Bar Ilan University

Computer Science Department

The Adaptive Multi-personality Agent

Shavit Talman

Advisor: Prof. Sarit Kraus

Submitted as partial fulfillment of the requirements for the Master's degree

in the Department of Computer Science, Bar Ilan University

Ramat Gan, Israel 2004

Acknowledgments

First and foremost, I would like to thank my advisor Prof. Sarit Kraus. The long meetings, the patient assistance and the unwavered support she provided me, not only contributed greatly to the success of this research but also enriched me substantially. Thank you.

To Meirav Hadad - thank you for always being there, being willing to direct me toward the right places and providing feedback and reactions unfailingly.

Abstract

Negotiating agents are an increasing factor in large-scale multi agents systems today. Thus, the need to design a high-performance agent, able to interact with its surrounding in order to achieve its goals, is in demand. It is especially beneficial to design an agent able to perform well in all environments - a cooperative environment where all agents work together to achieve a joint goal (for example RoboCup), a competitive environment where each agent has its own set of goals and competes with the other agents in the system (for example auctions) or an intermediate environment where agents have their own set of goals but also share a joint one (for example branches of the same company working to maximize their own income, while keeping the company interests in mind as well). Throughout the years many models of agents have been developed, most of which have been designated to act in a specific type of environment, either a cooperative or a competitive environment. Furthermore, several techniques for dealing with agents that compromise the foundations of the cooperative/competitive environment were developed, in order to increase the agents' gain and/or protect them from exploiters.

Our solution is to design an Adaptive Multi-personality agent that consists of a set of sub-agents. Each sub-agent is in charge of interacting and negotiating with one of the other agents coexisting in its surrounding. By modeling agents it interacts with, and by learning which sub-agent best-suits each agent, the Adaptive Multi-personality agent can interact with other agents in an optimal manner. As a result, the Adaptive Multi-personality agent is able to perform well in all types of environments, cooperative, competitive and intermediate alike. Moreover, it copes well with different strategies agents deploy - it doesn't yield to exploiters while taking advantage of the selfless.

The first part of this thesis describes the Adaptive Multi-personality agent: its design and motivation, its different modules, its special instance as an adaptive one-personality agent and our hypotheses regarding its performance.

The second part introduces the domain in which we would evaluate the Adaptive Multi-personality performance. This domain is the Colored Trails game, a complex game with numerous parameters, which allows us to conduct a large number of different experiments. In this part we also discuss the additional agents we have at our disposal, which were designed by other designers, and will be used to evaluate the Adaptive Multi-personality performance as well.

The last part presents the experiments we executed, some of which were precursory experiments, which "trained" the agent, and the others were evaluation experiments. All in all we executed over 58,000 games, which translate into ~3800 hours of computation. In all those experiments the Adaptive Multi-personality agent proved to be significantly better than its adaptive one-personality instance and the additional agents alike, and reached higher scores.

Contents

1 Introduction

In the area of multi-agent systems, the need for more adaptive intelligent agents arises frequently. In these types of systems agents have to achieve their goals by interacting with other agents in the system. Moreover, usually they have to negotiate with one another, either on resources they possess or skills they maintain, so that their goals are met. In order to cope with this constraint several models were proposed, some of which are designated for completely cooperative environments (see for example [11]), and some for completely competitive environments (see for example [2]). The agents have been built to be rational and able to adapt to the changing environment.

When there are other agents in the environment, which deploy multiple unknown strategies for interacting, and/or there is incomplete information about their goals and capabilities (their skills, their resources, inter alia), the problem becomes more difficult. For instance - some environments are more cooperative while others are more competitive, in the sense that the agents that operate within those environments have a more cooperative or a more competitive interacting behavior. As a result, an agent acting in that environment has to match his decision-making mechanism to the environment's "atmosphere". In particular, in a competitive environment, filled with agents whose interacting behavior aims to exploit their opponent agents, a generous behavior will not be beneficial. Such an agent would probably be exploited - either it would be exhausted of its resources or it would be given tasks that take advantage of its skill and occupy it completely. Skills can be reduced to resources by associating a skill with each resource; possession of a certain resource corresponds to having the skill. In view of this reduction, in the remaining sections of this thesis, we will assume agents have resources alone.

For every opponent agent, the Adaptive Multi-personality agent chooses the most suitable personality (or behavior) to interact with that opponent agent from a variety of possible personalities. Our agent works in an environment of non-anonymous agents, that is, each agent is recognizable by a unique ID number. Nevertheless, the agents' strategies of acting and interacting are unknown as well as their resources. The motivation is to learn the other agents' decision-making mechanism (or personality) and then to match each one of them a personality that suits them most, in order to maximize our agent's gain.

This notion greatly differs from the solutions currently existing in the multi-agents systems (some of which will be discussed in length in section 7.1). Although their agents adapt to their opponents, they do not change their very own personalities in the process. Their solution can be demonstrated with a real life situation - suppose a tour guide in a museum takes several groups to a tour around the museum, and suppose his salary depends on their enjoyment. Their agent is likened to a guide that knows how to approach each of the different groups. This is usually done by guiding each group in a different way, by adapting to their different compositions. Nevertheless, this adaptation can be done only up to a certain level - if that guide can not keep a straight face, a group of serious historians is bound to get irritated with him, no matter how hard he tries to adapt. In contrast, our agent is comparable to a group of entities, each maintaining a different personality. When the group has to interact with the outer world, it studies the characterizations of the outer beings, and confronts each with its most suitable entity simultaneously. If we return to the guided tour service example: Here the Adaptive Multi-personality agent is likened to the service manager, and its personalities are the different tour guides in the museum - their goal is to maximize the museum's income by providing extraordinary guides. The manager receives simultaneous requests for guided tours in the museum. He should learn the composition of the different groups, and match a guide that is most likely to make them take interest in the museum. For instance, a group of young school children should be guided by a person who is able to interact with children, and knows how to keep them entertained, whereas a group of university art students should be guided by a person who understands art, and is able to draw their attention to different art techniques. Each of these guides will further adapt to the specific group he leads. We believe that by adopting a different personality during the interactions with every opponent agent, our agent could succeed in the same way and benefit the most from the other agents.

Moreover, in order to know which personality to adopt while interacting with a specific opponent agent, that opponent agent's interacting strategy must be comprehended. To this end we implemented a simple learning module, which enables the Adaptive Multi-personality agent to comprehend those strategies, and to assign the best-suited personality to each opponent. As a result, the Adaptive Multi-personality agent can be used in any environment, cooperative, competitive and intermediate alike, as long as the environment allows negotiations. It simply changes the matching of its personalities to the opponent agents coexisting in the domain.

The following section describes the Adaptive Multi-personality agent in detail, and section 3 describes the design of one of its opponent agents (AOP agents), created to evaluate its performance. Section 4 describes the test-bed we chose to evaluate the Adaptive Multi-personality agent's performance. It is the Colored Game domain - a new computer game described in [9] (a detailed description of the game is given in section 4.1). In that section we also present the design of additional agents (PDAs) able to act in the CT domain. They were designed by other designers and deploy different playing strategies. Section 5 describes all the precursory experiments, including the experiments that tested the correctness of the Adaptive Multi-personality's learning module, and the matching experiments, which determined which personality of the Adaptive Multi-personality agent best-suits each opponent type. Section

6 depicts the evaluation experiments of the Adaptive Multi-personality agent, by comparing it to both the performance of the AOP agents and the performance of the PDAs. Section 7 describes the existing solutions to designing an automated agent, as well as to agent modeling and decision strategies. Lastly, section 8 summarizes and suggests several possibilities to continue this research.

2 The Adaptive Multi-personality agent

2.1 Environment assumptions

In this work we focus on environments in which there is a set of automated agents acting dynamically and using negotiations in order to maximize their utility. Each agent may have different capabilities and its own goals. In addition, it has a set of resources to use as it finds useful. Each agent may be able to achieve its goals unassisted or may be dependent on the help of the other agents in the environment in order to do so. The assistance required from other agents is mainly their resources, but also information in cases where there is incomplete information on several aspects of the domain. We also assume that each agent operating in the environment receives a score at the end of its life corresponding to its performance in achieving its goals (its utility). The score is determined by a scoring rule, which is provided to the agents upon their creation. As a result, the purpose of the agents is to maximize their score.

We assume that the activity within the environment may be divided into phases: either the environment consists of consecutive phases, as in many computer games, or it allows the agents in it to artificially divide their lifespans into phases, as in some real-time systems. Another constraint is that the domains we consider have finite horizons. This constraint is compatible with most of the multi-agents classical problems, and forces the agents to act instead of remaining idle for long periods of time.

2.2 Personality

2.2.1 The Five Factor Model

The definition of personality in the Oxford dictionary is "the complex of all the attributes - behavioral, temperamental, emotional and mental, that characterize a unique individual". Personality theorists, in the psychology domain, have strived for decades to define those basic dimensions of personality, which create the differences between people.

The approach that was adopted by the AI researchers (for example [1, 13, 6]) is the one defined by Norman [14], namely the *Five Factor Model* (FFM). This model suggests that there are five major factors in the analysis of personality: Extraversion (E), Agreeableness (A),

Positive Pole	Nature of Factor	Negative Pole
	EXTRAVERSION (E)	
talkative, optimistic, sociable, affectionate	Capacity for joy, need for stimulation	unartistic, conventional
	AGREEABLENESS (A)	
good-natured, trusting, helpful	One's orientation along a continuum from compassion to antagonism unreliable, lazy, careless, negligent	rude, uncooperative, irritable
	CONSCIENTIOUSNESS (C)	
organized, reliable, neat, ambitious	Individual has degree of organization, persistence, and motivation unreliable, lazy, careless, negligent	unreliable, lazy, careless, negligent
	NEUROTICISM (N)	
worrying, insecure, emotional, nervous	Proneness to psychological distress, excessive carvings or urges, unrealistic ideas	calm, secure, unemotional, relaxed
	OPENNESS (O)	
creative, original, curious, imaginative	Toleration for & exploration of the unfamiliar	unartistic, conventional

Table 1: The Five Factors of FFM

Conscientiousness (C), Neuroticism (N) and Openness (O). Each factor can be tested and be assigned a number between 0 and +1. Table 1 lists a short description of the factors and definers of both negative (closer to zero) and positive (closer to one) poles of each of them.

2.2.2 Personality of an automated agent

The agents in our domain apply negotiations to maximize their utility. Usually they have a certain task to carry out or a set of goals to achieve and they are given a score which corresponds to their performance. They try to achieve it by using their own resources and / or attaining required resources by negotiating with other agents in their domain. Factors A and C are the only two factors that affect the automated negotiation domain (the other factors are more related to emotional behavior). So, in our research the agent's personality will be defined by assigning two values to these factors: a higher value for factor A means that the agent is more cooperative, and a higher value for factor C means it is more reliable. For simplicity, we will refer to factor A as Cooperation and to factor C as Reliability. Moreover, we measure these factors by the following definitions:

- *Reliability* is measured by the truthfulness of the agent, particularly, its willingness to keep its commitments. A reliable agent is one that keeps almost all of its commitments (we allow a small degree of untruthfulness for reasons of force majeure).

- *Cooperation* is measured by the willingness of the agent to share its resources with others. In [9] we defined four types of cooperation-related behaviors: (1) the *reciprocal* agent - one that is involved in the exchange of resources (2) the *give* agent - one that only sends resources to others (3) the *take* agent - one that only receives resources from others (4) the *idle* agent - one the neither sends nor receives resources at all.
  A cooperative agent is an agent that is more *reciprocal* and *give* than *take* and *idle*.

We will divide the range of the factors' values into three categories: low, medium and high. For simplicity, we will use L for low, M for medium and H for high. Moreover, a personality will consist of two letters, the first one is the cooperation level and the second is the reliability level. For instance, *MH* is medium cooperation - a highly reliable personality.

If we consider all possible combinations, nine different personalities can be defined from these two factors. However, an agent with a low reliability factor entails that the agent is of type "take", since it never keeps its commitments. That is, the agent has a low cooperation level as well. Thus, personalities ML and HL can not be created. Consequently, there are seven different personalities altogether - {LL, LM, LH, MM, MH, HM, HH}. These personalities are not rigid, in the sense that they may and will adapt to their surroundings, within their personality constraints.

All agents in the domain, both our agent and the opponent agents, can be viewed as having a personality - their actions can be easily characterized based on the cooperation and reliability factors' categories. As a result, one can try to estimate another agent's personality by observing its behavior and recording their past interactions. Incidentally, these personality estimations may change over time, as the agent collects more data on its opponent. In the remainder of this thesis we will use *"type"* to refer to an opponent agent's personality (true personality or estimated one) in order to distinguish it from the Adaptive Multi-personality agent's different *"personalities"*, but the meaning is identical.

2.3 Supporting definitions and notations

We will use the following terms throughout this thesis:

- **Agent modeling** - the process of estimating the opponent's traits and/or strategy from observations.

- **Adaptiveness** - the ability of an agent to change its behavior due to changes it observed in its surrounding.

- **Exchange commitment** - the process of promising to exchange specific resources with a certain agent in the domain. Usually the commitment will specify which resources are expected in return, if any. If two agents commit to a certain exchange, from their point of view they are bound to it.

- **Deceitful behavior** - an agent's strategy that involves untruthful behavior. In our domains this behavior involves committing to resources exchange and not keeping commitments.
- **Exploiting behavior -** a behavior that contains a deceitful behavior, with the addition of resources accumulation. An agent possessing this kind of behavior will try to attain as many resources as possible in its commitments, and, almost always, will not keep those commitments.
- **Opponent agent** - all other agents in the domain. Although those agents may not compete with a specific agent in some domains, in order to distinguish between that agent and every other agent, we will address them as opponents.
- **Exchange benefit** - the potential gain of a specific commitment. This will be calculated by using the scoring rule and some subjective amounts (for instance reputation). If only the scoring rule is used, than the agent deploying it is being rational.
- **Exchange risk** - the potential risk of a specific commitment. This will be calculated by using the scoring rule and some subjective amounts (for instance reputation).
- **Task dependency** - arises whenever agents lack the resources they need to achieve their goals or to perform their tasks, and must depend on other agents supplying those resources. The term *task independence* reflects the opposite situation, when the agents have all the required resources in order to achieve their goals or to perform their tasks.

2.4 Agent design

The Adaptive Multi-personality agent uses negotiations in order to maximize its score. When it does not have all the resources required to achieve its goals, it exchanges resources with other agents in the domain - exchanging its own redundant resources with required ones from the other agents. It has multiple personality-driven negotiation modules (PDNM) within itself - it interacts with each of the other agents (opponents) using one of its personality-driven negotiation modules. Thus, if it has one opponent agent it creates one PDNM, and if it has N opponent agents it creates N PDNMs. This enables the agent to maintain approximately parallel negotiations with all the opponent agents coexisting in the domain. It decides which personality to assign each of its PDNMs by consulting a "Matching Table", which holds for each opponent type the personality that best-suits it, if available (the table was built empirically as described in section 5.2; a tit-for-tat exemplary table is presented in table 2). If the opponent agent's type is unknown it assigns a default personality to it.

While interacting with other agents, the Adaptive Multi-personality agent models the types of its opponents. That is, it assigns updated estimations of their cooperation and reliability factors (see more on opponent modeling in section 2.4.2). Once the estimated type of a certain opponent has changed, the Adaptive Multi-personality agent consults its Matching table, and matches a PDNM with a better personality to that opponent. To demonstrate a similar notion imagine a customer service of a small company. When a new customer turns to the service, he is modeled or characterized according to different parameters (e.g. his sector - private or business, the assistance he requires - technical support or administrative, to name a few), and his call is directed to the customer service person that is the most qualified to assist him. At first, a "default" customer service person takes his call and models the customer, but once this is done, the best person to assist interacts with the customer. Similarly, the Adaptive Multi-personality agent changes its very own traits upon detecting changes in the modeling of its opponents.

The Adaptive Multi-personality agent can be viewed as a shell consisting of 4 modules:

- Domain Manager module who is in charge of all aspects of the domain apart from the negotiation-related ones: calculating which resources are needed or redundant, executing all actions which are not related to negotiations and so on.
- Learning Handler module who learns the types of the opponent agents.
- Personalities Coordinator module who is in charge of the communication to and from all personalities and their opponents, the distribution of the resources between them, and of updating the personalities-opponent types matching.
- Personality-driven negotiation modules (PDNMs) that are responsible for the interactions with other agents. Each PDNM interacts with a single opponent agent in the domain, and negotiates with it in order to increase the Adaptive Multi-personality agent's performance.

The shell of the Adaptive Multi-personality agent simply directs communication between the Personalities Coordinator to and from the opponent agents, and also between the internal modules. Figure 1 illustrates the hierarchical structure of the Adaptive Multi-personality agent's modules. The following sections review these four modules - the hierarchical realtions between them, and their main responsibilities.

