

Multi-Agent Negotiation Strategies for Task Allocation Process in E-Commerce System

Ndayizeye Prudence
Dept. of CS & IT
SHITS Allahabad UP INDIA

W. Jeberson
Dept. of CS & IT
SHITS Allahabad UP INDIA

BD Mazumdar
Dept. of CS
ICST Varanasi

Abstract-In this model we have tried to analyze the cooperative behaviour in absence of any central authority. It clearly indicates that on increasing the number of cooperative entities the performance of the system improves. Also on varying the behaviour of agents interesting results in terms of personal gain and throughput are observed to business the products in E-Commerce. Though non-dedicated agents seem to draw benefits in terms of monetary unit but they are covering more distance per package delivery the products to destination agents. Hence it results in overall degradation in system performance of model. We look forward to work in the direction of comparing the performance of the system when the agents are cooperative with the case of agents working individually (not seeking help while delivering the package from other agents).

Keywords: Multi-agent, e-commerce, negotiation, BDI, NetLogo

INTRODUCTION

The process of estimating the cost of software has been of interest to researchers for decades. Some have developed sophisticated algorithms calibrated with historical data to improve the estimation process [1]. Others have found ways to combine different estimation methods such as bottoms up and analogy to arrive at estimates with a high degree of confidence [7]. While this research has helped shift the field of software cost estimation from an art to more of a science, the process of estimation remains prone to human errors and biases. These can be especially problematic when there is little information available about the people, technologies, development environment, and process used for developing software. Even in the face of missing information, humans make assumptions that help them develop software cost estimates. While these assumptions are not always justified, they certainly influence the outcome and accuracy of software cost estimates. The fields of human decision making and cognitive science help to further inform this issue. Tversky and Kahneman (1974) proposed that many human decisions are based on beliefs concerning the likelihood of uncertain events. Occasionally, beliefs concerning uncertain events are expressed in numerical form as odds or subjective probabilities. Their work showed that people rely on a limited number of heuristic principles which reduce the complex task of assessing probabilities and predicting values to simpler judgmental operations. Multi-agent paradigms have been developed for the negotiation and brokering in software of E-Commerce for task allocation method in DPS framework. Few of the models consider the

mental states, social settings and trust values but rarely any model depicts their combination.

In this work a combined model of belief, desire, intention (BDI) model [2] for agent's mental attitudes and social settings are used to model their cognitive capabilities. The mental attitudes also include preferences, commitments along with BDI. These attributes help to understand the commitment and capability of the negotiating agent.

In this work, we present mathematical model of MAS in B2B E-Commerce. In this chapter, we apply the task allocation algorithm for DPS framework to business the product in E-Commerce system. In the first section, we describe the MAS concept in E-Commerce software and different types of E-Commerce system for doing business in the world wide web, and the work & applications of MAS in B2B E-Commerce system are described in second section; agent based task allocation method of DPS framework in B2B business for this model is described in third section. The problem description, algorithms, experimental setup & implementation, results, and observation of the work for modeling the multi-agent systems (MAS) in B2B E-Commerce have been presented in section four. The discussion of our work is shown in section five and finally conclude of this chapter in section six is of conclusion of the experimental work.

MAS IN E-COMMERCE

"E-Commerce provides ways to exchange information between individuals, companies, and countries and, most important of all between computers". More simply put, E-Commerce is the movement of business onto the World Wide Web (WWW). This movement has been broken up into two main sectors: business-to-business (B2B) and business-to-customer (B2C). An agent in E-Commerce is a software program that acts flexibly on behalf of its owner to achieve particular objectives. Buyer agent and customer agent give instructions to its agent to fulfill his all needs for communicating in E-Commerce [8]. An agent for communication in E-Commerce must be a good listener, analyzer and cooperative in nature; as well as has the quality of good coordination, good communication and negotiation with other agents [7].

The agent software in E-Commerce must exhibit the following properties:

It should be autonomous: capable of making decisions about what actions to take without constantly referring back to its user.

It should be reactive: able to respond appropriately to the prevailing circumstances in dynamic and unpredictable environments.

It should be proactive: able to act in anticipation of future goals so that its owner's objectives are met.

The process of working of brokering as often occurs in E-Commerce involves a number of agents. Basic components of E-Commerce which are working in brokering process: buyer agent or client agent, broker agent, seller agent, and Courier_service agent. A buyer agent looking for products may be supported by a broker agent that takes its buyer agent's queries and contacts other agents and seller or Courier_service agents or looks at the web directly to find information on products within the buyer agent's scope of interest.

