

# An Analysis of the Effect of Communication for Multi-agent Planning in a Grid World Domain

Satyendra Singh Chouhan

Department of Electronics & Computer Engineering, Indian Institute of Technology, Roorkee  
Roorkee-247667, Uttarakhand, INDIA  
Email: saty1pec@iitr.ernet.in

Rajdeep Niyogi

Department of Electronics & Computer Engineering, Indian Institute of Technology, Roorkee  
Roorkee-247667, Uttarakhand, INDIA  
Email: rajdpfec@iitr.ernet.in

**Abstract**— Agent-based technology has generated a lot of attention in recent years because of its promise as a new paradigm for conceptualizing, designing, and implementing software systems. Some problems in real world cannot be handled by a single agent. Multiple agents work together to accomplish some task. Although multi-agent systems (MASs) provide many potential advantages, they also present many difficult challenges. This paper illustrates the importance of communication for planning in a multi-agent setting by considering a grid world domain that consists of obstacles at different locations. This paper provides a theoretical framework that is validated by the experimental results. Performance analysis with respect to plan size and execution time is also reported.

**Index Terms**— Grid world Domain, Communication, Multi-agent system

## I. INTRODUCTION

Research in multi-agent systems have led to its applicability in varied real world scenarios such as e-commerce [1], supply chain management [2], robotics [3], and also developing complex game application [4]. It is widely being advocated for use in networking and mobile technologies, to achieve automatic and dynamic load balancing, high scalability, and self-healing networks. Such systems are becoming increasingly important as they draw together a number of important trends in modern technology [5].

A multi-agent system (MAS) consists of a collection of loosely-coupled interacting autonomous agents working in an environment. Here agents are usually software agents and can perform actions, have some computational abilities, and may communicate with each other. Multi-agent systems support modularization i.e., a large complex problem is handled by developing a number of functionally specific and

modular components that are efficient to solve a specific problem aspect. This decomposition allows each agent to use the most appropriate paradigm for solving its particular problem. When dependent problems arise, the agents in the system coordinate with one another to ensure that interdependencies are properly managed. Cooperation refers to distributed and communicated group of agents that share a common interest and work together to achieve a common goal in an environment [6], [7].

The remaining paper is structured as follows. In Section 2 we discuss about agent communication and grid world domain. Section 3 describes a model used for communication. Section 4 describes a theoretical framework for multi agent communication. In Section 5 the experimental results are presented along with the analysis. Related work is discussed in Section 6. Section 7 concludes the paper.

## II. AGENT COMMUNICATION

The information exchange between agents is termed as communication between agents. Communication improves the behavior of agents and discourages any regard to other agent's internal structure. The communication between agents may be peer to peer, broadcast or mediated. In recent years several techniques evolved for the communication between agents. Some of these include Blackboard system [8], message passing via communication standards like knowledge query manipulation language (KQML) [9], and FIPA-ACL [10]. Blackboard system is an indirect approach of communication between agents. In this technique there is a common blackboard which is shared between all agents. For peer to peer communication KQML and FIPA-ACL are two most popular languages. Since these languages are wrapper languages and have high level of abstraction they can be extended depending on the need

of particular system [11]. Therefore in most of cases either these languages are extended according to need of application or engineers develop communication techniques suited for a particular application. In this paper we used a shared variable for agent communication. In our work the agents are collaborative in nature.

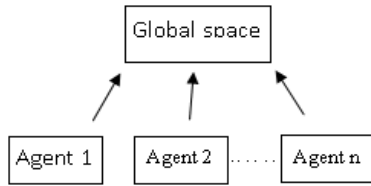


Figure1. Conceptual model for communication

**A. Grid World Domain**

In our work we used a two dimensional grid world domain that consists of cells arranged in a matrix. Some cells may be obstacles, represented by gray color in the figure shown below.

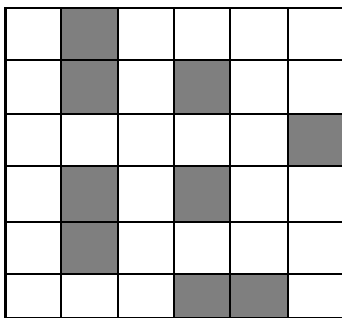


Figure2. A simple grid world domain

The size of the grid is 6X6. Cell that are darken in the figure are obstacles. No cell can be occupied by two or more agents simultaneously. Since in a multi-agent system the knowledge of an agent is limited and restricted to only local information, it only knows the information of its adjacent cells. The task of an agent is to reach the given goal state.

