Proceedings of the 2008 IEEE
International Conference on Robotics and Biomimetics
Bangkok, Thailand, February 21 - 26, 2009

A Multi-Agent Planning Approach Integrated with Learning Mechanism

Tao Zhang and Liang Zheng
Department of Automation
Tsinghua University
Beijing 100084, China
taozhang@tsinghua.edu.cn

Haruki Ueno
Intelligent Systems Research Division
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

***Abstract* - This paper presents a multi-agent planning approach integrated with learning mechanism. This method involves in task allocation, path planning, avoiding conflicts, cooperation, parameter learning, pattern learning, etc. In addition, with this method a multi-agent Sokoban platform is defined. With some simulations on this platform, the advantages of multi-agent planning approach with learning mechanism are illustrated comparing with single-agent approach. Since the proposed method can improve the efficiency and capability of multi-agent planning, by referring to the results of this research, the proposed method will be adopted for multi-robot system in the future research.**

***Index Terms* – Multi-agent planning, knowledge model, learning mechanism, Sokoban platform**

I. INTRODUCTION

With the development of multi-robot system, multi-agent system is regarded as one of important approaches to realize the modeling and control of multi-robot system. Each robot is represented by an agent. The behavior of multi-robot system can be described by that of multi-agent system. Therefore, various researches on multi-robot system are carried out on multi-agent system.

In the research of multi-agent system, planning is one of important topics. It is normally based on appropriate strategies and interaction among agents to realize task allocation, path planning, etc. So far, there are two types of planning approaches. One is centralized planning method and another is distributed planning method. There are already plenty of researches on the distributed planning method because multi-agent system is more close to a distributed system. Actually, centralized planning is also a kind of good methods because it has good global view of planning and it is easy for making learning process. Therefore, this research is focus on proposing a new centralized planning method.

Concerning about centralized planning method, there are following literatures to show their research achievements. Baki *et al* proposed a centralized planning technique concerning temporal constraints and uncertainty [1]. In this technique probability analysis and inference processing are adopted for planning. Dimopoulos *et al* introduced their multi-agent coordination and cooperation through classical planning [2].

This research focuses on avoiding conflicts by means of capability compliment. Matthew *et al* studied on the planning issue of Sokoban. Although the above literatures present various approaches for centralized planning, they all have a certain limitations to the variation of planning due to shorting of learning mechanism. Therefore, they have no capability to introduce new knowledge by learning during the planning process. This issue is just the aim of this research to solve.

In this paper, a centralized multi-agent planning approach integrated with learning mechanism is proposed. This method involves in task allocation, path planning, avoiding conflicts, cooperation, parameter learning, pattern learning, etc. In addition, with this method a multi-agent Sokoban platform is defined. With some simulations on this platform, the advantages of multi-agent planning approach with learning mechanism are illustrated comparing with single-agent approach. Since the proposed method can improve the efficiency and capability of multi-robot planning, by referring to the results of this research, the proposed method will be adopted for multi-robot system in the future research.

II. MULTI-AGENT PLANNING INTEGRATED WITH LEARNING MECHANISM

A. A Multi-Agent Planning System Integrated with Learning Mechanism

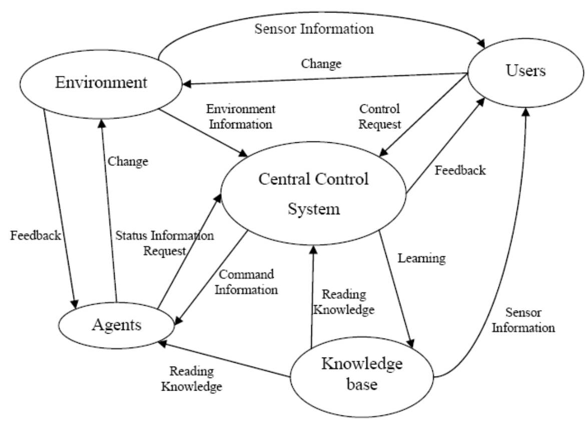

Fig.1 Multi-agent planning system integrated with learning mechanism

This research is supported by the Fund of State Key Laboratory of Industrial Control Technology.

In order to realize the multi-agent planning approach integrated with learning mechanism, a relative system is first defined. As illustrated by Fig.1, this system is organized by five parts: central control system, knowledge base, agents, users and environment.

