MULTI-AGENT PLANNING BY LEARNING ENVIRONMENT

SRAJAN DONGRE, AMAR NATH DSILVA, SWATI ARORA, SATYENDRA SINGH CHOUHAN AND RAJDEEP NIYOGI

Department of Computer Science and Engineering, IIT Roorkee, ROORKEE - 247667, U.K., INDIA

Email: {srajan.dongre, amardishilva, aroraswati.88, satyendra.chouhan}@gmail.com, rajdpfec@iitr.ernet.in

ABSTRACT: The problem with the Multi-Agent Systems (MAS) is the uncertainty associated with the environment being operated under. The agents can sense the part of environment which lies in its immediate neighborhood. However, these agents are ignorant about the complete information about their world. Hence, it becomes very tedious to avoid any unforeseen hurdles which an agent might face during its future course of actions. This study propose a novel idea to approach the multi-agent planning problems by distributed learning" of the environment. In distributed-learning each agent learns an individual model and shares the results with each other. A solution to this problem has been proposed by first learning the environment i.e. Possible hurdles, free paths, etc. and then finding the plan which when followed will lead each agent to its goal state.

KEYWORDS: Artificial intelligence, Grid world domain, Multi- agent planning.

INTRODUCTION

Artificial agents are replacing humans in wide range of tasks, from Metal sheet bending machines to NASAs Mars Land Rover, for tasks which can be performed by autonomous agents. These agents can be programmed to work in different types of environment and perform the required task efficiently. Agents need to think-before-they-act and then schedule their actions in order to reach their goal state, or in other words it needs to find the right "sequence of actions" to perform the task and reach the goal state. Part of this agent which performs this task of planning is termed as a planner. Most of the work in planning domain is done in a relaxed planning problem called "classical planning problem". Classical planning problem works under some assumptions about the system, which eases the general planning problem to something more solvable, like single agent working in a static environment with duration-less and deterministic actions, etc. However the real world problems are fairly complex with multiple agents working concurrently to achieve separate goals in a highly dynamic and partially observable environment. Future application of automated planning lies in actually employing autonomous intelligent agents for real world problems.

The study has focused on the problems faced by multi- agent planners and how it can be tackled using sensing and communication by learning the environment. In this paper a model has been proposed for the situation where agents communicate, perceive the world using its actuators (sensing) and plan the course-of-action accordingly. Here focus is on using learned-information in planning so that multiple agents can plan and coordinate with their resources and activities. Planning under such circumstances requires revision and re- generation of plans until agents reach a universal plan which, when acted upon, will meet individual goals without causing any kind of conflict.

Rest of the paper is organized as followings. Section II is all about the related work of the study. Section III describe about the Multi-agent planning problem. Section IV describes the proposed algorithm for proposed problem of multi-agent. The result and conclusion has been detailed in section V and VI respectively.

RELATIVE STUDY

This section briefly describes some works in the grid world domain for multi-agent planning. Various communities of researchers in multi-agent planning have different views on multi-agent planning. Therefore multi-agent planning can be considered in different ways [5]. Some Artificial Intelligence (AI) planning problems are inherently multi-agent planning problem. In this case problem is decomposed into different sub problems and is assign to different agents [8], [3]. In other way multi-agent planning can be thought as multiple agents planning together to achieve their goal such that their plan should not conflict [7], [4]. In both scenario multi-agents coordinate together via communication or defined policies [7], [11]. Also multi-agent planning can be restricted to classical planning [1] or it can represent planning problem with uncertainty [6]. But if there is no prior knowledge about environment then agent have to learn the environment before planning.

In this study, emphasized has been on the coordination between multiple agents with communication using centralized approach. Basic assumptions that are consider for this study are similar to classical planning i.e. environment is static and actions are deterministic. Here centralized planning agent is used to plan for multiple agents. Here centralized planning agent is used to plan for multiple agents. Additionally, agents do not have prior information about environment. Agent learns the environment and sends the information to planner. There- after planner generates the plan for agents.