Opponent agent type		Best-suit personality	
Cooperation level	Reliability level	Cooperation level	Reliability level
Low	Low	Low	Low
Low	Medium	Low	Medium
Low	High	Low	High
Medium	Medium	Medium	Medium
Medium	High	Medium	High
High	Medium	High	Medium
High	High	High	High

Table 2: A *Tit-For-Tat* Matching Table

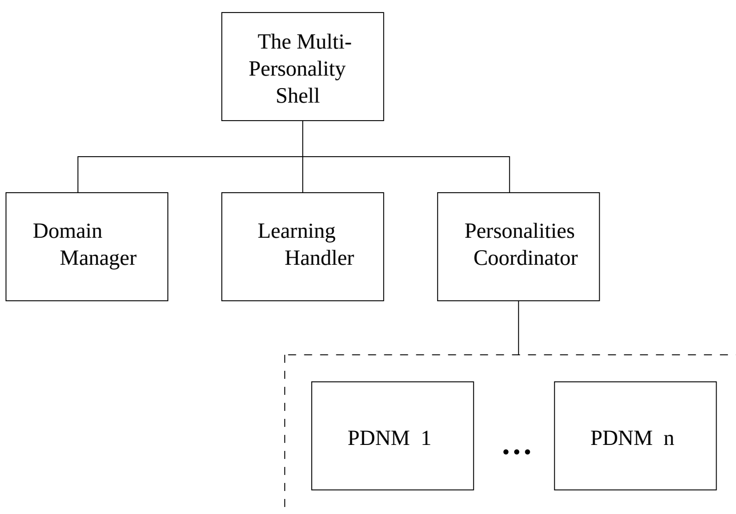

Figure 1: The Multi-personality Structure

2.4.1 Domain manager

This module is responsible for all the domain-dependent operations available in a specific domain, except negotiations. All actions related to negotiations are carried out by the PDNMs. For example - assume the Adaptive Multi-personality agent sells in a wholesale market. Its Domain Manager, thus, is responsible for keeping track of its resources (stock), and for telling its PDNMs which items should be sold, and how much of those items are available at any moment. Its role resembles the one of a quartermaster. In contrast, the negotiations (contacting several / all buyers, determining the price, deciding on barter deals and so forth) are done by the different PDNMs, which can be viewed as salesmen in this context. Incidentally, the Adaptive Multi-personality agent's shell in this scenario should be the General Manager: it activates all other parts and directs the communication between them.

2.4.2 Learning handler

There are two stages of learning in our system:

1. Learning the opponent agents types, while playing against them. This aspect will be discussed in length in this section.

2. Learning which personality best-suits each opponent type. The type-personality pairs were determined empirically, as will be described in section 5.2.

The opponent agents' types, that is their reliability factor and their cooperation factor, are learnt throughout the agents' lives. We assume that the agents' lives are divided into phases, and we allow the Adaptive Multi-personality agent to update the type estimators for each opponent from one phase to the other. Namely, at the end of each phase k, the Learning Handler of the Adaptive Multi-personality agent holds for each opponent j, updated estimators of its reliability factor and cooperation factor. We denote these two quantities at phase k by *reliability*(j,k) and *cooperation*(j,k).

The estimators are calculated as described in the reliability and cooperation definitions mentioned in section 2.2.2, and are based on the works presented in [18] and in [16] (also see section 7.2). These works suggest to establish the type estimation by incorporating the agent's own observations with testimonies received from other opponent agents in the domain (witness agents) about their experiences. Since testimonies may not be reliable, they are adjusted according to the witness agent's reliability estimator. In addition, as we believe agents evolve over time, higher weight is associated with recent observations and testimonies.

Formally, during each phase k, the Learning Handler quantifies its own observations of the type estimations of opponent agent j as follows:

$$reliability\_quantity(j,k) = \frac{\#\,times\,opponent\,j\,kept\,commitments\,at\,phase\,k}{\#\,of\,commitments\,made\,by\,opponent\,j\,at\,phase\,k} \quad \in [0,1] \tag{1}$$

$$cooperation\_quantity(j,k) = \begin{cases} 1 & if\,opponent\,j\,was\,reciprocal\,or\,give\,at\,phase\,k \\ 0 & otherwise \end{cases} \tag{2}$$

Suppose there are N opponent agents in the domain, each may give testimony on the type of other opponent agents. Then, the Learning module summerizes up all testimonies, but weighs each testimony with the reliability estimator of the witness agent -

$$test\_reliability\_quantity(j,k) = \frac{1}{N}\sum_{t=1}^{N}(reliability\_testimony(t,j,k) \cdot reliability(t,k-1)) \tag{3}$$

$$test\_cooperation\_quantity(j,k) = \frac{1}{N}\sum_{t=1}^{N}(cooperation\_testimony(t,j,k) \cdot reliability(t,k-1)) \tag{4}$$

Then, the Learning Handler summarize these quantities into the two updated estimators of the phase k (we will focus on the reliability estimator since the cooperation one is similar)-

$$reliability(j,k) = p \cdot observation\_value(j,k) + (1-p) \cdot testimony\_value(j,k) \tag{5}$$

where,

p- is the importance of the subjective observation value,

$$observation\_value(j,k) = \frac{1}{k}\sum_{l=1}^{k}(\delta \cdot reliability\_quantity(j,l)) \tag{6}$$

- δ is the time discount factor

$$testimony\_value(j,k) = \frac{1}{k}\sum_{l=1}^{k}(\delta \cdot test\_reliability\_quantity(j,l)) \tag{7}$$

2.4.3 Personalities coordinator

The personalities coordinator is in charge of -

1. Creating all the PDNMs and monitoring the suitability of their personalities to their

opponents. When a change occurs in the type estimators of an opponent agent, it is the Coordinator's role to refresh its PDNMs, that is, to create new PDNMS with a more suitable personality and initialize this factor.

2. Directing communication between the PDNMs and their opponent agents. It serves as their "gate" to communicating with all opponent agents. When the Adaptive Multi-personality agent receives a message, it is sent to the Personalities Coordinator, who passes it along to the proper PDNM. Similarly, whenever a PDNM wishes to send a message to an opponent agent, the message is collected by the Coordinator, which serves as the body that sends it to the target opponent.

3. Propagating all knowledge gathered from all other modules, including: initial data related to the domain (e.g. number of opponents, number and type of initial resources, to name a few), the opponents' types estimated by the Learning Handler, needed and redundant resources calculated by the Domain Manager, as well as the current domain state.

The resources propagation responsibility of the Personalities Coordinator is quite problematic: there is only one set of resources but several PDNMs that wish to use it for their needs. To clarify the dilemma, say the Adaptive Multi-personality agent is in need of resource of type A and that it has N PDNMs. As a result, two or more PDNMs ask for resource A at the same phase, from two or more different opponent agents. Potentially, this will result in the receipt of up to N resources of type A, although the agent was in need of only one.

A graver risk lies in the following scenario: this time the Adaptive Multi-personality agent has one spare resource of type B. Then, two or more PDNMs offer resource B at the same phase, to two or more different opponent agents. Consequently, two or more resources of type B should be exchanged in the phase although the agent has only one spare B. As a result, the agent will be regarded as unreliable, even if the personalities of the PDNMs that commit on that resource have a high level of reliability.

In [12] the dilemma was solved by assigning portions of the available resources and portions of the required resources to each PDNM. Thus, each PDNM pursued a subset of the required resources by exchanging a subset of the available resources. However, this formulation is too limiting for our problem: the resources cannot be divided into distinct groups as in their domains. Moreover, distributing the set of resources into distinctive subsets might damage the potential exchanges the PDNMs can offer or commit to. Accordingly, we suggest that the restriction be avoided by assigning each PDNM the whole resources set and the whole required resources set, and by appointing the Personalities Coordinator to solve any contradictions, if and when they arise.

The overall performance of the Personalities Coordinator is presented in figure 2, with emphasis on resource contradiction solving. If we return to the example from before, since there are N opponent agents in the domain, as soon as the agent is created, the Personalities Coordinator creates N PDNMs to interact with them. The Adaptive Multi-personality agent's shell activates the Personalities coordinator on two occasions: at the beginning of a domain phase (begin phase signal) and upon receipt of message from an opponent (message receipt signal). The first leads to a series of initialization actions: refreshing the PDNMs and propagating all other data gathered during the previous domain phase, as well as activating the PDNMs serially and gathering any initiated messages. The latter causes the activation of the PDNM matched to the sending opponent, asking for a reply message. In both cases, the Coordinator accumulates all the messages the PDNMs wish to send and checks if they contain any contradictions. Then, it calculates the potential gain of all the contradicting messages and chooses the subset that yields the maximal gain. The gain of an exchange is related to the exchange ratio and to the value the Coordinator gives to the resources in that exchange. Formally, if an agent sends R resources and expects to receive S resources in return in a certain exchange, then the exchange ratio is $\frac{S}{R}$. By assessing the sent resources and the received resources in the exchange, the Coordinator multiplies the ratio by c for every required resource in the received resources set. On the other hand, it divides the ratio by c for every non-redundant resource in the resources set to be sent. Then, the potential gain is calculated by multiplying the quantity by the opponent's reliability estimation.

Lastly, it sends this subset to their destinations, and updates all the PDNMs on the reduction in the redundant and in the required resources.

2.4.4 Personality-driven negotiation modules (PDNMs)

The heart of the Adaptive Multi-personality agent's PDNMs is their decision mechanism, which enables them to decide which action to execute at any given time. We assume that the actions in the domain can be defined in advance, and can be summarized into a final set of actions.

We consider domains, which require four major types of decisions:

1. Decide with which opponent to cooperate. This includes the decision of which opponent to contact when in need of resources, and which request to handle when receiving several requests from several opponents.

2. Decide on the resources exchange rate - the rate of an exchange can be viewed in two ways: if all resources have the same value, it compares how many items one sends to how many items it receives in return. When resources have different values, the rate

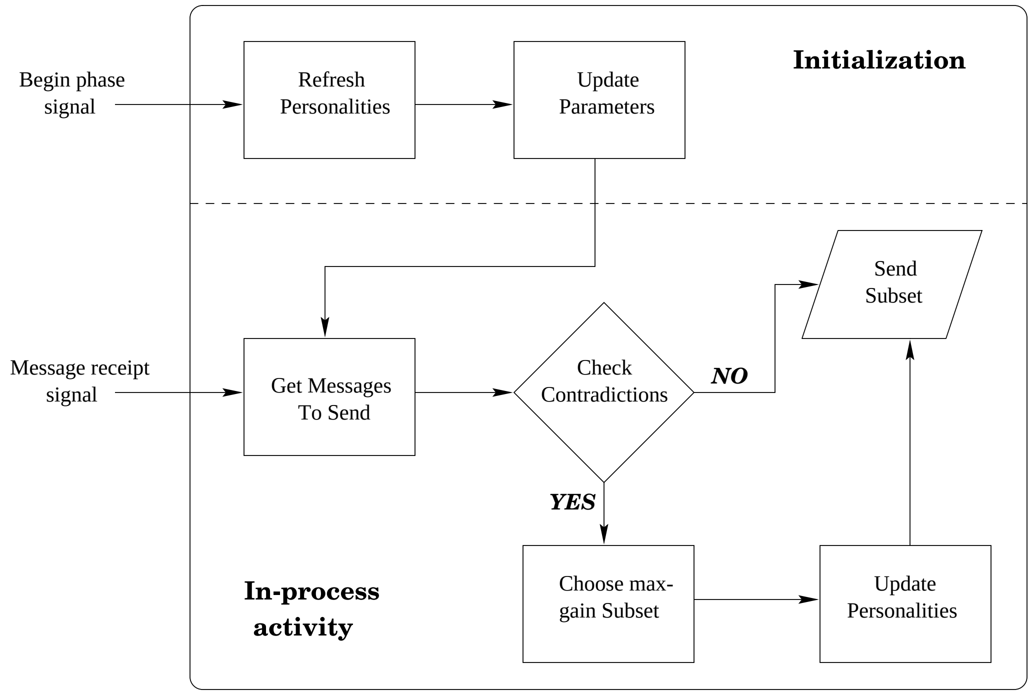

Figure 2: Flow chart of the Personalities Coordinator

will be expressed by the total value sent to the total value received, instead of number of resources sent and received. For simplicity, we assume all resources have the same value.

Naturally there are three subjective possibilities,

- **Balanced exchange** - resources exchanges in which agents send and receive the same number of resources. If we denote the exchange as a ratio between the number of resources sent and the number of resources received, this exchange is of type S:S, where S is a positive integer. We will use a *"1:1 exchange rate"* notation to describe this possibility, but, again, it includes any other rate in which the number of the sent resources is equal to the number of the received ones.
- **Unbalanced exchange to my advantage** - resources exchanges in which the recipient gets more resources than it sends. This exchange is of type S:R, where S and R are positive integers, and R>S. We will use *"1:2 exchange rate"* to describe this possibility, but it includes any other exchange rate of this kind (1:3, 4:5, inter alia).
- **Unbalanced exchange to my opponent's advantage** - exchanges in which the recipient receives less resources than it sends. This exchange is of type M:N, and the notion *"2:1 exchange rate"* will be used to describe it.

3. Once a commitment is made, decide whether to fulfill it or not. Furthermore, one may decide to keep only part of its commitment.

4. Decide to pass on information regarding the world and the agents in it. This includes the decision what information to provide, and to which agent.

Decision 1 and 2 are strongly related to the cooperation factor of the agent. This factor determines the agent's interaction level and generosity. On the other hand, decisions 3 and 4 are actually related to the reliability factor of the agent. That factor determines how many commitments the agent would keep, and how truthful it is in the information it provides to other agents in the world.

The decision mechanism can be activated when the agent wishes to initiate an action or as a response to an action committed by other agents in the domain. From psychological studies (see [**?**, **?**, **?**]) we know that the factors that influence one's decisions are its personality and its surroundings. So, when facing a decision each of the Adaptive Multi-personality agent's PDNMs takes under consideration the following factors:

- Its personality, as described by the cooperation and reliability factors. Rather than enumerate the set of PDNMs, we will refer to each PDNM by the number of the opponent it is matched with. For instance, the PDNM interacting with opponent j will be dented as P_j, its cooperation level as $P_j.w_C$ and its reliability level as $P_j.w_R$.

- The current state of the domain, as it perceives it (is it task dependent? how far is it from achieving its goal? and so on). The current state of the domain will be denoted as s, and the function evaluating its quality according to the domain's parameters will be $E(s)$. This function is domain-specific and takes under consideration a subset of the parameters in the domain. Equation 10 is an example from the Colored Trails domain, which will be discussed in section 4.2.

- The ramification of its actions - all agents are identified by a unique ID number known to all. Consquently, one's reputation is a factor that must be considered when deciding which action to take, especially when it may be conveyed from one agent to another. Specifically, when executing a "positive" action (for instance keeping a commitment) the opponent agent will know it, and perhaps will be more inclined to cooperate in the upcoming negotiations. On the other hand, when executing a "negative" action (for instance ignoring a request for resources) the other agent will know this too, and will adapt its behavior accordingly and/or convey this information to other agents. Nevertheless, if the opponent agent does not model the other agents in the domain, there are no ramifications to any action. $Gain(a, j)$ is the function that returns the quantified ramification of committing action a in relation to opponent agent j (either a positive number for a “positive” action, or a negative number for a “negative” one). The actual function we used is presented in section 4.2.

- The opponent agent’s type - if the Adaptive Multi-personality agent does not know its type, it may estimate it (as described in section 2.4.2). *Type*(*j*) is the function that returns the type of opponent agent *j*. It is either its true type, or its estimated value. Similar to the PDNM’s factors, $Type(j).w_C$ is the opponent’s cooperation level and $Type(j).w_R$ is the opponent’s reliability level.

Moreover, each PDNM keeps a list of available responses for any possibility (either its wish to initiate an action or its response to an action of the other agents in the domain). When it is activated, it calculates the quality of each response-action a in relation to its matched opponent *j* according to the following formula:

$$\begin{aligned} D(a,\, s,\, j) \quad &= \quad \alpha \cdot E(s) + \beta \cdot \{P_j.w_C + type(j).w_C\} + \\ &\quad\quad \gamma \cdot \{P_j.w_R + type(j).w_R\} + \delta \cdot Gain(a,\, j) \end{aligned} \tag{8}$$

$\alpha, \beta, \gamma, \delta$ - are weights which are determined according to action a. These weights give stronger emphasis to the personality or the domain state according to action a, and they were determined based on psychological studies and tuning. $\alpha, \beta, \gamma, \delta \in [0, 1]$. For instance: as mentioned before, when the action is concerned with keeping a certain commitment, the weight associated with the reliability factor will be higher than the one associated with the cooperation factor. The values we used for each action are detailed in appendix B.

The output of the *D* function is then multiplied by an applicability weight (*APwt*) to produce the final value of executing action a. The *APwt* is aimed to eliminate contradictions between the personality of the PDNM and the output of function *D*. For instance, the weight is zero if a PDNM with a high reliability factor wants to break a commitment it made.

After computing the value of all possible actions each PDNM chooses the action with the highest value, or a subset of non-contradicting actions that yields the highest combined value. Then each PDNM sends its recommended action(s) to the Personalities Coordinator, who chooses which messages to actually send (see section 2.4.3 for more details).

Descriptive example: Table 3 summarizes the behavior of all PDNMs of all personalities in their negotiations - it holds the main guidelines of their responses, so naturally, in some situations their behavior may change. Again, their behavior depends on the domain state and the personality of their opponents, as well as their own personality, so all factors influence the behavior. The table was created by recording the real-time responses of the different PDNMs, as was outputted by equation 8. It is divided according to the goal achievement prospects of the agent: either it has all resources to achieve its goal (task-independent) or it requires additional resources from its opponents in order to do so (task-dependent). Then it is divided according to the three possible ranges of the reliability and cooperation factors. The reliability factor is not divided according to the goal achievement prospects since only the medium reliability personality is affected by this factor (as stated in the table).

