Chapter 1

Introduction

Multi-Agent Systems (MAS) is the subfield of Artificial Intelligence (AI) that aims to provide both principles for construction of complex systems involving multiple *agents* and mechanisms for coordination of independent agents' *behaviors*. While there is no generally accepted definition of "agent" in AI, for the purposes of this dissertation, an agent is an entity with perceptions, goals, cognition, actions, and domain knowledge, situated in an environment. The ways it acts, or its mappings from perceptions over time to actions, are called its "behaviors."

When a group of agents in a multi-agent system share a common long-term goal, they can be said to form a *team*. Team members (or teammates) coordinate their behaviors by adopting compatible cognitive processes and by directly affecting each others' perceptory inputs including via communicative actions. Other agents in the environment that have goals opposed to the team's long-term goal are the team's *adversaries*.

This thesis contributes several techniques for generating successful team behaviors in real-time, noisy, collaborative and adversarial multi-agent environments. As of yet, there has been little work within this type of multi-agent system.

Because of the inherent complexity of this type of multi-agent system, Machine Learning (ML) is an interesting and promising area to merge with MAS. ML has the potential to provide robust mechanisms that leverage upon experience to equip agents with a large spectrum of behaviors, ranging from effective individual performance in a team, to collaborative achievement of independently and jointly set high-level goals. Using a hierarchical task decomposition, multiple ML modules can be combined to produce more effective behaviors than a monolithic ML module that learns straight from sensors to actuators.

By learning portions of their cognitive processes, selectively communicating with each other, and coordinating their behaviors via common knowledge, a group of independent agents can work together towards a common goal in a complex, real-time, noisy, adversarial environment.

1.1 Motivation

The line of research addressed in this dissertation was originally inspired by the Dynamite test bed [Barman *et al.* 93]. Their successful implementation of two non-holonomic robots pushing a ping-pong ball towards opposite goals on a walled field was the first real robotic soccer system. While limited to a single robot on each team in 1994, Dynamite demonstrated the feasibility of creating robotic soccer *agents* for studying real-time, collaborative and adversarial multi-agent systems. Inspired by their success, we set out to create a robotic soccer system with several robots on each *team*. In so doing, it was immediately clear that the research issues introduced by the extension from single to multiple robot teams would be numerous and challenging.

Dynamite operates in a real-time, noisy, adversarial domain. This thesis focuses on a domains with those same three characteristics plus an additional one:

Real-time domains are those in which success depends on acting in response to a dynamically changing environment.

Noisy domains are those in which agents cannot accurately perceive the world, nor can they accurately affect it.

Collaborative domains are those in which a group of agents share a common goal.

Adversarial domains are those in which there are agents with competing goals.

While no previous system has addressed a domain with all of these characteristics, there have been previous systems examining each of these characteristics individually. Pengi [Agre and Chapman 87] is a real-time system in which an agent in a video game environment must act quickly in response to a dynamically changing environment. All robotic domains (e.g. [Brooks 86, Arkin 87]) are noisy as real-world sensors perceive the world imperfectly and robotic actuators are typically unreliable. Collaborative environments have been examined in the field of MAS [Grosz 96, Sycara *et al.* 96, Cohen *et al.* 99], and adversarial domains have been considered in AI game playing systems such as for checkers [Samuel 59] and chess [Newell and Simon 72]. However these adversarial domains are turn-taking as opposed to real-time, they are not noisy, and there are no collaborative agents.

Some of these four domain characteristics have also been considered in combination. Several multi-agent robotic systems [Arkin 92, Mataric 94a, Parker 94] consider collaborative, noisy environments. Dynamite, mentioned above, operates in a real-time, noisy, adversarial environment. In this thesis, I consider real-time, noisy, collaborative and adversarial multi-agent domains, using simulated robotic soccer with multiple agents on each team as a representative test bed. The challenges that arise in this domain have motivated my thesis research.

One of the first observations made was that creating all of the behaviors and agent interactions directly would be an impossible task in such a complex multi-agent domain. Thus, a primary research goal of this thesis is the investigation of whether and how ML techniques can be used to create a team of cooperating agents.

1.2 Objectives and Approach

1.2.1 Thesis Question

The principal question addressed in this thesis is:

Can agents learn to become individually skilled and to work together in the presence of both teammates and adversaries in a real-time, noisy environment with limited communication?

More specifically, the thesis contributes an agent structure enabling the use of ML techniques to improve an agent's behavior in domains with the following characteristics:

- A need for real-time decision-making;
- Sensor and actuator noise with hidden state.
- Several independent agents with the same well-defined high-level goal: *teammates*
- Several agents with a conflicting well-defined high-level goal: *adversaries*

Note that the intermediate or low-level goals of teammates and adversaries can differ indeterminately.

The agents are assumed to have at their disposal the following resources:

- Sensors of the environment that give partial, noisy information;
- The ability to process the sensory information and use it to update a world model;
- Noisy actuators that affect the environment;
- Low-bandwidth, unreliable communication capabilities.

This thesis contributes a successful method—using machine learning—of equipping such agents with effective behaviors in such a domain.

1.2.2 Approach

The general approach to answering the thesis question has been to create an existence proof: a full-fledged functioning multi-agent system that incorporates learning in a real-time, noisy environment with both teammates and adversaries.

Robotic soccer is a domain which fits the above characteristics while being both accessible and suitably complex. Insofar as the main goal of any test bed is to facilitate the trial and evaluation of ideas that have promise in the real world, robotic soccer proved to be an excellent test bed for this thesis. All of the thesis contributions were originally developed in *simulated* robotic soccer. However, some were then applied in either real robotic soccer or in network routing.