In other words, E-Commerce or electronic commerce, a subset of e-business, is the purchasing, selling, and exchanging of goods and services over computer networks (such as the Internet) through which transactions or terms of sale are performed electronically. Contrary to popular belief, E-Commerce is not just on the Web. In fact, Guttman et al (1998) describes the E-Commerce was alive and well in business to business transactions before the Web back in the 70s via EDI (Electronic Data Interchange) through VANs (Value-Added Networks). E-Commerce can be broken into four main categories: B2B, B2C, C2B, and C2C.

B2B(Business-to-Business)

Companies doing business with each other such as manufacturers selling to distributors and wholesalers selling to retailers. Pricing is based on quantity of order and is often negotiable.

B2C(Business-to-Consumer)

Businesses selling to the general public typically through catalogs utilizing shopping cart software. By dollar volume, B2B takes the prize, however B2C is really what the average Joe has in mind with regards to E-Commerce. Having a hard time finding a book? Need to purchase a custom, high-end computer system? With the advent E-Commerce, all things can be purchased literally in minutes without human interaction.

G2B (Government-to-Business), B2G (Business-to-Government), G2C (Government-to-Citizen), C2G (Citizen-to-Government) are other forms of E-Commerce that involve transactions with the government from procurement to filing taxes to business registrations to renewing licenses. There are other categories of E-Commerce out there, but they tend to be superfluous.

MAS FOR B2B E-COMMERCE

E-Commerce is the movement of business onto the World Wide Web (WWW). In MAS, an agent is a software program that acts flexibly on behalf of its owner to achieve particular objectives. MAS have also been used to represent

the clients and sellers / Courier_services as agents and the broker as a coordinator agent. In this model the job of the coordinator agent is to take the required items from the client agent and to find out the proper, best and trusty seller / Courier_service agents who can supply the items to satisfy the trust of client agent and constraints on the requirement of the client agent as well as on the seller / Courier_service agents in supply of the items. The client agent constraints are related with price, quality, quantity, brand, payment mode, trust of product, time, etc. The seller / Courier_service agent constraints are related with the price, quality, quantity, brand, trust, payment mode, payment type, address mode, etc. In MAS negotiation product brokering, cognitive parameter based selection, and monitoring have been incorporated by some of the researchers [7]. In this part, we define "Agent Based Model in MAS of B2B E-Commerce" in 2-stages: (1) need identification, (2) brokering (product brokering and merchant brokering). We first describe our models. The proposed model consists of two stages of CBB (Client Buying Behavior) model of B2B E-Commerce [8]. These stages are: need identification, seller / Courier_service selection and negotiation through broker agent.

In this model there are three types of agents with their different functionalities. Client Agent is the agent who needs to buy some items from another agent. Seller Agent / Courier_service Agent are the agent who sells items to the client. Broker Agent (Broker) is the agent who acts as a mediator between client and Courier_service [9]. He identifies all the need of the client agent and then selects the best seller / Courier_service agent for good product by evaluating the profile of the various agents and finally negotiates between client and Courier_service agent in B2B E-Commerce.

AGENT BASED SYSTEM IN B2B E-COMMERCE BY TASK ALLOCATION METHOD

Distributed artificial intelligence (DAI) [14] in E-Commerce is concerned with problem solving in which several agents interact in order to execute tasks of brokering process. Task execution in multi-agent environments may require cooperation among agents. Given a set of agents and a set of tasks which they have to satisfy for selling and buying the products, we consider situations where each task should be attached to a group of buyer / client agents and seller / Courier_service agents that will perform the task allocation in brokering process. Task allocation to groups of agents is necessary when tasks cannot be performed by a single agent. However it may also be beneficial when groups perform more efficiently with respect to the single agent's performance. Using the Distributed Problem Solving framework of agent-based simulation [10,11,12] we present solutions to the problem of task allocation in E-Commerce process among autonomous agents. We present task allocation algorithm which are appropriate for Distributed Problem Solving (DPS) cases where agents cooperate in order to increase the

overall outcome of the system and are concerned of buying and selling the products in MAS. We assume agents to be uniquely identifiable, an agent may help other agents in performing assigned tasks to buy or sell the product. We develop criteria for an agent to decide whether to help or not other agent when the latter requests for help. The task allocation decision criterion is such that it makes the overall system performance better. Finally using the simulated model of B2B E-Commerce, we tried to show the effect of agent's behaviour on the overall system throughput and deliver the products.

