Phoenix Team: A General Agent Paradigm for Rescue Simulation

Zijian Ren

Department of Electrical and Computer Engineering, The University of Iowa, 4016 SC, Iowa City, IA 52242, USA

zijian-ren@uiowa.edu

Abstract. Agents are autonomous entities which can sense from and act on environment. Agents could be programs, robot's control systems or entire robots. We propose a general agent paradigm, in which learning, communication, knowledge, and other key factors are clearly divided and integrated. The agent paradigm tries to integrate learning, communication, and knowledge in a general way regardless of problem domains. We apply this paradigm in rescue simulation with satisfactory results. This agent paradigm can be easily applied in many other domains such as rescue robot, robot foraging, and robot exploration.

1 Introduction

Agents are autonomous entities which can sense from and act on environment. Agents could be programs, robot's control systems or entire robots. In this paper, robot and agent are used without distinction. Intelligent robots play an increasingly important role in modern society in numerous fields such as manufacturing, designing, space exploration, human-assistance, entertainment, and so on. Robots are good for the "3D" jobs as it says in a well-known joke: Dirty, Dull, and Dangerous (Murphy, 2000).

The RoboCup rescue project was motivated by a real disaster: Hanshi-Awaji earthquake occurred on January 17, 1995 in Kobe, Japan. More than 6,500 citizens were dead, and more than 1 million people suffered, and 80,000 wooden houses were completely destroyed. The total loss is more than 100 billion US dollars (Tadokoro et al., 2000). Scientists and engineers face a serious question: How to mitigate the damage of future earthquakes or, further, how to avoid earthquake damage? Specifically for computer and robotic scientists, how robotics can help in the disaster situation?

The rest of the paper is organized as following. A general agent paradigm is presented in section 2. Section 3 investigates learning, communication, knowledge, and their relations. Section 4 describes our experimental results with the agent paradigm and conclusions.

2 A General Agent Paradigm

2.1 An Agent Paradigm

There are several ways to categorize intelligent robots. Murphy (2000) states three major robotic paradigms: hierarchical, reactive, and hybrid deliberative/reactive. We propose a general agent paradigm, which is more comprehensive and useful.

The agent paradigm consists of five major components: Sensing (S), Control Unit (CU), Knowledge Base (KB), Action (A) and Communication (C) in figure 1. The control unit can be roughly divided into three modules: action control, knowledge acquisition and usage, and communication control. The communication component includes two parts: messages from other agents and messages to other agents. The knowledge base is the most complex part in the whole paradigm.

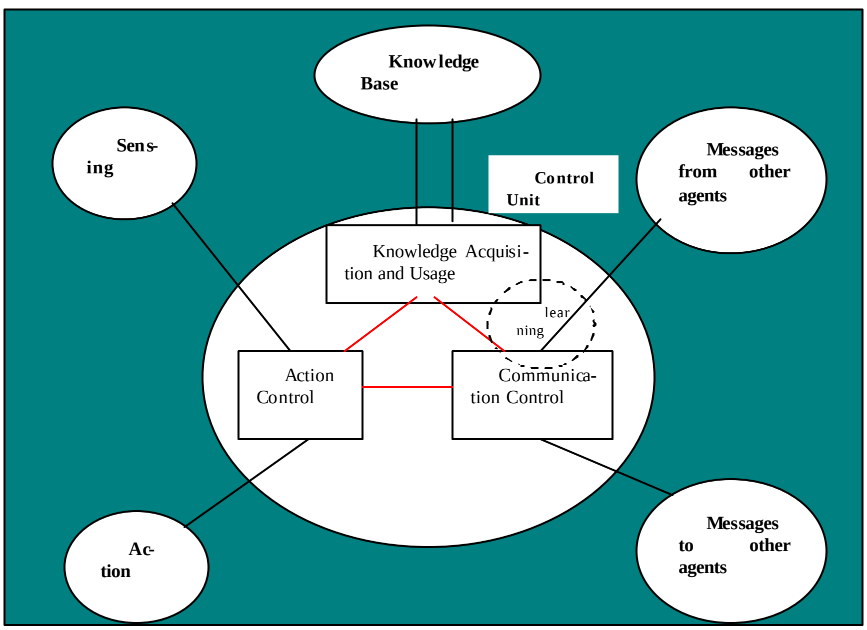

Fig. 1. An Agent Paradigm with Communication and Knowledge

Besides the above modules, robots also have goals and constrictions.

- Goals: To pick up more objects (foraging task), or rescue more humanoids

(rescue task), or higher score (soccer task), or others.

- Constrictions: computational abilities, storage space, communication loads, physical action limitations[1] such as action range and energy consumed, and errors in communication and action etc.