Central control system is the core of the multi-agent planning system. It composes of the modules for planning, control and so on. The main algorithms proposed in this research are integrated in this system. Knowledge base is the important part to support the planning process. In particular, the learning process is carried out on the knowledge stored in the knowledge base. Agents are the objects of the planning. Users and environment are necessary for planning in this system. In addition, all these parts are inter-connected with various data stream and task requirements.

The features of this system are summarized as below.

- In agent part, basic information can be exchanged between the neighbor agents, which is necessary for autonomous and real-time behavior of agents;
- Knowledge base can be only controlled by the central control system, which is an effective way to protect it;
- Knowledge base has no direct interaction with environment;
- Central control system will not directly change the environment without behaviors of agents;
- Users will not directly interact with agents. But they can understand the internal status of agents through central control system;
- Users can not change the environment directly. But they can affect the environment by means of platform of central control system;
- Users can not modify the contents of knowledge base. But users can give their suggestions to central control system when central control system will modify the contents of knowledge base;
- Central control system is the core of the system and it is responsible for controlling date stream, inspecting the errors of commands and protecting the system as well.

Based on the above definition, the centralized planning method for multi-agent system can be realized in this system.

B. Multi-Agent Sokoban Issue

Sokoban issue is a classical issue presented by Hiroyuki Imabayashi in 1980. Since its map is very variable but the operation rules are simple, and even small scale map also needs several steps to find the solution, therefore, it is often challengeable to human intelligence and becomes very popular.

The rules of sokoban are that in a map formed by many checks, agents need to push some boxes to the target positions. At each step agent or box can only move one check horizontally or vertically. In addition, agent can only push the box behind the box and he can only push one box as a step. Obstacle can not be over-passed by agent. If all boxes are pushed to the target positions, the game is over, otherwise the game is failed.

Fig.2 is an example of a complex Sokoban map. Even for an expert of solving Sokoban issue, he also spends long time with many attempts to solve this issue given in Fig.2. By use of computer, Sokoban is still a hard issue, which has been proved in 1995 that its complexity is NP-hard concerning to the scale of map [4]. In 1997, it also proved that this issue is a kind of PSPACE-complete [5]. The above conclusion has a preliminary that one agent in one game. Currently, there are still many mathematical researchers do their best to find solution under the condition of one agent.

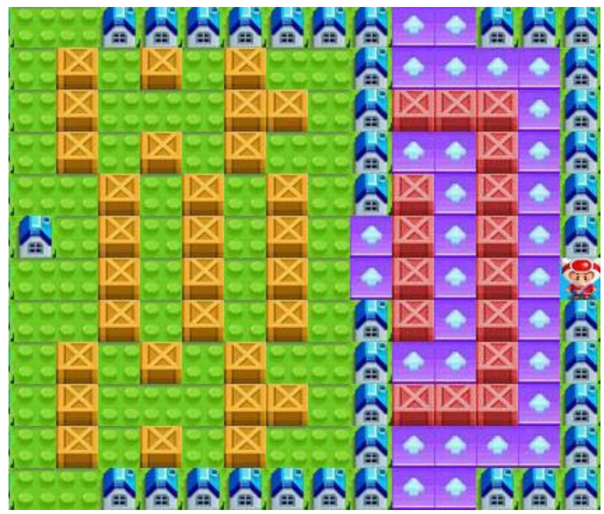

Fig.2 Example of a complex Sokoban map

Actually, if using multiple agents in the game (i.e., multiple agents), this issue can be remarkable changed. That is, some unsolved issues can be solved and the number of steps for solving one issue can be decreased. However, there appear many new problems of using multi-agents. One of important problems is planning. That is how to optimize the behaviors of multi-agent from the global point of view.

Besides, with the development of research on multi-robot system, researchers found that the research on the planning of Sokoban issue with multi-agent is very helpful for considering the planning of multi-robot system. In some senses, the behavior of multi-agent in Sokoban issue is quite similar to the behavior of multi-robot. Therefore, the conclusion from solving Sokoban issue can be directly adopted for multi-robot system. This is just the purpose of our research in this paper to develop a new planning approach for multi-agent system.

C. Task Allocation

In the proposed centralized multi-agent planning approach with learning mechanism, task allocation is one of important stages. Although there are many task allocation methods, optimal matching method is adopted for task allocation in this method.

Concerning about Sokoban issue, a distance calculation function is defined, by which a reasonable distance and its evaluation can be given. Therefore, the distance between any agent and box as well as the distance between any box and target position can be given. All these information are stored in the knowledge base. When using optimal matching method to realize task allocation, all these knowledge will be adopted for finding optimal solutions.

