Team Description of TsinghuAeolus

Yao Jinyi and Cai Yunpeng

State Key Lab of Intelligent Technology and System,
Deparment of Computer Science and Technology,
Tsinghua University , Beijing, 100084, P.R.China
Team Name:TsinghuAeolus
http://www.lits.tsinghua.edu.cn/RoboCup
Team leader:Yao Jinyi
yjy01@mails.tsinghua.edu.cn
Team members:Cai Yunpeng, Lao Ni, Yang Fan, Wang Qi
Caiyp01@mails.tsinghua.edu.cn
{lao_ni, albertyang} @263.net
wang_qi99@mails.tsinghua.edu.cn

Abstract. The aim of Tsinghua RoboCup Research Group covers both education and research. TsinghuAeolus as a simulation team, developed by this group, is the result of a project that belongs to Tsinghua's SRT (STUDENTS RESEARCH TRAINING) Program. We won the RoboCup2001 Simulation League in Seattle and ChinaRoboCup2001 in KunMing. In this paper we illustrate the motivation and research focus of our team.

1 Introduction

TsinghuAeolus is being developed on the basis of the champion team [1] in Seattle. Until last August we had built the competent simulation team with much effort on basic skills and agent architecture. The team had showed impressive skills, such as dribble and pass, in Seattle. And we designed the reactive agent architecture with a decision-making mechanism composed of action generator, evaluator and mediator. Much detail work had paved the way for subsequent research.

According to the experience of previous work, we feel that the team behavior is hard to control because of so many inordinate parameters that describes the characteristics of the system. It is impractical to use methods of machine learning to tune them directly because of difficulties such as immense state space, action space and reward hard to provide or delayed badly. So we turn to adjustable autonomy, which is a hot research field now.

From the viewpoint of a coach, he wants a player to follow his advices, but he doesn't want to manipulate the player fully. In other words, a player is required to be autonomous when without advices and to be half-autonomous when with advices. In more general cases for an agent, advices may come from its own brain, cooperators and even human. The mechanism of decision-making to support adjustable autonomy is our research focus now.

In the next section, we explain more why we turn to adjustable autonomy. In section 3 we briefly describe our work on it. In Section 4 we conclude with problems and our plan for future work.

2 Adjustable autonomy in RoboCup

Adjustable autonomy has been studied a great deal in human-robot applications, such as manned space flight [2]. But for a robot, commands may come not only from human users, but also from other intelligent robots. In RoboCup Simulation League, there exists such a typical case. A coach who can get full state information often hopes to control a player's behavior in varied levels. And teammates may also give some valuable instructions because they obtain some different infomation. These provide a stage for adjustable autonomy in RoboCup.

Adjustable autonomy provides the flexibility to adapt to changing environments, user goals and resources [3]. Its advantage in real applications is obvious. It is also crucial for developing a competitive simulation team in RoboCup. From many simulation matches, we have found a phenomenon that the same team often plays differently against different rivals, sometimes better and sometimes worse. It's same with real human soccer games. Hence, the ability to adapt to different rivals can improve a team's competence. Further more, this kind of flexibility allows a learning machine to improve team behaviors in a high level because with the support of adjustable autonomy, action space is transformed and usually reduced greatly.

3 Adjustable Autonomy in TsinghuAeolus

Behaviors of a soccer agent should include positioning, ball possession, ball recovery, communication and visual action etc. We choose the behavior of ball possession as the current emphasis of research on adjustable autonomy. We continue to use the three-layered decision-making architecture composed of action generator, evaluator and mediator.

3.1 Advice mode

As we have said, for a player agent, advices may come from the following sources.

- Coach
- Teammates
- Its own brain beyond the decision-making architecture.

The following are some representational forms of advice. They are anagalous with what a human coach may say.

- To designate one or several actions such as "Pass to some region"
- To give some kind of guidance such as "Attack along left more"

– To give some kind of inclination such as "Accelerate offense rhythm"

From these modes, we can see that the delimitation between different levels of autonomy is a little ambiguous. In fact, unlike general adjustable autonomy applications, no explicit selection of the level of autonomy is needed here. External advices may come in different forms from different sources. After parsing each advice, the agent decides by itself which layer of the decision-making architecture should be adjusted.

3.2 Flexible action generator

Action generator discretizes the action space and provides candidate actions for the subsequent decision-making. The optimality of candidate actions is contradictory to efficiency. External advices can often help to reduce the action space greatly. So it's possible for a flexible action generator that follows advices to improve the optimality of generated candidate actions. It's also an advantage of applying adjustable autonomy.

The lowest level of autonomy is implemented in this module. When the agent is expected to execute a designated action, the only thing needed to do is to limit the candidate actions to it. And the considered action space is reduced to a point in this case.

3.3 Biased Layered Evaluator

Evaluator is a core component for implementing adjustable autonomy in our method. When a coach gives the last two kinds of advices, it often means preference for some subset of client actions. Then preferential policy is taken for this subset to help them to win over other actions in the process of evaluation. So we call it biased evaulator.

The agent architecture includes modules reciprocal to each kind of behaviors. The competition among candidate actions in the same module is different with that among the winners from each module. So we design a layered evaluator. Accordingly, external advices are allowed to influence evaluation in each layer. It's similar to different levels of autonomy. For example, the coach can say, "pass more to No 10" which change the inclination in low layer competition among pass actions, or he can say, "dribble more" which influence the high layer competition among pass, dribble and shoot, etc.

3.4 Advices validation

An agent should be responsible for mission safety and success. In other word, it should validate the external advices to exclude wrong ones.

There are two kinds of validation. One is to validate the right of the adviser. It's a natural idea when different advisers are trusted to different extent. For example in TsinghuAeolus, a client agent can't force its teammates to do a specific action while a coach can. The other one is to validate the correctness of a received advice. It's simply implemented here. We only consider mistake that an advice doesn't match the agent's current situation. Some kinds of advice should be remembered and followed in the future and the others should be rejected.

3.5 Summary

Until now, we have provided a method to extend the decision-making architecture composed of action generator, evaluator and arbiter or mediator with adjutable autonomy.

4 Conclusions

In this paper we have discussed why and how we implement adjustable autonomy in TsinghuAeolus. We have realized several weaknesses in our current method. Firstly, it is still difficult for preferential policy to be carried out accurately. Secondly, the agent that is advised generates no feedback. These two points, especially the second one, will hinder learning above this architecture. Thirdly, external advices from different sources are added simply. In some cases more work is needed to dissolve conflict and merge them.

Anyway, we believe this method is a strong step for the future work of developing a learning machine based on adjustable autonomy. These weaknesses are just where our future work starts from.

References

1. YaoJinyi, ChenJiang, CaiYunpeng and LiShi:Architecture of TsinghuAeolus, Available in http://www.lits.tsinghua.edu.cn/RoboCup/RoboCup/Docs/Tsinghuaeolus.ps
2. David Kortenkamp, Debra Keirn-Schreckneghost, and R.Peter Bonasso:Adjustabl Control Autonomy for Manned Space Flight, in IEEE Aerospace Conference, 2000
3. Michael A.Goodrich, Dan R.Olsen Jr., Jacob W.Crandall and Thomas J.Palmer:Experiments in Adjustable Autonomy, Technical report version of paper to be presented at the IJCAI01 workshop on Autonomy, Delegation, and Control: Interacting with Autonomous Agents.