Simulation League, Team *Gongeroos99*, pages 40–44
http://www.ep.liu.se/ea/cis/1999/007/09/

Gongeroos'99 Team

Gongeroos99

Chee Fon Chang, Aditya Ghose, Peter Harvey, Justin Lipman

CC, AG, PH, JL: Decision Systems Laboratory, Department of Business Systems, University Of Wollongong

Abstract. *This article presents Gongeroos'99 approach to the Robocup simulator league challenge.*

1 Introduction

The RoboCup simulator league provides a unique real-time, multi-agent domain in which to experiment with research interests. The complexity of the task to be completed and the unreliable low-bandwidth inter-agent communication is of particular interest to us. This article presents Gongeroos'99 approach to the RoboCup simulator challenge. Section 2 introduces our overall agent architecture. Section 3 presents our team-oriented approach that attempts to solve the problems of task complexity and restricted communication. Section 4 describes our machine learning approach. Section 5 concludes.

2 Agent Architecture

Our agent architecture is a variant of the well-known BDI architecture. As with the BDI architecture, our agents are based on the following core data structures: beliefs B, desires D, intentions I and plans PL [3]. Therefore, our agent is denoted by a 4-tuple <B,D,I,PL>.

Definition 1: The belief state B of an agent consists of descriptions of the agent's environment E, a set S denoting the agent's playing skills, and a domain theory M. We denote the agent's belief state as a 3-tuple <E,S,M>. The environment E is denoted by a collection of sensory parameters including positions of moving and stationary objects (ball, players, field), time and information accuracy. The skill set S denotes the repertoire of actions available to the agent. These skills would incorporate atomic actions such as kicking and turning. S also contains complex skills that make use of other skills within the same set. These actions would be executed when a plan is executed. M is the domain theory that describes how actions affect states of the environment. It contains casual rules that determine the effects of actions in various states of the environment as well as (possibly partial) specifications of how actions map from one state of the environment to another.

Definition 2: The set D of desires of the agent is consisted of two categories of desire, an offensive or a defensive desire. Only one desire exists at any instance. The selection of offensive or defensive desires is derived from the team-oriented components of our agent (detailed later).

Definition 3: Intention I represents a set of intentions resulting from a single desire that the agent decided to pursue. These intentions are categorised into either tactical or reactive. This allows the agent to consider the intentions in the short-term Re and mid-term Ta. Re intentions share a common trait that their impact on the game is limited and their relevance is based on triggers that are derived directly from the world-state or changes that occur in the world-state. Re intentions are not expected to impact the game for more than a few seconds. Ta, however, tend to impact the game for a longer period and are based only partially upon triggers within the world-state. Triggers that influence the Ta intentions are derived from the team-oriented components of our agent mentioned above. This desire is then used as a trigger for a corresponding plan from a plan library PL.

Definition 4: Plan Library PL is a collection of plans. Each plan is a 4-tuple denoted by <Pr,Po,A,F> where Pr and Po describe the pre- and post-conditions for the execution of plan respectively. A is the actions that form the plan body. A is compiled using the set of S contained in the B. F is the frequency of success of the plan. Learning can be implemented within the Plan Library using the element F. It can be assumed that multiple plans will match the pre-conditions generated from the I. Thus, the selection of a particular plan will be based upon it's successful matching of pre-conditions with I and it's success F. The plan library is split into priority levels. These levels assist in plan interruption and recovery. A plan with a higher priority can interrupt a plan with a lower priority. When a plan is selected for execution, an expiration time is assigned to the plan. When a plan is interrupted, the agent will hold the current plan and execute the higher prioritized plan. At the completion of the plan, the agent will try to recover the previous plan and try to complete it within the given expiration time. Should the time expire, the plan will be discarded.

3 Team Coordination and Communication

Our approach to team coordination is to implement agent teams. We have adopted the concept of mutual belief, joint goal and joint intention [2] as well as dynamic role assignment [4] and team plans.

Definition 1: Team t represents a set of a finite individual agents [2].

Definition 2: Social structure s represents the agents' belief of belonging to a t. The social structure s as defined by Tidhar [2] denotes it as a pair of command and control team which we denote as <St,Ta> where St denotes the command team and Ta denotes the control team for a t, however, in our approach, there are no distinct command or control teams. Every agent would perform both the St and Ta functions. These functions are however emphasized when the agent becomes a leader. A leader is an agent that take charge of other agents' and are responsible for the their behaviors. The majority of the St component's authority is contained within the leader. Thus, a team's goal is centralized within the leader. The Ta component is distributed equally between each agent. Unlike the St component, very little inter-agent communication is used. Ta is concerned with an individual agent's contribution towards the team goal. That is, Ta takes into account the team goal. Ta could be considered the component of the agent that handles team intentions. Ta also communicates with St, which in turn communicates within the team to synchronize certain movements of players. St holds the concept of social structure and responsibilities where the agents know who is in command of it and who it is in command of. Ta adopts the role of coordinator. This approach allows the dynamic assignment of sub-teams, which we will denote as task groups. In theory there can multiple encapsulation of sub-teams however with the limited number of agents involved we are limiting to only one level of sub-teams. These individual task groups can only receive task from the agent that initiated their formation, in other words, if the Captain initiates the formation, then only the Captain can assign tasks to the task group however transfer of task group between the Captain and the Goalie are permitted. This result in t that can adopt two or more non-contradicting desires simultaneously.