2.2 Agent Categories with Knowledge and Communication

We classify agents with sensing, action, knowledge, and communication.
There are ten agent categories:

- S and A: Non-communicated reactive agent
- S, A, and KB: Non-communicated cognitive agent
- S and C: Communication agent
- S, C and KB: Intelligent communication agent
- C: Blind communication agent
- C and KB: Blind intelligent communication agent
- C and A: Blind controlled agent
- C, A and KB: Blind intelligent controlled agent
- S, C and A: Communicated reactive agent
- S, C, A and KB: Communicated cognitive agent

These categories are listed in table 1.

[1] Physical action limitations are common in real robots but less common in robot simulations.

Table 1. Agent Categories

	Sensing (S)	*Action(A)*	*Communication(C)*	*Knowledge Base(KB)*	*Comments*
Non-communicated Reactive Agent	●	●			standalone agent
Non-communicated Cognitive Agent	●	●		●	
Communication Agent	●		●		messages as actions
Intelligent Communication Agent	●		●	●	
Blind Communication Agent			●		message center
Blind Intelligent Communication Agent			●	●	
Blind Controlled Agent		●	●		controlled by other agents
Blind Intelligent Controlled Agent		●	●	●	
Communicated Reactive Agent	●	●	●		
Communicated Cognitive Agent	●	●	●	●	Fully functional agent

In the rescue domain, three rescue agents (FireBrigade, AmbulanceTeam, and PoliceForce) could be a type of non-communicated reactive agent, non-communicated cognitive agent, communicated reactive agent, or communicated cognitive agent. Three rescue centers (FireStation, AmbulanceCenter, and PoliceOffice) could be a type of communication agent or intelligent communication agent.

3 Learning, Communication, and Knowledge

Parker (2003) presents eight primary research topics in multi-robot systems — biological inspirations, communication, architectures, localization/mapping/exploration, object transport and manipulation, motion coordination, reconfigurable robots, and learning. With above paradigm, we only discuss three primary topics: learning, communication, and knowledge.

3.1 Learning

Learning is a process of improving individual performance, precision (or quality) of solutions, efficiency (or speed) of finding solutions and scope of solvable problems (Plaza et al., 1996). We define learning in a more general way: learning is to acquire new knowledge or update existing knowledge. Its purpose may just be curiosity instead of a pragmatic goal such as problem solving. Learning can be divided into several major categories: reinforcement learning, genetic algorithm, neural network, rule-based learning, and statistical learning (Ren and Williams, 2003).

However, it seems learning is not effective with current rescue simulation. Two factors are accounted for this phenomenon: fires spread fast so that there is no time to learn and the current competition rules use average score to evaluate a team. We hope learning will play an important role in future rescue with some changes.

3.2 Communication

Communication format or protocol is an agreement between senders and receivers. There are several common agent communication protocols: KQML, KIF, and FIPA ACL. We are more concerned with communication content and directions.

Rescue simulation limits the maximum length of each messages and the maximum number sending and receiving in each cycle. These limitations put most challenges in communication. The maximum length in one message requires transferring the most useful information or knowledge. The maximum message sent/received in each cycle forces a careful design of communication directions.

There could be some improvements for current simulation communication protocols. For example, "tell" command is a broadcast type. If a fire brigades uses "tell" command to send message to a fire station, this message is also sent to other fire brigades who may not be interested in it. If a fire station uses "tell" command to send message to a police office, the message also goes back to fire brigades. In the future, we propose a one-one type communication such as from a fire brigade to a fire station or from a fire station to a police station.

3.3 Knowledge

In general, the knowledge content can be distinguished into *knowing something* and *knowing how to do something* (Ferber, 1999). In brief, there is a *what/how* pair for above two categories. The "*what*" part is concerned with the descriptions of objects and phenomena in the universe. The "how" part is mostly concerned with the relationship between objects and phenomena so that laws of the universe are modeled for further prediction. The knowledge can be divided into several categories such as fact, rule, model, policy, and theory.

The knowledge should be hierarchical from simple to complex. Simple knowledge types used in rescue domain include burning buildings, blockade roads, building with buried persons. Some complex knowledge types used in the rescue domain might be team formation, communication methods, and crucial parameters but should not be limited to above types.

There is been long history to build a powerful and universe knowledge base. However, there are drawbacks in these knowledge bases. They are too big for a small problem such as current rescue simulation. How to effectively scale down one huge knowledge base is a key problem. Moreover, these knowledge bases completely use symbolic representation and reasoning, which suffers from many aspects including lack of flexibility, slow reasoning, and difficult handling of ambiguity. It might be promise to use connectionist knowledge representation and reasoning similar to brain knowledge processing. A powerful knowledge base should integrate symbolic and connectionist knowledge seamlessly. However, there is still a question: Does a universe knowledge base exist for any problems? The answer is No in my personal view.

