{-73}}$

Table A-4, Table A-5 and Table A-6 display the statistical result for case 3 with 9 running courses.

Table A-4: Statistical values of comparison between intelligent strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	57.291	0.994	0.988
Intelligent	57.096	1.084	1.175

t = 0.937 df =98 p =0.350

Table A-5: Statistical values of comparison between proportional strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	57.291	0.994	0.988
Proportional	57.252	1.025	1.050

t = 0.193 df = 98 p = 0.847

Table A-6: Statistical values of comparison between equal-share strategy and optimal

strategy	*mean*	*SD*	*variance*
Optimal	57.291	0.994	0.988
Equal-share	50.221	1.445	2.088

t = 28.504 df = 98 p = $\mathbf{3.127 \times 10^{-49}}$

A.2 Combination of Voting Strategies

Figure A-1 and Figure A-2 show the result for case 2 where a group of students are allocated to one strategy, while the rest are allocated to another strategy. The number of running courses was set 18.

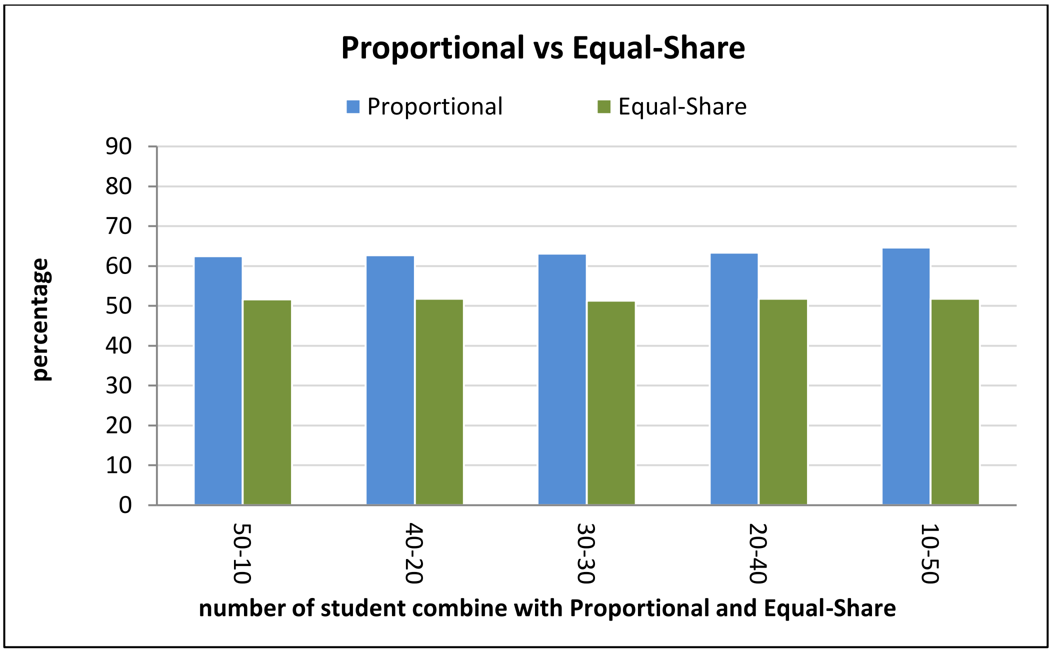


Figure A-1: Case 2: Proportional vs. Equal Share

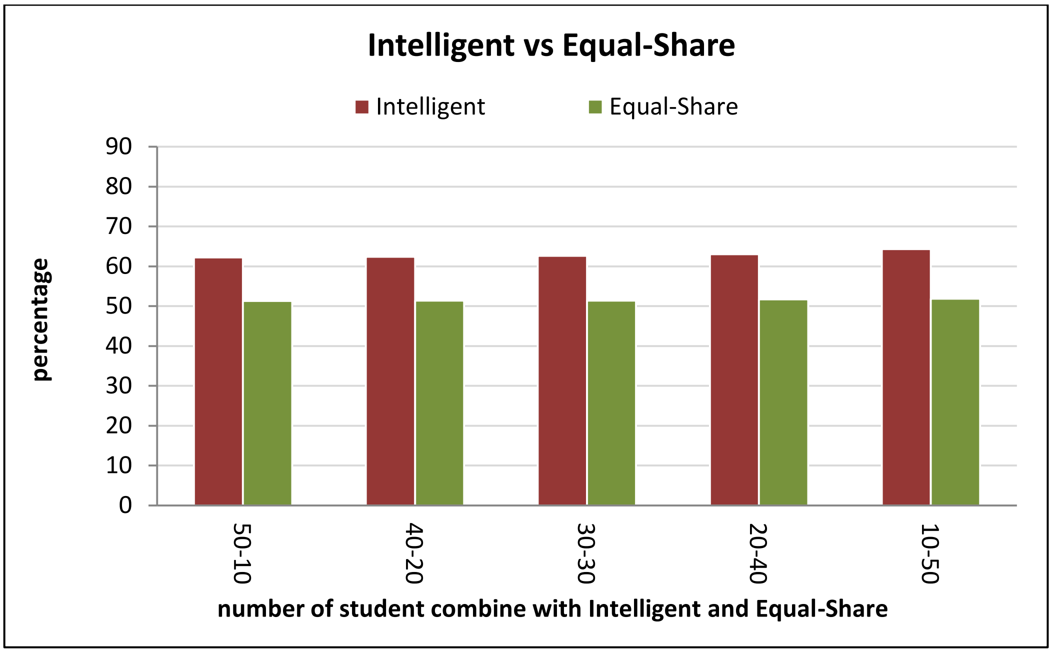


Figure A-2: Case 2: Intelligent vs. Equal Share

Figure A-3, Figure A-4 and Figure A-5 show the result for case 3 where a group of students are allocated to one strategy, while the rest are allocated to another strategy. The number of running courses was set 9.

