International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)

International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)

Multi-Agent Based Algorithm for Global Result Computation in Automated Searching

Gift Lee Jones.J, B.Tech., (M.E.)[1], R. Ramkumar M.E.,(Ph.D).[2]

M.E CSE-II year, St.Joseph's College of Engineering,Chennai, India.
Associate Professor, CSE department, St.Joseph's College of Engineering,Chennai, India.

giftleejones@gmail.com
hello_raam@yahoo.co.in

***Abstract*—The aim of this paper is to establish an automated search algorithm for global result identification which has exact patterns of data in warehouses. Automated MULBS, a defined version of the MULBS (MUltiple Local Bounded Search) algorithm is used for this computation. This process includes the usage of multiple agents that makes the search an automated one. These agents interact with each other forming various solution sets. The main agent controls all the agents and holds the authority to take major decisions. It also prevents the other agents from wasting time in unwanted execution. Usage of agents makes this defined algorithm an asynchronous and the most linear one. No human intervention is needed once the agents are started. The agents exchange information with neighboring ones to make an agreement resulting in the most complete and correct solution for search. The agents are also grouped together to take responsibility of various modules of the entire process dividing the search area into smaller pieces. The main agent refines the solutions from these different agents forming the most effective solution out of the list.**

***Keywords*— Agents, MULBS, computation, asynchronous, global result.**

I. INTRODUCTION

An agent is a piece of code that runs automatically. In recent years, the concept of Agents has attracted many fields with its asynchronous behaviour. They run independently without any human intervention avoiding wastage of time and man power. They are also known for making their automated decisions at the run time dynamically. Intelligent agents are used in many systems as they are found to be effective and handy. They not only communicate very easily and effectively but also coordinate and corporate within themselves in decision making. The agent architecture conceptualizes agents as being made of perception, action and reasoning components.

These allow them to take advantage of available services and facilities. The agents stop their execution only when the most correct global solution is obtained which makes the inference that using agents bring correctness to the solution. The complexity of the search technique is much linear due to the reduced number of messages, size of messages and throughput. The working together of multiple agents enhances productivity. The agents are made to agree with the meaning of concepts. So, various languages and channels are described for the purpose of effective agent communication. Examples of such communication protocols and languages are KQML or FIPA's Agent communication language ACL [6].

The cost of the whole process can be much reduced by using multiple cheaper agents and not one single agent. Also, the whole search space is divided into many blocks. These sub divided blocks are controlled by separate agent or a set of agents. They take care of their area of execution minimizing the entire search time and using the best time utilization. Their target is to complete their individual or shared goals called as local goals. All local goals lead to one single global goal which forms the global result. Even if the problem becomes so complex, the multi agent system is very handy in use. The usage of agents also works well and good in the distributed environments. The multi agent systems are generally loosely-coupled, problem solving entities that work together to find out effective and efficient solutions that are beyond individual capabilities or knowledge. They are generally used in systems where autonomous, heterogeneous systems need to intercommunicate and interoperate in order to achieve a common goal.

In addition to the effective solution, the whole system becomes so flexible and customizable that provides better management of the problem.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)
International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)

Cooperation is the process where several agents work together and draw on the broad collection of their knowledge and capabilities to achieve a common goal. Coordination is the process of achieving the state in which actions of agents fit in well within each other. Negotiation is a process by which a group of agents communicate with one another to try to come to a mutually acceptable agreement on some matter [5].

II. Objective

An effective searching method has to be formed by using the Agents which automate the entire system. The correct global result need to be computed out of the list of local solutions. The agents need to be controlled, used effectively and processed in parallel method for less time consumption. The time is efficiently managed by dividing the search area into smaller pieces.

III. Preliminaries

In this system in order to reduce the false positive, and true negative risk prediction all the possible breast cancer related proteins and their DNAs sequence are analyzed. Also improve the speed of the system different sequences are sending to different systems in the distributed parallel computing environment and parallel sequence analysis will be done. The Multi-Agent System (MAS) is one of the enabling technologies due to its scalability and efficiency in resource usage. In MAS, agents may vary in their capabilities: some agents have a single capability, while others have multiple capabilities. An agent group composed of multiple types of agents has far greater capability in complex environments than a monolithic group to different features of the environment to efficiently accomplish tasks available within a feature [2].