In this domain an agent can perform 4 actions, namely MOVE\_LEFT, MOVE\_RIGHT, MOVE\_UP, and MOVE\_DOWN. For example, after performing the action MOVE\_DOWN, the agent moves one cell below the current cell. When all agents achieve their goal state then we says that the multi-agent system has achieved the goal, otherwise it is said to be failure.

**III. ARCHITECTURE USED FOR COMMUNICATION**

Communication is done using a shared variable. When an agent takes an action it updates its information in shared variable (communication module) and the other agent may retrieve the information from the

communication module. The architecture of overall simulation system is shown below. The architecture of system consists of the following modules.

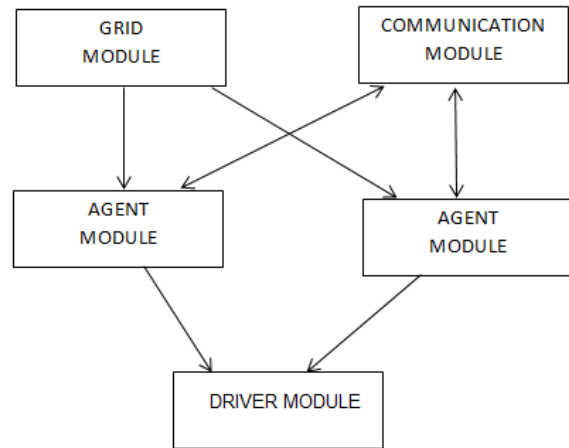


Figure3. Architecture of simulation system

**A. Grid module**

This module is used to design the grid world system on to which the multi-agent system is simulated.

**B. Communication module**

This module is used for communication between agents. In this communication module the agents share the information through the share variable. Agent interacts indirectly via the shared variable by broadcasting their current state.

**C. Agent module**

This module is used to describe the actions of the agents.

**D. Driver module**

This is used to run the program.

**IV. THEORETICAL FRAMEWORK**

**A. Notations**

- i, j: agents
- $s_{j,curr}$ : the current state of agent j
- $s_{j,next}$ : the next state of agent j
- P: probability

Given the current state of the agents, the probability that the agents move to the next states is given as:

$$P_{i,j(\text{without-info-sharing})} = P([s_{i,next}, s_{j,next}] | [s_{i,curr}, s_{j,curr}]) = P(s_{i,next} | s_{i,curr}) \cdot P(s_{j,next} | s_{j,curr}) \quad (1)$$

Since  $i$  and  $j$  are acting independently (i.e., there is no information sharing).

*Theorem 1.*  $Z_{i,j}(\text{with-info-sharing}) \geq Z_{i,j}(\text{without-info-sharing})$

Proof: Consider a term  $t$  in  $Z_{i,j}(\text{with-info-sharing})$  and the corresponding term  $t'$  in  $Z_{i,j}(\text{without-info-sharing})$ . Since the agents are communicating so  $P(s_{i,next} | s_{i,curr})$ .

$P(s_{j,next} | s_{j,curr})$  is more. Thus  $t \geq t'$ . Now  $Z_{ij}$  is the product of all the terms.

Since each term  $t \geq t'$ , so the product of the term in communication is also greater than equal to the product of term without communication. Hence the result.

We illustrate the above result by taking an example of the grid world structure in figure 2.

Let the initial position of agent  $i$  and  $j$  be  $(0, 2)$  and  $(2, 0)$  respectively. The final positions of the agents  $i$  and  $j$  are  $(4, 2)$ ,  $(2, 4)$  respectively.

Now, without information sharing between agents in the given grid world domain of figure 2, we find the individual probability values as follows. The probability of agent  $i$  taking first step from the initial position is  $1/2$  since there is available only two choices. Similarly for all the other steps to the goal state the probability values are  $1, 1/3$ , and  $1$ . For the agent  $j$  the probability values are  $1/3, 1, 1/3, 1$ .