MODELING THE AGENT BASED TASK ALLOCATION FORMATION IN B2B E-COMMERCE

Agent Based Models in Multi Agent Systems for E-Commerce are computational models of heterogeneous agents and their interactions. The emergence of ABM in MAS has enabled scientists to conduct a large number of computational experiments testing theories which may have been easy to conduct empirical experiments for task allocation method to drop the products.

Problem Description

The process of brokering as often occurs in B2B E-Commerce involves a number of agents. A client agent looking for products may be supported by a broker agent that takes its client agents' queries and contacts other agents or looks at the web directly to find information on products with in the client agent's scope of interest. The model, in this chapter addresses three stages: need identification, Courier_service selection and negotiation;

where the Courier_service selection stage is the need identification, the client agent recognizes a need for some product through a profile. This profile may be appearing to broker agent in many different ways. Secondly the Courier_service selection involves the broker agent to determine what product is to be bought to satisfy this need and finding the Courier_service that offered software project module at desired price. The main techniques used by the brokers in this stage are (i) feature- based filtering i.e. software project module based on brand and quality (ii) constraint- based filtering i.e. the agent specify price range and date limit. The client agent contacts with broker agent, the broker agent determines the desire (wishes) of the client agent. The following terms need to be defining in the context of B2B E-Commerce of DPS framework for task allocation method (in Fig. 6.1):

Client Agent (Client): the agent who needs to buy some software project module from another agent. The client gains quality when client buys software project module.

Courier_service Agent (Courier_service): the agent who sells these software project modules to the client. It devotes processing time and other resources to sells these software project module.

Broker Agent (Broker): The agent, who acts as a mediator between client and Courier_services, is broker agent. The broker agent identifies the need of the client agent and then selects / chooses the best and trusty Courier_service agent by evaluating the profile item of the various Courier_service agents.

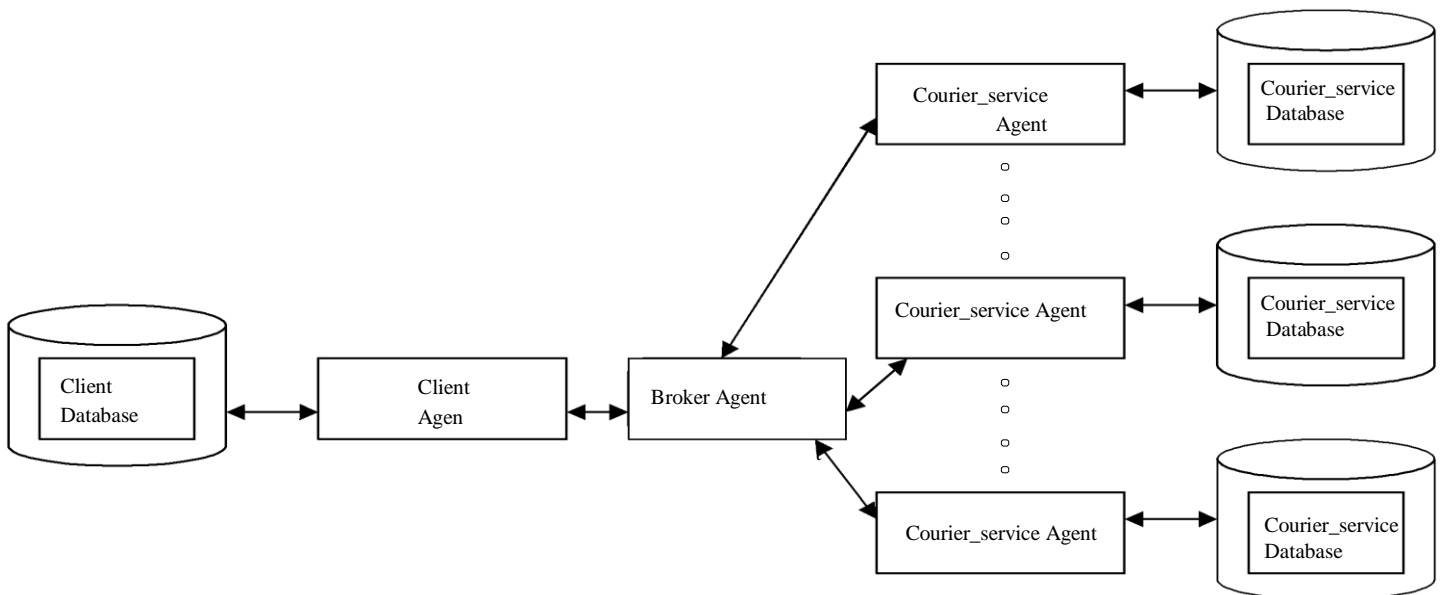


Fig. 1: Communication between Agents in B2B E-Commerce Mode

Algorithm

Most work in DPS framework involves starting with a team of negotiating agents and either providing a