Usually, an agent controls the value of one variable and exchanges information with neighboring agents to reach an agreement about the variable values that minimize the constrained costs. Usage of agents makes the system to be asynchronous, correct, complete and linear. The experiments show that MULBS finds the best solution very quickly for loosely connected constrained graphs but does not converge to the optimal solution in all cases for over constrained graphs. Experiments show that MULBS runtime is much better than that of other state of the art algorithms for the distributed meeting scheduling domain.

MULBS processing bottleneck is the local search procedure because when a graph node has a large number of neighbors the number of partial solutions available is also large. MULBS explores local search in order to reduce communication among variables [1]. To achieve optimal system performance, we should consider not only agent capability, but also agent communication costs. When an MAS wants to execute a task, the first step is to allocate the task to some agents (i.e., task allocation). The main goal of task allocation is maximizing the overall performance of the system and completing the task as soon as possible [3].

MAS systems belong to a relatively new genre of systems to solve complex real life problems. A set of autonomous, proactive, intelligent computer programs, which are called agents, share the responsibility of solving the problem. Each agent has its own share of tasks. The agents react to the environment and collaborate with each other to achieve the stated objective. The agent-based model that is proposed here is related to the model of social learning since the agents base their decisions [4]. Deploying multiple agents can help in collecting information quickly and provide robustness to the mission through redundancy. These agents have to coordinate with each other to improve search performance. In principle, real-time algorithms for MAS can be implemented with agents communicating their states to a centralized controller that has sufficient computational capability to execute cooperative control algorithm based on information received from agents [7].

The MAS approach tends to be effective and convenient in the control of a network of microgrids since; in this case, all agents are independent decision makers (DMs). In addition, a MAS approach is robust and scalable, since it can be reconfigured through adding and/or removing agents without leading to a global failure in the system of microgrids. In general, each agent has the capability to interact with other agents [8]. Intelligent searching agents employ traditional Web search engine spider technology in new ways. An intelligent agent can also be autonomous as it can make judgments about the material's likely relevance. Agents learn from past experiences, and a user has the option of reviewing search results and rejecting any information sources that are either irrelevant or not useful. The agent stores this information in a user profile to use when performing a search, learning from its initial forays into the Web and returning a more tightly defined searching agenda if requested [9].

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)
International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)

One of the approaches, which has emerged recently as a result of convergence of many technologies within computer science, is an agent-based approach. Agent technology has become an important enabling tool allowing for smooth hybridization of different techniques. An agent is understood here as any autonomous piece of software that is designed to use intelligence to automatically carry out an assigned task, mainly retrieving and delivering information. Among many features of multi-agent systems, like autonomy of agents, lack of global system control, decentralization of data, possibility of parallelization of computation threads, etc. Agents, cooperatively solving particular problem, exchange pieces of information about states, models, entire sub-problems, solutions or other search space characteristics [10].

IV. System Design

This paper is designed and developed to improve the searching mechanism. This system is also used to automate the search with the help of intelligent agents for the correctness of the global result.

A. Proposed System

An effective searching algorithm called Automated MULBS which is a defined version of the existing MULBS algorithm is used. Searching problem is not as complicated as the DCOP problems. DCOP problems are NP complete problems where the searching problems are NP problems. Centralized main agent is introduced in the automated MULBS which controls and coordinates other agents. This avoids unwanted execution or transfer of messages between the agents. This is done to make sure that the linearity of the algorithm is improved. In the first phase, the local solutions are framed by each groups of agents and a solution set is formed which is sent to the main agent. In the refining phase of the MULBS algorithm, instead of propagating the solution from the leaves to the higher priority agents, the centralized main agent announces the global result to all the immediate sub agents. Only the sub agents who have a better solution than the announced global solution must reply to the main agent. This process eradicates unwanted confusions and makes the agents to have a better controlled communication.

Advantages of MAS include the ability to survive single-point-failures and decentralized data processing, which leads to efficient task distribution, eventually causing faster operation and decision-making process.

Introducing a centralized main agent reduces the message complexity drastically as it controls the execution and communication of the other agents. The major decision making is only in the hands of the main agent as it computes the best solution from the suggested solution list by the sub agents which reduces the arithmetic complexity in other agents. The refining phase is made much simpler and better by introducing the announcing the global solution method.