To illustrate the differences between the behavior of all PDNMs, we will focus on a PDNM with an HH personality and on a PDNM with an LL personality, both playing against a certain opponent. For the sake of the example we assume both HH and LL estimated their opponent as type MM. In addition, some of the matching of the PDNMs with their opponents may not occur when we execute the whole agent (for instance HH playing against LL), and they too are only for the sake of the example. The Personalities Coordinator will maintain the proper matching. The PDNMs behavior is characterized by the suitable lines in table 3, namely -

HH and LL are task independent - the conduct of both PDNMs (with emphasis on the differences between them):

- LL will not initiate any exchange requests, whereas HH will initiate a exchange request in order to demonstrate its willingness to assist others.
- When LL receives a request for resources from opponent LL, it calculates the exchange rate - if it is of the form 1:2 it falsely commits to it. Then it waits to receive the resources from MM, but it does not send anything in return. If not, it ignores the request altogether.
  HH, on the other hand, receives the message and assesses it. If the rate exchange is of the form 1:1 or 1:2, bearing in mind the opponent is MM, HH will commit to the request and send the requested resources to MM.

HH and LL are task dependent - this setting leads to the following behaviors:

- LL will initiate an exchange request, containing the resources it needs. At first the exchange rate will be in its own advantage (1:2), but as the need prevails, it will increase the number of resources it is "willing" to exchange, until the exchange rate is of the form 1:1.
  HH will initiate an exchange request, containing the resources it is missing too, but with a 1:1 exchange rate. If it is unable to persuade MM to agree to its offers, it will increase the number of resources it is willing to exchange, but no more than a predefined ratio.
- When LL receives a request for resources from MM, it looks at the offered resources. If they include some of the needed ones, it "commits" to the request. If not, it calculates

Goals Prospect	Person-alities	Guidelines
All	Low reli.	Never keeps its commitments.
	Med. reli.	Keeps some of its commitments - the more reliablie its opponent is, the more reliable it becomes. In contrast, the more cooperative its opponent is, the less reliable it becomes, in order to maximize its gain. Moreover, when it is task dependent it tends to keep less commitments.
	High reli.	Always keeps its commitments
Task indep-endent	Low coop.	(1) Never initiates messages; (2) When accepting a request, always tries to attain resources exchange rate of 1:2, hoping to gain more resources. The rate of the exchange increases as the cooperativeness of its opponent increases; (3) Negotiates with all other agents in the world
	Med. coop.	(1) Initiates messages when estimating its opponent is cooperative; (2) Its exchange rate is also related to the estimated personality of its opponent. The more cooperative its opponent is the higher the rate the agent requests. Moreover, it seldom exchanges at a 1:1 rate; (3) Negotiates with high-reliability agents only.
	High coop.	(1) Always initiates exchange messages, to demonstrate its willingness to assist others; (2) Usually offers exchange messages on a 1:1 basis, but when faced with an un-cooperative opponent may offer exchanges of a 1:2 rate; (3) Negotiates only with agents whose total personality is more than medium (the sum of their cooperation and reliability estimated factors must reach a certain threshold level)
Task depe-ndent	Low coop.	(1) Always initiates exchange messages with unbalanced rates at its advantage, until its situation becomes desperate. Then, it is willing to exchange at a banalanced rate or even give more resources for needed ones (when it is faced with low cooperation opponents); (2) Ignores requests for resources from its opponents,unless they offer resources it needs. Another exception for this rule is when the rate of exchange is extremely unbalanced in its advantage; (3) Negotiates with all other agents in the world
	Med coop.	(1) Initiates exchange messages at different levels of exchange rates. If its opponent is cooperative it is likely to ask more resources in return. As its state in the domain worsens it offers exchanges at 2:1 rates; (2) Requests for resources are accepted if they contain the resources it needs in return; if they're of a 2:1 exchange rate; or if they are issued by a cooperative agent and are balanced; (3) Negotiates with medium-reliability agents (or above) only.
	High coop.	(1) Always initiates exchange messages at a fair rate. When its state in the game becomes desperate it begins offering exchanges at 2:1 rates; (2) Considers any requests for resources. The more cooperative the requesting agent is, the more likely it will accept the request; (3) Negotiates only with agents whose total personality is medium or above.

Table 3: Main guidelines of personalities

the exchange rate, and "commits" only if the rate is of the form 1:2.
HH, on the other hand, receives the message and assesses it. If the rate of exchange is 1:1 it will commit to the request and send the requested resources to MM, even if the resources it will receive are not required. Moreover, if the resources offered by the opponents are required, HH will be willing to a 2:1 exchange rate as well.

Say, over time, the opponent LL and HH estimated as being MM, did not keep its commitments and did not cooperate with them. As a result, LL and HH change their estimation of its type to LL. Now their behavior will be as follows:

<u>HH and LL are task independent</u> - the conduct of both PDNMs (with emphasis on the differences between them):

- Both LL and HH will not initiate any exchange requests.
- When LL receives an request for resources from the opponent LL, it nevertheless calculates the exchange rate - if it is in its own advantage (1:2) it falsely commits to it. Then it waits to receive the resources from LL, hoping its estimations are pessimistic, but it does not send anything in return. If not, it ignores the request altogether.
  HH, on the other hand, receives the message and, estimating the opponent's type as LL, refuses the request.

<u>HH and LL are task dependent</u> - the conduct of both PDNMs:

- The same as in the previous scenario, LL will initiate an exchange request, containing the resources it needs. At first the exchange rate will be 1:2, but as the need prevails, it will increase the number of resources it is "willing" to exchange, until the exchange rate is 1:1.
  HH will also initiate an exchange request, containing the resources it is missing. However, estimating its opponent as having a low cooperation level, it will start off with some sort of a 2:1 exchange request, and increase it with time, but no more than a predefined ratio.
- When LL receives a request for resources from MM, it looks at the offered resources. If they include some of the needed ones, it "commits" to the request. If not, it calculates the exchange rate, and "commits" only if the rate is of the form 1:2.
  HH, on the other hand, receives the message and assesses it. Since the opponent has a low reliability level it will assess the risk in committing to the request, bearing in mind its opponent may try to exploit it. Only if the risk is below a predefined threshold it will commit and send the resources. Otherwise, it will refuse.

2.5 Modules combination - flow of data in the Adaptive Multi-personality agent

The Adaptive Multi-personality agent's lifespan is divided into phases, each phase is divided into two sub-phases: a negotiation phase and an action phase. As describes in section 2.1, this division may be artificially constructed by the Adaptive Multi-personality agent, or it is built-in within the domain. In the first sub-phase the Adaptive Multi-personality agent negotiates with its opponent agents: its PDNMs request resources, offer resources to others, decide which commitments to keep and which to break, send and receive resources and so on. In the second sub-phase it executes all the possible domain-operations, which are not related to negotiations but crucial to its performance in the domain. Each sub-phase is limited in time.

As described in figure 3, at the beginning of the negotiation phase, the Adaptive Multi-personality agent initializes or updates all its parameters from all the information gathered in its modules. This includes updating the number of its resources and the opponents' types estimations, and also creating PDNMs with the most suitable personalities to interact with those opponents. Then, it activates the Personalities Coordinator, which "awakens" all the PDNMs. The Coordinator sends the parameters updates to the PDNMs and waits for their recommended course of action. At this point the PDNMs decide whether to initiate a message, which type of message to initiate and so on. As soon as a message is received, the shell of the Adaptive Multi-personality agent passes it along to the Coordinator, which, in turn, conveys it to the proper PDNM. Again the PDNM's decision mechanism is activated, and a suitable response is recommended. This process is repeated again and again during the negotiation phase.

At the action phase two modules are activated: the Learning NHandler keeps track of the messages and commitments which were made during the past negotiation phase and produces updated estimations of the opponents' types. The Domain Manager is also activated, and it executes any set of domain-specific non-negotiation related actions, which should increase the agent's utility.

This procedure occurs repeatedly until all agents existing in the domain have achieved their goals, or until another predefined ending was reached. An example for a predefined ending is when all agents in the domain have run out of resources (without achieving all the goals).

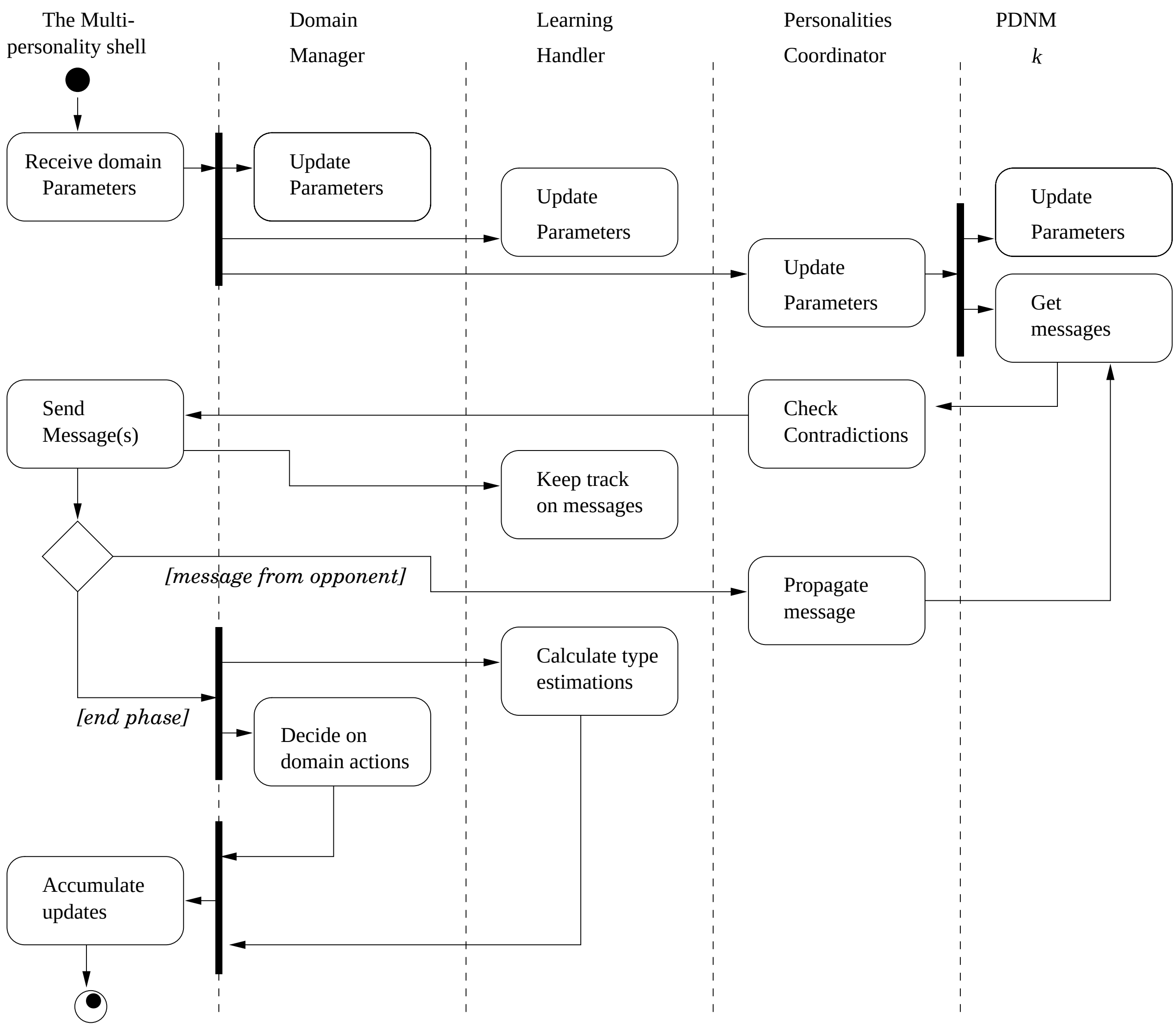

Figure 3: Activity diagram of the Multi-personality agent's modules

3 Adaptive one-personality agent (AOP agent)

In order to evaluate the performance of the Adaptive Multi-personality agent, we designed adaptive one-personality (AOP) agents as well, for the Adaptive Multi-personality to interact with. These agents are in fact instances of the Adaptive Multi-personality agent, differing in the following aspects:

- They have only one personality whereas the Adaptive Multi-personality agent comprises several personalities.
- Their personality weights are static and never changes. Nevertheless, since their PDNMs' actions are dictated by their decision mechanism (equation 8), and are influenced by their surroundings in addition to their personality factors, they are capable of adapting to changing states in their surroundings. This adaptation is limited by their personality - a high reliability AOP will not be able to adopt a deceitful behavior even if its surroundings demand it.
  In contrast, the personalities of the PDNMs of the Adaptive Multi-personality agent dynamically change as it models its opponents.

All other modules (described in section 2.4) are identical. Specifically, the AOP agents also employ the learning procedure, and, as a result, they too adapt to their opponents. However they do not change their very own factors as the Adaptive Multi-personality does, only their behavior: the w_C and the w_R in equation 8 stay constant in the AOPs while changing in the Adaptive Multi-personality agent. In a sense, the AOPs can be viewed as the social agents currently exisiting in the multi-agents systems. The AOPs also implement the Personalities Coordinator module, and a PDNM to interact with each of its opponents. The difference is that the PDNMs of the AOP agent has only one static personality, while the PDNMs of the Adaptive Multi-personality agent may have different personalities, which may dynamically change as the agent evolves.

Since Each AOP has one personality only, seven different AOP agents can be created - LL, LM, LH. MM, MH, HM and HH AOP agent. To illustrate the difference between the AOPs and the Adaptive Multi-personality agent, consider the following example (main differences appear in table 4 too):

The Adaptive Multi-personality agent and an MH AOP agent interact with an LL AOP and with an HH AOP. At the beginning of their interaction they assign default estimations for their opponents' type, say MM. The Adaptive Multi-personality agent also needs to decide which personality to assign each of the PDNMs that interact with each of its opponents. It consults the "Matching Table" and assigns each PDNM the personality that best-suits opponents of type MM, say also MM. At this point, the Adaptive Multi-personality agent is analytically

Event	The MH AOP agent response	The Multi-personality agent response
Beginning	Assigns default estimations for its two opponents (say MM).	(1) Assigns default estimations for its two opponents (say MM); (2) Creates two PDNMs with personalities that best-suit opponents of type MM (say MM too).
In process activity	Plays against LL and HH as MH.	Plays against LL and HH as MM.
Opponents types estimations converged to their true type	(1) No change in personality weights; (2) Adapts to its opponents. For example, against LL, although medium cooperative, it will disingage negotiations.	(1) Changes the personality weights against LL to LL weights; (2) Changes the personality weights against HH to HH; (3) Plays in a totally adjusted way against its opponents.

Table 4: Illustration of main differences in the performance of the Multi-personality agent and an AOP agent

identical to the MH AOP - both estimate their opponent the same, and both negotiate with them having one uniform personality.

Nevertheless, over time both the Adaptive Multi-personality agent and the MH AOP agent learn their opponents and update their estimations of their types. At this point the MH AOP agent still has an MH personality (its w_C and w_R stay the same) whereas the Adaptive Multi-personality agent could have changed its PDNM with the changes it detected in the behavior of its opponents. Say the matching scheme the Adaptive Multi-personality agent employs is *tit-for-tat* (table 2), and that it was able to detect the true type of its opponents. As a result it will create an LL PDNM for its interactions with its LL AOP opponent (its w_C and w_R change from MM to LL) and an HH PDNM for its interactions with its HH AOP opponent (its w_C and w_R change from MM to HH). The MH AOP, on the other hand, will play the same in all of its interactions, despite its adapting to its new state in the domain and to the new estimations of its opponents' types. This may cause a potential resource loss to the MH AOP - it may commit to its LL AOP opponent, and having a high reliability, it would send the resources, and not receive anything in return.

Although this illustrated difference between the AOP agents and the Adaptive Multi-personality agent seems negligible, it causes a huge difference in their performance, as will be presented in section 6. There we will evaluate the performance of the Adaptive Multi-personality agent against each of the different AOPs by conjuring several domain scenarios with different participants in each scenario, and recording the performance of all participants.

Our hypothesis is that the performance of the Adaptive Multi-personality agent will outweigh the performance of all other AOPs.

4 Test-bed to evaluate the Adaptive Multi-personality agent's performance

4.1 Colored Trails Game

We chose to evaluate the Adaptive Multi-personality agent performance in the Colored Trails (CT) domain. Colored Trails (CT) is played on an NxM board of colored squares with a set of chips in colors chosen from the same palette as the squares. For each game of CT, one square is designated as the goal square and there are two or more players. Each player has a piece on the board in one of the non-goal squares and a set of colored chips. To move a piece into an adjacent square, a player must turn in a chip of the same color as the square. Each player can see the whole board layout. The goal of the game is that all the players end up in the goal square. Chips can be exchanged during the game by means of negotiations.

The game is divided into steps. Each step is composed of two phases: the communication phase and the movement phase. In the communication phase, the players are allowed to communicate with each other, but are not allowed to move. This phase can be used by a player for negotiations with other players, providing them with information or requesting information from them. In the movement phase players are only allowed to move their corresponding piece across the board, or to pass on the movement possibility. In order to prevent the situation in which players remain idle for long periods, a constraint has been added that restricts the number of consecutive steps that a player is allowed to pass. In case a player exceeds this limit, he becomes irrelevant to the game, meaning he cannot move until the game has ended, but can communicate and negotiate freely with other players still in the game. Adding this specific constraint imposes a finite time for each game. During both phases of each step, players are allowed to make private calculations and decisions concerning the next actions to be taken during the game.

In CT, the need for decision-making strategies on collaboration between agents arises when a player lacks chips required in order to reach a goal square. Helping another agent is interpreted as providing him with chips he may lack, or uncommon information about other agents. The circumstances under which one agent decides to help another may be varied depending on whether the agents are on a team or members of a group (but not currently collaborating) or acting completely independently.

