Applying a new Communication Technique among Agents in Hospital Management System

Eng. Ahmed Moustafa Elmahalawy,Ph.D
Computer Science and Engineering Department,
Faculty of Electronics Engineering, Minufiya University
Menouf, 32952, El-Minufiya, Egypt

ABSTRACT

Nowadays, Multi Agent System (MAS) play a central role in the field of distributed artificial intelligence. MAS' applications go from electronic commerce, aircraft maintenance and business process management to e mail filtering. One of the important features in Multi Agent System is the communication among agent. The communication methods enable cooperation and coordination of agents in the system. The standard and important one is the Contract Net Protocol. Also, Round Contract method is created to overcome the limitations of Contract Net Protocol. This paper proposed a new communication technique among agents in MAS. This technique combines the advantages of both previous methods and avoids their disadvantages. All these techniques are applied To Hospital Management System that uses Multi Agents Systems. There is a comparison among these methods according to their performance.

General Terms

Distributed Artificial Intelligence: Coherence and coordination, intelligent agent, Multi agent system.

Keywords

Communication protocol, Contract Net protocol, Hospital Management System, Multi Agent System, Round Contract method.

1. INTRODUCTION

Recently, Multi Agent System (MAS) has become one of the dominating topics of research in Artificial Intelligence (AI). The MAS is a collection of agents co-operating with each other in order to fulfill common and individual. In MAS, different agents often have different roles and individual goals [10]. This interaction requires that they are provided with communicative capabilities. In a distributed environment agents managing applications and sources need to coordinate and cooperate with each other in order to achieve a goal, this can be done using one of the communication methods. It is helpful to find a method for communicating the agents to achieve their goals.

Communication is an important, basic concept in Distributed Artificial Intelligent, because it is the process of interaction, via some sorts of communication that makes it possible for several agents to combine their efforts towards solving an overall problem. The main reason for communication is to resolve conflicts between agents.

Contracts are a powerful communication mechanism in distributed systems. The Contract Net Protocol (CNP) has been

applied since about 1980 by Smith and Davis. The CNP has been developed to specify problem-solving communication and control for nodes in a distributed problem solver. Task distribution is affected by a negotiation process, a discussion carried on between nodes with tasks to be executed and nodes that may be able to execute those tasks [9].

Hospitals are service providers with the primary aim to improve the health state of their patients, where the treatment of the patients is the main value-adding-process in hospitals. Hospitals consist of several autonomous, administratively distinct wards and ancillary units. During hospitalization, the patients reside at the wards and visit the ancillary units for treatments according to their individual disease [12].

Hospital Management System (HMS) is a process of implementing all the activities of the hospital in a computerized automated way to fasten the performance. This system gives the procedural approach how a patient gets treatment, details about date of treatment and finally depending on different criteria like room allocated, lab reports, treatment and medicine taken…..etc.

The remainder of this paper is organized as follows: section 2 states overview of communication and different techniques for it. Section 3 illustrates the related work for Hospital Management system. In section 4, the proposed communication technique is suggested and the architecture for agents and system is presented. Section 5 shows some results of different communication protocol and the suggested one. Finally a conclusion is briefly described in section 6.

2. OVERVIEW OF COMMUNICATION PROTOCOLS

Since agents act in an environment that contains other agents and various services, architecture is needed to standardize communication between agents and access to common services. Agents can communicate by sending messages. Messages can be addressed to individual agents, to groups of agents, or to all agents of a particular class. Agent system communication architecture should provide support for: agent to agent communication, communication across a network, and security mechanisms [2].

One of the most important features of an Agent and Multi Agents System is the ability of the communication. There are many methods for communication among agents. In this section, only two of them are discussed as:

2.1 Contract Net Protocol (CNP)

The famous and standard protocol for communication among agent is called Contract Net Protocol (CNP). In the CNP, the network is assumed to consist of loosely coupled asynchronous agents; each agent can communicate with every other agent by sending messages. In the role of contractor, an agent can decompose a task into subtasks, send requests for bids on each specific subtask to all the other agents select the most appropriate bid and allocate the task to that subcontractor. The CNP is based on some assumptions. The announcement request message is not cancelled or changed; the contractor or subcontractor does not die; counter proposing isn't allowed, and an agent can decide whether or not to bid to an announcement [11].