B. System Architecture

The proposed system contains a set of agents that act in separate groups that has their own set of goals to be satisfied. The total search area is subdivided into smaller areas for which a group of agents are made responsible for. They also work together for computing the local solution in their respective areas. Once the local solution is computed, it is send to the main agent. Thus the main agent will have various solution sets that are suggested by these sub agents. The main agent computes the most correct and specific solution of the list. Once the local solution is computed, it is send to the main agent. Thus the main agent will have various solution sets that are suggested by these sub agents. The main agent computes the most correct and specific solution of the list.

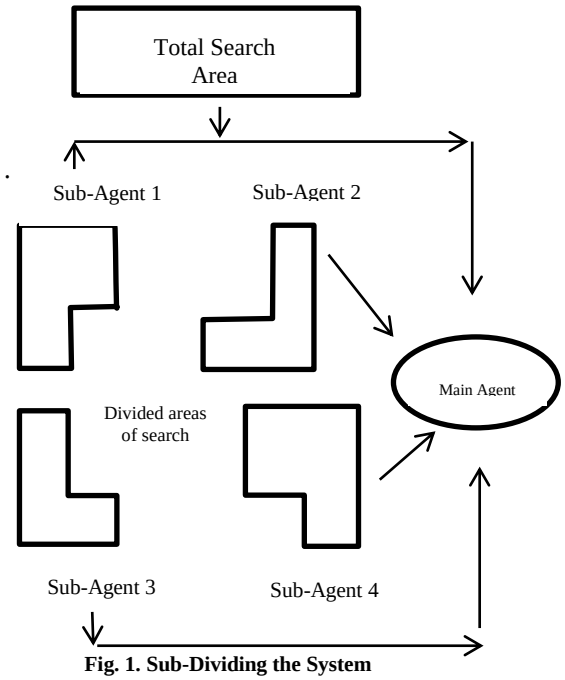
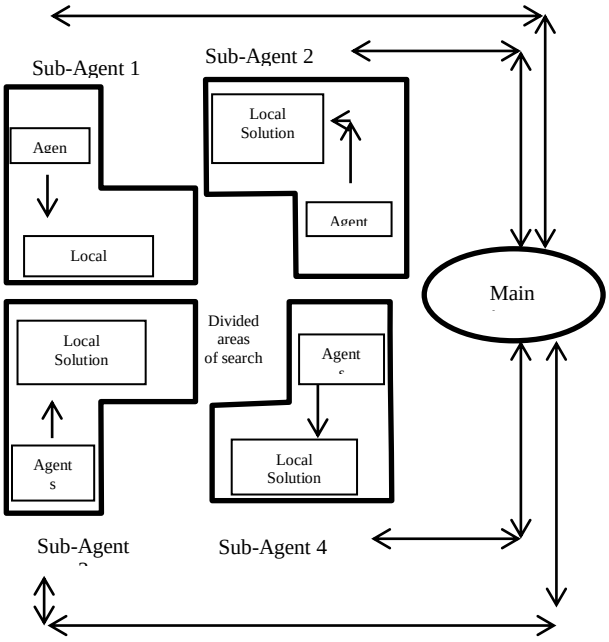


Fig. 1. Sub-Dividing the System

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)
International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)



Solution Set in Main Agent

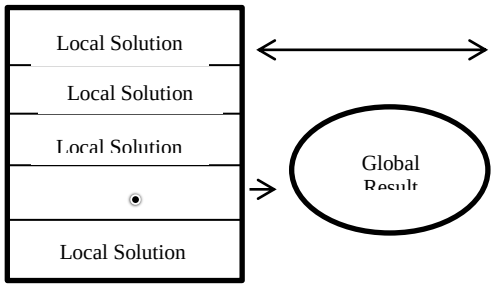


Fig. 2. Overall System Architecture

C. Algorithm

Input: The search pattern.
Output: The Global result.
Algorithm:
1: procedure Initialize()
2: start the main agent
3:main agent starts all other agents
4:started agents are brought to the background
5:all agents wait for the call from the main agent for their job.
6:end procedure
7:procedure getInput()
6: main agent starts the input agents
7: read the input search pattern
8: Transfer the control to the main agent along with the inputs
9:end procedure
10:procedure searchRefinePhase1()
11:main agent signals the local agents to start searching the corresponding areas
12:local agents refine the solution from the leaves and forward them to their respective parents
13:end procedure
14:procedure searchRefinePhase2()
15:the parent nodes check the best of the forwarded solution with their own solution.
16:if (parent has a better solution) then
17:continue;
18:end if
19:else
20:replace the parent's solution with the local solution
21:respective agents forward their solutions to the main agent
22:end procedure
23:procedure searchRefinePhase3()
24:main agent forms a solution list with the solutions from the parents
25:main agent chooses the best global solution
26:main agent announces the best solution to its immediate parents
27: parents reply only of they have a better solution
28:end procedure.