methodology for allowing them to work together, or providing a mechanism for improving their group performance to business the product. In our model we

have taken the issue of *task allocation process*: the whole process of forming groups of agents in B2B E-Commerce. In our implementation of group formation in DPS we have taken a problem of delivering packets (products). The domain involves the set of agents that receive packets to deliver to specified addresses. The client agents receive their initial packets from any of five depots (from Courier_service agents) where the packets are placed initially. A client agent may ask for help to broker agent in performing assigned tasks. The criteria for an agent to decide whether to help or not is its current load and the additional distance that it will have to cover. Speed is dependent on the current load of packets (products). Here we can have overlapping of tasks as one agent may help in delivering more than one packets. We then evaluate the system performance in terms of number of packets delivered (if the agents are not dedicated) and the distance covered in delivering packets. The step by step methods / algorithm are described below shows the task allocation method of DPS framework for dropping the packet to client agents in B2B E-Commerce Multi Agent System.

Algorithmic Steps:

Step1: Setup the package depots, client agents, broker agents and Courier_service agents randomly in the environment.

Step 2: Initializations:

- Initialize agents by randomly setting the value of capacity / load of packets and travelled speed for each agent.
- Initialize packets by setting weight / load, source (one of the package depot), destination (patch) and the time-limit (t) randomly.
- Assign number of packets / 5 to each depot.

At every tick (t):

Step 3: Agents are at random location wandering with their initial speed for travelling the distance to drop the packet in B2B E-Commerce system.

Step 4: Agents check for a package depots. For checking the package depot, to pick one of the packets randomly from that depot.

Step 5: Loading decisions are made for each agent:

(i) Searching possible cooperators for uploading.

If current load of agent is less than or equal to 0.7 percent of its total capacity.

If other agent is at same location.

If current load of other agent is greater than or equal to 0.7 percent of its total capacity.

If the package to be offloaded lies in the vicinity of any of the packets that are carried by this agent.

(ii) Searching possible off loaders for each agent:

If current load of agent is more than or equal to 0.7 percent of its total capacity.

If other agent is at same location.

If current load of other agent is less than or equal to 0.3 percent of its total capacity

If the package to be offloaded lies in the vicinity of any of the packets the other agent is carrying.

Step 6: Final calculations for each agent:

- Calculate the number of packets that it delivered from Courier_service agents to client agents successfully.
- Calculate the total weights / load of packets.
- Calculate the total distance traveled for delivering the packets from Courier_service agents to client agents.
- Create a list containing that helped this agent in delivering the package to destination.

Step 7: Count the members that helped in delivering a particular package are put together in a task allocation method for B2B E-Commerce model.

Step 8: Reporting results for delivering the packets:

- Number of agents required for dropping packets vs Average time taken (t) to deliver the packets.
- Percentage loss of the overall system in case of both committed / dedicated agents and non-committed / non-dedicated agents.
- Individual agent's gain (in percentage) in case of committed / dedicated and non-committed / non-dedicated agents.
- Average distance covered per successful delivery of a package in case of committed and non-committed agents to each client agent.

Experimental Work and Implementation

We use the Netlogo programming language for simulating the models of packets delivery. Netlogo is particularly well suited for modeling complex systems developing over time. Modelers can give instructions to hundreds/thousands of agents all operating independently. This makes it possible to explore the connection between the micro-level behaviour of individuals and the macro-level patterns that emerge from the interaction of many individuals in E-Commerce.

Experimental Setup

Similar to the previous model using a torus of 32 x 32, we have randomly placed package depots and the Courier_services. Each packet has weight / load, source (depot), destination address (patch label) and the time limit (for delivering the packet). The agents form coordination for successfully delivering the packets. The model is run till all the deliveries are made (reach to their destination patch) or till 1000 ticks in case of non-dedicated Courier_services (as these agents may drop packets on their journey). We report the results by varying the number of Courier_services and changing the agent's behaviour from dedicated to non-dedicated. Snapshots of the NetLogo implementation of the agent task allocation method in DPS for B2B E-commerce to deliver the packets to client agents on a 32 X 32 grid, with number of packets = 25 (in Fig. 2 – Fig. 4).