The Foundation for Intelligent Physical Agents (FIPA) Contract Net Interaction Protocol (CNIP) is a minor modification of the original CNIP pattern1 in that it adds rejection and confirmation communicative acts. In the CNIP, one agent (the Initiator) takes the role of manager which wishes to have some task performed by one or more other agents (the Participants) and further wishes to optimize a function that characterizes the task. This characteristic is commonly expressed as the price, in some domain specific way, but could also be soonest time to completion, fair distribution of tasks, etc. For a given task, any number of the Participants may respond with a proposal; the rest must refuse. Negotiations then continue with the Participants that proposed. Figure 1 represents the FIPA Contract Net Protocol flow [3, 4].
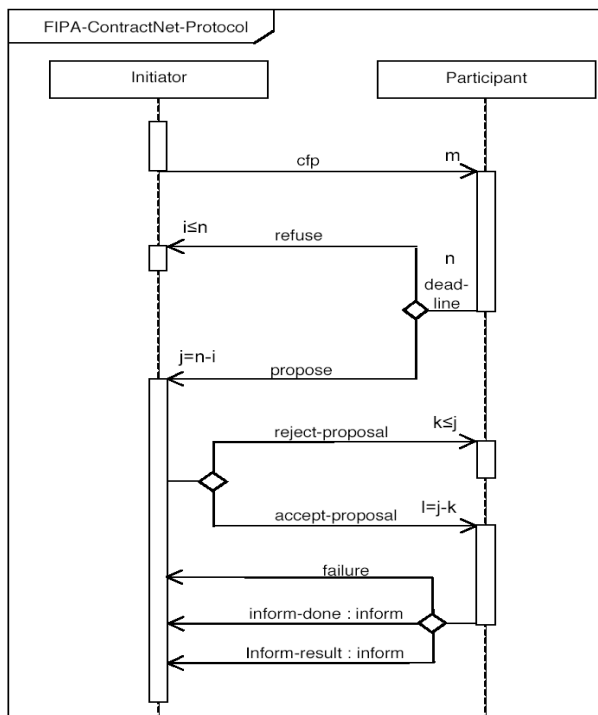

Fig 1: FIPA Contract Net Interaction Protocol

2.2 Round Contract (RCont) Method

Round Contract (RCont) is a developed method for communication among agents in MAS. It is based on the Contract Net Protocol (CNP) and the Acquaintance Model [2].

Figure 2 is the proposed system to explain my new method for communication among agent, RCont method. This system contains 5 general agents {A1, A2,…, A5}. These agents are called normal agent and contain the information for each part of

the application that MAS is used in it. The communication path between two agents has a weight where Wnm = Wmn. Also, there is an agent that is called Round contract Agent (RContA). Figure 3 shows the internal Architecture of the Round Contract Agent (RContA). The flowchart of RCont method is presented in Figure 4.
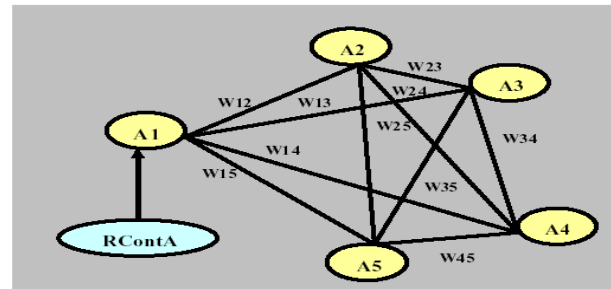

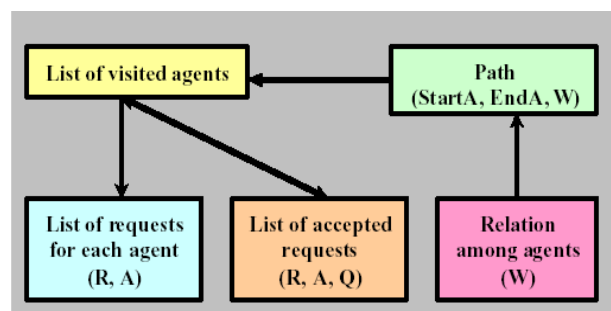**Fig 2: The proposed MAS for Round Contract method**