CT allows for performance of an individual player to be measured according to different criteria, corresponding to different ways of measuring performance of team activities. We include the following criteria in the evaluation of different game scenarios:

(a) the number of moves that a player makes to reach a goal square (corresponding to the number of resources consumed);

(b) the number of chips that a player possesses when the game ends (related to the cost of a particular recipe) - the score increases as the number of chips increases;

(c) the distance of the player from a goal score, if the goal is not reached (representing credit for partial fulfillment of the task). We neglect the cost of communication, since we are interested in the decision-making aspect of the game.

The scoring rule for a particular instance of CT is given to the players by the manager of the game. Formally, the scoring rule of an agent is:

$$scoring\_rule(agent_i) = \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right\} RGwt - \frac{NMwt \cdot num\_moves_i}{M+N} + \frac{CLwt \cdot num\_chips_i}{M+N} - \frac{GDwt \cdot dist_i}{M+N} \tag{9}$$

where,

i - is the serial number of the agent,

RGwt, *MNwt*, *CLwt*, *GDwt* - are coefficients provided to all agents at the beginning of the game (Reached Goal weight, Number Move weight, Chip Left weight and Goal Distance weight respectively). If the agent reached the goal than *RGwtA* is multiplied by 1. If not it is multiplied by 0,

M,*N* - are the dimensions of the CT board,

num_moves - is the total number of moves agent *i* has made in the game,

num_chips - is the total number of chip agent *i* holds at the end of the game,

dist - is the distance of agent *i* to the goal, measured in number of squares. If the agent reached the goal square than *dist*=0.

The automated game manager was developed by the students of Prof. Barbara Grosz from Harvard University. The negotiations are performed by using a semi-formal language, and accommodate the following message types:

1. Chips request - specifies number and color of the chips without offering chips in return.

2. Chips exchange - specifies number and color of the chips requested and the chips offered.

3. Exchange commitment - serves as an unenforceable promise and specifies the amount of chips of each color to be sent and to be received.

4. Refusal - of a request or an exchange.

5. Liar testimony - specifies the ID of the agent.

Each message is sent from an agent to a single agent, and both the sender and the recipient IDs are specified in the message. In addition, two functions were implemented and are at the disposal of the agents. The first function is a parser that decodes the messages written in the semi-formal language to their atomic parts. The second function is a path finder that calculates all possible paths from a given square to the goal, and provides the lists of chips required for these paths as well.

Furthermore, in the 4-player evaluations two different CT game boards were used. One board, the "all dependent" board was constructed so that every player was task dependent. In particular, in order to reach the goal, all players needed at least one chip from another player, and each player had some chips it could offer to help other players. The second board, the "one self-sufficient" board was constructed such that one player was not task-dependent. This self-sufficient player was able to reach the goal without any help from other players. In addition, the three other players were task-dependent on the self-sufficient player and on each other; All needed a chip that only the self-sufficient player could provide, and an additional chip from one of the other two players. In the remainder of this thesis, we will refer to the first board as *AllDep* board, the second as *OneSelf* board and the player on the OneSelf board who was self-sufficient as the *SelfS* role player. Screenshots of the two boards are available in figure 4 and 5 (taken from a user interface software developed at Harvard University). In addition, in all games we executed, all agents were able to see the game board and the initial locations of all agents, but they were provided with no information on the chips held by other agents (incomplete information on the resources).

We find this domain suitable for evaluating the Adaptive Multi-personality agent performance since the CT game provides a good and complex environment for negotiating with other agents, of all types. It serves as a good analogy for the environments in which we may investigate the Adaptive Multi-personality agent's performance: Finding the best playing strategy is quite hard in this case, and we believe an Adaptive Multi-personality agent can perform better than an Adaptive one-personality agent.

4.2 The Adaptive Multi-personality agent in the CT domain

The Adaptive Multi-personality agent is one of the players in the CT game. Its goal is to maximize its gain according to the scoring rule it is given at the beginning of a game, and in compatibility with its personality. That is, the Adaptive Multi-personality agent is not purely rational in the respect that it does not execute the best action to serve its goals at any given moment, but it acts according to its personality traits as well. As a result if one of its PDNMs

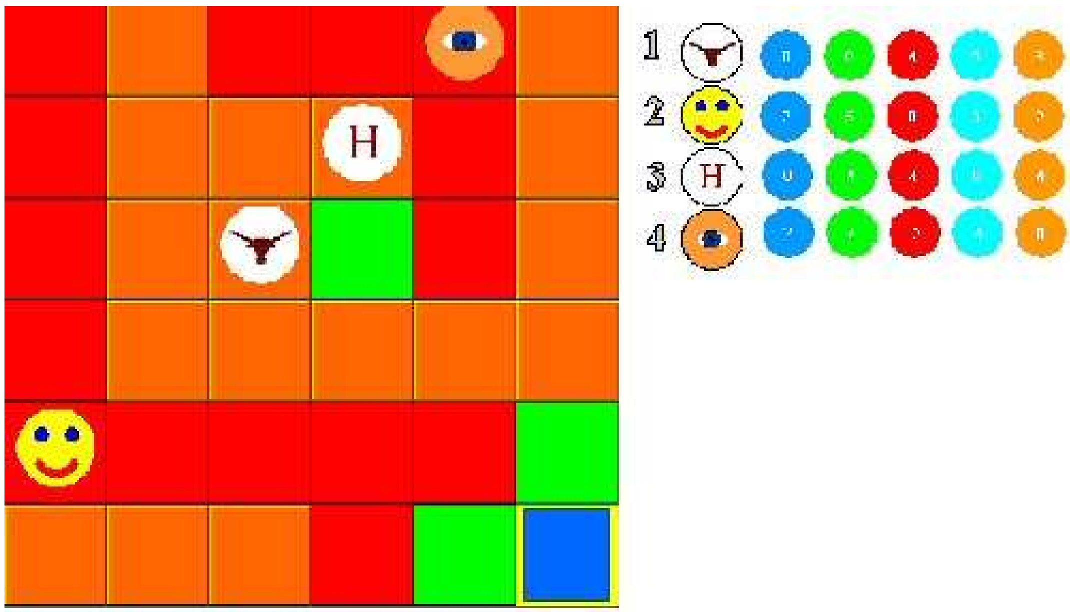

Figure 4: AllDep Game board

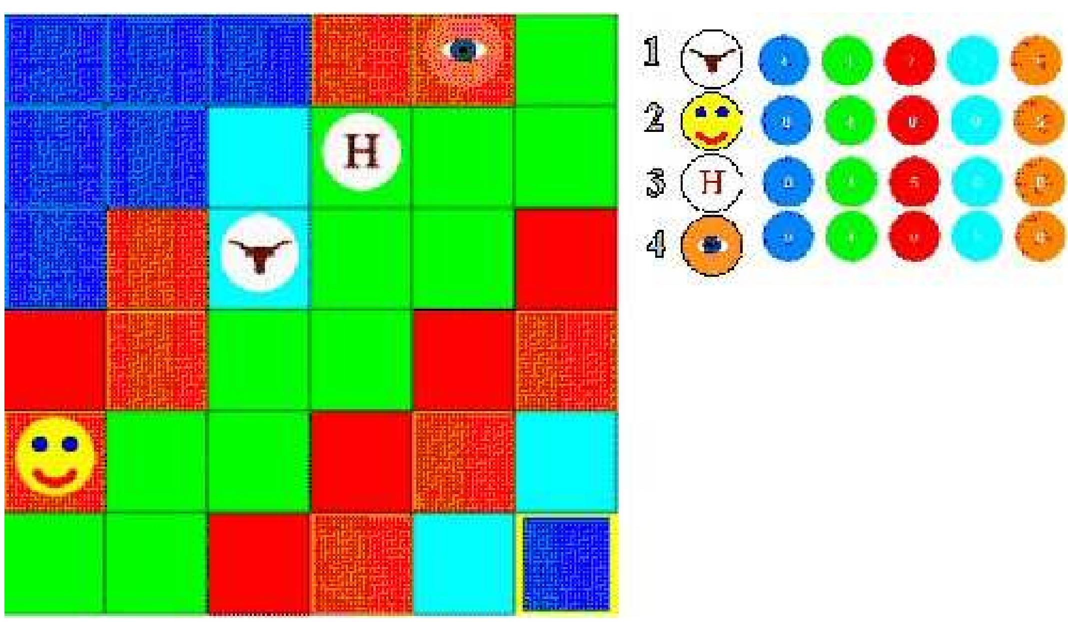

Figure 5: OneSelf Game board

has a personality that is cooperative and reliable, it may agree to cooperate with opponent agents even if it seems irrational, for example.

The playing manner of the Adaptive Multi-personality agent in the CT domain is identical to the playing manner in general domains as presented in section 2.5. However, sevral of its domain-specific functions are unique to CT, in particular:

- **State in the CT domain:** One of the variables that influences the PDNM's decision mechanism is the agent's state in the game. In CT the state of each agent is specified in equation 9. As a result, the quality the PDNMs assign to state s in the CT domain ($E(s)$ in equation 8) is -

$$E(s) = pr \cdot RGwt - \frac{NMwt \cdot num\_moves}{M+N} + \frac{CLwt \cdot redunt\_chips}{M+N} - \frac{GDwt \cdot dist}{M+N} - e^{num\_passes} \tag{10}$$

  where -

  pr - is the probability of the Adaptive Multi-personality to reach the goal, namely -

$$pr = 1 - \frac{num\_of\_chips\_to\_obtain}{length\_path\_to\_goal} \tag{11}$$

  When the opponent agents lack the chips the Adaptive Multi-personality agent must obtain, the probability should be zero. Nonetheless, this information is not provided to the agent due to chips-invisibility.

  redunt_chips - is the number of redundant chips the Adaptive Multi-personality agent holds at state s (similar to num_chips in equation 9),

  num_passes - is the number of consecutive steps that the Adaptive Multi-personality agent has passed until state s. Since a player exceeding this limit becomes irrelevant to the game, and can no longer move, it is crucial to pay close attention to this parameter. The quality of the CT state of the Adaptive Multi-personality agent deteriorates with the number of consecutive passes it commits.

- **Ramifications of the Adaptive Multi-personality's actions:** We used the following simple function for the quantified ramification in equation 8 in the CT domain. We defined which actions reflect well on the agent ("positive" actions), which result in a bad reputation ("negative" actions) and which are "neutral" (a detailed list is provided in appendix C). And -

$$Gain(a,j) = \left\{ \begin{array}{cc} 5 & a\,is\,positive \\ 0 & a\,is\,neutral \\ -5 & a\,is\,negative \end{array} \right\}. \tag{12}$$

Although this function is dependent on the strategic ability of opponent agent j, for simplicity we assumed all agents have the same power as the Adaptive Multi-personality agent, and thus, the function is the same for all the opponents.

- **Overall performance of the Adaptive Multi-personality agent in the CT domain:** The implementation of the overall performance of the agent, described in section 2.5, in the CT domain is as follows:
  When a game begins, the Adaptive Multi-personality agent searches the last opponent type estimations it calculated for all opponent agents. If the information is unavailable, it assigns default values to these estimations. Then, it matches each of the opponent agents with a PDNM having the best-suited personality, according to the Matching Table it maintains. During the communication phases of the game it interacts with the opponent agents in accordance with the following rules:

  1. At the beginning of a new communication phase the Coordinator activates the PDNMs serially and collects the messages they wish to send (if any). Two types of messages can be issued - either a chips request message or a chips exchange message (though all possible exchange rates are accepted).
  2. Upon receipt of a message, the Coordinator sends it to the proper PDNM, which in turn issues a response message. The types of responses allowed depend on the received message, in particular -
     (a) Chips request or chips exchange messages - the possible responses to these messages are a different chips request, a different chips exchange (different in the exchange rate or in the chips offered), refusal, chips commitment, false chips commitment or ignorance.
     (b) Refusal - the possible responses here are a different chips exchange, with a more generous exchange rate or another set of chips offered, or ignorance.
     (c) Chips commitment - may respond with a commitment, if it agrees to the commitment, or a refusal. In addition, it may send a false commitment, or a partial one. If the opponent agent issued the commitment in response to the PDNM's commitment, no reply is issued.
     (d) Liar message - no reply is issued.

Throughout the communication phase the Learning Handler keeps track of the chips commitments made by all agents in the domain.

During the movement phase, which proceeds to communication phase, the Adaptive Multi-personality agent -

1. Addresses the Domain Manager module for instructions. In CT the Domain Manager is responsible for finding the best path to the goal, in accordance with the agent's chips, and for managing them. Thus, the Manager instructs the agent which move to take, and then updates the chips lists according to the chips received from the opponents, the chips sent to the opponents and the chip that was used to move.
2. Activates the Learning Handler to update its opponents' type estimations. The Handler matches the chips received to the commitments made by the opponents, and calculates their new reliability factor estimate (using equation 5). Then, it characterizes their cooperation level based on the past communication phase: it is "reciprocal" if the opponent sent and also received chips; "give" if the opponent only sent chips; "take" if the opponent only received chips; and "idle" if it neither sent nor received chips. Then it calculates the new cooperation factor estimate of all opponents.

We investigate the Adaptive Multi-personality agent performance in two game-playing strategies:

1. A single game: each game is played as if it is the first time the game participants encounter each other. Information accumulation about the participating agents is not allowed to be passed on from one game to the other, but it can be used in the course of a single game.

2. Repeated games: the agents were allowed to use all information they have accumulated during all previous games. Since all agents are recognizable by their permanent ID numbers, any agent can keep statistics on its opponent agents, estimate their game playing strategy and assess their moves, from one game to the next game in which it encounters them. The length of the series of the repeated games is unknown, i.e. at the end of every game in the series there is a probability that the series will be over.

4.3 Peer-designed agents (PDAs)

The Peer-designed agents (PDAs) are agents designed by upperclass undergraduate and graduate computer science students from Bar Ilan university, which were built to play the CT game. Ten different PDAs were used in the evaluations of the Adaptive Multi-personality agent. We used these agents in order to evaluate the performance of the Adaptive Multi-personality agent when interacting with other agents, whose design and game playing manner were totally unknown or unpredictable by the Adaptive Multi-personality agent. Unlike the AOP agents, which were designed by us and interact smoothly with the Adaptive Multi-personality agent, the PDAs were created by other people, using other design ideas. We found it interesting to observe the behavior of the Adaptive Multi-personality agent in its interactions with the PDAs.

Each designer submitted a design paper and an agent. By reading the papers and inspecting the PDAs' games transcripts, the main characterizations the PDAs playing strategies are presented in table 5. The table is divided according to several main parameters in the playing strategy of an agent playing CT: firstly, it describes how their personality factors change according to their task dependency. Note that if an agent is deceitful only the reliability factor is stated in the table, and if it stops negotiating for any reason, only the cooperation factor is stated. When it employs all possible values a "dynamic" value is stated. Secondly, the table describes the main aspect of the PDA's adaptation. Lastly, it presents the PDA's rationalism by stating whether it uses the CT scoring rule in its decision-making process. The PDAs detailed strategy is as follows:

- **PDA A**: The agent interacts with its surroundings only if it is task dependent. Otherwise, it uses its own chips in order to achieve its goal without helping others. When it is task dependent it may commit to chips exchanges even if it does not have the chips requested. It does not agree to exchanges where the exchange rate is not 1:1. Moreover, when it detects untruthful behavior on the part of its opponents, it becomes unreliable too. Specifically, the designer built an excessive decision tree which the agent follows throughout the negotiation phase. The tree's nodes are related to the reliability of its opponents and to the PDA state in the game.
  The agent models its opponents - it is mainly concerned with estimating their reliability, but it also gathers information on their locations, number of idle phases, possible paths to the goal square, and so forth. This information is saved in an external file and reused in future interactions.

- **PDA B**: The agent is involved in chips exchanges only when it is task dependent, and only with opponents it estimates reliable. Then, it is willing to exchange up to three chips for any chip it requires. When it is task independent it employs deceitful behavior, that is, it commits to chips exchanges, but never sends any chips (in this case it is similar to LL AOP). In addition, it is untruthful when it is task dependent and does not have any of the chips it is asked in return.

PDA	When task independent		When task dependent		Adaptation	Use of Scoring Rule
	Cooperation	Reliability	Cooperation	Reliability		
A	Low	Tends to low	Dynamic	Dynamic	Lies to liars	No
B	—	Low	Dynamic	Dynamic	Cooperates only with reliable opponents	Semi-use
C	Low	—	Dynamic	Dynamic	Cooperates only with reliable opponents	No
D	Dynamic	Dynamic	Dynamic	Dynamic	Yes	Semi-use
E	Dynamic	Dynamic	Dynamic	Dynamic	Does not negotiate with opponents that owe it chips	Yes
F	Low	Dynamic	Dynamic	Dynamic	Takes advantage of task indep- -endent opponents	No
G	—	Low	Dynamic	Dynamic	Cooperates only with reliable opponents	Yes
H	Low	Tends to low	Medium-low	Tends to low	May keep commitments to reliable opponents	No
I	Low	—	tit-fot-tat	tit-fot-tat	Similar to tit-for-tat	No
J	Low	High	Dynamic	High	Strives to form a community of traders with a high reliability level	Yes

Table 5: PDAs characterization

The agent models its opponents (their reliability, their distance from the goal square and number of idle turns) but does not pass this information on from game to game.

- **PDA C**: Similar to PDA A, the agent interacts with its surroundings only if it is task dependent. Otherwise, it ignores requests from its opponents. When it is task dependent it may commit to chips exchanges even if it does not have the chips requested. The agent ranks its opponents by their reliability and state in the game, and addresses only the top ranked ones. The information it gathers throughout the game is passed on to the next one.