A MULTI-AGENT PLANNING PROBLEM

Multi-agent planning is a broad field with many applications, but its subfields remain mostly dispersed and uncoordinated. Multi-Agent planning can be defined as; the problem of finding a plan of actions for a set of agents in order to achieve a set of goals, in which specification of their initial state and actions has been given. The issue of learning and adaptation in multi-agent systems has been given increasing attention in artificial intelligence research. It is becoming clear, given the dynamic environments in which we want our agent teams to interact, that behavioral repertoires and activities cannot simply be defined in advance. Multi-agent learning (MAL) refers to settings in which multiple agents learn simultaneously. Usually defined in a game theoretic setting, specifically in repeated games or stochastic games, the key feature that distinguishes multi-agent learning from single-agent learning is that in the former the learning of one agent impacts the learning of others [9]. Agent in Multi-agent system is a computer that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives. Multi-agent planning must take into consideration the fact that the activities of agents can interfere with one another - their activities must therefore be coordinated [10].

Consider a general multi-agent planning problem: Let there be a set of agents N = 1, 2,, k, Where each agent Ai can be identified by the number e.g. A_1, A_2, A_n and likewise. Let S be a finite set of states, S = S_0, S_1, S_2...S_g where S_0 is the initial state, S1 after that, and so on, till S_f which is the goal state. S0 being the initial state is known to all the agents. Now, all the agents are associated with some finite number of actions that they can perform. An action can be defined as a mapping from S to S. When an action from a set of actions A_i be formalized in the form of a mapping from S→S. For a set of action X, there can be more than one action that can be performed by an agent. A plan, therefore, is denoted by a series of action from a set of actions A which when acted upon, leads the state from S_0 to S_f.

Grid World Domain
For the purpose of further discussion in this paper, we will be discussing a multi-agent planning problem in grid world. The grid world domain consists of cells. The size of the grid is [6X6]. Cell that are darken in the figure 1 are obstacles. No cell can be occupied by two or more agents simultaneously.

Since in multi-agent system the knowledge of agent is limited and restricted to only local information, it only knows neighbors information and this scenario is considered in this grid world domain therefore in this grid world agent knows information about four neighbor cells i.e. left, right, up and down cell. Agent does their planning by moving in grid world architecture autonomously to reach the destination termed as goal state of agent. Here in this domain agent has the information only about the local obstacle i.e. it only knows the presence and absence of the obstacle in the four neighbor cells [2].

- Agents can only move in 4 directions UP, DOWN, LEFT, RIGHT with respect to their present cell. Therefore, set of permissible actions X, for every agent Ai in A, X= MOVE_UP, MOVE_DOWN, MOVE_RIGHT, MOVE_LEFT.
- All agents have some initial position, from where they are to start their moves. So the (overall) initial state of the grid world is where, S_i = all agents are at their initial cell for every Agent Ai belonging to A, Position (Ai) = initial position.
- All these agents have goal position where they need to reach and the goal state of the system is supposed to be attained only when each agents has reached its respective goal position, so the (overall) goal state of the system is where,
  S_g = all agents at their final/goal cell for every Agent Ai belonging to A, Position (A_i) = goal position.
- All agents in the grid can communicate with the planning agent (PA), i.e. exchange information freely with the planning agent.
- All agents are ignorant about the complete information of the grid, i.e. agents do not know the positions of all the hurdles in grid.
- Agents can only sense their neighbouring cells in the grid, i.e. an agent can check if the cell above it ,below it ,right to it and the cell in its left side is free to be occupied or not.
- A cell in the grid can be occupied by only one agent at a time.

Figure 1 depicts the grid world problem for three agents, A,B and C, in a 6x6 grid with their initial and final position marked, e.g. initial and final positions of A are marked as Ai and Af and likewise for other B and C, and blackened cells represent obstacles.

PROPOSED ALGORITHM FOR MULTI-AGENT PLANNING

We propose to solve the multi-agent problem by learning the environment which is being operated upon. This needs exploration of the surrounding by the agents. In order to optimize this

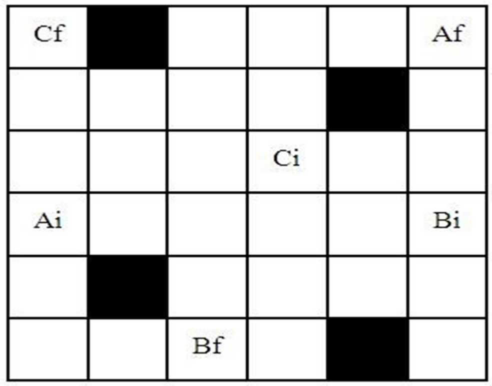

Figure1. Initial state and goal state of agent A, B and C in grid world

exploration, it makes sense to distribute the work load among the agents such that all the agents are allotted separate territories of exploration. It will ensure that each part of the environment is visited exactly once. To solve this problem we introduce a planning agent PA. Planning agent is separate entity very different from the acting agents. Although it helps in planning and execution, planning agent isn't directly involved in the actions performed to reach the goal state. The main functions performed by the planning agent will be as following.