Thus,

$Z_{i,j}(\text{without-info-sharing}) = P(s_{i,next} | s_{i,curr}) \cdot P(s_{j,next} | s_{j,curr})$   
at each step is  $1/2 \times 1/3, 1 \times 1, 1/3 \times 1/3, 1 \times 1$ .

Now with information sharing the probability of agent  $i$  for each step will be  $1/2, 1, 1/2$ , and  $1$ . The probability of agent  $j$  for each step will be  $1/3, 1, 1, 1$ .

Thus,

$Z_{i,j}(\text{with-info-sharing}) = P(s_{i,next} | s_{i,curr}) \cdot P(s_{j,next} | s_{j,curr})$

at each step is  $1/2 \times 1/3, 1 \times 1, 1/2 \times 1, 1 \times 1$ . We can see that that the value of each term in this case is more than the corresponding term in the previous case. Thus,

$Z_{i,j}(\text{with-info-sharing}) \geq Z_{i,j}(\text{without-info-sharing})$ .

### B. Definition

Chain Probability: let  $s_1, s_2, \dots, s_r, s_g$  be a sequence of states that an agent makes. Where  $s_1$  is the initial state and  $s_g$  is the final (goal) state. We call the probability of moving from  $s_1$  to  $s_g$  via the sequence as the chain probability, denoted as  $P(s_1, s_2, \dots, s_r, s_g)$ . Now,

$P(s_1, s_2, \dots, s_r, s_g) = P(s_1) \cdot P(s_2 | s_1) \cdot \dots \cdot P(s_g | s_r)$

*Theorem 2.* Chain probability (with sharing)  $\geq$  chain probability (without sharing)

Proof: let us denote the LHS by  $A$  and the RHS by  $B$

Thus  $A = \prod_{\forall j} \prod_{\forall i} Z_{ij}(\text{with info sharing})$   
 $B = \prod_{\forall j} \prod_{\forall i} Z_{ij}(\text{without info sharing})$

By Theorem 1

we have  $Z_{i,j}(\text{with-info-sharing}) \geq Z_{i,j}(\text{without-info-sharing})$

Thus the product of the terms for  $Z_{i,j}(\text{with-info-sharing})$  is more than that for  $Z_{i,j}(\text{without-info-sharing})$

Hence  $A \geq B$

For the example considered, by substituting the values obtained before, we get

$$\begin{aligned} A &= \prod_{\forall j} \prod_{\forall i} Z_{ij}(\text{with info sharing}) \\ &= 1/6 \times 1 \times 1/9 \times 1 \\ &= 1/54 \\ B &= \prod_{\forall j} \prod_{\forall i} Z_{ij}(\text{without info sharing}) \\ &= (1/6 \times 1 \times 1/2 \times 1) \\ &= 1/12 \end{aligned}$$

We can see that  $A \geq B$ .

*Theorem 3:* Expected time of convergence with information sharing ( $E_{\text{info-sharing}}$ )  $\leq$  Expected time of convergence without information sharing ( $E_{\text{without-info-sharing}}$ ).

Proof: Let  $m$  be the number of steps (transitions) for convergence. Expected convergence time = search time  $\times$  number of free cells. Now search time depends on the local information available. Moreover,

Search time (with information sharing)  $\leq$  search time (without information sharing). Therefore,  $E_{\text{info-sharing}} \leq E_{\text{without-info-sharing}}$ .

## V. EXPERIMENTS AND RESULTS

The purpose of our experiments is to demonstrate the importance of communication in a multi-agent system. For this we used the grid world domain in which two agents are acting together to achieve their goal. The experiments were carried out on a 2.26 GHz Intel Pentium machine with 2 GB RAM. The programs are written in C++ and executed on Windows 7. We used the convention of naming each cell of the grid as a coordinate starting from 0 to 6 from left to right and top to bottom.

Abbreviations:

IA:-Initial position of agent A.

IB:-Initial position of agent B.

FA:-Final position of agent A.

FB:-Final position of agent B.

N:-Number of steps taken by agent to move from IA to

FA.

ET:-Time taken by CPU to perform the simulation (in milliseconds).  
 NA: - Number of steps taken by Agent A to reach the goal position.  
 NB: - Number of steps has taken by Agent B to reach the goal position.  
 N\_max = maximum (NA, NB)

**A. Single agent system without communication**

For a single agent in the grid world domain the experimental results for the different initial-final states are given in Table 1 and figure 4.