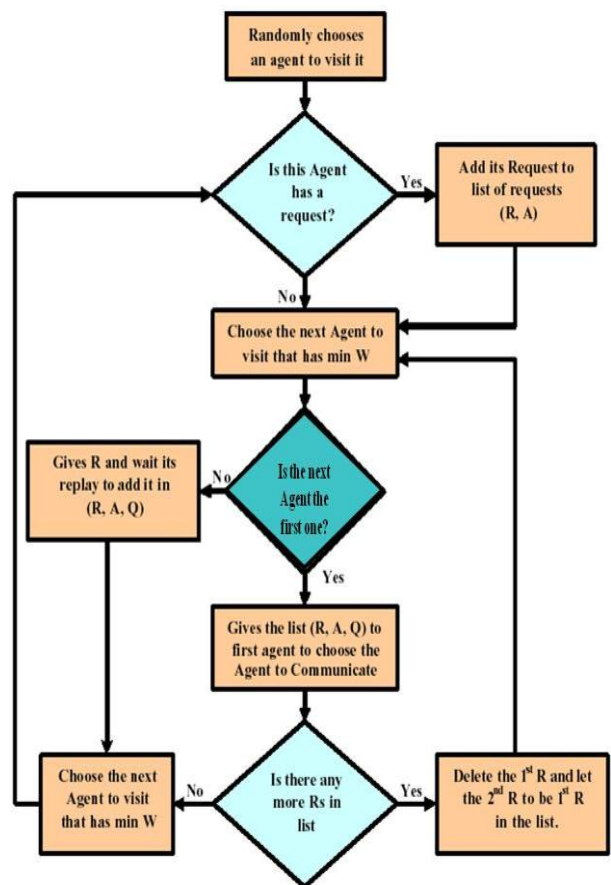**Fig 3: The Internal Structure of Round Contract Agent**

Fig 4: The Flowchart of my New Round Contract Method

3. THE RELATED WORK

Many researchers have used software intelligent agents in various applications that automatically perform the task on behalf of users as follows. One of these applications is Hospital Management System (HMS). In this section, there is an overview about some of these previous works as follows:

1- The architecture of the hospital system is presented with the help of the Jade platform. It gives the idea about different agents used in hospital and how the communication occurred between them and how to manage different agents. Multi Agent System (MAS) provides an efficient way for communicating agents and it is decentralized. Prototype model is developed for the study purpose and shows the impact of the agent's platform and how it is used to solve the complex problems arise into the hospital domain. The work discusses prototype model of Multi Agents in hospital environment and its impact. It is an attempt to make the system more flexible by using the MAS paradigm. It provides the more stability in the process [8].

2- In this work, intelligent agents are initially installed at user's computer at the development time of the HMS. These intelligent agents sense and gather user's new requirements automatically. In this paper patient agent, doctor agent, nurse agent and environment agent are designed to collect respective user requirements for HMS and send the collected requirement report to developer for enhancement of HMS. These agents interact separately with the user and automatically understand and gather user's requirements. After gathering the requirements, agents generate a report for user requirements collaboration and send that to the developer for the HMS enhancement [7].

3- This work describes the formation of agents of hospitals with intelligent and coordinative characteristics. They also discussed patient agent, doctor agent, nurse agent and environment agent working for their proposed hospital management system. These agents can perform hard coded task. This is a useful attempt at the innovation of hospital's modeling method [5].

4- The other development demonstrated how software agents incorporate learning, personalization, pro-activity, context-sensitivity and collaboration will lead to a new generation of medical applications that will streamline user interfaces and enable more sophisticated communication and problem-solving [6].

5- The proposed MAS based approach is used to model the drugs management processes and to solve the limited resources assignment problem through a combinatorial auction mechanisms [1].

4. PROPOSED TECHNIQUE AND SYSTEM'S ARCHITECTURE

The CNP and RCont method that describes in previous, each one has its own advantages and disadvantages. CNP is time consuming where it waits a time for replaying all agents in the system for its message and if the timeout and there is one agent doesn't replay so it resends the message and wait a particular time again. Although RCont overcomes this disadvantage by going round to each agent and for a particular time if the agent doesn't replay so it leaves it to next one and return to it in next round. This is good for a small number of agents in the system but for the large number of agents so the round takes a long time. If the there is an emergent task to do, it takes a long time.