RESULTS

The results obtained after several runs of simulation of the models to deliver the packets, number of agents and average time to drop the packets are shown below (in Table 1 and Fig. 5):

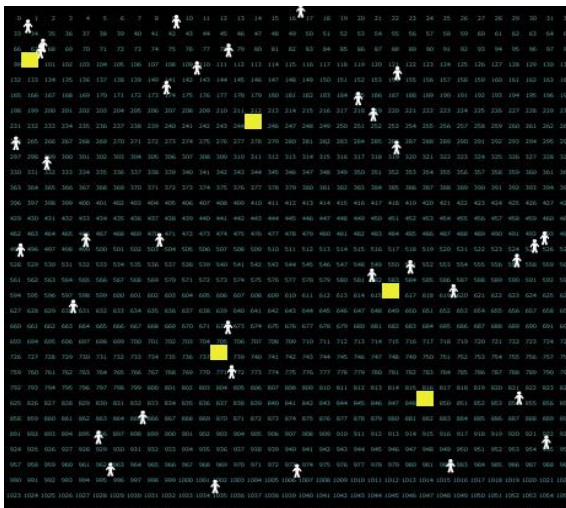


Fig. 2: Snapshot of the Initial Random Placement of the Depots of Courier_services, Broker Agents and the Client Agents.

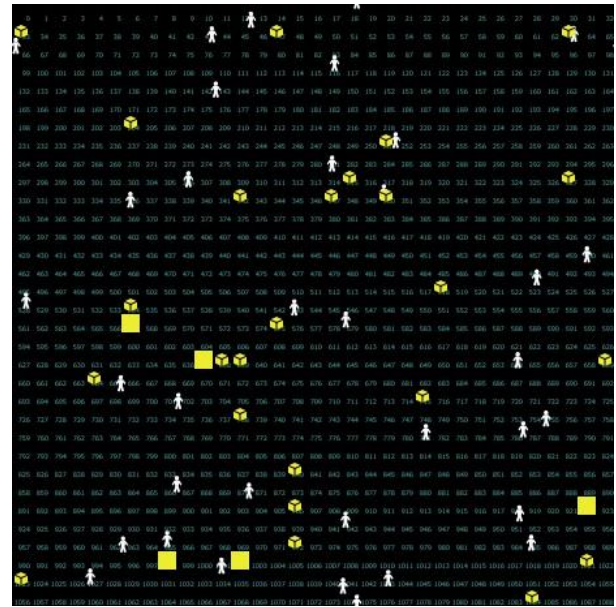


Fig. 3: Snapshot After the Successful Delivery of All the Packets (In Case of Dedicated Agents)

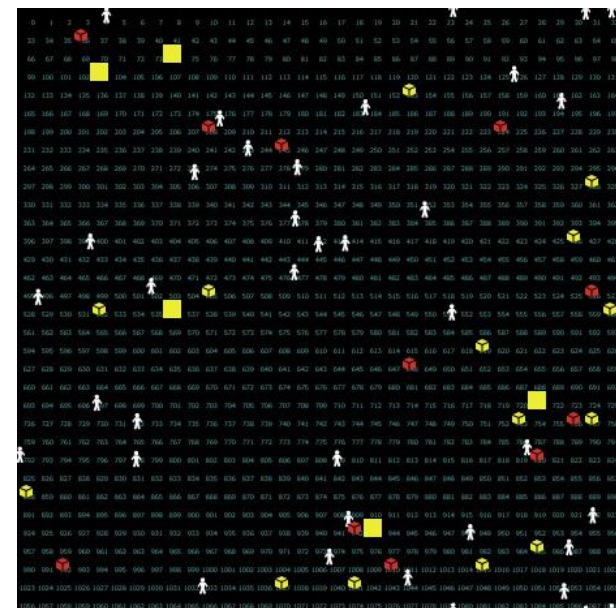


Fig. 4: Snapshot Taken After 1000 Ticks in Case of Selfish Agents who are not Dropped the Packets to Client Agents. The Packets that are Red Colored are Undelivered Packets

Table 1: Number of Agents for Dropping Packets vs Average Time Taken to Deliver the Packets

Number of Agents	Run 1	Run2	Run 3	Run4	Run 5	Average Time Taken
16	471	796	1000	573	646	697.20
25	513	288	447	414	299	392.20
30	341	312	546	512	214	385.00
40	262	246	189	333	270	260.00
50	220	149	211	283	230	218.60
70	178	146	130	149	132	147.00
100	117	145	187	147	133	145.80