If the number of boxes is larger than that of agents, task allocation will be carried out by several stages or one agent will be allocated several tasks. On the other hand, if the number of boxes is smaller than that of agents, spare agent can be free of the task. If the number of boxes is equal to that of agents, optimal solution can be found out.

D. Path Planning

After task allocation, normally an agent will perform its task right now. However, the actual situation is different from the expectation. Agent always can meet some constraints. Concerning about the Sokoban issue, obstacle is one of constrains. If allocating one agent to push a box, but he meets an obstacle, it has to change the results of task allocation. Therefore, the result of path planning is also factor to be considered in the task allocation. A function called *getto* is defined to provide the results of generating the concrete route. These results are also stored in the knowledge base. When making task allocation, it must consider the results of path planning. Particularly, when there is an obstacle, the result of path planning is more important to choose appropriate route.

In addition, in the proposed method, this *getto* function is not a simple function to make simple path planning. It also possesses the function of considering collaboration between agents. For example, when an agent meets an obstacle, this agent itself can not push the box continuously to realize the original results of task allocation. At this time, this *getto* function can give the suggestions to ask another agent to push this box so that this agent can continue his task without changing remarkably of original results of task allocation. Therefore, this path planning method, so-called collaborative path planning, can be more powerful than path searching method for path planning.

E. Avoiding Conflicts

If several agents are pushing the boxes simultaneously, conflicts among them often occur. Therefore, avoiding conflicts are very important for keeping stability. In [3], dividing map is proposed to avoid conflict. Since each agent is limited into a small area of map and can not move into another area, it causes low efficiency of the method. Additionally, it may also bring non-solutions.

In the proposed method, a simple way is introduced as below. After task allocation, the arrangement of each agent is stored into the data array. In each step, movements of agents are virtually carried out in the memory. If the return value is true, it means there is no conflict among agents in this step. If the return value is false, it has conflict. The agents which have conflicts will not move in this step. In the next step, these agents will try again to move as allocated in the last step. If there are so many conflicts, the task allocation will be made again with different conditions.

F. Cooperation

Cooperation in the proposed method is centralized managed by the central control system, which is different from the distributed multi-agent system. The cooperation can be realized in the several ways. One is arranged in the task allocation process. Another is when one agent asks for cooperation during the operation process, a free agent can be allocated by central control system. Since the central control system is playing the role of dispatcher and it holds the global information of the system, therefore, it is easy for the central control system to arrange the cooperation among agents. In addition, it can optimize the arrangement of cooperation, which is hard to realize by the distributed multi-agent system.

G. Learning Mechanism

In the proposed method, learning mechanism is one of important features. Comparing with the traditional multi-agent planning method, it is also a new powerful function for planning. Actually, learning in multi-agent system is very complex due to their interaction. In addition, learning methods which can be adopted by single agent may not be appropriate for multi-agent systems. By investigation, the following learning methods can be used in the proposed multi-agent planning method.

- Reinforcement learning: by means of a reward value, agent can attempt to improve itself behaviors in order to obtain the maximal reward value. The most attractive feature of this method is that this method needs not depending on human's supervision.
- Parameter learning: there are two parameter learning method: generic algorithm based learning method and case based learning method. By use of above two methods, parameters of multi-agent system can be captured.
- Pattern learning: comparing with parameter learning, pattern learning is very abstract in the multi-agent system. However, by pattern learning, it will be easy to make judgment on some phenomena of multi-agent system, which is previously made by human. For example, deadbolt lock phenomenon is often appeared in the multi-agent system. It can be regarded as a kind of patterns. By pattern learning, it will assist agent easily to judge this phenomenon.

By above summary, these three types of learning are often adopted in the proposed centralized multi-agent planning method. For example, by means of pattern learning, the cooperation feature can be learned. When several agents are cooperating for one task, if each box can be put onto another one box, this pattern can be learned by pattern learning. This patter can be directly used in the future planning.

Also there are many cases which are appropriate for using parameter learning. For example, if an agent pushes a box to the edge of the wall, the game can not be continued because the box can not be moved. If this situation can be described by a parameter, such as the position of edge, the value of this parameter can be learnt and it will be very helpful for planning. The game will not be easily stopped due to such case.]