The proposed Technique, called Central-Round Contract (CRCont), is developed. The CRCont Technique bases in two things as:

A) Central Agent (CAgent) has a copy of all jobs that each agent can do it and the weight for doing every job. It has a direct connection to each agent that it can use this connection in case of emergent requests. Also, it has a connection to Round Agent.

B) Round Agent (RAgent) is a mobile agent that goes round to each agent. It takes requests from agents and gives a job for them. It informs the Central Agent with any changes in agents' jobs.

Figure 5 presents the proposed system to explain the method for communication among agent, CRCont method with both CAgent and RAgent.
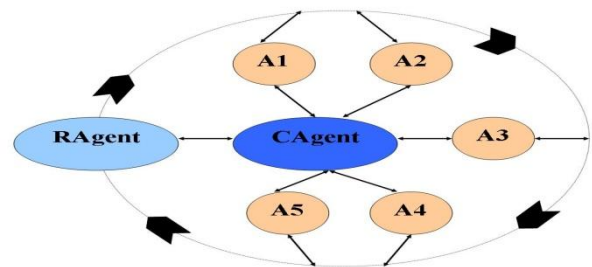

Fig 5: The proposed MAS for Central-Round Contract (CRCont) method

The important thing in this method is the priority of each job. This priority takes '0' for emergent and '1' for normal job.

The internal structure of both CAgent and RAgent are shown in figure 6 and figure7 respectively.
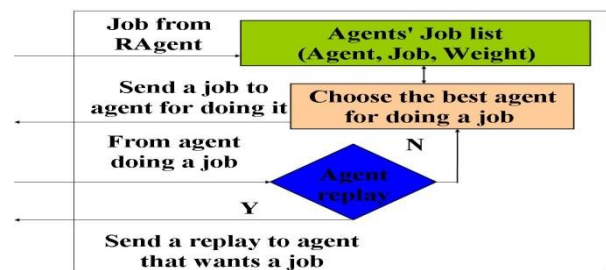

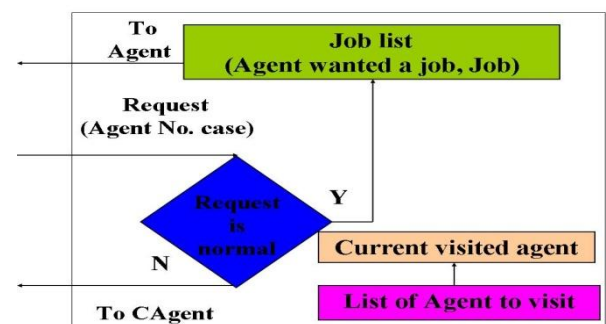**Fig 6: The Internal Structure of Central Agent (CAgent)**

Fig 7: The Internal Structure of Round Agent (RAgent)

The steps for the CRCont technique are as follows (as in figure 8)

1- The RAgent begins its round to all agents by randomly choosing the starting agent to visit it. When it reaches, it

asks this agent to do the job, according to the information from CAgent, and takes requests and any changes from this agent.

2- For each request with a normal priority, RAgent saves it and for a request with an emergent priority, it sends to CAgent.

3- If the agent can do a job, it replay to RAgent by a result, else its replay is no. In this case, a RAgent asks the next agent to do the job.

4- In case of emergent request, CAgent finds from its list the best agent to do the job, according to the weight of doing job, and sends it to this agent.