TABLE1. EXPERIMENT RESULTS WITH DIFFERENT CASES

Sr. no.	IA	FA	N	ET
1	2,4	3,4	1	0.009
2	0,2	2,2	2	0.015
3	0,4	4,4	4	0.032
4	2,2	3,5	4	0.032
5	1,5	4,3	5	0.046
6	1,0	5,1	7	0.078
7	0,0	5,2	7	0.078
8	4,4	1,0	7	0.078
9	1,5	5,1	8	0.093
10	0,5	5,0	10	0.109

On performing this experiment the path chosen by agent is shown below. And no. of steps taken by agent is 10. And the average time taken by agent to reach up to desire state was 0.109 msec.

Similarly number of experiment is performed by taking different position [Initial state] and goal state of agent. Since there is only single agent and if there is certainly a path is available between initial position and final position then there is no need of implementing communication model.

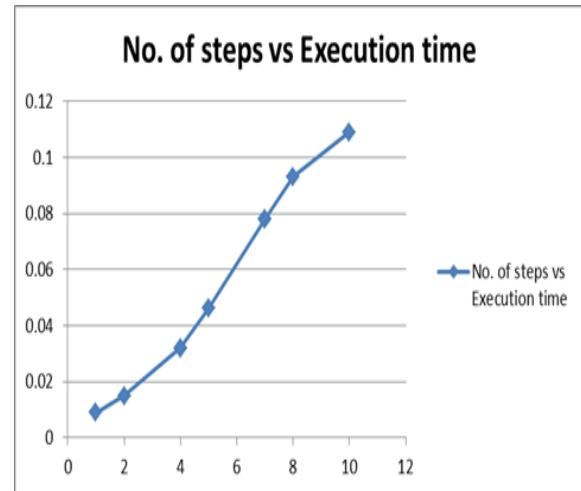


FIGURE4. NO. OF STEPS VSEXECUTION TIME

**B. Two agent system without communication**

**1) No U-turn is allowed**

TABLE 2. TWO AGENT SYSTEM (NO COMMUNICATION)

sr. no.	IA	FA	IB	FB	NA	NB	N-max	ET
1	2,4	3,4	2,2	4,2	1	2	2	0.047
2	0,2	2,2	0,4	3,4	2	3	3	0.063
3	0,4	4,4	0,0	2,3	4	5	5	0.109
4	1,0	5,1	0,5	5,5	5	7	7	0.172
5	0,0	5,5	0,5	5,0	10	10	10	0.234
6	0,0	5,0	0,2	5,2	5	5	5	0.108
7	2,2	4,5	0,1	5,2	5	6	6	0.136
8	2,0	5,0	0,0	5,5	3	10	10	0.234
9	3,2	4,5	2,3	4,2	4	3	4	0.084
10	4,0	2,2	0,4	4,4	4	4	4	0.087

For two agents in the grid world domain the experimental results for the different initial-final states are given in Table 2 and figures 5, 6.

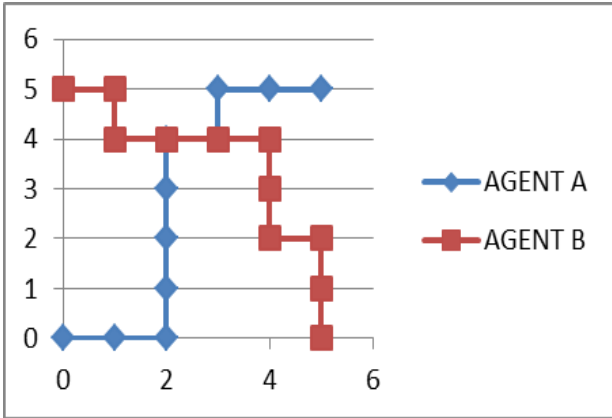


Figure5. A graph representation of movement of agent A and Agent B

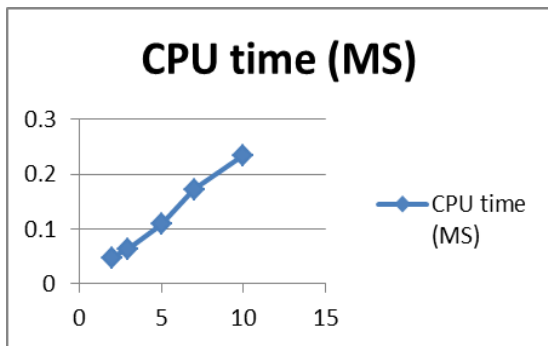


Figure6. No. of steps VS CPU time(ms)

a) Collision

We have observed that some situations there are collision of the agents. For instance, suppose that the agents are at initial positions (0, 2) and (2, 0); the corresponding final states of the agents are (4, 2) and (2, 4). This is shown in figure 7.