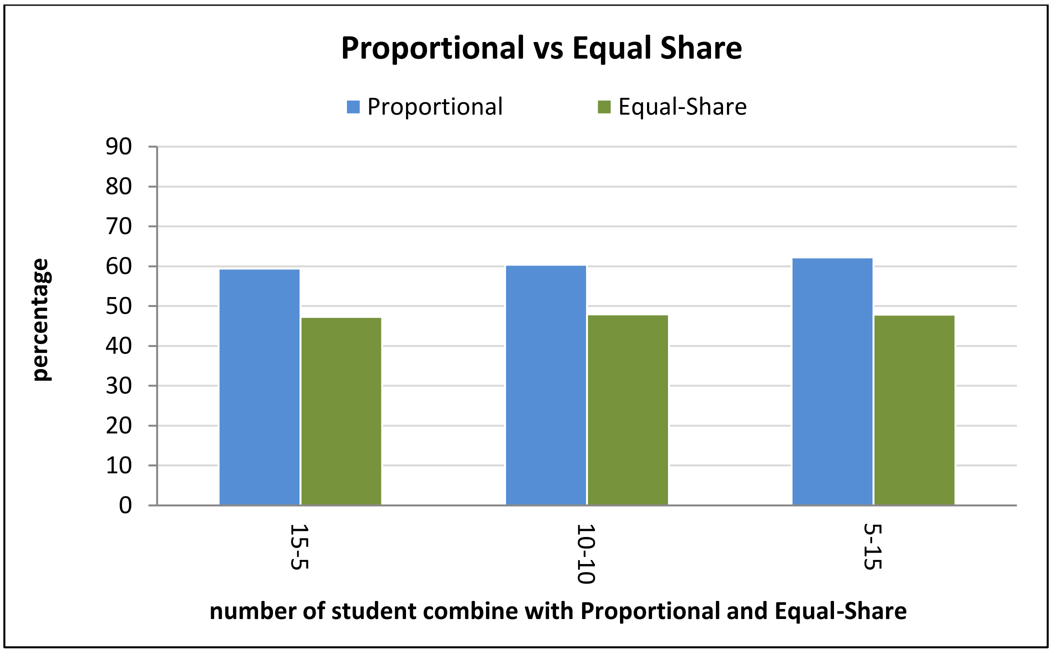


Figure A-3: Case 3: Proportional vs. Equal Share

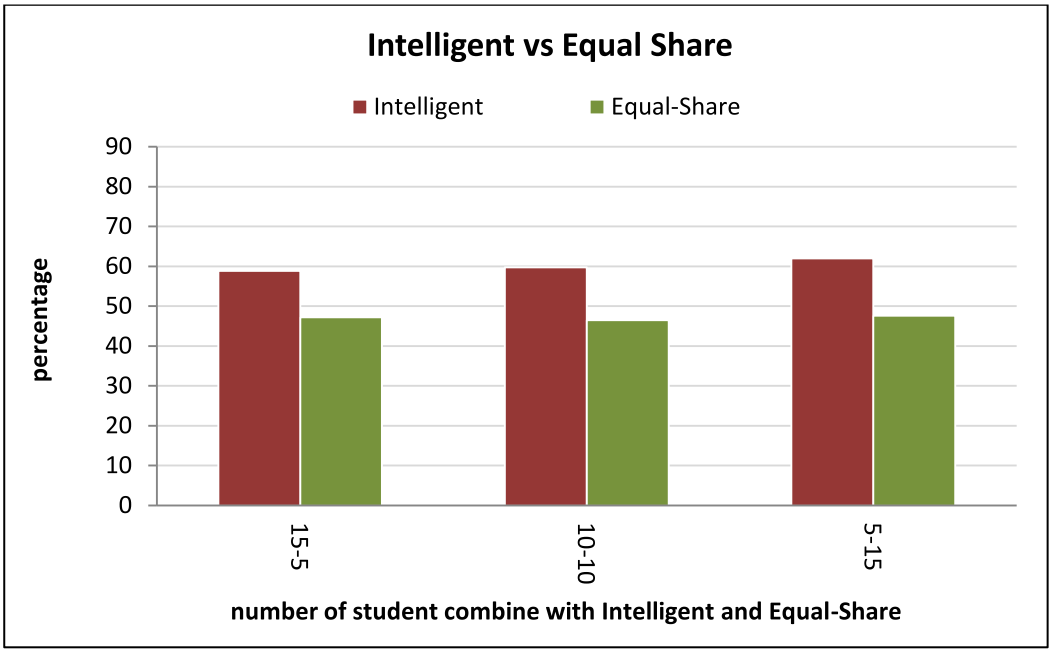


Figure A-4: Case 3: Intelligent vs. Equal Share

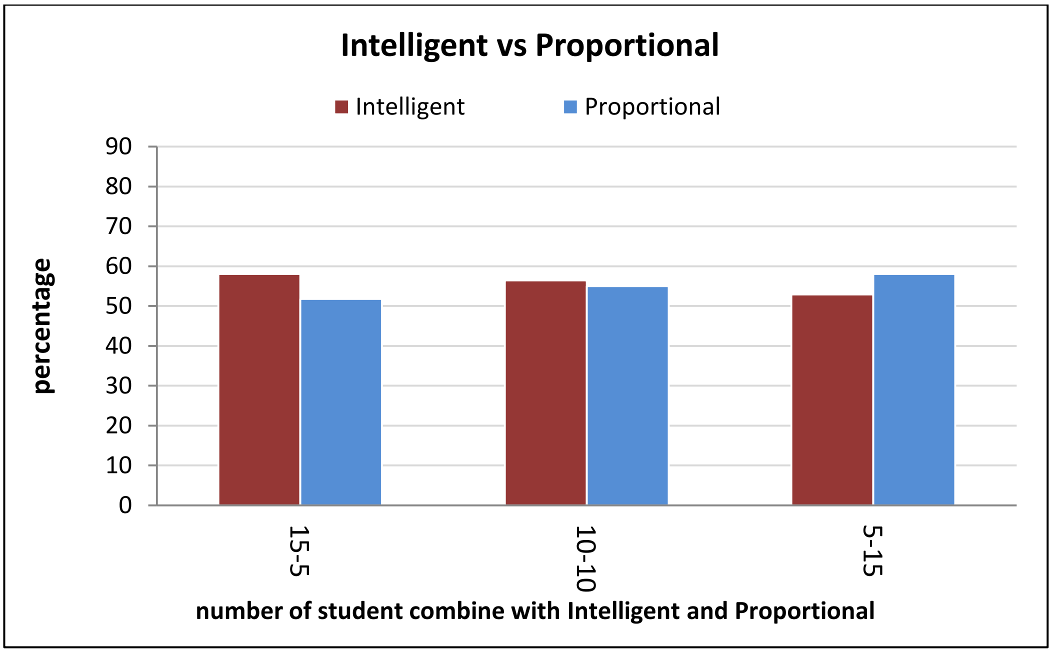


Figure A-5: Case 3: Intelligent vs. Proportional

Appendix B

Figure B-1 and Figure B-2 show the student satisfaction for group of agents using a particular strategy, as well as the overall average of student satisfaction for case 1 where proportion of students using a particular strategy for a different percentage of the applied rules. The number of running courses was set 40.