- **PDA D**: The agent interacts with its opponents throughout the game. It keeps its commitments when it finds it beneficial (it needs the chips offered, the exchange rate is in its favor, inter alia). All other occasions it does not keep them. The agent keeps score on its opponents reliability, and upon receiving a request from any of them, it compares the potential benefit from committing to the request to the potential risk. If the potential benefit exceeds the potential risk, it commits to the request and keeps it. The agent passes the reliability estimations of its opponents on from one game to another.

- **PDA E**: Very similar to PDA D. One difference is that it commits only to exchanges in which it receives equal amount or more chips than the amount of chips it sends. Another difference is that this agent refuses to interact with any opponent agent that "owes" him chips. The agent keeps reliability listings of its opponents, and updates their values in the course of the games.

- **PDA F**: When the agent detects a desperate need for a certain chip by its opponents, it adopts a deceitful behavior and (1) asks for three times as many chips as it offers (2) often does not keep its commitments. This agent also keeps dynamic reliability listings.

- **PDA G**: The agent keeps its commitments only when it is task dependent, and only if the offering opponent is trustworthy and the exchange rate is fair or to its own advantage. When it is task independent it adopts deceitful behavior and never keeps its commitments.

- **PDA H**: The agent's objective is to reach the goal while acquiring as many chips as possible. As a result its playing strategy is very similar to the strategy of LL AOP. Nevertheless, it may share its chips from time to time (for instance, one scenario of cooperation is when the opponent is estimated as a highly reliabe agent and has offered a required chip).
  The agent stores the reliability estimations of its opponents, and reloads them at the beginning of new games.

- **PDA I**: The agent's design is a simplified version of the tit-for-tat strategy - it cooperates with agents who cooperate with him (to a certain point though) and deceits the ones who deceit it. Nevertheless, it stops interacting with its surrounding when it is task independent. When it is task dependent it is willing to accept all possible exchange rates.

- **PDA J**: The agent negotiates with its opponents only when it is task dependent. Otherwise, it is willing to exchange chips when the exchange rate is in its advantage. Nevertheless, once it commits to an exchange, it always keeps it. The designers were inspired by the stock exchange system model, and tried to build a simplified version of it. The agent strives to establish a closed community of traders with a high reliability level. It assumes there are some high reliability agents in the game, which are potential members of the community. The community will allow exchanges of large amounts of chips at a low risk of being cheated. The designers' motivation was that, eventually, this methodology will lead to high game scores. The behavior of this agent is similar to the HH AOP.

The main difference between our design and the designs specified above (apart from the notion of multiple personalities) is that all designers took notice of the reliability level alone, and disregarded the cooperation factor altogether. In our previous work ([9]) we discovered that the level of cooperation is a major factor in the design of an automated agent. That work investigated the properties of decision-making strategies in multi-agent situations and we used these PDAs to draw our conclusions. In that work we showed that the total performance of these PDAs can be characterized as low cooperation, which is also evident from the characterization made above. Most PDAs also possess low levels of reliability. As a result our hypothesis is that their overall performance in the game will be similar to the performance of the LL AOP agent, or at least will have the same behavioral trend.

Results from current work are presented in section 6.2.

5 Precursary experiments

The Adaptive Multi-personality agent may achieve good results only if it estimates its opponents types correctly and quickly, and then assigns them PDNMs with the most suitable personalities. As a result,the Learning Handler module must -

1. Be very accurate - otherwise, the matching of personalities to each opponent type will be incorrect.

2. Converge as quickly as possible to the true types of the opponent agents - up until that convergence time, the personalities of the PDNMs do not best-suit their opponents, and might damage the Adaptive Multi-personality's performance.

Furthermore, the matching schemes the Personalities Coordinator module uses must be proven to be efficient against each opponent type.

The next sections describe the two types of experiments we conducted in order to bring the agent to this level. The first set was aimed to measure the correctness of the Learning Handler module. The second set's goal was to build the Matching table of the Personalities Coordinator module: to find out which personality best-suits each type, both in the single game environment and in the repeated-games environment.

Another aspect to be evaluated is the importance of learning from testimonies. We believe that incorporating testimonies into the types estimators contributes to their accuracy. Nevertheless, only four out of the ten PDAs use this possibility, and only partially through the "liar" message. The Adaptive Multi-personality and the AOP agents use it very rarely, so we will leave the analysis of its importance for future work.

5.1 Learning experiments

The correctness and convergence time of the Adaptive Multi-personality Learning Handler was measured by using the HH AOP agent. Both use the same Learning Handler, but the HH AOP agent maintains a constant personality throughout the game while the Adaptive Multi-personality agent has multiple personalities, and they may be dynamically changed. We chose the HH AOP to be a constant player in these experiments since all AOPs and PDAs are adaptive, and in order to reveal their true type their opponent must allow them to do so. Since he HH AOP is highly reliable, it does not "encourage" deceitful behavior by players who are not deceitful by nature. Moreover, it is highly cooperative, thus it encourages communication, and gives its opponents enough opportunities to reveal their type.

The evaluation of the correctness and convergence time of the Learning Handler module was done by comparing the estimations the AOP had calculated to the true type of its opponent. The HH AOP played 2-player games in the CT domain against all AOPs and PDAs. During the games the HH AOP estimated the cooperation level and the reliability level of its opponent agent, using equation 5. It quantified its observations after each communication phase into an external file. If no communication was made between the two agents a "Not Available" (NA) sign was written. After a game ended, we analyzed the estimations file as follows:

1. We translated the quantified cooperation and reliability estimations into three possible values: low, medium or high. The corresponding numerical ranges were:

- Low cooperation: [0.0,0.3) ; low reliability: [0.0,0.4)
- Medium cooperation: [0.3,0.6) ; medium reliability: [0.4,0.8)
- High cooperation: [0.6,1.0) ; high reliability: [0.8,1.0)

The numerical ranges were determined according to the analytical analysis of the two factors. Thus, in regard to the reliability factor, wider equally-spaced ranges were assigned to the low and medium values and narrower range was assigned to the high value. Only agents that keep almost all their commitments are regarded as having a high reliability personality. In regard to the cooperation factor, a wider range was assigned to the high value, and the other values were equally spaced. Here, agents that cooperate more than half the time are regarded as having a highly cooperative personality.

2. We compared the estimated cooperation level to the true cooperation level of the opponent agent: when the estimation converged to the true value, we recorded the first phase number for which this occurred. If no data was available throughout the game an NA sign was recorded. And, if HH AOP never converged to the true value, an error sign (X) was recorded. Then, we repeated the process for the reliability level estimation.

3. After all the games were played, we counted the number of "X"s, the number of "NA"s and the number of correct estimations and calculated the correctness of the Learning Handler module. Then we calculated the convergence time by averaging the phase number in which the convergence occurred.

HH AOP played 30 different games against each AOP and PDA. The games varied in respect to several parameters. We know, from our previous work and from analyzing the PDAs design papers, that the most significant parameter affecting an agent's behavior is task dependency. Moreover, almost all designers also considered the amout of time this dependency prevails - the longer it is, the more "desperate" their agent becomes, and the more it is willing to perform extreme actions (lie more, offer more chips in an exchange, negotiate more and so forth). We also incorporated this in the Adaptive Multi-personality/AOP agent design (equation 10, the *num_passes* parameter). As a result we created three settings for this parameter:

(a) Task independency - the agent has all chips to reach the goal

(b) Real task dependency - the agent needs chip(s) from its opponents in order to reach the goal, and one or more of its opponents have those chips for exchange. In the remainder of the thesis we will refer to this term simply as "task dependency".

(c) Unassisted task dependency - the agent does not have all the chips in order to reach the goal, but none of its opponents have them for exchange. The agent is not aware of it due to chips invisibility. We will refer to this term as "unassisted dependency".

In 2-player games, this parameter allows nine different scenarios: three possibilities for the first player and three possibilities for the second player. In addition, we varied two secondary parameters, namely -

1. The game board - we used five different boards, two boards of size 6X6, two of size 7X7 and one board of size 8X8. There were five possible colors for the squares (and chip sets) of all boards, and the goal square was identical for all players, the most bottom right square.

2. The players' initial squares - here we varied two features:

   - The distance of the initial square from the goal square - it was either the maximal distance or an "ordinary" distance (was chosen to be 1-4 squares less than the maximal distance).
   - The number of possible shortest paths to follow - when the initial location is directly above or beside the goal square there is only one short path, but when it is located diagonal to the goal square potentially there are several shortest paths to choose from.

This setting was chosen to be relatively simple but complex enough to examine the learning capacity of the Learning Handler module. The board sizes were chosen to restrict the complexity of path-finding, and the palette was set large enough to enable the three possibilities of task dependency.

All in all, HH AOP played 210 games against the AOPs and additional 300 games against the ten PDAs. The results against the AOPs, by AOP, are presented in table 6. The total correctness rate of HH AOP was 71%, its error rate was 12.8%, and the rest (6.2%) were times when no data was available. The error rate is the highest when cooperation is high because due to the incomplete information, it is very hard to detect it: if the opponent, although cooperative, failed to reach an agreement with our agent (did not have the chips requested, needed them for its own path and so on), the HH AOP calculated it as idleness and considered the agent as uncooperative. In addition, the NA is the highest when the cooperation of the opponent agent is low, since not a lot of negotiations take place, and, thus, the agent cannot estimate the type of its opponents. In contrast, the correctness rate is perfect (100%) when playing against the LL AOP. It was rather straight-forward to detect an LL behavior - this AOP never keeps its commitments. Cosequently, after only two phases, in which LL AOP promised to send chips and never did, HH AOP converged to its true type. HM was the hardest behavior to detect because of its slightly contradicting nature - on the one hand, it is highly cooperative, but on the other hand, it keeps only a portion of its commitments.

	LL	*LM*	*LH*	*MM*	*MH*	*HM*	*HH*
% correct	100	86.67	78.33	73.33	83.33	60	85
% NA	0	0	21.67	5	6.67	5	5
% error	0	13.33	0	21.67	10	35	10

Table 6: Learning results by personality

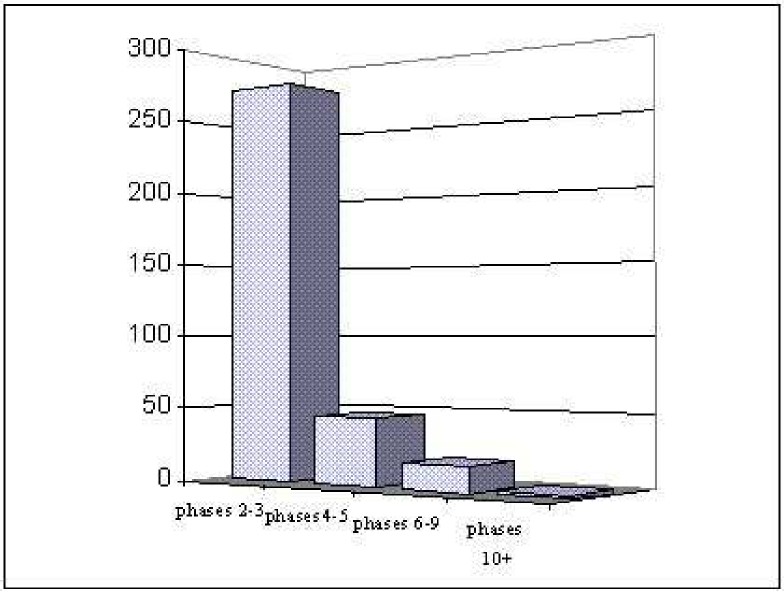

Figure 6: Frequency of phase number HH AOP converged to true type

The average convergence time in all games was 2.8. Figure 6 illustrates the frequency of the different convergence times. This means that after merely three phases on average the Adaptive Multi-personality agent is already able to assign the most suitable personalities to its PDNMs. The elaborated results are listed in Appendix A.

In the additional 300 2-player games in which HH AOP played against the PDAs, it erred only in four of its estimations, resulting in an error rate of only 0.67%. Most of the PDAs are adaptive so verifying the HH AOP's estimations was now done by comparing its output to the game transcripts.

Most of the PDAs designers took special notice of unreliable opponents - they instructed their agents either not to cooperate with liars or to lie in return, meaning that their agent's cooperation and their agent's reliability should be lower. To test this behavior we executed 300 games where LL AOP played against the PDAs, using the same 30 different scenarios described above. Again, we compared the type estimated by the LL AOP at the end of the game, to the type of the PDA. By analyzing the game transcripts, LL AOP did well in estimating its opponents, achieving an error rate of 0.33% (2 errors). Moreover, the PDAs that declared that would act differently towards unreliable opponents, indeed changed their behavior, and the total playing manner of the PDAs was less cooperative and more deceitful.

5.2 Opponents' matching experiments

The goal of this experiments phase was to establish which personality best-suits each opponent type, in the single game environment and in the repeated-games environment. Generally speaking, there is a trade-off between the benefits of being cooperative with the benefits of being exploitive. One must be cooperative in order to encourage its opponents to negotiate with it. However, if one is faced with a highly cooperative opponent, it may be beneficial to exploit it.

Our hypotheses on the best matching schemes are as follows:

1. In a single game environment one must be very careful with its commitments. Since all ramifications of one's actions are ignored as soon as the game ends, we believe that a low cooperative matching scheme will be the best choice.

2. In the repeated-games environment, a more cooperative approach should bring better long-term results. It should encourage communication between the players and establish cooperation amongst them.

3. In all environments, a hybrid matching scheme, which cooperates with the high cooperation agents, and is cautious with the low cooperation agents, should lead to the best performance.

Although we believe these should be the trends of the matching schemes in each environment, the actual personality-type pairs must be determined empirically.

To test our hypotheses and to find the best pairs, we executed four sets of experiments, two sets in the single game environment and two in the repeated-games one. The experiments' goal besides establishing the best matching schemes was also to verify them. Accordingly, in each environment separately, we first executed several 2-player games which served as the basis of the best matching schemes. We expected the results in both environments to be essentially the same since there are limited negotiation opportunities with only one partner in a 2-player game. Moreover, even if an opponent exploits the other agent, it can not be "punished" in the following games (the exploited agent has only the exploiting opponent to interact with). As a result, we created two additional matching schemes in the repeated games environment, based on the results of the 2-player games. They were constructed by increasing the cooperation level, making them a highly cooperative schemes.

Then, we executed verification experiments, testing the suitability of the empirical matching schemes in each environment. The verification experiments were tested on small groups, which were later expanded to include all possible settings, in order to evaluate the Adaptive Multi-personality performance (section 6). In the repeated-games environment, this experiments set also chose the best scheme out of the three we established. Lastly, we created a combined matching scheme - the hybrid matching - merging the best results from the single game environment (section 6.1.1) and the best results from the repeated-games one (section 6.1.2). This scheme is evaluated in section 6.1.3.

5.2.1 Single game environment

We executed 1470 2-player games in which the two players were AOP agents. Additional 60 games involved an AOP agent and one of the two PDAs that expressed a constant type (PDA H and J). Each pair played the 30 different 2-player CT games, discussed in the previous section. For convenience, we denote the first player in the game as PlayerA, and the second one as PlayerB. We recorded PlayerA and PlayerB's scores, and then sorted the pair of scores by PlayerB type (remember that PlayerB is either an AOP or a PDA). At this stage we had a scores-table divided according to all games played against type LL, against type LM and so on. Since we were interested in determining which personality best-suits each type, we averaged the scores of the different PlayerA within the PlayerB types. The PlayerA that reached the highest average score against a certain PlayerB was outputted as the personality that best-suits that type. The results are presented in the second column of table 7.