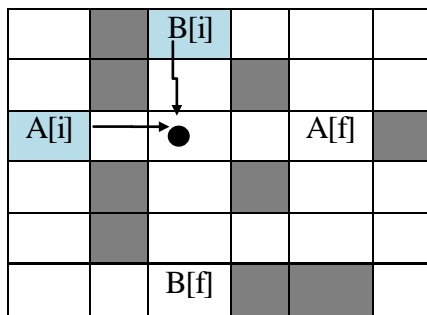


Figure7. Collision scenario in grid

b) Deadlock

There may be the possibility of a deadlock. This arises when any one agent get stuck somewhere, unable to make any further move, even though the other agent has achieved its goal. For instance, suppose that the initial position of agent A and B are (0, 0) and (4, 5) respectively and the goal state of the agents A and B are (5, 0) and (5, 2) respectively. At some point of execution

we find that A has achieved its goal state but B has got stuck at the cell (5, 5).

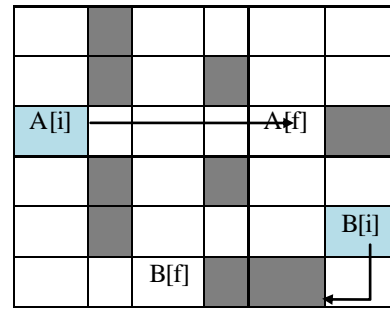


Figure8. Deadlock scenario in grid

2) U-turn is allowed

We have found that in this case deadlock cannot arise. However collision may still occur. An example situation is shown in the figure where the agent A now being able to take an U-turn prevents the deadlock.

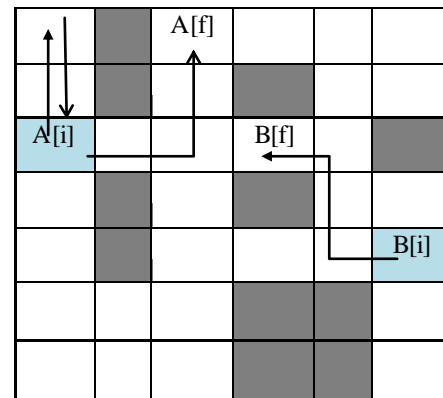


Figure9. A sample case of agent action

The following Table 3 corresponds to situations where deadlock occurs. Table 4 corresponds to situations where no deadlock occurs.

TABLE3. EXPERIMENTS AND RESULTS WITH DIFFERENT CASES FOR TWO AGENT SYSTEM

Sr no.	Agent A (initial position),(final position)	Agent B (initial position),(final position)	No. of steps taken (without U turn)	Result
1.	(1,0),(0,2)	(0,5),(5,0)	2	Agent A blocked
2.	(0,0),(4,4)	(4,5),(5,2)	1	Agent B blocked
3.	(1,0),(5,0)	(5,0),(0,0)	2	Agent A & B blocked
4.	(0,3),(3,0)	(2,2),(0,3)	1	Agent A&B blocked

a) Results with U-turn

TABLE4. EXPERIMENTS AND RESULTS WITH DIFFERENT CASES FOR TWO AGENT SYSTEM (with U-turn)

Sr no.	Agent A (initial position),(final position)	Agent B (initial position),(final position)	No. of steps taken (with U turn)	Min No. of steps needed	Exec ution time	Result
1.	(1,0),(0,2)	(0,5),(5,0)	10	10	0.245	successful
2.	(0,0),(5,0)	(4,5),(5,2)	6	5	0.162	successful
3.	(2,0),(5,0)	(5,0),(0,0)	10	5	0.251	successful
4.	(0,3),(3,0)	(2,2),(0,3)	10	6	0.248	successful
5.	(2,4),(3,4)	(2,2),(4,2)	2	2	0.062	Successful
6.	(1,0),(5,1)	(0,5),(5,5)	7	7	0.184	Successful
7.	(2,2),(4,5)	(0,1),(5,2)	6	6	0.158	Successful
8.	(3,2),(4,5)	(2,3),(4,2)	5	5	0.124	Successful
9.	(4,0),(2,2)	(0,4),(4,4)	4	4	0.103	Successful
10.	(0,0),(5,5)	(0,5),(5,0)	10	10	0.252	Successful
11.	(0,2),(4,2)	(2,0),(2,4)	-	-	-	Collision