An initial assumption was that, due to the complexity of the environment, agents in domains with the above characteristics would not be able to learn effective direct mappings from their sensors to their actuators, even when saving past states of the world. Thus, the approach taken was to break the problem down into several behavioral layers and to use ML techniques when appropriate. Starting with low-level behaviors, the process of creating new behavior levels and new ML subtasks continues towards high level strategic behaviors that take into account both teammate and opponent strategies.

In the process, a key advantage of such an approach surfaced: learning low-level behaviors can facilitate the learning of higher level behaviors. Either by creating the behavioral components that make up the new learned behavior or by manipulating the input space of the new learned behavior, previously learned behaviors can enable the creation of increasingly complex learned behaviors. This new approach to multi-agent machine learning is called *layered learning.*

Layered learning assumes that the appropriate behavior granularity and the aspects of the behaviors to be learned are determined as a function of the specific domain. Layered learning does not include an automated hierarchical decomposition of the task. Each layer is learned by applying or developing appropriate ML algorithms. The thesis methodology consisted of investigating several levels of learning to demonstrate the effectiveness of combining multiple machine learning modules hierarchically.

1.3 Contributions

This thesis makes four distinct contributions to the fields of MAS and ML.

Team Member Agent Architecture. The team member agent architecture is suitable for domains with periodic opportunities for safe, full communication among team members, interleaved with periods of limited communication and a need for real-time action. This architecture includes mechanisms for task decomposition and dynamic role assignment as well as for communication in single-channel, low-bandwidth communication environments. It is implemented in both simulated and real robotic soccer.

Layered Learning. Layered learning is a hierarchical ML paradigm that combines multiple machine learning modules, each directly affecting the next. Layered learning is described in general and then illustrated as a set of three interconnecting learned behaviors within a complex, real-time, collaborative and adversarial domain.

Team-Partitioned, Opaque-Transition Reinforcement Learning (TPOT-RL). TPOT-RL is a new multi-agent reinforcement learning method applicable in domains in which agents have limited information about environmental state transitions. TPOT-RL partitions the learning task among team members, allowing the agents to learn simultaneously by directly observing the long-term effects of their actions in the environment. TPOT-RL is demonstrated to be effective both in robotic soccer and in another multi-agent domain, network routing.

The CMUnited Simulated Robotic Soccer System (CMUnited). The CMUnited simulated robotic soccer team is a fully implemented and operational team of simulated robotic agents. Coupled with the detailed descriptions and source code in the appendices, CMUnited's algorithmic details presented throughout the thesis should enable future researchers to build upon CMUnited's successful implementation.

While the last of these contributions is inherently domain-specific, for each of the first three contributions, this thesis provides a general specification as well as an implementation within a complex domain: simulated robotic soccer. In addition, opportunities for generalization to other domains are discussed, illustrated, and implemented.

1.4 Reader's Guide to the Thesis

Following is a general description of the contents of each chapter as well as guidelines as to which chapters are relevant to which contribution. Since CMUnited is described throughout the entire thesis, all chapters are relevant and it is not mentioned in the guidelines.

Chapter 2 introduces the domains used as test beds within the thesis: simulated and real robotic soccer as well as network routing. The simulated robotic soccer domain is used as an example throughout the thesis and all contributions are implemented in this domain. The team member agent architecture is also implemented in the real robotic soccer domain, while TPOT-RL is also implemented in network routing.

Chapter 3 describes the team member agent architecture. It is defined generally and then implemented in both simulated and real robotic soccer. This architecture is used within the implementations of both layered learning and TPOT-RL.

Chapter 4 introduces the general layered learning method. The implementation of layered learning described in Chapters 5–7 also contribute to the complete description of layered learning.

Chapter 5 describes a learned, individual robotic soccer behavior. It is presented as the first learned layer in our layered learning implementation.

Chapter 6 presents a learned multi-agent behavior built upon the individual behavior from Chapter 5. This second learned layer in our layered learning implementation also serves as the basis for the team learning algorithm described in Chapter 7, namely TPOT-RL.

Chapter 7 introduces the novel multi-agent reinforcement learning algorithm, TPOT-RL. TPOT-RL builds upon the multi-agent behavior described in Chapter 6 to create a team-level collaborative and adversarial behavior. This team-level behavior is the highest implemented learned layer in the simulated robotic soccer implementation of the layered learning paradigm. The chapter also includes a description and implementation of TPOT-RL in network routing, demonstrating its generalization to other multi-agent domains.

Chapter 8 provides general results of the implemented robotic soccer systems in international competitions. While focussed evaluations are provided in several of the previous chapters, anecdotal evidence of the success of the overall approach is provided in this chapter.

Chapter 9 provides an extensive survey of work related to the research in this thesis. While several chapters contain their own related work sections describing the research most relevant to their contents, this chapter surveys the field of MAS from a machine learning perspective, and presents a broad overview of robotic soccer research around the world.

Chapter 10 summarizes the contributions of this thesis and outlines the most promising directions for future work.

Table 1.1 summarizes the relevance of each chapter to the thesis contributions.

	Chapter								
Contribution	2	3	4	5	6	7	8	9	10
Team Member Agent Architecture	$*$	$*$	$-$	$-$	$-$	$-$	$+$	$+$	$+$
Layered Learning	$*$	$+$	$*$	$*$	$*$	$*$	$+$	$+$	$+$
TPOT-RL	$*$	$+$	$-$	$-$	$+$	$*$	$+$	$+$	$+$
CMUnited	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$+$

$*$: essential; $+$: relevant; $-$: irrelevant

Table 1.1: Correspondence between the thesis contributions and the dissertation chapters.