3.4 Relations among Learning, Communication, and Knowledge

Sen and Weiss (1999) propose two major relationships between learning and communication: learning to communicate and communication as learning. Our general agent paradigm reveals the relationship between learning/communication with knowledge. Learning is inevitably involved with knowledge. Knowledge is closely related to communication content.

A proverb summarizes the relations among learning, communication, and knowledge though it may not fully cover the truth:

> "***Learning acquires knowledge; Communication transfers knowledge; Knowledge makes action rational and efficient***."

4 Results and Conclusions

Using above paradigm with knowledge and communication, we compare four programs in table 2.

- Program A is a sample program (Morimoto, 2002). This program has several simple reactive rules such as extinguishing any known fires or rescuing any known injured person. Fire brigades don't refill water after running out of water. Communication only transfers the road blockade messages. The result of this sample program is used as a benchmark.

- Program B is an improved sample program. A water refill rule is added for fire brigades.

- Program C removes communication from program B in order to evaluate the communication effect.

- Program D improves in knowledge and communication. The information is divided into several categories. For example, fire has two categories: urgent fires which can spread other unburned buildings and ordinary fires which not. Of course, fire agents extinguish urgent fires at first. This program packs several similar small messages to one large message to handle the communication restriction.

Table 2. Experimental Results

	Alive Humans (Mean)	Remaining Health Points (Mean)	Unburned Building Areas (Mean)	Score (Mean)	Score Scope	Comments
Program A	52	426,944	83,431	38.85	38.06-41.16	Simple Reactive Robots, with communication and without water refill
Program B	53	434,686	104,295	44.03	42.21-50.36	Simple Reactive Robots, with communication and water refill
Program C	40	366,840	66,980	26.68	24.81-28.67	Simple Reactive Robots, with water refill and without communication
Program D	73	576,786	144,123	71.71	70.86-71.81	Cognitive robots, with efficient communication

With these results, we have following conclusions:

- Communication is a crucial factor in rescue domain. With the real communication limitation or simulation restriction, efficient communication becomes the key of success. Efficient communication includes reducing communication load, set communication directions, defining communication content, and so on.
- Knowledge improves the performance. It demonstrates that cognitive agents with knowledge perform better than reactive agents in this domain.
- Graphic user interface (GUI) is an effective tool to investigate the system performance. With the graphic viewer, designers (humans) can analyze the rescue agent actions to find the problem and improve the performance.

There are my personal expectations for rescue simulation projects.

1. Modify simulation rules or setting for learning and communication.

2. More powerful GUI tool for debugging and developing effective strategy with human intelligence.

3. Design rescue simulation as a general platform for other problems AND/OR a realistic simulation for rescue robots.

References

1. Ferber, J. 1999. Multi-agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley Press.
2. Morimoto, T. 2002. YabAPI: API to develop a RoboCupRescue Agent in Java. http://ne.cs.uec.ac.jp/~morimoto/rescue/yabapi
3. Murphy, R. R. 2000. Introduction to AI Robotics. MIT Press.
4. Parker, L. E. 2003. Current Research in Multi-Robot Systems. Journal of Artificial Life and Robotics.
5. Plaza, E., Arcos, J. L., and Martin, F. 1996. Cooperative Case-Based Reasoning. In Gerhard Weiss, editor, Distributed Artificial Intelligence Meets Machine Learning, Lecture Notes in Artificial Intelligence, Vol. 1221, pages 180-201. Springer.
6. Ren, Z. and Williams, A. B. 2003. Lessons Learned in Single-agent and Multiagent Learning with Robot Foraging. Proceedings of 2003 IEEE International Conference on Systems, Man & Cybernetics, Washington, D.C.
7. Tadokoro, S., Kitano, H. Takahashi, T., Noda, I., Matsubara, H., Shinjoh, A., Koto, T., Takeuchi, I., Takahashi, H., Matsuno, F., Hatayama, M., Nobe, J., and Shimada, S. 2000. The RoboCup-Rescue: An International Cooperative Research Project of Robotics and AI for the Disaster Mitigation Problem, Proceedings of SPIE 14th Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls (AeroScience), Conference on Unmanned Ground Vehicle Technology II.
8. Sen, S. and Weiss, G. 1999. Learning in Multiagent Systems. In Gerhard Weiss, ed. Multiagent Systems, MIT Press