5- For a particular time, if the agent doesn't send a result for the job to CAgent. CAgent finds another agent to do the job, if there is another one to do it. CAgent sends the job again to the previous agent if there is only one agent can do this job.
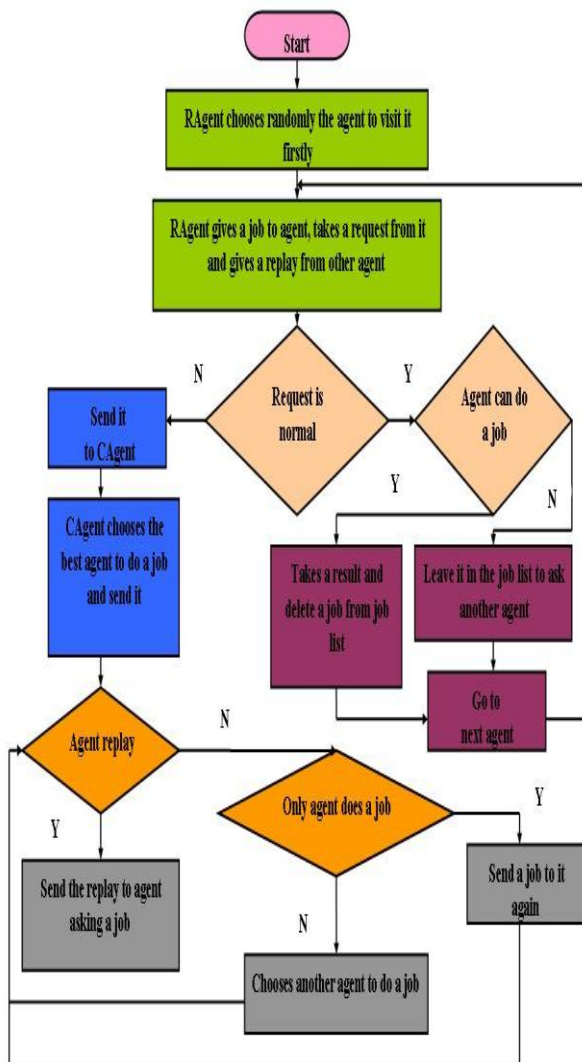

Fig 8: The Flowchart of the proposed Central Round Contract (CRCont) Technique

From these steps, it is seen that this method takes its power from using the principles of both serial and parallel control for doing the jobs as:

I) The serial control comes from RAgent that it goes round to all agent in serial manner.

II) The parallel control comes from CAgent that it connects directly to a particular agent and RAgent.

The Hospital has many departments as: Outpatient Clinics, Pharmacy, Inpatient& Nursery Room, Operations Theater, Laboratory & X-Ray, Storeroom, and Reception. Also, it treads with people like patients and staff (doctors and nurseries). According to the proposed technique, the system's architecture is suggested as follows, figure 9:

1- Every department is represented by an agent in the system.

2- The Agent for reception is considered as CAgent in proposed technique.

3- CAgent creates the RAgent (Request Agent) which is responsible for communication among agents in system.

The Department Agent (DAgent) is created for each department in the hospital. The hospital has eight departments as indicated before. Each DAgent has the following components, figure10:

1- Data Base contains all the data related to the department.

2- Classification and Clustering module. It responsible for organization of data in data base using a cluster and classification algorithms.

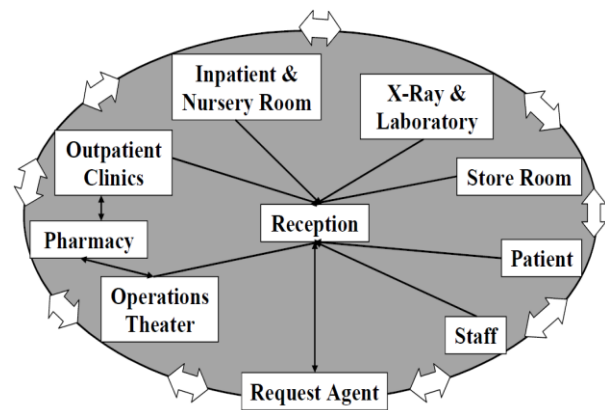3- Decision Module prepares the request and data to be sent for reception and then Request Agent.

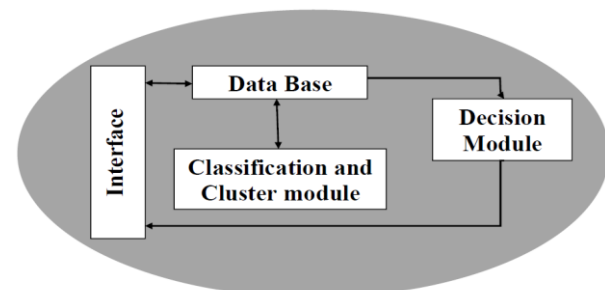**Fig 9: The Structure of Multi Agent System for Hospital Management**

Fig 10: The Internal Structure of Department's Agent (DAgent)

The Request Agent (RAgent) is created by reception to perform some requests from other departments of hospital. It has the following components, figure 11:

1- Request Information has all information about the requests as (the department wants the request, the department can

perform the request, the needed data to perform the request and the priority of request (urgent or normal))

2- Life Cycle has the ordering of agents that RMA must visit them in its trip. This ordering may change according the priority of request when there is urgent request.