To verify the results, we executed the same games, but now the Adaptive Multi-personality agent, using the matching scheme, played against one of the six PDAs that were not involved in the matching experiments. Then, the AOPs played the same games against the same PDAs (a total of 1440 games, which is ~96 CPU hours). We compared the performance of the Adaptive Multi-personality agent to the performance of each of the PDAs. The results are presented in figure 7, by the different PDAs (A, B, D, F, G and I). The bars represent the average relative score of the Adaptive Multi-personality agent calculated as follows -

$$Multi\_avg\_relative\_score = \frac{Multi\_avg\_score - AOP\_avg\_score}{AOP\_avg\_score} \tag{13}$$

Thus, if the relative average score is positive, the Adaptive Multi-personality agent average score was higher than the AOP, both playing against the same PDA. Moreover, the higher the bar is, the greater the differences between them (for better or worse). The total average

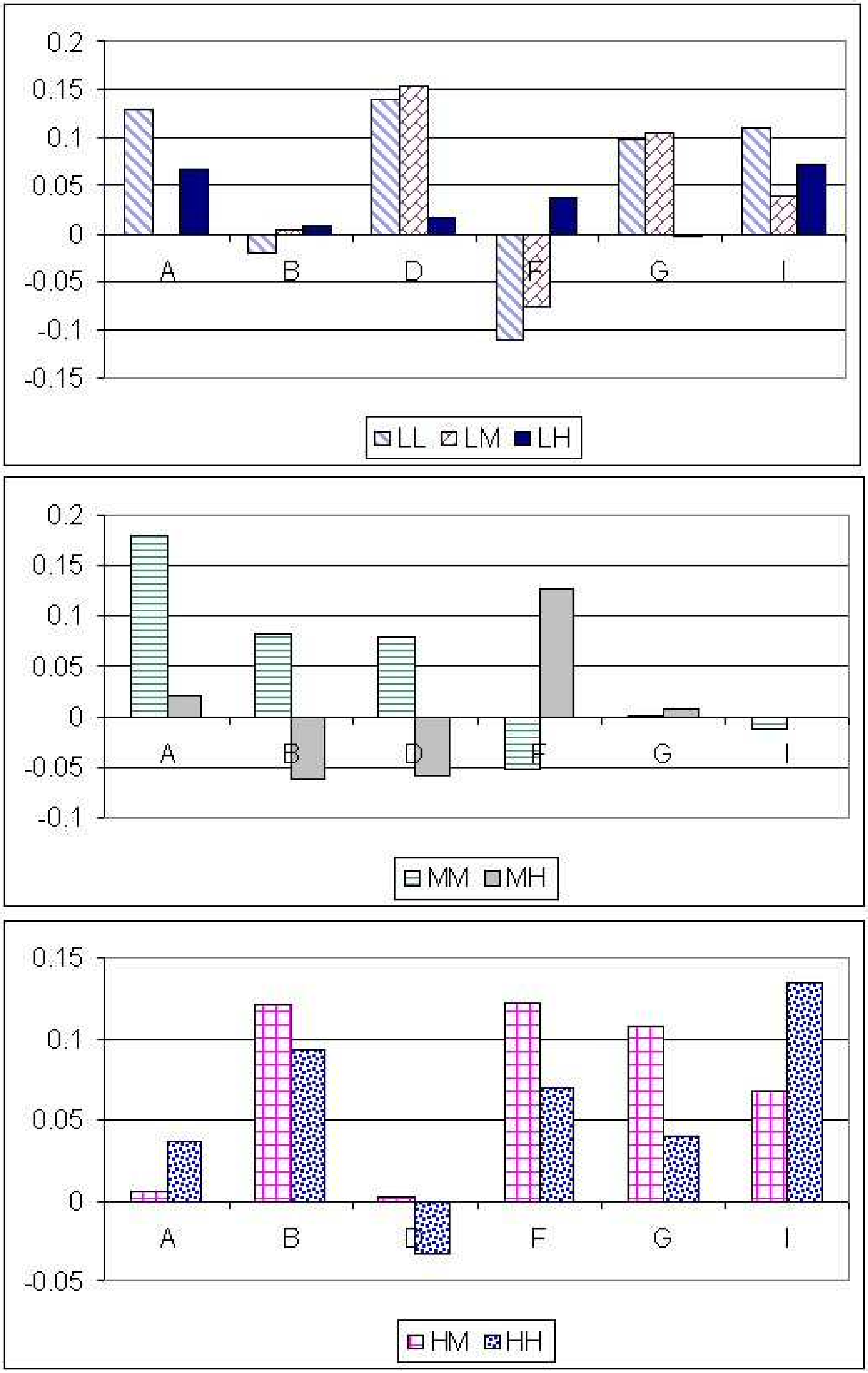

Figure 7: Relative difference between the average score of the Adaptive Multi-personality agent and the average score of the AOPs, by PDA (verification experiment)

relative difference between the average score of the Adaptive Multi-personality agent and the average score of the AOPs was 0.044, meaning it was slightly better than all the AOPs.

5.2.2 Repeated games environment

We executed a 2-player series of three consecutive games. Again, the two players were AOP agents or an AOP agent and a PDA (one of the two mentioned earlier), but this time they were allowed to pass on information from game to game within the series. The variations from one series to another was in the arrangements of its games. As discussed in section 5.1, one can vary several parameters in the CT game (such as board, chips set, possible shortest paths, and so forth). The most significant parameter concerning the agents behavior is the chips sets which determines the task dependency. Thus, in this experiments set we only varied this parameter, which resulted in nine different possibilities for each game within the series, i.e. a total of 729 different series combinations. In conclusion, each pair played 729 series of three consecutive games, totaling 20,412 games (~1360 CPU hours).

The results are shown in the third column of table 7. As expected, the matching was not that different from the single game matching. We devised two adjusted matching schemes, both increase the cooperation level (table 7, column 4 and 5). The first adjusted matching scheme relies on the 2-player results of the single-game. It promotes the cooperation level of that matching scheme up one level, while leaving the reliability level the same (thus it is named single-game adjusted matching). The second adjusted matching adjusts the 2-player results of the repeated-games, by promoting the cooperation level of that matching scheme up one level (repeated-games adjusted matching).

In order to determine which matching scheme produces the best results in the performance of the Adaptive Multi-personality agent, the verification experiment tested all three of them: the matching scheme found in the 2-player games, the single-game adjusted matching and the repeated-games adjusted matching. The design of this experiment was as follows:

- A series of three 4-player games in a row. We used game boards OneSelf, where one player is task independent and the other three are task dependent on it, and AllDep, where all players are task dependent (mentioned in section 4.1), alternatively - either OneSelf-AllDep-OneSelf or AllDep-OneSelf-AllDep. This setting allows us to fully appreciate the quality of the different matching schemes by varying the task dependency of the payers from one game to the other.

- Out of the ten PDAs we used four - the four that achieved the highest average scores in the single game scenario, and learn from game to game (PDA A,C,D,I).

Vs. opponent of type	Single-game Matching	Repeated-games Matching	Single-game Adjusted Matching	Repeated-games Adjusted Matching	Combined Matching
LL	LL	LL	LL	LL	LL
LM	LM	MM	MM	HH	LM
LH	LM	LH	MH	MH	LM
MM	LM	LH	MM	MH	LM
MH	MH	LM	HH	MM	MM
HM	LM	HM	MM	HM	HM
HH	LL	LM	MM	MM	MM

Table 7: Matching Tables

- The other players were the Adaptive Multi-personality agent and two AOPs, two different ones in each game.
- Each player played each role once - a total of four possibilities. This constraint was added so that any advantage given to a player due to its role in the game is eliminated (for instance SelfS player in the OneSelf board).

We executed 168 series of three 4-player games for each of the three matching schemes (the total number of games was 6048, ~403 CPU hours).

The results are summarized in table 8, by AOP type. The Adaptive Multi-personality agent average score in the repeated-games matching was 127.4, in the single-game adjusted matching 138.09, and in the repeated-games adjusted matching 141.09. The results support the notion that being cooperative in repeated games is beneficial (a cooperative matching received the highest average score). Consequently, the repeated-games adjusted matching scheme was assigned to the Adaptive Multi-personality agent in the first evaluation experiments set.

Finally, we constructed the combined matching scheme, which merged the best results from the single game environment and from the repeated-games environment. These results are provided in the last column of table 7, and concur with our hypothesis on the best matching scheme one should use in any environment. This scheme is evaluated in section 6.1.3.

6 Evaluation experiments

We evaluated the performance of the Adaptive Multi-personality by comparing it to the performance of the AOPs and the PDAs in the CT domain. We examined it in two game-playing

Repeated-games Matching Scheme								
	LL	LM	LH	MM	MH	HM	HH	Total
Multi's avg.	75.83	104.46	123.73	130.89	132.92	157.04	166.96	127.4
AOP avg.	62.46	76.38	94.73	91.35	93.31	79.23	80.66	82.59
Single-game Adjusted Matching Scheme								
	LL	LM	LH	MM	MH	HM	HH	Total
Multi's avg.	98.17	132.59	139.56	146.27	149.13	146.68	154.21	138.09
AOP avg.	64.46	80.94	92.98	81.42	90.91	77.16	74.1	80.28
Repeated-games Adjusted Matching Scheme								
	LL	LM	LH	MM	MH	HM	HH	Total
Multi avg	71.59	130.55	140.58	146.73	166.07	170.44	158.69	141.09
AOP avg.	70.28	102.42	101.57	119.12	116.35	118.38	129.9	92.01

Table 8: Comparison between the matching schemes for the repeated games environment

environments - a single-game environment and a repeated-games environment. In each environment the Adaptive Multi-personality agent used a suitable matching scheme.

Our hypothesis is that the Adaptive Multi-personality agent will achieve higher scores than the AOP agents, especially in the repeated-games environment. We believe that an adaptive agent that interacts with each of its opponents with a personality, which was designed to best-suit that opponent, will do much better than an adaptive agent, which has only one personality, especially in the long run. As a result, we expect the Adaptive Multi-personality agent to perform well in the single game environment, but not to reach its full potential. In order to do that, the Adaptive Multi-personality agent must establish its estimations on its opponents' types, and it is hard to accomplish this process in a single game. Furthermore, in this environment less cooperative strategies should achieve higher scores, and thus, LL, LM and LH AOPs, as well as most of the PDAs should do better than in the repeated-games environment.

In the repeated-games environment, on the other hand, the scores of the Adaptive Multi-personality agent should increase significantly, and should reach their full potential. Here it will use a more cooperative matching scheme and due to playing several games in a row, it will be able to succeed in -

1. Establishing cooperative relationships with the medium and high cooperation AOPs and PDAs.
2. Identifying the exploiters and preventing itself from being exploited.

We expect to gradually detect the improvement, as the series of repeating games proceed. In the first game of the series, the Adaptive Multi-personality agent should perform equivalently to the way it did in the single game environment, but in the next few games its scores should increase considerably. Similarly, the medium and high cooperation AOPs and PDAs should do significantly better in this environment than in the single game environment.

In addition, since the PDAs introduced noise to the system (as well as challenge), we expect that in homogeneous games, where all players are the Adaptive Multi-personality agent and the AOP agents, all players will achieve higher scores. Our hypothesis is that the overall behavior of the Adaptive Multi-personality agent performance will reoccur in this settings, with the exception that all scores will be higher. Considering all players in this setting are basically the same, the communication between them should be smoother. Moreover, since the PDAs are not too cooperative, by excluding them from this experiments set, both the Adaptive Multi-personality agent and the AOPs should do much better.

To test our hypotheses we executed thousands of CT 4-player games. The design of the experiments is identical to the one presented in the previous section: we used the AllDep and the OneSelf game boards. The players in each game consisted of the Adaptive Multi-personality agent, one PDA out of the set of Ten PDAs and two AOP agents from the seven available.

In the single game environment, each game was executed 24 times, varying the role of the players in the game, and accommodating all 4! role permutations. In the repeated-games environment, we executed a series of three games in a row, either AllDep-OneSelf-AllDep or OneSelf-AllDep-OneSelf - each series was executed four times. In this environment we also executed several 6-game series, using the boards alternatively as before, in order to examine the robustness of our results. In addition we executed additional games, consisting of the Adaptive Multi-personality agent and the AOPs alone, in order to examine its performance in a more homogeneous surrounding.

The next sections present the comparison results between the Adaptive Multi-personality agent and the AOP agents, and between the Adaptive Multi-personality agent and the PDAs separately. Each section is further divided into the results of the games played in the single game environment, the repeated-games environment with two different matching schemes and the results-robustness games. The comparison to the AOPs also discusses the results of the homogeneous games.

6.1 Comparison to the AOP agents

6.1.1 Single game environment games

In this experiments set 10,080 games were executed, which translates to about 672 CPU hours. The results are summarized in table 9. We compared the performance of the Adaptive Multi-personality agent to the performance of each of the AOP agents. The table's rows represent the different AOPs. Each row holds the average scores of the Adaptive Multi-personality

AOP	Total average		OneSelf board		AllDep board	
	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.
LL	95.88	118.6	91.86	114.94	97.72	120.28
LM	180.18	196.56	180.86	196.96	179.86	196.38
LH	144.13	148.15	131.76	137.82	149.55	152.68
MM	169.77	144.84	165.6	136.84	171.64	148.42
MH	131.18	119.79	117.41	111.08	137.51	123.80
HM	120.33	108.57	111.66	111.92	124.29	107.06
HH	118.18	105.69	119.78	106.68	117.46	105.25
Total Avg.	137.1	134.5	131.28	130.89	139.72	136.27

Table 9: Single game results

and then the average score of the corresponding AOP, in the subset of games where both participated (each AOP participated in $\frac{1}{7}$ of the total number of games). The first column holds their scores in all the games, and the next two columns provide their scores for the two game boards separately.

The low cooperation AOP agents - LL and LM played significantly better than the Adaptive Multi-personality agent (T-test, p-value 0.05). LH achieved a higher average score but it was not significant. In other words, the low cooperation AOP agents succeeded in exploiting the other players in the game. However, if we examine homogeneous games, consisting only of four instances of the same AOP agent, or four instances of the Adaptive Multi-personality agent, The Adaptive Multi-personality agent did much better: LL's score was 54, LM's was 104, LH's was 180 and the Adaptive Multi-personality agent's score was 205. In order to investigate the matter further we executed two more series of experiments with different player arrangements: the first experiment consisted of three homogeneous AOP agents and one Adaptive Multi-personality agent, and the second consisted of one AOP agent and three Adaptive Multi-personality agents (figure 8). The differences between the Adaptive Multi-personality agent and the AOPs are not statistically significant. This suggests that the Adaptive Multi-personality is as good as the low cooperation AOPs in those settings.

Our conclusion is that being low cooperative is beneficial in a single game domain, as long as there is at least one player who is cooperative in the game. Since this cannot be guaranteed, the Adaptive Multi-personality playing strategy proved to be very adequate.

6.1.2 Repeated games environment - Original matching games

In this set, the Adaptive Multi-personality used the repeated-games adjusted matching scheme discussed in section 5.2. Actually this experiments set is the continuation of the matching ex-

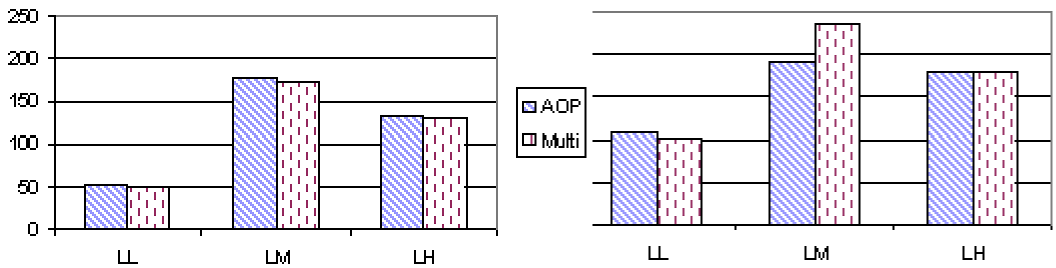

Figure 8: (left) Average scores of three AOP agents and one Adaptive Multi-personality agent (right) Average scores of one AOP agent and three Adaptive Multi-personality agents

periments presented there. While deciding between the three matching schemes, 40% of the games were executed and their scores recorded. The games involving the six remaining PDAs were executed in order to complete the set. Here, 3024 games were played (execution time was ~201 hours). After adding the games in the matching experiment, a total of 5040 games were analyzed.

The results are summarized in table 10. The columns represent the different AOPs. Each column in the table presents the average score of the Adaptive Multi-personality agent and then the average score of the corresponding AOP, in the subset of games in which both participated.

The average score of the Adaptive Multi-personality agent was significantly better than the AOP agents' averages: 147.55 versus 108.57 (T-test, pv<0.001). We can see the improvement of the average score within the series of the games (table 11). The average score of the Adaptive Multi-personality agent improved by 2.45 on average from game 1 to game 2, and by 6.38 on average from game 2 to game 3. The greatest improvement was achieved by the HM AOP agent - by 6.97 on average points from game 1 to game 2, and by 21.67 points on average from game 2 to game 3. The LL AOP agent was the only one whose average score decreased considerably.

Despite the conclusive results, the average score of the Adaptive Multi-personality agent in game 1 is considerably lower than its average score in the single game environment (table 9), in the games where the low cooperation AOP agents and MM participated. The differences are due to the different matching schemes the Adaptive Multi-personality agent used: in the repeated-games environment the matching scheme was more cooperative, and thus, the agent was exploited by the low and medium cooperation AOPs. Consequently, we constructed the combined matching, combining the single matching with the repeated games matching (table 7): the matching of LL, LM, LH and MM was determined according to the single game matching, and all other matching were determined by the repeated games matching. The

	LL	LM	LH	MM	MH	HM	HH
Multi' Avg.	83.54	139.19	138.76	152.25	170.95	175.46	172.69
AOP's Avg.	73.89	105.84	104.40	118.12	111.47	115.46	131.84

Table 10: Repeated Games results (original matching)

AOP	Game 1		Game 2		Game 3	
	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.
LL	72.17	79.12	84.39	81.47	94.09	61.08
LM	140.06	107.94	137.04	100.84	140.46	108.92
LH	135.92	99.12	135.16	99.48	145.21	114.62
MM	154.17	117.81	150.09	116.11	152.49	120.44
MH	163.76	101.23	170.09	107.71	179.00	125.48
HM	171.78	103.59	175.82	110.56	178.76	132.23
HH	168.64	123.97	171.09	131.33	178.34	140.23

Table 11: Repeated games - average scores by game (original matching)

results are presented in section 6.1.3.

In addition, the average scores of the AOP agents in the single game are much higher than in game 1 of the repeated games. This is also true for the Adaptive Multi-personality agent in the games in which low and medium cooperation AOP agents participated. Although this seems inconsistent, there is a significant difference between the two - the Adaptive Multi-personality agent is much more cooperative in the repeated games environment. Thus, on one hand, in the single games it exploited the other players in the game more, and achieved higher scores. The low and medium cooperation AOPs enjoyed this situation as well - since the Adaptive Multi-personality agent did not cooperate with the other two players in the game they were able to further exploit the other players in the game. On the other hand, in the repeated games environment, being more cooperative encouraged more resource exchanges, and as a result, the Adaptive Multi-personality agent's scores increased when playing in games in which high cooperation AOPs participated. In contrast, the averages of the AOPs decreased since the Adaptive Multi-personality agent cooperates more with the other players, and they preferred to negotiate with it.

This surprising outcome suggests that one should use the repeated games adjusted matching scheme not only in the repeated games environment but also in the single game environment. Being cooperative guaranties higher results. This conclusion is consistent with the findings in our previous work ([9]).