Besides, we have introduced some other learning mechanism in the proposed method. One is heuristic learning. By use of the knowledge stored in the knowledge base as well as new knowledge learning by above methods, new knowledge can be also found out by means of some knowledge mining rules. Besides, by some commands from human, the system can be automatically generated some new knowledge based on the old knowledge. In some senses, this knowledge is more important than the old knowledge and they are hard to get by traditional learning methods.

Another learning mechanism different above method is knowledge re-organization due to the change of environment. Since the environment can not be remarkably changed in a short time, this learning mechanism is just causing the change of knowledge gradually. The new knowledge is not quite different from the previous knowledge.

In a word, learning mechanism is implemented by many learning methods. Their selection is based on the requirement of multi-agent planning process.

III. APPLICATION: MULTI-AGENT SOKOBAN PLATFORM

A. *Features of Multi-Agent Sokoban Platform*

In order to verify the proposed centralized multi-agent planning method integrated with learning mechanism, a multi-agent Sokoban platform is created. This platform is programmed by use of Java language. The editor environment is by use of Eclipse 3.3.2 + Java 2 Platform Standard Edition 6.0. Fig.3 is the interface of this platform.

All factors of this platform can be summarized as Fig.4. Each factor in turn represents empty position, box out of target position, target position, box in the target position, obstacle, agent out of target position and agent in the target position.

The functions of this platform can be summarized as below.

- Create/modify map: when out of operation, user can easily create/modify the map by use of keyboard or mouse. User can easily redefine the position of agents or set up a new obstacle, and so on.

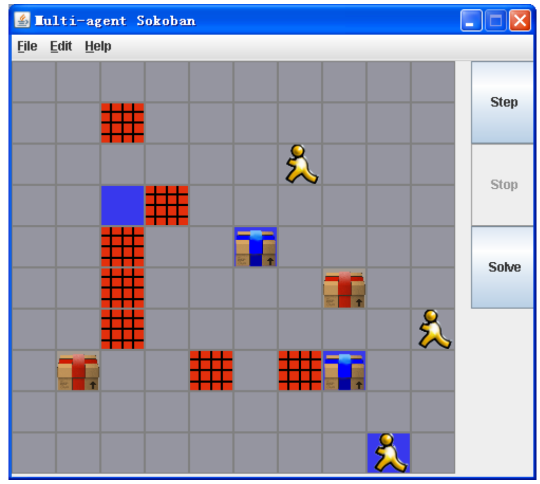

Fig.3 Interface of multi-agent Sokoban platform

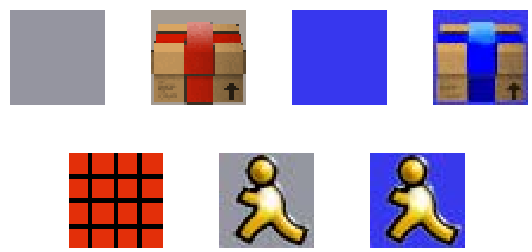
Fig.4 Factors in the platform

- Adjust the size of map: by setting new size of map, the map can be changed according to human's requirement.
- Install a new map: by use of "Load map…" in the menu of "File", user can easily load a new map defined previously in the platform.
- Save the current map: by use of "Save map…" in the menu of "File", user can save a map he wants to keep.
- Judge the end of game automatically: according to the rule of game, if the number of boxes in the target positions are same as the number of all boxes, this game can be ended successfully.

B. *Implementation of Multi-Agent Sokoban Platform*

The source code of multi-agent Sokoban platform is several Java files and image files. In the Jave files, it includes two classes: Mainframe and SizeSetting. In the class of Mainframe, it defines the modules for interface, rules, control, solving problems, etc. The class of SizeSetting is for changing the size of windows of platform.

The solution process in the platform is given in Fig.5. In the next section, some simulation results are introduced in detail.

IV. SIMULATIONS AND DISCUSSIONS

In order to verify the effectiveness of the proposed centralized multi-agent planning method integrated with learning mechanism, simulation was made by using the developed multi-agent Sokoban platform. In the simulation, the efficiency and capability of the proposed method are mainly verified as below.