3- Data Store has the data that transfer among departments and the results from processor.

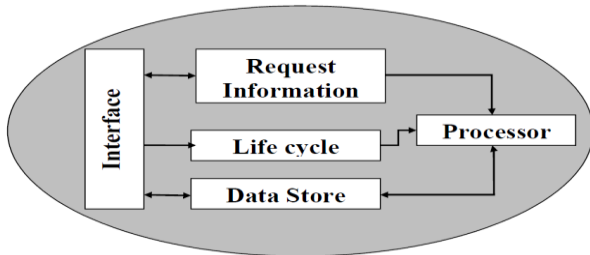4- Processor performs the requests when the RMA reaches to the dedicated Department's agent.

Fig 11: The Internal Structure of Request Agent (RAgent)

5. EVALUATION

Many experiments are presented using Hospital Management System to compare the performance of the proposed method (CRCont) with the performance of both CNP and RCont methods. The performance is evaluated according to the number of requested from agents in the system verses time for detecting the agent for doing the request.

In this case, the number of requests from agents in the system is varying from 1 to 100. Then, the time that the system takes to detect which agent can perform the request is calculated. The time for each method can be calculated as follows and is shown in figure 12.

1- For CNP, the time is calculated according to the replay of the last agent in the system. Time = time of the last agent's replaying.

2- For RCont, the time is calculated according to the visit of all agents in the system. Time= time of visiting the last agent in the system.

3- For CRCont, the time is calculated according to the search in the list of agent's job. Time = the time of searching in the jobs' list.

From figure 12, it is dedicated that the CRCont has the best performance rather than the other methods especially with large number of request in system. The CRCont can detect the agents that can do all requests in short time. But for RCont, the time is approximately constant because it depends on visiting all agents in system. In CNP, it has the worst time because it must wait the replaying of last agent in the system to know which agent can do the request. For large number of requests, this will be very bad.

6. CONCLUSION AND FUTURE WORK

The CRCont has a better performance than both CNP and RCont. It bases on serial (as in RAgent) and parallel (as in CAgent) concepts in its operation. Also, CRCont can perform both emergent and normal jobs; this is an advantage of it over other methods. It can deal with large number of requests in the system with suitable time. RCont has a good performance near to CRCont. It is consider that CRCont is RCont with facility of performing emergent jobs. CNP is the oldest method of communication and is considered as the basic for almost

communication method. It can't distinguish between emergent and normal job. Both of them are preformed as the same. It is not suitable for large number of requests in system because it takes a long time.

The future work will focus on two branches. The first one is to evaluate the performance of previous methods according to number of agents in MAS. The second is to apply the proposed method (CRCont) to many different real life applications to see its power.

7. REFERENCES

[1] Baffo, I. Stecca, G. and Kaihara, T. 2010. A multi agent system approach for hospital's drugs management using combinatorial auctions. In: IEEE Int. Conference on Industrial Informatics - INDIN, PP. 945-949.

[2] Elmahalawy, A. 2010. Multi Agent Systems for Artificial Life Domain. In: Lambert Academic Publishing AG & Co.KG, Germany. ISBN: 978-3-8383-4590-1.

[3] FIPA Communicative Act Library Specification. 2001. FIPA Contract Net Interaction Protocol Specification. URL:< http://www.fipa.org/specs/fipa00029/XC00029F.html>.[Cit. 2008-7-27]

[4] FIPA Communicative Act Library Specification. 2002. FIPA Contract Net Interaction Protocol Specification. URL:< http://www.fipa.org/specs/fipa00029/SC00029H.html>.[Cit. 2008-7-27]

[5] Hongqiao, Y. Xihua, L. Fei, W. and Weizi, L. 2009. Multi-agent based modeling and simulation of complex System in Hospital. In Proc. Sixteenth Int. Conference on Industrial Engineering and Engineering Management, PP.1759-1763.

[6] Hood, C. and Ji, C. 1998. Intelligent agents for proactive fault detection. In Proc. Conference on Internet Computing, Vol.2, Iss.2, PP. 65-72.