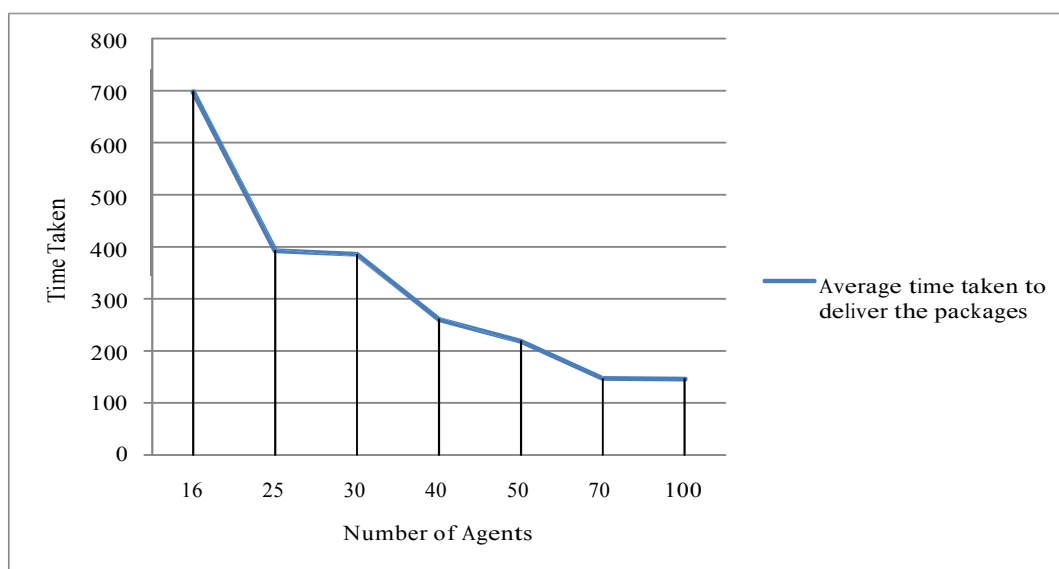


Fig. 5: Graph for Average Time Taken in Delivering Packages (In between Number of Agent's Required and Average Time Taken to Drop the Packets to Client Agents by Using Courier_service Agents)

The above graph seems to be in agreement with the cooperative nature of the agents in this model. As the number of agents was increased the total time taken in successfully delivering packets decreased sharply. With a higher number of agent population, Courier_service agents had more opportunities for selecting cooperative partners/broker agents to communicate with client agents. Now the percentage loss of the overall system of B2B E-Commerce model in case of dedicated and non-dedicated agents for delivering the packets to client agents (in Table 2 and Fig. 6):

Table 2: Loss Percentage of the System between Dedicated Agents and Non- Dedicated Agents

Number of Agents	In case of Dedicated Agents (in percentage)	In case of Non-Dedicated Agents (in percentage)
16	0.42	2.39
25	0.04	5.42
50	0.18	1.70
70	1.26	1.43
100	0.24	1.29

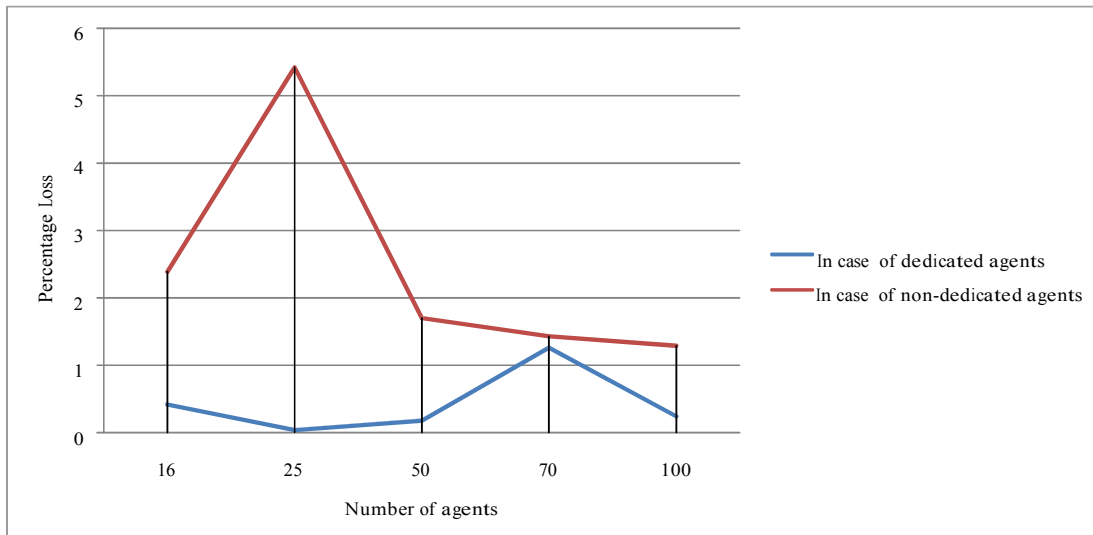


Fig. 6: Graph for Loss Percentage of the System between Dedicated and Non- Dedicated Agents