- Communicate exploration details to acting agents.
- Collect and consolidate information received from agent.
- find path for agent's, and
- Help in synchronizing during execution.

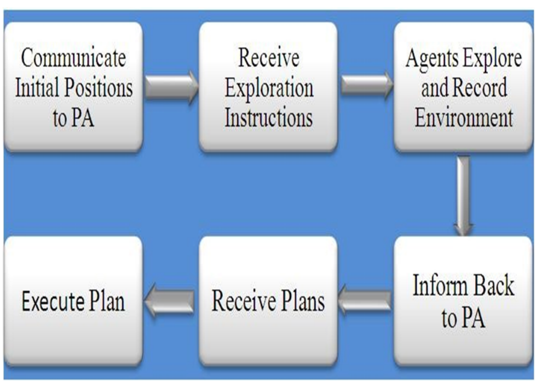

Figure2. Flow diagram for algorithm

Figure2 shows the flow diagram for proposed algorithm. Here, the planning agent acts as a central repository for all the information to be collected and then find the path for the agents to be followed. Firstly, the planning Agent knowing the size of the grid distributes the grid area into parts for each agent. The grid is divided into four parts in such a way that, these four areas form an exhaustive set and no two of them have any part in common. Once all the agents are informed about their area for exploration, they start freely exploring their respective areas and start recording its features, e.g. obstacles in the grid. This way one also can deduce which of the spaces are also available to be occupied. Once, finished with the exploration, all these agents inform the planning agent (PA) about their observation. Meanwhile the planning agent has been waiting for all the agents to send back the recorded information about their area. Once it receives this information from all the agents, it combines all these individual pieces of information to conceive the complete view of the grid. Now, the planning agent can decide upon the individual path to be followed by each agent.

We can visualize this with 4 agents and a planning agents in a grid world as depicted in Figure 3 , where the shaded portion represents the territory for exploration allotted to the agent residing in that area.

EXPERIMENTAL RESULTS

The purpose of our experiment is to demonstrate the need of communication in multi-agent system and then shown the proposed communication model. For this we used the existing grid world domain in which multiple agent are acting together to achieve their goal. All of below experiments that we perform on 2.26 GHz Intel Pentium using 2 GB of its RAM and on windows operating system. Consider each cell of grid as a coordinate starting from 0 to 6 from left to right and top to bottom.

Two results are shown in Table I and Table II. The situation may exist where two of the agents may try to visit the same place and in situation one of the agents has to wait. This type of waiting actions are shown in Table II, in which agents have to wait before executing their plans as both of the agent are trying to reach the same location. The results clearly are tabled with information of initial state, goal state and corresponding plan for agents. The execution time taken by CPU to execute the plan with different initial and goal state scenario is also mentioned with plan valid or not.

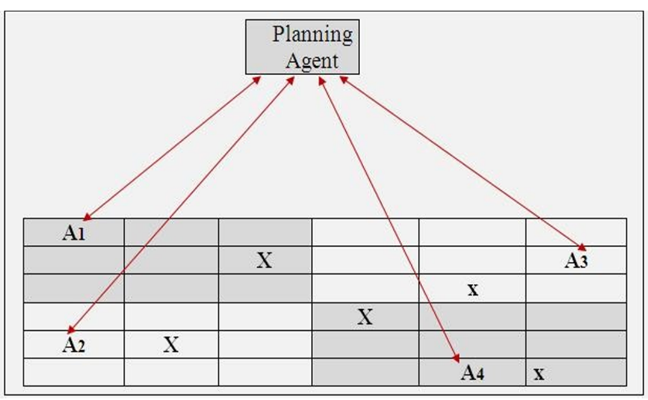

Figure3. Different territory of all agents shown by shaded area

All agents communicate with planning agent

Table I. Planning with 3 agents in a 6X6 grid with some obstacles and planning agent