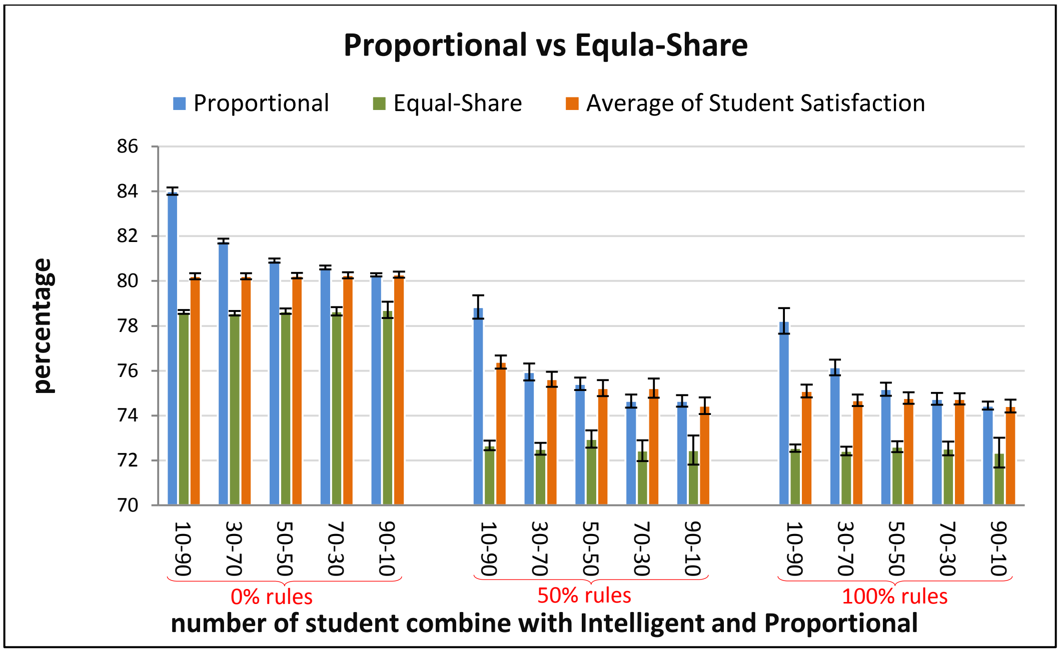


Figure B-1: Case 1: Proportional vs. Equal Share

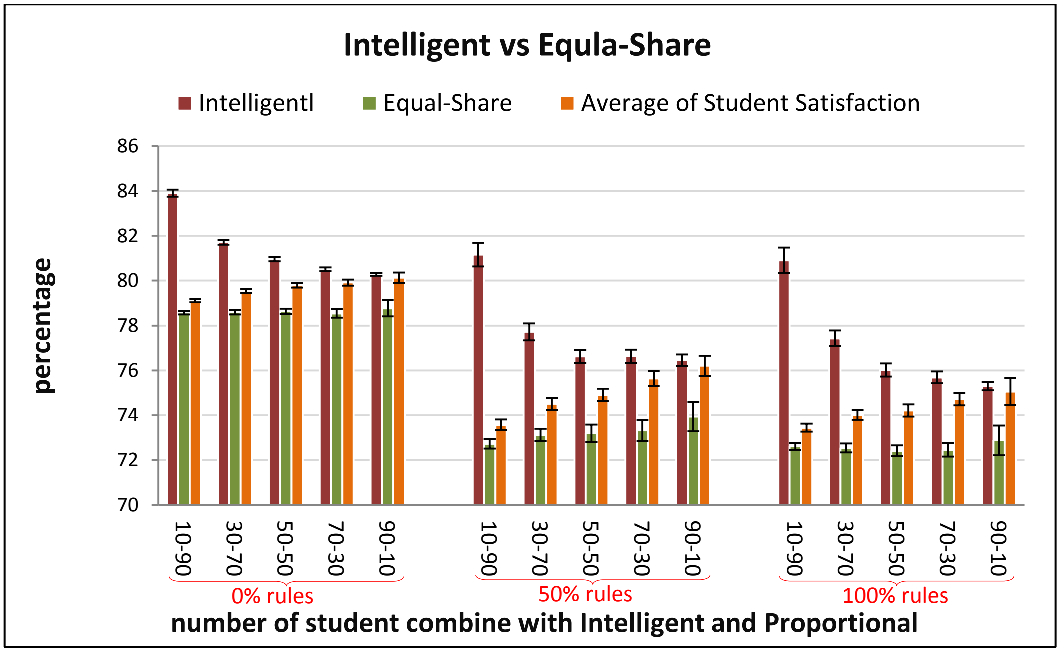


Figure B-2: Case 1: Intelligent vs. Equal Share

Figure B-3 and Figure B-4 show the student satisfaction for group of agents using a particular strategy, as well as the overall average of student satisfaction for case 1 where proportion of students using a particular strategy for a different percentage of the applied rules. The number of running courses was set 9.