On analyzing the system performance based on the observations obtained after simulation runs, we observed that in case of non-dedicated agents (selfish agents) loss percentage was higher than that of dedicated agents (active and working agents). Selfish agents drop packets if they foresee any penalty. Because of these dropped packets, the overall system performance is degraded.

Individual agent’s gains (in percentage) in case of committed / dedicated and non- committed agents for the model of B2B E-Commerce model are shown below (in Table 3 and Fig. 7):

Table 3: Agent’s Gain in both Cases Committed / Dedicated and Non-Committed / Non-Dedicated Agents

Number of Agents	Individual agent’s gain (in percentage) in case of Dedicated Agents	Individual agent’s gain (in percentage) in case of Non -Dedicated Agents
15	6.39	6.66
25	3.90	4.00
45	2.17	2.22
55	1.78	1.81
100	1.00	1.00

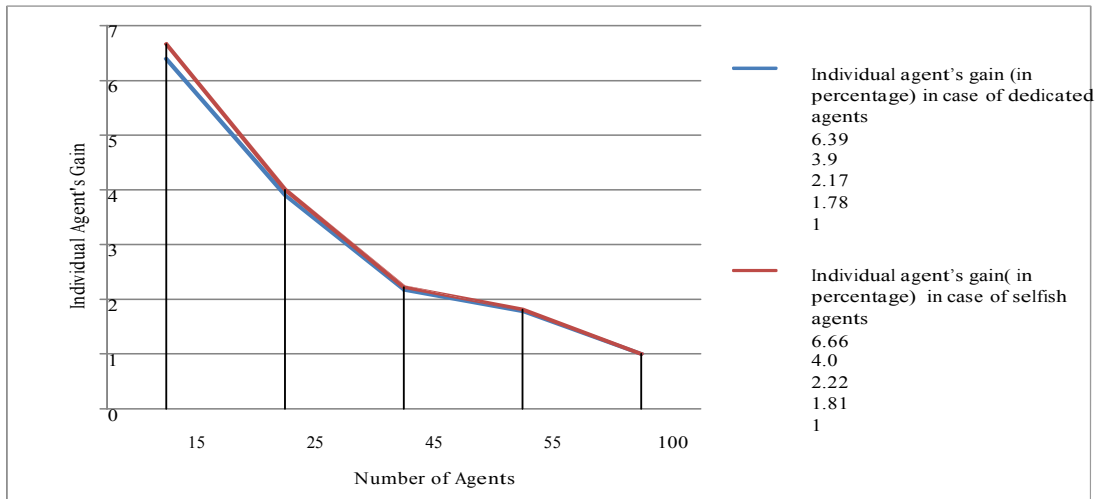


fig. 7: Graph for Individual Agent's Gain in between of Dedicated and Non- Dedicated Agents

Selfish agents (non-dedicated agents) try to improve their weights regardless of the system performance. The above graph to business the products also validate this behaviour as the personal gain in selfish agents was found to be higher than the dedicated agents. The penalty charged on the selfish agents after dropping packets was compensated by the gain earned because of the early delivery of other packets that it had to deliver. Hence these agents manage to earn more profit despite of the fact that they delivered lesser packets to their destination. Average distance covered per successful delivery of a package in case of both dedicated agents and non-dedicated agents are shown below (in Table 4 and Fig. 8):

Table 4: Average Distance Covered for Successful Delivery of Packets to Each Client Agents

Number of Agents	Average Distance covered per successful delivery of a package in case of Non - Dedicated Agents	Average Distance covered per successful delivery of a package in case of Dedicated Agents
15	14.05	11.70
25	17.50	12.00
45	14.61	13.32
55	15.61	13.64
100	15.50	13.00



Fig.8 : Graph for Average Distance Covered for Successful Delivery of Packets to

The above graph for average distance covered per successful delivery of packets shows that selfish agents (non-dedicated agents) had to cover a longer distance per package in comparison to dedicated agents. This increase in distance traveled per package also results in degradation of net system performance for B2B E-Commerce model. These observations are found because selfish agents were not able to draw benefit of the vicinity condition applied in the model.