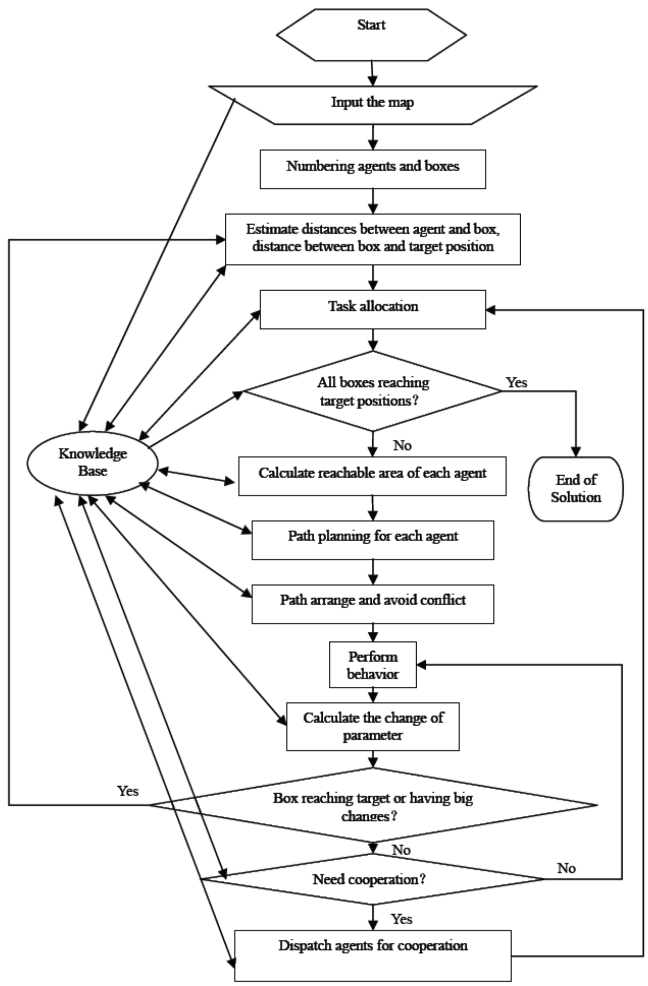

Fig.5 Solution process of platform

A. Simulation for Verifying the Efficiency

As illustrated by Fig.6, this is a simple map. But if there is only one agent, it is difficult to go over the long “wall” at the left side of the map. If we use two agents, this problem can become very simple to solve.

The cooperation is conducted at the key position. For example, at the gab of the wall, when one agent pushes the box to the gab, another agent then pushes it to pass the gab. Then this work can be very easily finished.

In the platform, when planning the route for one agent, the central control system can find the cooperation position by learning mechanism and save this knowledge into the knowledge base. When performing the job, central control system will guide two agents to move simultaneously and fulfill the cooperation at the cooperation position. Fig.7 illustrated the cooperation position of this map.

By the above simulation, the efficiency of the proposed method is verified.

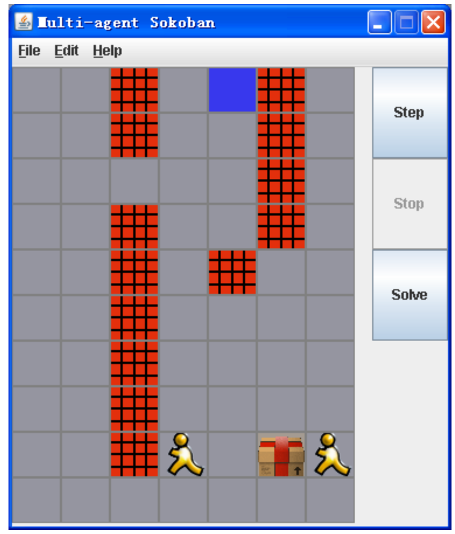

Fig.6 Initial map for simulation 1

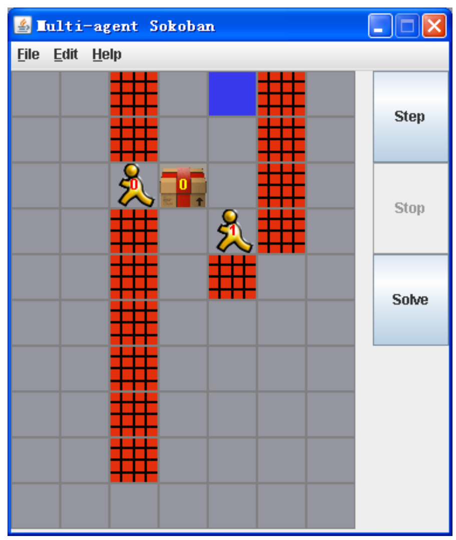

Fig.7 Cooperation position at the map

B. Simulation for Verifying the Capability

In this simulation, a problem which can not be solved by single agent is solved by the proposed method. Fig.8 illustrates the original map which can not be solved by single agent. It has two obstacles, two boxes and two target positions. By single agent, it has no solutions. If using two agents, this problem can be easily solved.