[7] Kushwaha, N. Sahu, S. and Ahmed, P. 2012. Gathering Requirements for Hospital Management System Using Intelligent Agents. In: Int. Journal of Engineering and Innovative Technology (IJEIT), Vol.1, Iss.3. ISSN: 2277-3754

[8] Mistry, M. and Shah, D. 2011. Impact of Multi-Agents in Hospital Environment. In Proc. Int. Conference on Intelligent Systems and Data Processing (ICISD) 2011, published by International Journal of Computer Applications (IJCA32).

[9] Smith, R. 1980. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. In: IEEE Transactions on Computers, vol. C-29, no.12, pp.1104-1113.

[10] Wernstedt, F. 2005. Multi-Agent Systems for Distributed Control of District Heating Systems. Doctoral Dissertation: Blekinge Institute of Technology, Series No 2005:10, Kaserntryckeriet AB, Karlskrona.

[11] Xu, L. and Weigand, H. 2001. The Evolution of the Contract Net Protocol. In: Wang, X. et al (Eds.): WAIM 2001, LNCS 2118, pp. 257-264. Springer-Verlag Heidelberg. ISBN:978-3-540-42298-3

[12] Zöller, A. Braubach, L. Pokahr, A. Rothlauf, F. Paulussen, T. Lamersdorf, W. and Heinz, A. 2006. Evaluation of a Multi-Agent System for Hospital Patient Scheduling. In: Int. Transactions on Systems Science and Applications, Vol.1, Num. 4, PP. 375-380.

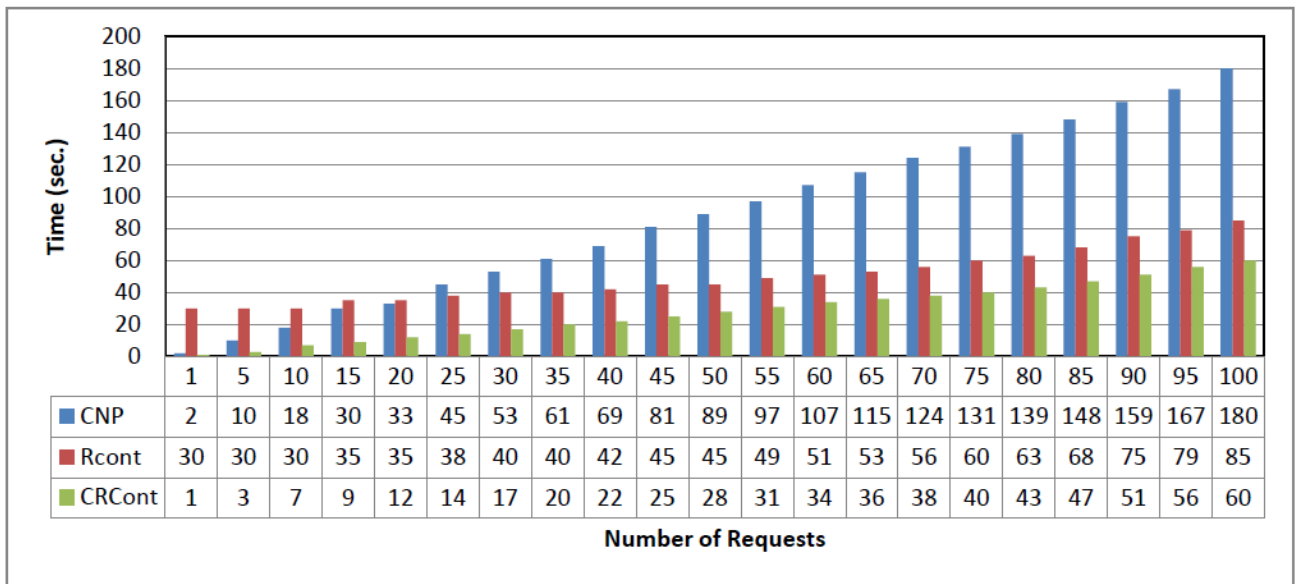|	1	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
CNP	2	10	18	30	33	45	53	61	69	81	89	97	107	115	124	131	139	148	159	167	180
Rcont	30	30	30	35	35	38	40	40	42	45	45	49	51	53	56	60	63	68	75	79	85
CRCont	1	3	7	9	12	14	17	20	22	25	28	31	34	36	38	40	43	47	51	56	60

Number of Requests

Fig 12: The comparison among the three methods according to number of requests