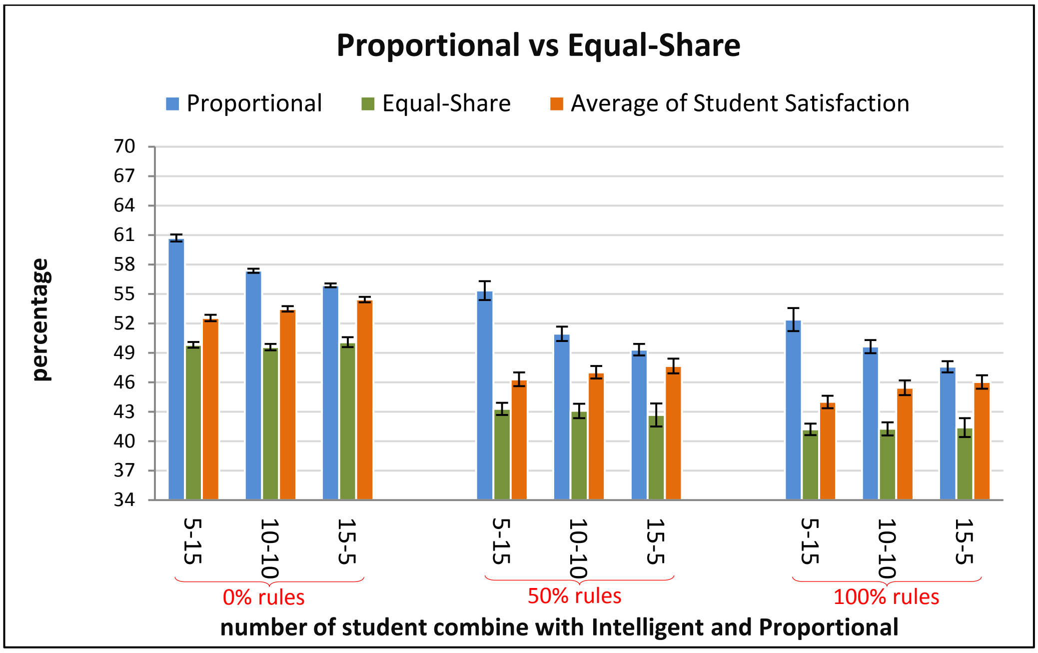


Figure B-3: Case 3: Proportional vs. Equal Share

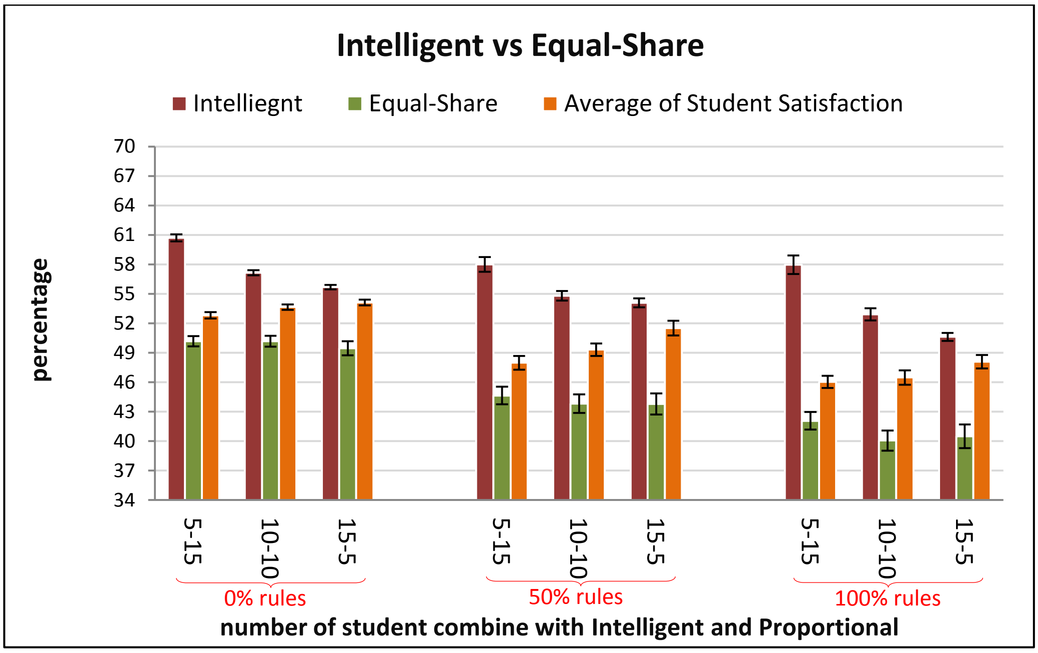


Figure B-4: Case 3: Intelligent vs. Equal Share

Appendix C

Figure C-1, Figure C-2 and Figure C-3 how the mean, median and range for utilitarian, egalitarian and equalitarian respectively as measures for fairness for the case3 when mixing equal-share and proportional.