With the proposed method, when making multi-agent planning, cooperation position can be found by learning. One agent will move to the position near the cooperation position. When another agent pushes the box to the cooperation position, this agent can assist it to push the box into the target position. Fig.9 and Fig10 illustrate the map that two agents cooperate to push the boxes into two target positions.

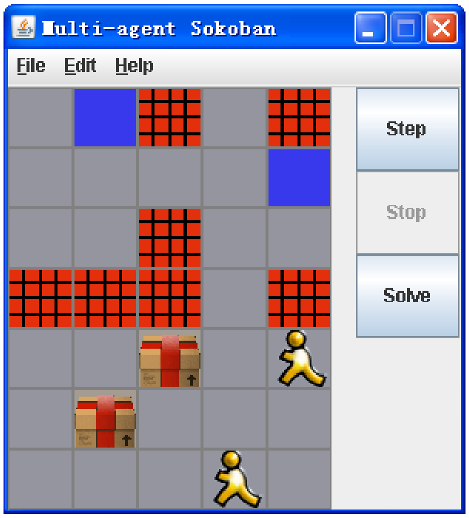

Fig.8 Original map which can not be solved by single map

Fig.9 Map for pushing one box into target position 1

C. *Extension of Multi-Agent Sokoban Platform*

Since this multi-agent Sokoban platform is organized by several modules, it can be easily extended to further test more functions of the proposed method. In addition, some modules of this platform can be replaced by new modules. Since the algorithms of the proposed method can be improved with the future research, this platform can be adopted as a good simulation platform for verifying the functions of method. In the future research, the size of map, boxes, obstacles, etc., will be changed. The capability of agent will be also improved so that we can consider more complex tasks.

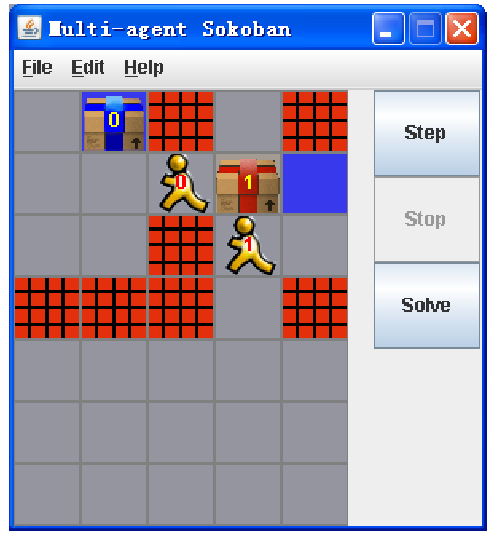

Fig.10 Map for pushing one box into target position 2

D. *Application of the Proposed Method*

As explained in the introduction, the study on the planning of multi-agent system is aiming for multi-robot system. Therefore, the proposed method is eventually adopted for multi-robot system. In our research group, we are doing the study on the collaboration of multi-robot system. The proposed method will be integrated into the collaboration system of multi-robot system. With the actually application, it will prove the effectiveness of the proposed method quite well.

V. Conclusions

This paper proposes a centralized multi-agent planning approach integrated with learning mechanism. This method involves in task allocation, path planning, avoiding conflicts, cooperation, parameter learning, pattern learning, etc. In addition, with this method a multi-agent Sokoban platform is defined. With some simulations on this platform, the advantages of multi-agent planning approach with learning mechanism are illustrated by comparing with single-agent approach. In the future research, we want to adopt the proposed method for planning the behaviour of multi-robot system.

References

[1] B. Baki, M. Bouzid, A. Ligza, et al, “A centralized planning technique with temporal constraints and uncertainty for multi-agent systems,” *Journal of Experimental & Theoretical Artificial Intelligent*, vol. 18, no.3, pp. 331-364, 2006.

[2] Y. Dimopoulos, and P. Moraitis, “Multi-agent coordination and cooperation through classical planning,” *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp.398-402, 2006.

[3] S. B. Matthew, and H. L. James, “Multi-agent planning in Sokban,” *Multi-Agent Systems and Application V*, vol.4696, pp.334-336, 2007.

[4] M. Fryers and M.T.Greene, Sokoban, Eureka, vol.54, 1995.

[5] J. C. Culberson, “Sokoban is PSPACE-complete”, *Technical Report TR 97-02*, Department of Computing Science, University of Alberta, 1997.