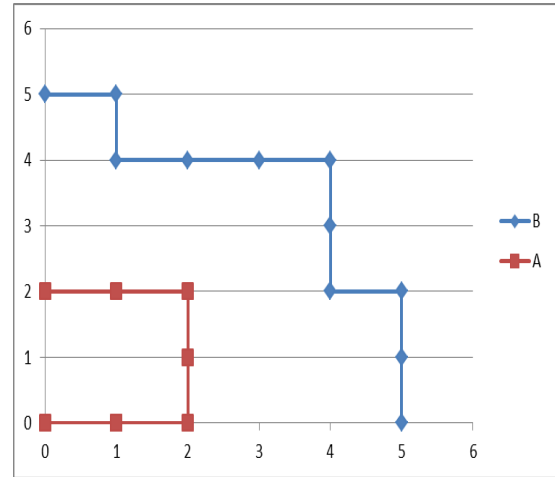


Figure10. Movement plan of agent A & agent B for Sr no. 1 (With U turn)

b) Analysis of result:

The above results can be summarized in Table 5.

TABLE5. ANALYSIS OF RESULTS

	COLLISION	DEADLOCK
Without Communication (no U turn )	Yes	Yes
Without Communication (U turn )	Yes	No

C. Two agent system with communication model

In a two agent system with communication, each agent communicates their current position and the next position that will arise for some action. Here next position indicates the intention of the agent to move to this position but it does not mean that the agent has actually moved to this position.

Before taking an action every agents will check the “other’s next position”. If the “other’s next position” is the same as “its next position” then they will look at a signal variable. Otherwise it takes the action according to local information available.

The purpose of the signal variable is to give a signal to the agents so that they can cross without any collision. The variable is a tuple (a, b) that can either be (1, 0) or (0, 1). For instance, (1, 0) means agent A is waiting; (0, 1) means agent B is waiting.

After taking the actions, the agents update the shared variable indicating the current position and next position.

In Table 6 the experimental results with communication are given. Now with communication there cannot be deadlock or collision. This is summarized in Table 7.

TABLE6. TWO AGENTS WITH COMMUNICATION

Sr. no.	IA	FA	IB	FB	N	ET
1	2,0	2,4	0,2	4,2	5	0.133
2	2,1	3,5	0,5	4,4	6	0.167
3.	0,4	4,4	0,0	2,3	5	0.141
4	1,0	5,1	0,5	5,5	7	0.186
5	0,0	5,5	0,5	5,0	10	0.264
6	0,0	5,0	0,2	5,2	5	0.137
7	2,2	4,5	0,1	5,2	6	0.158
8	2,0	5,0	0,0	5,5	10	0.265
9	3,2	4,5	2,3	4,2	5	0.138
10	4,0	2,2	0,4	4,4	4	0.118

Consider row 1 of Table 7. This result suggests that now with communication there is no collision; previously when there was no communication, collision occurred (refer to the last row of Table 4).

TABLE7. ANALYSIS OF RESULTS

	COLLISION	DEADLOCK
Without Communication (U turn)	Yes	No
Using communication	No	No

## VI RELATED WORK

In this section we briefly report some works in a grid world domain for multi-agent planning. A continual

planning technique is developed and implemented for the grid world in [12]. Our method is not based on continual planning.

Another similar domain based on a grid world is the packet world environment—that consists of a number of agents, packets, and baskets occupying different cells of a grid. The task of the agents is to cooperate among each other to pick up the packets and place them in the

Baskets In [13] a cooperative multi-agent system for solving the packet world problem is proposed.

In [14] Packet-World domain is considered. It consists of a number of differently colored packets that are scattered over a rectangular grid. The task of the agents is to cooperate among themselves to place these packets in the corresponding colored destination (cell). Architecture for such a multi-agent setting is suggested in [14].