The Adaptive Multi-personality agent's average score in the games in which unlearning PDAs participated was significantly higher than its average score in all other games (154.7

	LL	LM	LH	MM	MH	HM	HH
Multi's Avg.	95.92	136.23	135.82	160.15	179.36	183.84	191.55
AOP Avg.	84.26	111.03	106.47	118.49	116.43	117.46	135.46

Table 12: Repeated games, original matching - Average scores when playing with unlearning PDAs

	LL	LM	LH	MM	MH	HM	HH
Multi Avg.	89.35	142.72	143.68	157.27	169.16	180.13	185.7
AOP Avg.	70.25	103.3	106.35	118.15	106.74	107.84	122.45

Table 13: Repeated Games results (combined matching)

vs. 144.49. T-test p-value<0.001). The average scores of the Adaptive Multi-personality agent are presented In table 12. We conclude that using prior knowledge on one's opponents is a very beneficial feature, and despite the computational overhead, it should be incorporated in the agent's design.

6.1.3 Repeated games environment - Combined matching games

In this set, the Adaptive Multi-personality agent used the combined matching scheme. 5040 games were executed (~336 CPU hours).

The results are summarized in table 13. The average score of the Adaptive Multi-personality agent was significantly better than the AOP agents averages (T-test, pv<0.001). We can see the improvement of the average score within the series of the games (table 11). The average score of the Adaptive Multi-personality agent improved by 2.94 on average from game 1 to game 2, and by 5.63 on average from game 2 to game 3. It did not improve as much as in the original matching scheme since it was less cooperative and made a "bad impression".

AOP	Game 1		Game 2		Game 3	
	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.
LL	88.26	74.68	88.03	67.36	91.76	68.73
LM	145.75	109.25	140.84	95.96	141.54	104.41
LH	138.55	102.05	145.06	94.55	147.44	101.24
MM	152.55	119.72	158.2	114.44	161.06	120.3
MH	169.17	104.4	168.96	104.72	169.19	111.26
HM	174.22	117.05	181.99	99.62	184.17	106.86
HH	180.6	127.88	188.34	112.94	188.07	126.7

Table 14: Repeated games - average scores by game (combined matching)

The average score of the Adaptive Multi-personality agent in game 1 is closer to the single game results, and moreover, it significantly improved the results of the original matching (an increase of five points, from 147.5 to 152.5). Moreover, the performance of the Adaptive Multi-personality agent improved against all AOPs (except MH which incurred a slight decrease).

The average score of the Adaptive Multi-personality agent in game 1 was higher than the one achieved in the original matching experiments, and from the one in the single game environment (149.9 vs. 143.8 and 137.1, respectively). As a result, in the following experiments sets the Adaptive Multi-personality agent was assigned the combined matching scheme. This superiority is compatible with our hypothesis regarding the best matching scheme one should deploy.

6.1.4 Results robustness games

In all the experiments in the repeated games environment we limited the number of games in the series to three. We were interested in investigating the performance of the Adaptive Multi-personality agent in longer intervals, and in testing the strength of our previous results. Our hypothesis is that at some point the Adaptive Multi-personality agent's average score will converge into a maximal average score, and will not deviate from it significantly. Furthermore, we believe that the convergence value is close to the average scores of the Adaptive Multi-personality agent received in previous sections, and that our results from previous sections will prevail.

To this end, we executed a total of 4032 games (~269 CPU hours) of the same design, but using only 4 PDAs: the two best ones, against which the Adaptive Multi-personality agent achieved the highest average scores, and the two worst ones, against which the Adaptive Multi-personality agent achieved the lowest average scores.

The results are summarized in table 15. T-test on the differences between the additional three games and game 3 showed that there is not a significant change in the performance of the players, except for the PDAs performance in game 5 and 6 (p-values are 0.046 and less than 0.001 respectively). Nevertheless, the average of the PDAs in those games were 113.23 and 107.2 respectively, which are lower than the average they achieved in game 3. Thus, if there is any difference in the performance of the PDAs with time, it is only for the worst.

Table 16 contains the same results by AOP.

6.1.5 AOP games

In the single game environment, we executed a total of 4032 games composed of the Adaptive Multi-personality agent and three AOPs (~269 CPU hours), and in the repeated games

	Game 1	Game 2	Game 3	Game 4 (pv)	Game 5 (pv)	Game 6 (pv)	Total
AOPs	98.84	82.67	91.5	96.67 (0.65)	102.64 (0.34)	98 (0.87)	95
Multi	149.05	154.81	154.12	148.87 (0.94)	149.32 (0.89)	152.09 (0.5)	151.9
PDAs	122.12	119.11	117.99	112.83 (0.96)	113.23 (0.046)	107.2 (<0.001)	115.46

Table 15: Series of six games results - combined matching

		LL	LM	LH	MM	MH	HM	HH
Game 1	Multi	81.36	130.69	137.74	166.73	172.97	168.65	186.56
	AOP	70.06	101.68	87.06	110.74	103.77	100.44	118.27
Game 2	Multi	93.18	147.39	142.19	154.85	170.81	192.77	183.57
	AOP	79.01	84.95	81.78	99.29	90.77	87.45	86.59
Game 3	Multi	92.26	153.13	145.96	163.18	174.26	182.03	170.11
	AOP	58.22	98.57	85.89	100.51	97.08	90.83	109.28
Game 4	Multi	64.47	139.81	147.77	164.13	173.67	174.2	180.59
	AOP	66.39	108.54	95.84	101.56	95.83	98.78	109.95
Game 5	Multi	69.51	128.89	148.3	150.81	181.48	184.33	183.62
	AOP	74.84	104.13	101.51	112.2	98.52	100.73	126.38
Game 6	Multi	78.95	138.16	146.99	162.06	177.52	179.58	183.16
	AOP	75.31	105.59	90.06	107.38	94.35	98.24	115.17

Table 16: Series of six games in a row - elaborated results by AOP (combined matching)

environment settings, we executed 672 series of three consecutive 4-player games (~134 CPU hours). The Adaptive Multi-personality agent used the combined matching scheme in both environments, which proved to be superior.

The results of the single game environment are summarized in table 17. The Multi personality agent's average score was 174.56, and the AOPs' was 155.65. In the previous set of experiments, when the PDAs participated in the games, their average scores were 137.1 and 134.5, respectively. As expected, the increase is statistically significant (t-test, p-value < 0.001 for both). This supports our hypotheses regarding the ease of communication between the players in this experiments set, and the more cooperative environment they succeeded to create.

The results of the repeated games environment are are summarized in table 18, and in table

	LL	LM	LH	MM	MH	HM	HH
Multi's Avg	104.98	190.66	169.9	199.24	191.99	174.27	190.86
AOP Avg	90.06	170.12	142.1	169.1	165.68	170.14	182.34

Table 17: AOP experiments results - single game environment

	LL	LM	LH	MM	MH	HM	HH
Multi's Avg	100.26	196.28	175.51	191.87	191.75	181.41	192.29
AOP Avg	88.06	160.09	132.72	155.87	159.17	161.75	175.55

Table 18: AOP experiments results - repeated games environment

AOP	Game 1		Game 2		Game 3	
	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.	Multi's Avg.	AOP's Avg.
LL	99.44	87.15	103.78	82.13	97.57	94.9
LM	185.95	158.85	201.8	153.72	201.09	167.7
LH	169.35	136.73	176.47	128.32	180.73	133.13
MM	194.03	151.14	185.51	146.47	196.06	170.01
MH	187.2	161.05	190.74	159.74	197.31	156.73
HM	172.05	164.21	182.56	154.7	189.61	166.34
HH	185.91	182.23	191.84	168.02	199.12	176.4

Table 19: AOP experiments results - repeated games environment, by game

19 by game. As expected both average scores, of the Adaptive Multi-personality agent and of the AOPs, increased significantly: from 152.5 to 175.62 for the Adaptive Multi-personality agent (t-test p-value < 0.001) and from 104.02 to 147.61 for the AOPs (t-test p-value < 0.001). All other conclusions of previous sections prevail here too - the improvement of the Adaptive Multi-personality agent was gradual, as the series proceeded, but it was superior to the AOPs performance nevertheless.

6.2 Comparison to PDAs

The average scores of each PDA is presented in table 20, for the single-game and the repeated games environments, with the original matching scheme and with the combined matching scheme. In all environments the Adaptive Multi-personality agent performed significantly better. In the single game environment the Adaptive Multi-personality agent achieved an average score of 137.1 while the PDAs' average score was 96 (T-test p-value<0.001). This also occurred in the repeated games environment - the average score of the Adaptive Multi-personality agent was 147.5 whereas the PDA's average score was 107.2 in the original matching scheme, and 149.8 versus 105.96 in the combined matching scheme (T-test p-value<0.001).

The results of the 6-game-series experiments are summarized in table 21. The performance of the PDAs seems to deteriorate with the prolongation of the series' length. This can be explained by their overall uncooperativeness - the AOPs and the Adaptive Multi-personality agent "preferred" to interact between themselves if it was possible.

PDA	Single game		Repeated games, original matching		Repeated games, combined matching	
	Multi avg.	PDA avg.	Multi avg.	PDA avg.	Multi avg.	PDA avg.
A	135.36	74.78	118.75	85.91	133.71	113.88
B	142.13	120.17	142.42	116.24	161.98	112.84
C	126.54	84.32	186.58	93.39	138.29	106.18
D	95.08	81.67	99.32	80.53	142.13	119.76
E	136.39	77.9	137.36	94.16	154.99	91.33
F	154.81	100.82	143.75	113.84	171.96	127.56
G	146.94	103.42	188.35	93.36	144.71	101.25
H	139.89	119.51	175.13	104.44	137.09	109.73
I	176.83	122.91	134.5	103.04	138.94	98.53
J	116.54	84.32	149.09	128.75	184.86	129.21

Table 20: Comparison to PDAs

	Multi Average	PDA Average
A	142.12	112.68
F	163.09	107.16
G	132.92	107.34
J	181.23	129.16

Table 21: Series of 6 games in a row - comparison to PDAs (combined matching)

7 Related work

7.1 Agent Design

The notion of developing rational and adaptive agents has been discussed in length in the agent research field [15]. Game theory introduced many models to represent two or more agents' interactions [17]. Our research is based on the fundamental ideas of these theories.

The Adaptive Multi-personality agent was designed according to the logical hierarchy presented by Kraus and Lehmann in [12]. They designed an automated negotiating agent that consists of five modules: a Prime Minster, a Ministry of Defense, a Foreign Office, a Headquarters and Intelligence. These modules were implemented by local agents associated with each module. The structure was used to develop a *Diplomacy* agent which had negotiating capacities - it explained, promised, kept and broke promises. The similarities and differences between the Diplomacy modules to the Adaptive Multi-personality agent is as follows:

- The Prime minister corresponds to the shell of the Adaptive Multi-personality agent. Both encapsulate all other modules and activate them.
  The main difference between the modules is that the Prime Minister is the only module that has personality traits in Diplomacy, whereas the shell of the Adaptive Multi-

personality agent is an empty shell - it simply directs communication to and from the Personalities Coordinator. Another major difference is that the personality of the Prime Minister is constant during the game.

- The Ministry of Defense together with the Foreign Office correspond to the Personalities coordinator, and the Foreign "Desks" (with the information the Ministry of Defense supplies them) are the PDNMs: the "Desks" are responsible for negotiating with the opponent agents, each "Desk" negotiates with one opponent agent - which is quite similar to the PDNMs responsibilities.
  The Foreign Office accumulated the data from the different "Desks" and decides whether to sign agreements. The Coordinator, similarly, accumulated the messages from the PDNMs but it does not decide which offer to accept - this decision is made by the PDNMs. It interferes in the decision making only when a conflict is found in the exchanges the PDNMs are interested in. Then, the Coordinator decides which PDNM will be eventually allocated with the requested resources.
  Another significant difference is that the Foreign Office and its "Desks" are influenced by the personality of the Prime Minister. That is, they do not have their own personality, and, furthermore, they all are influenced by one and only one personality. The personalities of the PDNMs may all be different, and usually are changed when changes in the opponents are detected.

- Intelligence is the learning module. Both are responsible for modeling their opponents in relation to their own observations and to testimonies received by opponent agents.

- The Military Headquarter corresponds to the Domain Manager - both of them are responsible for non-negotiations actions.

Our work is strongly connected to the work done by Jennings and Hogg [10], which also investigated the performance of different personality traits. In particular, their work dealt with socially intelligent agents acting in a social environment. They investigated how agents can make decisions rationally in a social environment, by using social laws that specify to the agents what to do according to a balance between individual and social utilities (Harsanyi's social welfare function). That function simply accumulates all of the other agents' utility functions multiplied by a weight, which determines how important the welfare of the environment is, and adds the utility of Jennings and Hogg's agent multiplied by another weight. Thus, by changing the two weights Jennings and Hogg's agent can vary its character from selfish (the individual agent's weight is the decisive one) to selfless (the other agents' weight is the decisive one). That is, it can change its personality according to our notation.

Their research changed the decision-making mechanism from static to dynamic by adding a Metacontrol above that mechanism. The Metacontrol helped the agent to change the weights in run-time, and thus, to adapt to the society's state and to determine how many resources it can spend in the process of making a decision. And lastly, they inserted a learning feature (Q learning) of the other agents, so that their agent, based on the past interactions, will know which opponent agent to approach when it needs assistance. We also model the opponent agents in order to improve negotiation outcomes. Apart from these strong similarities our research differs from Jennings and Hogg's in the following aspects:

1. Their agent works in a social society - that is, every agent in the environment works to achieve a common goal and to maximize the group gain. At the same time, their agent is also offered some goals of his own, whose gains are granted to it alone. Their agent, thus, is faced with the decision of which goal to pursue first. The environment in which our agent acts is completely unknown, there is no common goal and the opponent agents can follow any known or random policy.

2. The aims of Jennings and Hogg were to examine what is the correct balance of the individual-social utilities while our objectives are to examine whether multiple personality agent performs better than an adaptive one-personality agent. Consequently, their agent changes his strategy (which is determined by the "personality" in our notation) according to the resources available in the society. In contrast, the Adaptive Multi-personality agent holds several personalities, each best-suits an opponent agent in the environment individually.
   Moreover, to evaluate the performance of the Adaptive Multi-personality agent, we used other agents written by other designers (PDAs), in addition to different instantiations of the multi-personality agent (AOPs).

3. Their agent can change its strategy ("personality") at any stage whereas ours almost always keeps a constant personality per opponent agent throughout a given phase. Then, the performance of each opponent in that phase is evaluated and it is decided whether to change the personality facing a specific opponent agent or not. Our approach is more structured. Moreover, agents adapt to their surroundings and act differently to different opponent types. As a result, keeping a constant personality per opponent agent throughout a given phase, gives a more reliable insight to the true type of any opponent, and thus, results in better opponent modeling.

4. Our model is more complex then the model Jennings and Hogg used. The decisions the multi-personality agent has to make are more complicated and it can also interact with several opponent agents simultaneously. Their agent had a limited number of actions to choose from, and it could only interact with one other agent at any given stage. They

tested its performance in the Phoenix fire fighting multi-agent simulation.

Another work that involves an agent that adapts its personality to its environment is the work conducted by Sandip Sen et *al* [19]. Their research focused on developing cooperation strategies within homogeneous groups of agents. Moreover, he researched the behavior of homogeneous agents against exploiting agents in that environment. To this end, a probabilistic reciprocity schemes were developed, in order to help the agents decide when to assist or contact another agent. The experiments involved agents of four types varying from selfish to reciprocative, and the goal of the experiments was to determine in which environments agents find it beneficial to aid other agents. Sen focused this research on homogeneous environments and his probabilistic schemes were used to learn other agents' behavior as a whole, and not individually as we suggest. Another work of Sen and Haynes [20] was generating programs for the coordination of cooperative autonomous agents in pursuit of a common goal. They evolved behavioral strategies to guide the actions of agents in a given domain. In this work they were more interested in developing a model for solving the problem of coordinating cooperative autonomous agent, which was a version of genetic programming.

Glass and Grosz [8] investigated how a social commitment incentive schemes affects agents performance in pursuit of a common goal in a cooperative society. In their model, an individual agent can choose whether to perform a task common to the entire society, which rewards all agents or to perform an independent task, which yields an individual gain. Consequently, an agent makes a decision based on a weighted combination of the gain of the common action with an incentive gain of the individual task. Their aim was to examine how changes in the domain state and the social conscious influence the individual agent and the society. Although they discussed various social strategies, they concentrated on cooperative societies and did not actually grant a personality to their model. They focused on the optimal balance between the individual gain and the social gain.

A similar work was carried out by Zhang and Lesser [21]. They introduced a negotiation mechanism which enables agents to choose attitudes varying from the extremes of self interested and cooperative. They investigated environments of heterogeneous agents of different organizational positions, some of which are of competing companies while the others are from the same group. In their model an agent allocates assignments to another agent, which in turn decides whether to comply or not, depending on the units of MQs (units of value, or resources) the assignments withhold. With time the agent dynamically changes its attitude depending on the agent it negotiates with, that agent's organizational goals, the relationships between them and so forth. Although their work learns the agents in the system individually, their attitude in general is known in advance (their organization and their relationships with the agent is known). A major difference between their work and ours is the reliability notion. They were solely interested in the personality cooperation factor of personality. Moreover, in their model the agents allocate each others assignments while we consider more complex decision problems - in addition to choosing which agent to address and whether to comply to a request or not. Our agents have to decide on exchange rates, decide whether to keep their commitments or not, consider conveying information to other agents and so forth.