Initial state	Goal State	Resulting Plan	Execution Time(ms)	If plan Valid?
A[0,0], B[1,0], C[4,3]	A[0,3], B[4,0], C[4,5]	A: 00 01 02 03 B: 10 20 30 40 C: 43 44 45	18	Yes
A[2,2], B[3,1], C[1,3]	A[5,2], B[3,2], C[2,5]	A: 22 21 20 30 40 50 51 52 B:31 32 C: 13 14 15 25	15	Yes
A[2,1], B[0,1], C[4,2]	A[5,3], B[2,3], C[4,5]	A: 53 52 51 50 40 30 20 21 B: 01 02 03 13 23 C: 42 43 44 45	19	Yes
A[5,4], B[4,3], C[2,1]	A[3,4], B[4,5], C[2,3]	A: 54 53 52 42 32 22 23 13 14 15 25 35 34 B:43 44 45 C: 21 22 23	25	Yes
A[13,], B[05,], C[21,]	A[15,], B[54,], C[23,]	A:13 14 15 B: 05 04 03 02 01 11 21 22 32 42 43 44 C: 21 22	20	Yes
A[43,], B[31,], C[50,]	A[45,], B[01,], C[05]	A:43 44 45 B: 31 21 11 01 C: 50 51 52 42 32	23	Yes

CONCLUSION

This paper presented a centralized learning framework to address the problem of model learning. The algorithm presented in this study shows how communication among agents is made and then plan is made based on the communication. The result that are shown in Table II and I describes the overall action's scenario with and without wait. The actions with wait indicate the situation where multiple agents may try to visit the same place at the same time. Hence this scenario is also mention here in the result so that agent could perform action without disturbing others.

REFERENCES

R. I. Brafman and C. Domshlak., "From one to many: Planning for loosely coupled multi-agent systems", In *International Conference on Automated Planning and Scheduling*, 2008, pp 28–35.

S. S. Chouhan and R. Niyogi, "Multi-agent planning in a grid world domain". In *Second International Conference on Advanced Computing & Communication Technologies*, volume 10, 2012.

Table 2. Planning with wait action

Initial state	Goal State	Resulting Plan	Execution Time(ms)	If plan Valid?
A[0,0], B[1,0], C[4,3]	A[0,3], B[4,0], C[4,5]	A: 00 01 02 03 B: 10 20 30 40 C: 43 44 45	20	Yes
A[2,2], B[3,1], C[1,3]	A[5,2], B[3,2], C[2,5]	A: 22 W 32 42 52 B: 31 W 32 C: 13 14 15 25	15	Yes
A[2,1], B[0,1], C[4,2]	A[5,3], B[2,3], C[4,5]	A: 21 22 32 W 42 43 53 B: 01 02 03 13 23 C: W 42 W 43 44 45	20	Yes
A[5,4], B[4,3], C[2,1]	A[3,4], B[4,5], C[2,3]	A: 54 W 44 34 B: 43 W 44 45 C: 21 22 23	23	Yes
A[13,], B[05,], C[21,]	A[15,], B[54,], C[23,]	A: 13 14 W 15 B: 05 W 15 25 35 34 44 54 C: 21 22 2323	18	Yes
A[43,], B[31,], C[50,]	A[45,], B[01,], C[05]	A: 43 W 44 W 45 B: 31 21 1101 C: 50 51 52 53 54 W 44	22	Yes

B. J. Clement and E. H. Durfee. “Top-down search for coordinating the hierarchical plans of multiple agents”. In *Proceedings of the third annual conference on Autonomous Agents*, 1999, pp. 252–259.

Y. Dimopoulos and P. Moraitis. “Multi-agent coordination and cooperation through classical planning”. In *Intelligent Agent Technology*, 2006, pages 398–402.

E. H. Durfee. “Distributed problem solving and planning”. In Multi-agent systems and applications. Springer, 2001, pp. 118–149.

Guestrin, D. Koller, and R. Parr. “Multiagent planning with factored mdps”. In NIPS, volume 1, 2001, pp. 1523–1530.

R. B. Larbi, S. Konieczny, and P. Marquis. “Extending classical planning to the multi-agent case: A game-theoretic approach”, In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Springer, 2007, pp. 731–742

R. Nissim, R. I. Brafman, and C. Domshlak. “A general fully distributed multi-agent planning algorithm”. In *Proceedings of the 9th International Conference on Autonomous Agents and Multi agent Systems*, 2010, pp. 1323–1330.

Y. Shoham and R. Powers. “Multi-agent learning: Problem definition” .In *Encyclopedia of Machine Learning,*. Springer, 2010, pp. 694–696.

M. Wooldridge. “ An introduction to multiagent systems”. John Wiley & Sons, 2009.

F. Wu, S. Zilberstein, and X. Chen. “Online planning for multi-agent systems with bounded communication”. Artificial Intelligence, 175(2), 2011, pp. 487–511.