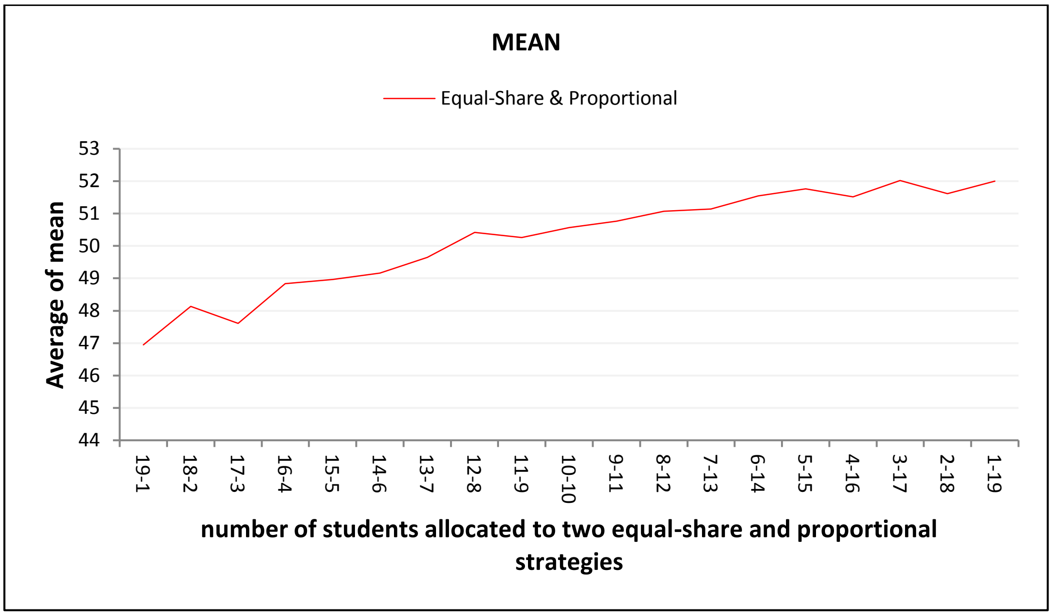


Figure C-1: Case 3: mean when Equal-Share and Proportional

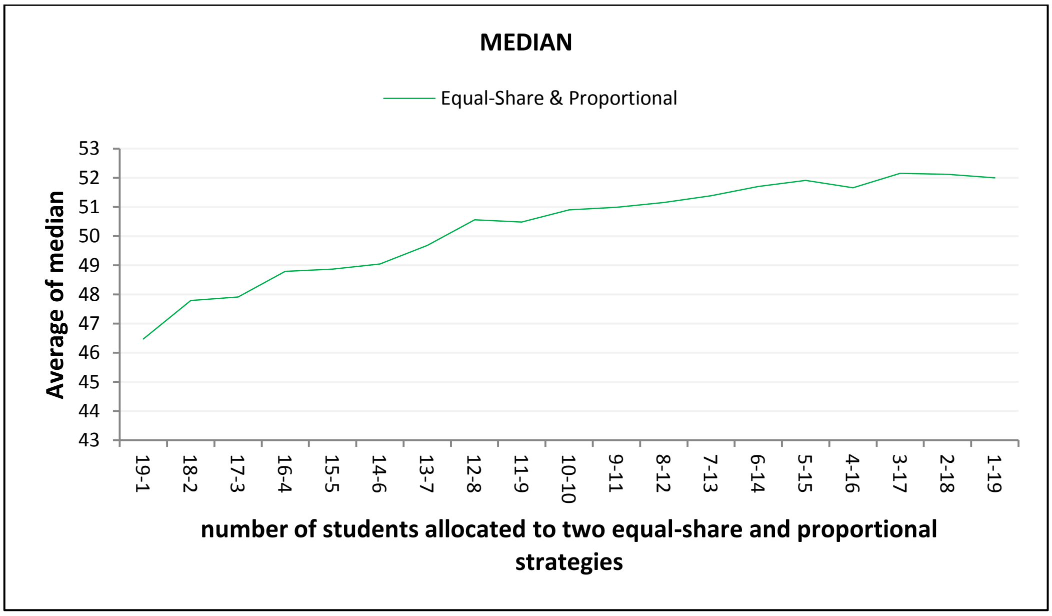


Figure C-2: Case 3:median when Equal-Share and Proportional

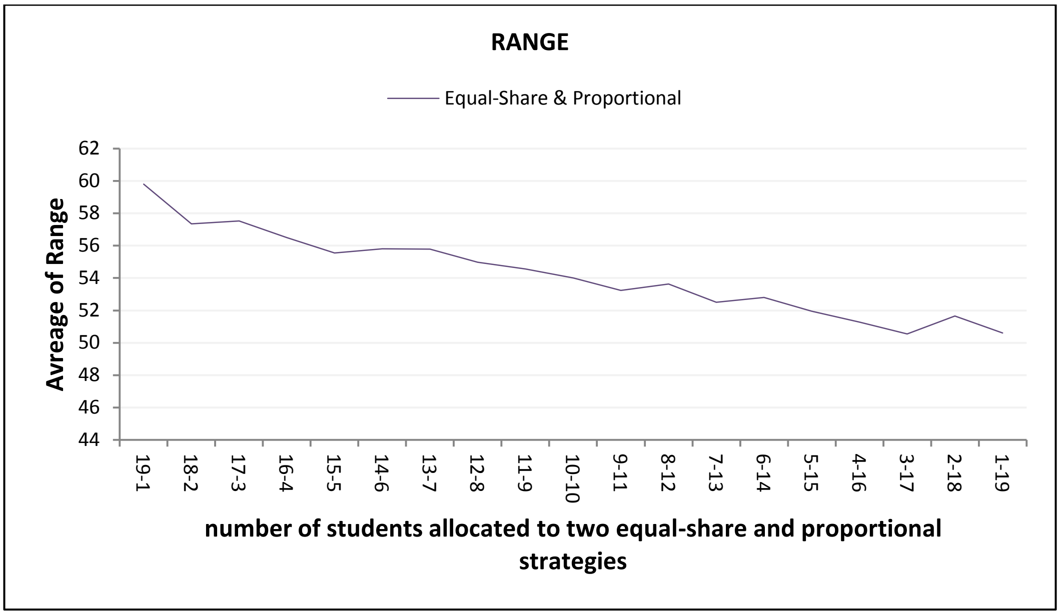


Figure C-3: Case3: range when Equal-Share and Proportional

Figure C-4, Figure C-5 and Figure C-6 show the mean, median and range for utilitarian, egalitarian and equalitarian respectively as measures for fairness for the case 3 when mixing proportional and intelligent.