The work performed by Castelfranchi [5] involved giving computer systems, and agents in particular, character traits. Castelfranchi tried to examine how a personality could affect the cooperation in a multi-agent system. He tested the effect of various personality combinations, including those of goal conflict. Accordingly, he designed a multi-agent world, which encapsulated different social attitudes and personalities in interacting and in delegating tasks to other agents, called GOLEM. In continuation to this work Castelfranchi investigated how deception can work towards the benefit of a personality-based agent, in cases of goal conflicts [4]. Similarly, the Adaptive Multi-personality agent's PDNMs were assigned deception and cooperation abilities based on Castelfranchi results. In [7] Castelfranchi continued to investigate the effect of designing personality-based agents, this time by interfacing with human beings. He designed several systems with a personality that matched its purpose: a cooperative multi-agent system, with such traits as selfish, lazy, cooperative and so on; a conflict resolution dialog whose agents were anxious, conservative altruistic and so on ; or an instruction giving system, in which the personality traits were unique for this domain. Our work expands his designs - we implement the existence of multiple personalities in one agent, which evolve over time. Castelfranchi's agents had one personality, which stayed constant throughout its life.

Carlsson and Johansson [3] investigated three types of strategies - generous, even-matched and greedy, as concepts for analyzing games. They granted the participating agents a strategy and examined their performances in two types of games. Their aim was to determine which kind of strategy was preferable and in which environments. They kept the strategy of an agent constant throughout a game, although they realized that adopting a more adaptive strategy could be used. Our personality-type matching experiments design relies on their work. We also established which type of strategy (personality) was preferable against each opponent type. However, in a way, we continue their work by designing an agent, which uses all possible strategies, often simultaneously, depending on its surroundings.

7.2 Learning

The learning module of the Adaptive Multi-personality agent combines all information available on its opponent agents. It mainly learns from its own observations, but when testimonies are sent to it, it incorporates them into its estimations.

The learning model presented by Schillo et *al* at [18] did just that - the model combined the knowledge gathered by the agent and testimonies sent by *witness* agents. In that paper they formulated the concept of trust (which can be translated into reliability in our notation after mild modifications) and presented an algorithm for learning trustworthiness of others by the combination of observations and testimonies (here deception plays a role as well). Their model resembles our environment in the following aspects (in their notation):

1. Each agent knows only his own behavior / social attitude.
2. They introduced fuzziness into the social roles by not assigning an absolute trait but by replacing it with a probability factor to act according to that trait.
3. They focused on two social attitudes ranging from egoistic to altruistic, from honest to dishonest.

Their research differs from ours in the following aspects:

1. Although they inserted fuzziness, their agent still has the same basic personality, with which it faces every opponent.
2. They chose to implement their ideas in the disclosed prisoners dilemma with partner selection, which strongly varies from our environment
3. They focused on the question of how to choose the best partners, while we try to find out how to play in the best manner. However, we dealt with this question implicitly as well. By modeling the Adaptive Multi-personality agent's opponent, preference was given to the cooperative over the uncooperative ones.

Sabater and Sierra in [16] presented a slightly different method - they developed a system called REGRET, which calculates reputation of agents by incorporating the use of social network analysis (*sociograms*). In their domain there were several types of agents, all having full visibility of each others actions and agreements. The system was based on three types, or dimensions, of reputation:

- Individual dimension - models the direct interaction between two agents, from one of these agents' subjective point of view. In their model, this dimension was calculated as the weighted mean of all the agent's observations, giving more relevance to recent observations. In our domain, since there isn't full visibility, the Adaptive Multi-personality agent calculates this dimension from its own observations (equations 1 and 2)
- Social dimension - uses information that arises from other agents in the system. They defined three types of social reputation:

- Witness reputation - is based on the information other agents deliver on a certain agent (*target* agent). They also assumed that testimonies may be false, or correlated. As a result they defined fuzzy rules to determine the reliability of a witness agent, given its social relations with the target agent. Then they multiplied the normalized value of the reliability of the witness agent with the value they assigned to its testimony. We did just that in our model (equations 3 and 4).
- Neighborhood reputation - uses the neighbors of the target agent in the sociogram, and their relations with it. In our domain this data is not available.
- System reputation - a default reputation value which is based on the target agent's role in the system. The REGRET system has a institutional structure in the sense that all agents in the system have observable features by which their role in the structure can be identified. Thus, a default reputation value can be associated with each role. In our model such assumptions are not feasible.

• Ontological reputation - is used to combine different aspects of reputation into a complex one. In the REGRET system this notion was represented with a graph, and the complex reputation of a given aspect is the weighted sum of the reputation of all its related aspects. Again, in our domain this data is not available

Finally, they combined all those dimensions into one reputation value by calculating the weighted sum of all the dimensions. The Learning Handler module of the Adaptive Multi-personality agent was implemented using these ideas: in our domain only the individual dimension and the social dimension (witness reputation) are valid. The Learning handler combines the two into one quantity, which serves as the estimator of the opponents types. (equation 5).

8 Summary and future work

In this thesis we have proposed a new design for an automated agent that is able to perform well in multi-agents environments. Unlike the designs available nowadays in the multi-agents systems, the Adaptive Multi-personality interacts simultaneously with a large number of agents by matching each such agent a personality that best-suits it. As a result, the negotiations between each personality-type pair ensue a great gain for our agent. The design of our agent can be implemented in any multi-agents system allowing negotiations. For instance, it can be used in a commerce site, where each buyer's behavior is modeled, and, accordingly, a suitable marketing strategy is matched. We investigated the performance of the agent in two environments, a single-game environment and a repeated-games one, and against 17 different opponent agents using different strategies and designs. Experimental results have shown that the agent is able to cope successfully with all opponent agent types and in all environments. Moreover, the different matching schemes we have presented support the notion we also stated in our previous work: when interacting with agents of an unknown type a cautious cooperative approach should yield the best results.

There are several directions for further research: firstly, it would be interesting to determine how well the Adaptive Multi-personality agent will perform in different domains and against several more types of opponent agents. We are especially interested in inserting the notion of social dependency by adding a social dependency weight to the scoring rule. Then, the players' score would combine the players' individual-performance scores and a weighted average of the individual scores of the other players. Introducing a social dependency weight should increase helpful behavior. For instance, in the CT domain, this should result in more frequent chips exchanges and even several "free gives" (sending chips without expecting anything in return). The Adaptive Multi-personality agent should embrace more cooperative personalities in that domain. Nevertheless, we strongly believe that the Adaptive Multi-personality agent could do well in that domain as well, after small modifications. Another change in the domain is the agents acting within it. The question of how well the Adaptive Multi-personality agent will do after introducing better opponents agents is still open. Although we had the PDAs for possible comparison with the Adaptive Multi-personality agent to, they did quite poorly, and thus did not provide a good insight.

Secondly, there are several improvements we would like to make in the Adaptive Multi-personality agent design. For now the matching of personalities-opponents' types is done offline. Consecutively, the matching scheme stays constant in all the games the agent plays, without any possibility to change it if a certain pair produces bad results. We would like to improve the performance of the Adaptive Multi-personality agent by making it online: while the series of games is in progress, the agent may view its scores and make adjustments to its "Matching Table". Further work is required to implement a module for information transfer. Although the CT domain allowes this feature, our current design does not support this module yet. The information may be true or false, regarding the other agent's types and resources or the Multi personality's type and resources. After assimilating this module in the agent, we would bw interested in investigating what sort of information would be worthwhile to transfer, when to do the actual transfer, to which opponent agents and so on.

9 Appendix A - learning experiments results

The table below presents the results of the learning experiments discussed in section 5.1. It holds the output of the Learning Handler module in 30 different games it played against all AOPs. If the Handler converged to the true type of its opponent, the corresponding cell holds the first phase number the convergence occurred: first the convergence time for the cooperation level and then the convergence time for the reliability factor. If no data was available on the opponent's type, a NA symbol appears. Otherwise, the Handler failed to detect the opponent's true type, and an error notation (X) is applied.

i	Scenario	LL	LM	LH	MM	MH	HM	HH
1	B1 RO-RO MAX-STR	2	2 / 4	2 / NA	5 / 3	4 / 2	2 / 3	2 / 2
2	B1 RO-NA MAX-SEV	2	2 / 10	2 / 3	4 / 3	4 / 2	2 / 3	2 / 2
3	B1 CR-CR ORD-STR	2	2 / X	2 / NA	X / NA	X / NA	X / NA	X / NA
4	B1 CR-RO ORD-SEV	2	2 / X	2 / NA	7 / 8	4 / 2	X / 8	3 / 2
5	B1 NA-NA MAX-STR	2	2 / 3	2 / 3	4 / 2	5 / 2	X / 2	2 / 2
6	B1 NA-CR MAX-SEV	2	2 / 5	2 / 3	4 / 3	3 / 2	X / 3	2 / 2
7	B2 CR-RO ORD-STR	2	2 / X	2 / NA	X	X / NA	3 / X	3 / 2
8	B2 CR-NA ORD -SEV	2	2 / 6	2 / 3	X	3 / 2	2 / 2	X / 2
9	B2 RO-CR MAX-STR	2	3 / X	2 / 3	9 / 13	3 / 2	X / 2	X / 2
10	B2 RO-RO MAX-SEV	2	7 / 5	2 / NA	4 / 2	5 / 2	3 / 2	2 / 2
11	B2 NA-NA ORD-STR	2	2 / 6	2 / 3	5 / 3	8 / 2	2 / 3	2 / 2
12	B2 NA-CR ORD-SEV	2	2 / 5	2 / 3	3 / 3	9 / 2	2 / X	2 / 2
13	B3 RO-RO ORD-STR	2	2 / 3	2 / NA	5 / 3	4 / 2	2 / 3	2 / 2
14	B3 RO-NA ORD-SEV	2	2 / 5	2 / 3	3 / 7	7 / 2	2 / 3	2 / 2
15	B3 NA-CR MAX-STR	2	2 / 3	2 / 3	4 / 3	3 / 2	X / 2	2 / 2
16	B3 NA-RO MAX-SEV	2	2 / 5	2 / NA	5 / 3	4 / 2	X / 4	2 / 2
17	B3 CR-NA ORD-STR	2	2 / 7	2 / 3	X	3 / 2	2 / 3	2 / 2
18	B3 CR-CR ORD-SEV	2	2 / X	2 / NA	X / NA	X / NA	X / NA	X / NA
19	B4 RO-RO ORD-STR	2	2 / 5	2 / NA	5 / 3	4 / 2	2 / 3	2 / 2
20	B4 RO-NA ORD-SEV	2	2 / X	2 / 3	4 / 2	6 / 2	X / 2	2 / 2
21	B4 CR-CR MAX-STR	2	2 / X	2 / NA	X / NA	X / NA	X / NA	X / NA
22	B4 CR-RO MAX-SEV	2	2 / 4	2 / NA	X / 3	X / 2	4 / 4	3 / 2
23	B4 NA-NA ORD-STR	2	2 / 4	2 / 3	2 / 2	5 / 2	X / 3	3 / 2
24	B4 NA-CR ORD-SEV	2	2 / 8	2 / 3	6 / 6	5 / 2	X / 2	2 / 2
25	B5 CR-RO ORD-STR	2	2 / 4	2 / NA	6 / 2	3 / 2	X / 4	X / 2
26	B5 CR-NA ORD-SEV	2	2 / X	2 / 3	2 / X	2 / 2	X	2 / 2
27	B5 RO-CR MAX-STR	2	2 / 5	2 / 3	X / 4	X / 2	X	2 / 2
28	B5 CR-RO MAX-SEV	2	2 / 4	2 / NA	X / 2	4 / 4	2 / 3	2 / 2
29	B5 RO-NA ORD-STR	2	2 / 5	2 / 3	4 / 3	4 / 2	X / 2	2 / 2
30	B5 NA-CR ORD-SEV	2	2 / 4	2 / 3	5 / 2	3 / 2	X / 4	2 / 2

Legend:

B1-B5 - CT game boards
RO - Reach Alone (Task independent)
NA - Need Assistance (Task dependent)
CR - Can't Reach (Unassisted dependent)
MAX - MAXimal distance from goal square
ORD - ORDinary distance from goal square
SEV - SEVeral shortest paths to goal square
STR - One shortest path to goal square (STRaight one)
X - error
NA - estimation Not Available
m / n - number of phases cooperation / reliability estimations converged to true type. If only one number appears, both estimations converged at the same specified phase.

10 Appendix B - values of the $\alpha, \beta, \gamma, \delta$ weights in the CT game

The table below summarizes the $\alpha, \beta, \gamma, \delta$ weights for all possible actions in the CT domain:

Action	Alpha	Beta	Gamma	Delta
Initiate request chips message	0.50	0.30	0.75	0.10
Initiate a 1:1 exchange message	1.00	0.50	0.95	0.10
Initiate a 2:1 exchange message	1.00	0.50	0.95	0.30
Initiate a 1:2 exchange message	1.00	0.50	0.95	0.30
Send a refuse message	0.50	0.50	0.75	0.05
Commit to an exchange when having the chips	1.00	0.25	0.95	0.50
Commit to an exchange while not having the chips	1.00	0.25	0.95	0.50
Send chips as promised	0.25	0.95	0.50	0.40
Send partial subset of the chips promised	0.25	0.95	0.50	0.40
Do not keep a commitment	0.25	0.95	0.50	0.40
Initiate a "liar" message of an unreliable opponent	0.05	0.25	0.75	0.20
Initiate a "liar" message of a reliable opponent	0.05	0.25	0.75	0.20
Ignore message / stay idle	0.25	0.75	0.95	0.10

11 Appendix C - list of "positive", "negative" and "neutral" actions in the CT domain

The table below presents the different types of actions in the CT domain:

"Positive"	"Negative"	"Neutral"
Initiate a 2:1 exchange message	Initiate a 1:2 exchange message	Initiate a 1:1 exchange message
Commit to an exchange when having the chips	Commit to an exchange while not having the chips	-
Send chips as promised	Do not keep commitments	Send subset of the chips promised
Initiate a "liar" message of an unreliable opponent agent	Initiate a "liar" message of a reliable opponent agent	-
-	-	Initiate a request chips message

References

[1] Creating socially interactive non player characters: The ţ-sic system.

[2] Frances M. T. Brazier, Pascal van Eck, and Jan Treur. Modelling competitive co-operation of agents in a compositional multi-agent framework. In *Knowledge Acquisition, Modeling and Management*, pages 317–322, 1997.

[3] B. Carlsson and S. Johansson. Generous and greedy strategies. In *Proceedings of Complex Systems*, 1998.

[4] C. Castelfranchi, R. Falcone, and F. de Rosis. Deceiving in golem: How to strategically pilfer help. *In Autonomous Agent '98: Working notes of the Workshop on Deception, Fraud and Trust in Agent Societies*, 1998.

[5] C. Castelfranchi, D. Fiorella, F. Rino, and P. Salvatore. A testbed for investigating personality-based multiagent cooperation. In *in Proceedings of ESSLLI'97 Symposium on Logical Approaches to Agent Modelling and Design*, pages pp.23–35, Aix-en-Provence, France, August 1997.

[6] L. Chittaro and M. Serra journal=. Behavioral programming of autonomous characters based on probabilistic automata and personality.

[7] F. de Rosis and C. Castelfranchi. How can personality factors contribute to make agents more 'believable'? In *Proceedings of the I3 Spring Days Workshop on 'Behaviour Planning for Lifelike Characters and Avatars'*, Sitges, 1999.

[8] A. Glass and B. Grosz. Socially conscious decision-making. In Carles Sierra, Maria Gini, and Jeffrey S. Rosenschein, editors, *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 217–224, Barcelona, Catalonia, Spain, 2000. ACM Press.

[9] B. Grosz, S. Kraus, S. Talman, B. Stossel, and M. Havlin. The influence of social dependencies on decision-making: Initial investigations with a new game. In *Proceedings of AAMAS-2004*.

[10] L.M.J. Hogg and N.R. Jennings. Socially intelligent reasoning for autonomous agents. *IEEE Trans on Systems, Man and Cybernetics - Part A 31(5)*, pages 381–399, 2001.

[11] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 8(3):223–250, 1993.

[12] S. Kraus and D. Lehmann. Designing and building a negotiating automated agent. *Comput. Intell. 11*, pages 132–171, 1995.

[13] S.S. Nemani and V.H. Allan. Mates: A conversational agent system. In *Affective Computational Entities at the 17th European Meeting on Cybernetics and Systems Research*, 2004.

[14] W. T. Norman. Towards an adequate taxonomy of personality attributes: Replicated factor structure in peer nomination personality ratings. *Journal of Abnormal and Social Psychology*, 66:574–583, 1963.

[15] S. Russell. Rationality and intelligence. *Artificial Intelligence*, vol.94, no. 1:55–57, 1997.

[16] J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 475–482. ACM Press, 2002.

[17] T.C. Schelling. *The strategy of conflict*. Cambridge, MA: Harvard University Press, 1963.

[18] M. Schillo, P. Funk, and M. Rovatsos. Using trust for detecting deceitful agents in artificial societites. *In Applied Artificial Intelligence, Special Issue on Trust, Deception and Fraud in Agent Societies*, 2000.

[19] S. Sen, A. Biswas, and S. Debnath. Believing others: Pros and cons. In *Proceedings of the 4th International Conference on MulitAgent Systems*, pages 279–285, Boston, MA, July 2000.

[20] D. Schoenefeld T. Haynes, S. Sen and R. Wainwright. Evolving a team. In E. V. Siegel and J. R. Koza, editors, *Working Notes for the AAAI Symposium on Genetic Programming*, Cambridge, MA, 1995. AAAI.

[21] V. Lesser X. Zhang and T. Wagner. Integrative negotiation in complex organizational agent systems. In *the Proceedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2002), research abstract*, pages 503–504, Bologna, Italy, July 2002.