DISCUSSION

In this model we have tried to analyze the cooperative behaviour in absence of any central authority. It clearly indicates that on increasing the number of cooperative entities the performance of the system improves. Also on varying the behaviour of agents interesting results in terms of personal gain and throughput are observed to business the products in E-Commerce. Though non-dedicated agents seem to draw benefits in terms of monetary unit but they are covering more distance per package delivery the products to destination agents. Hence it results in overall degradation in system performance of model. We look forward to work in the direction of comparing the performance of the system when the agents are cooperative with the case of agents working individually (not seeking help while delivering the package from other agents). The above experimental work can be applied in the "Courier Service for dropping the packets". The "Courier Service" is the best example to cover the

minimum distance from Courier_service agents to client agents for using to drop the packets, those agents who are covered minimum distance to travel, take minimum time to deliver the product, fewer prices offered for service, and fast delivery to drop the packet that agent's Courier Service will be preferred by client agents.

CONCLUSION

We have shown the application of this method for the purchase domain in agent's coordination and cooperative system. The selection of Courier_service agent is based upon his cognitive, social and trust characteristics. The client agent has set of requirements of items for which it needs some best and trusty Courier_service agents. To perform this, the Courier_service agent can choose from several alternatives that produce different qualities, best offer and price, and consume different resources. This context requires a negotiation that leads to a satisfying solution with increasing combined utility. We first examined a search for the index of negotiation of the Courier_service as a mechanism to find a compromise between the histories of different Courier_service agents. This mechanism helps to evaluate a good solution to fulfill all the requirements of client agent and the best Courier_service agent is selected by client agent for doing business in B2B E-Commerce System.

REFERENCES

- [1] Brazier, F., Dunin-Keplicz, B., Treur, J. and Verbrugge, R., 1997. *Beliefs, Intentions and Desire*. John Wiley.
- [2] Brooks, C.H. and Durfee, E.H., 2002. Congregating and market formation. Proceedings of the First International Joint Conference on Autonomous Agents and Multi Agent Systems, pp. 1-100.
- [3] Jennings, N.R., 1993. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, Vol. 8, Issue 3.
- [4] Jennings, N.R., 2001. An agent-based approach for building complex software systems. *Journal of Comm. ACM*, Vol. 44, Issue 4, pp. 35-41.
- [5] Jennings, N.R., Faratin, P. and Sierra, C., 2002. Using similarity criteria to make trade-offs in automated negotiations. *Journal of Artificial Intelligence*, Vol. 142, Issue 2, pp. 205-237.
- [6] Jennings, N.R., Faratin, P., Johnson, M.J., Norman, T.J., O'Brien, P. and Wiegand, M.E., 1996. Agent-based business process management. *International Journal of Cooperative Information Systems*, Vol. 3, Issue 1.
- [7] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C. and Wooldridge, M., 2001. Automated negotiation: Prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, Vol. 10, Issue 2, pp. 199-215.
- [8] Jennings, N.R., He, M. and Leung, H., 2003. On agent-mediated electronic commerce. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, Issue 4, pp. 985-1003.
- [9] Jung, J. and Jo, G., 2000. Brokerage between buyer and seller agents using constraint satisfaction problem models in e-commerce. *Journal of Decision Support Systems*, Vol. 28, pp. 293-304.
- [10] Smith, R., 1980. *The Contract Net Protocol: High Level Communication and Control in Distributed Problem Solver*. New York, IEEE Transactions on Computers.
- [11] Smith, R.G., 1977. The Contract Net: a formalism for the control of distributed problem solving. Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, USA.
- [12] Sousa, P. and Ramos, C., 1999. A distributed architecture and negotiation protocol for scheduling in manufacturing systems. *Journals of Computers in Industry*, Vol. 382, Issue 2.
- [13] Subramani, M. and Walden, E., 2000. Economic returns to firms from business-to-business electronic commerce initiatives: An empirical examination. Proceedings 21st International Conference Information Systems, pp. 229- 241.
- [14] Weiss, G., 2001. *Multi-Agent Systems – A Modern Approach to Distributed Artificial Intelligence*. MIT Press, pp. 83-129, 242, 356, 360.
- [15] Wilensky, U., 1999. *NetLogo*. Available on the Web at: <http://ccl.northwestern.edu/netlogo>, Centre for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.
- [16] Wilkes, J., 2008. *Utility Functions, Prices, and Negotiation*. Technical report HP Laboratories, HPL-2008-81, Available at: www.Hpl.hp.com/techreports/2008/HPL-2008-81.pdf.