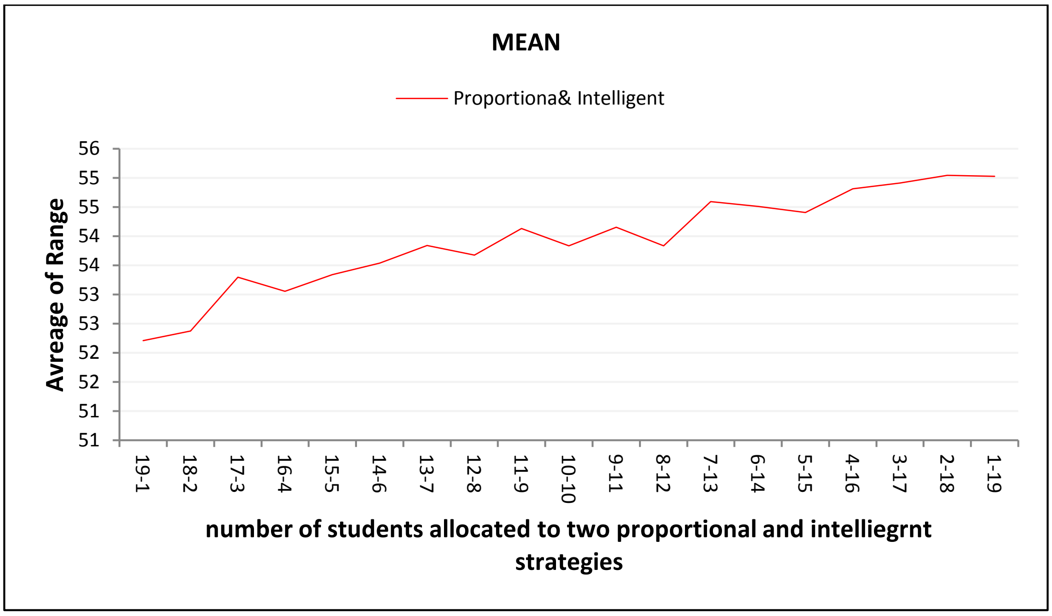


Figure C-4: Case3: mean when Proportional and Intelligent

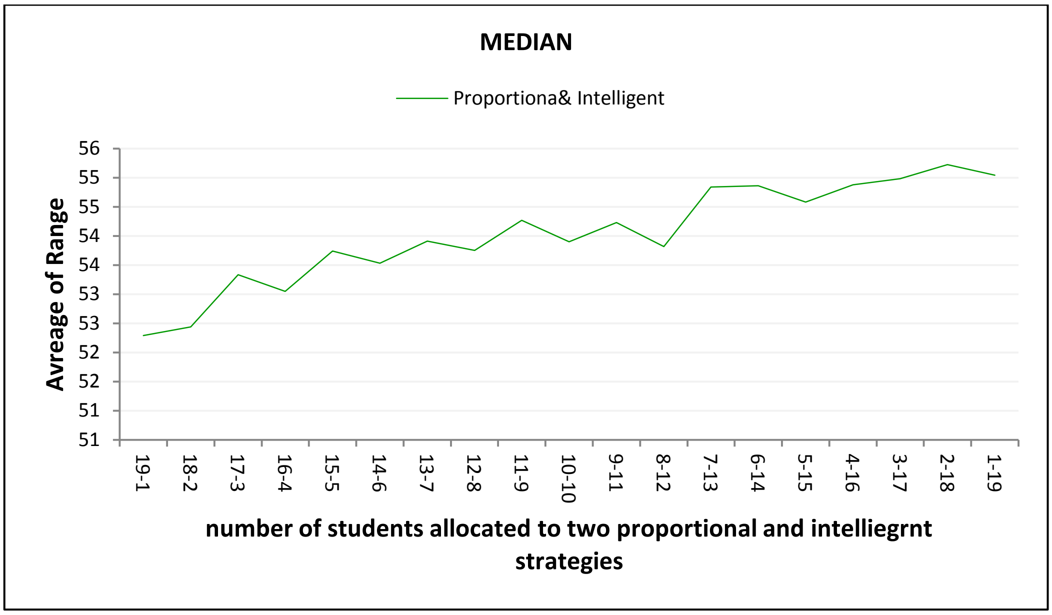


Figure C-5: Case3: median when Proportional and Intelligent

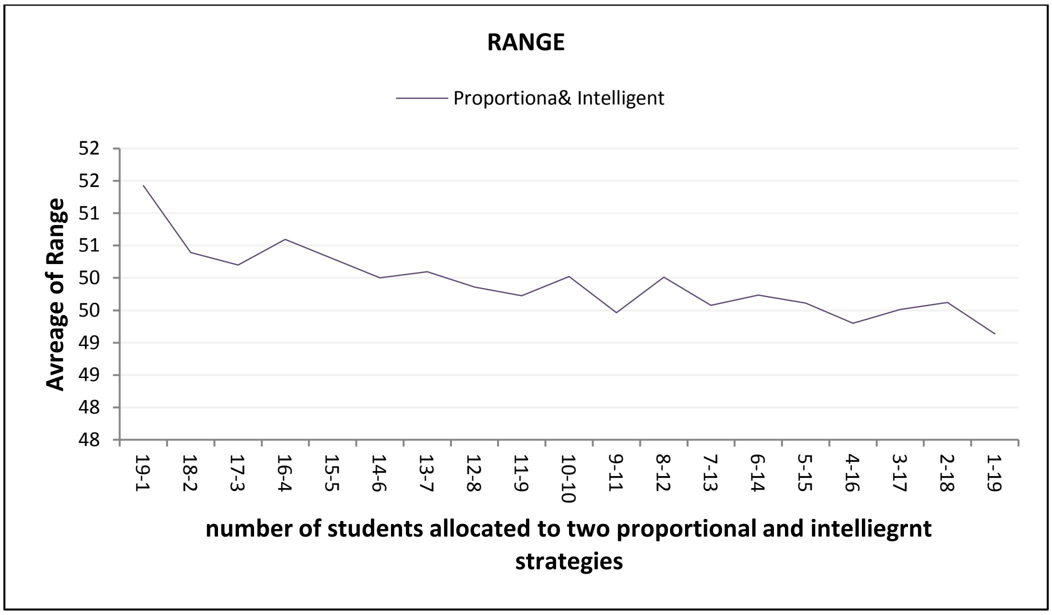


Figure C-6: Case3: range when Proportional and Intelligent