In our approach, we have adopted a social structure where the Coach is at the top of the chain, followed by the Captain and Goalie. The agent structure of the coach is yet to be decided, however, it might be a variance of the Case-Based BDI Agents [5]. Before the start of the game, the Coach plays an important role where it will make decisions regarding the type of strategy to deploy as well as the composite of the initial team plan library similar to the locker room agreement [4]. Due to the time constants, we have limited the number of team plans that are available to the team. The team plan is a 4-tuple similar to the make up of the agent plans. The team plan library selection will be done based upon the success frequencies. During the game, the coach will analyze the game, determine where the plans are failing and make modifications or in extreme cases generating new team plans. These plans will then be uploaded to the player when allowed. At the initial start of the game, an agent can only be in either one of the two teams, the offensive team where the leader is the Captain or the defensive team where the leader is the Goalie. As the game progress, both the Captain or Goalie can assign agents as task group leaders and other agents can be requested to join task groups. A task group will consist of at least two members and no more than four. Each task group will have a leader, and only this leader will remain a member of t. The Captain and Goalie will always be leaders and their teams will always exist. Many leaders may exist at an instance. When an agent is assigned as a leader, they will perform the function that is similar to the command team and control team mentioned previously. As the leader, the agent will start considering intentions that will encompass the task group. It will also coordinate with other members of the task group. Individual agents can reject a request to join a task group when the current task group that they are in has a high priority. As members join the task group, the leader is informed and the world-states of all members of the task group are synchronized. Each element of the world-state has an expiration time associated with it, so the most reliable world-state elements are retained. Each task group member also retains the right to leave the task group. When an agent leaves a task group or team, the leader of the team is informed. When a particular task is achieved, the task group can be disbanded or a new task assigned to the group. A task would consist of either an offensive or defensive desire, a team plan i.e. defend a particular part of the field and an expiration time in which to complete the task. Should the task fail, the leader could request for more agents, time or for the task to be deemed a failure. When a task group is disbanded, its' members will return to either the Captain's team or Goalie's team.

As this approach relies heavily on agent communication, we have adopted the concept of Multihop Ad Hoc networking [1] to overcome the limited broadcast radius, low bandwidth and unreliable communication channel. A large part of the Gongeroo's 'strategic' and 'tactical' behavior is reliant upon effective inter-agent communication. In order for a leader to coordinate offensive or defensive agent behaviors, messages need to be passed across the range of the soccer field to agents inside and outside the broadcast radius. This provides a unique domain in which to apply current research in the areas of Multihop Ad-hoc networking [1]. Ad-hoc networks are wireless networks consisting of multiple mobile hosts. They form an unstructured, dynamic and temporary mobile network. In situations where a source host wishes to send a packet to a destination host, which is not within broadcast range, the packet may be sent via other hosts within broadcast range. Hosts forward this packet until it reaches its destination.

4 Learning

Our approach to learning is a layered approached. Learning will be done within an agent via the Plan Library's F component and at the team plan level via the Coach mentioned previously.

5 Conclusion

The Robocup environment provides an interesting range of challenges within the multi-agent domain. We have described our approach to the development of a BDI multi-agent team that utilizes a Multihop Ad-Hoc network to assist in the implementation of a team-oriented programming approach.

References

[1] Johnson, D.B. "Routing in AD Hoc Networks of Mobile Hosts", Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, December 1994.

[2] Tidhar, G. "Team-Oriented Programming: Preliminary Report", Tech. Rep. 41, Australian Artificial Intelligence Institute, Melbourne, Australia, April 1993.

[3] Rao, A.S. and Georgeff, M.P. "BDI Agents: From Theory to Practice", In Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95), San Francisco, USA, 1995.

[4] Stone, P. and Veloso, M. “Task Decomposition and Dynamic Role assignment for real-time Strategic Teamwork”, (ICMAS-98).

[5] Chang, C.F. Ghose, A. Enguix, C.F. Olivia, O. “Case-Based BDI Agents: An Effective Approach For Intelligent Search On the World Wide Web”, In Proceedings of AAAI Spring Symposium on Intelligent Agents, Stanford University, USA, 22-24 March 1999.
http://budhi.uow.edu.au/postgrad/carlos/publications/AAAI_WebCBR_BDI.ps