## VII CONCLUSION AND FUTURE WORK

In this paper we considered the problem of multi-agent planning in a grid world domain. Our experimental results demonstrate the importance of communication in a multi-agent setting. As part of our future and ongoing work we would like to study similar grid world domains of different sizes and structures. We would also like to analyze the behavior of several agents in such domains.

## REFERENCES

- [1] Zhiyong Sui, Lixin Huang, "Study on an E-Commerce System based on Multi-Agent Technology," International Symposium on Electronic Commerce and Security, pp.714-717, 3-5 Aug. 2008
- [2] Hellingrath, B, Bohle, C, van Hueth, J, "A Framework for the Development of Multi-Agent Systems in Supply Chain Management," 42nd Hawaii International Conference on System Sciences, pp.1-9, 5-8 Jan. 2009
- [3] Banik, S.C, Watanabe, K, Habib, M.K, Izumi, K, "An emotion-based task sharing approach for a cooperative multi-agent robotic system,". IEEE International Conference on Mechatronics and Automation, pp.77-82, 5-8 Aug. 2008
- [4] Pena, L, Ossowski, S., Pena, J.-M., "vBattle: A new framework to simulate medium-scale battles in individual-per individual basis,". IEEE Symposium on Computational Intelligence and Games, pp.61-68, 7-10 Sept. 2009

- [5] Ahmad, H.F., "Multi-agent systems: overview of a new paradigm for distributed systems," 7<sup>th</sup> IEEE International Symposium on High Assurance Systems Engineering, pp. 101- 107, 2002.
- [6] N. Vlassis, A Concise Introduction to Multi-agent Systems and Distributed Artificial Intelligence, 1<sup>st</sup> ed., Morgan & Claypool, 2007.
- [7] I. J. Timm, D. Pawlaszczyk, "Large scale multi-agent simulation on the grid," IEEE International Symposium on Cluster Computing and the Grid, pp. 334-341 2005.
- [8] Song Qiang, Lingxia Liu; , "The Implement of Blackboard-Based Multi-agent Intelligent Decision Support System," Second International Conference on Computer Engineering and Applications, pp.572-575, 19-21 March 2010
- [9] Xiaojun Wu, Jinhua Sun, "Study on a KQML-Based Intelligent Multi-agent System," International Conference on Intelligent Computation Technology and Automation, pp.466-469, 11-12 May 2010
- [10] Foundation for Intelligent Physical Agents(FIPA). "Fipa2000 agent specification", <http://www.fipa.org>.
- [11] Yaqiu Liu, Xue Zhang, Qu Wu , "Optimizing KQML for usage in wood drying multi-agent coordination system," Control and Decision Conference, pp.3725-3730, 23-25 May 2011.
- [12] M. Brenner, B. Nebel, "Continual planning and acting in dynamic multivalent environments," Autonomous Agents and Multi-Agent Systems, vol. 19, no. 3, pp. 297-331, 2009.
- [13] Abusnaina, A.A.A, Chan Huah Yong, "Cooperative Multi-Agent System for solving Packet World Problem in Grid," IEEE 9<sup>th</sup> Malaysia International Conference on Communications, pp.754-758, 15-17 Dec. 2009.
- [14] D. Weyns, A. Helleboogh , T. Holvoet, "The Packet-World: A Test Bed for Investigating Situated Multi-Agent Systems, Software Agent-Based Applications, Platforms and Development Kits", [online], 2005.

**Dr. Rajdeep Niyogi** is an Assistant Professor in the Indian Institute of Technology (IIT) Roorkee. He did his Ph.D. in Computer Science and Engineering from IIT Kharagpur. His research interests include Automated Planning, Formal Methods and Distributed Systems. He is a member of ACM.

**Satyendra Singh Chouhan** is an MTech student of Computer Science and Engineering at IIT Roorkee. His research interests include communication in multi-agent systems and distributed systems.

**How to cite this paper:** Satyendra Singh Chouhan,Rajdeep Niyogi,"An Analysis of the Effect of Communication for Multi-agent Planning in a Grid World Domain", International Journal of Intelligent Systems and Applications(IJISA), vol.4, no.5, pp.8-15, 2012. DOI: 10.5815/ijisa.2012.05.02