D. Phases Of The System

1. Getting the Inputs:

The input agents prompt the inputs to the user. The inputs can be of any type of pattern that the system needs to search from the warehouse. As the user enters the input, the input agents get the input and forward it to the main agent for further processing.

2. Searching the Local Areas:

The main agent triggers the sub agents which are responsible for the divided part of the whole search area. These sub agents have other agents under them which are initiated and the whole process begins. The sub agents wait in the background till the main agent sends them the input.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)
International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)

When the input is received by the sub agents, the agents that work under them search the local area and return the solution to the sub agents if any hit is found. If no pattern matching the input is found, the sub agents return the status as no result found to the main agent. If any hit is found, the sub agents return the hit as the local solution to the main agent.

3. Computing Global Solution:

The global solution which is also called as the global result is the target result of the entire system. Proper care must be taken as the process to compute the global result must be an effective one. The main agent forms a solution list which has the set of all the local solutions. The intelligence to implement the global result computation is implemented only in the main agent, which reduces the load on all agents.

4. Refining the Global Solution:

The demand for the correctness of the global result that is calculated is very high. So in order to obtain a highly correct global result, the refining process must be done. Even if the complexity of the whole calculation is increased considerably, the process is maintained for the correctness of the solution. Once the global result is calculated, there must be no other most possible result in the warehouse. If this is ensured, the correctness of the solution can be increased to the maximum. After the result is calculated as in the previous module, the solution is announced to all the sub agents by the main agent. These agents need not reply anything if they have a same or a worse solution. But, if they have a better solution, they can reply it to the main agent. The main agent then changes the global result announces this to other agents. This process continues till all the other sub agents fail to have a better solution.

V. CONCLUSION

An effective System with needed search technique is designed. The intelligent software agents that are generally difficult to maintain is effectively managed. The usage of single centralized main agent is done effectively and the whole process is automated for increased efficiency. The most correct Global result is obtained and is found to be correct by the refining process.

Such an effectively controlled distributed system can be framed in the future and can be worked on turning the globe on the power of agents. Future work can be done on dealing with the agents in the multi systems environment. Through this the effectiveness and efficient usage of resources can be defined. This problem can also be extended to the environments like grid and the agents can be used effectively for better search solutions.

REFERENCES

[1] Enembreck, Jean-PaulAndre Barth, 2011, "Distributed constraint optimization with MULBS: A case study on collaborative meeting scheduling", ELSEIVER, Journal of Network and Computer Applications, pp.1-12.

[2] Wang Lu, Wang Zhiliang, Hu Siquan, Liu Lei, 2013, "Ant Colony Optimization for Task Allocation in Multi-Agent Systems", IEEE, China Communications, pp.125-132.

[3] Khamis A, Elmogy A, Karray .F, 2011, "Complex Task Allocation in Mobile Surveillance Systems [J]", SPRINGER, Journal of Intelligent & Robotic System, No. 64(1), pp.33-55.

[4] Ritesh Kumar and Subir Bhattacharya, 2012, "Cooperative Search Using Agents for Cardinality Constrained Portfolio Selection Problem", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—Part C: Applications and Reviews, Vol. 42, No. 6, pp.1510-1518.

[5] Kwang Mong Sim, 2012, "Agent-Based Cloud computing", IEEE TRANSACTIONS ON SERVICES COMPUTING, Vol. 5, No. 4, pp.564-577.

[6] Sanskruti Patel, Tejas Thakkar and Sajja, 2012,"A knowledge Depiction in Digital Library by Integrating Multi-Agent System for Distributed Database Grid", IJGCMAS Vol.3, No.2.

[7] P. B. Sujit and Debasish Ghose, 2011, " Self Assessment-Based Decision Making for Multiagent Cooperative Search", IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, Vol. 8, No. 4, pp.705-719.

[8] Hanane Dagdougui and Roberto Sacile, 2013, "Decentralized Control of the Power Flows in a Network of Smart Microgrids Modeled as a Team of Cooperative Agents", IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, pp.1-10.

[9] Anita Ferreira, John Atkinson, 2005, "Intelligent Search Agents Using Web-Driven Natural-Language Explanatory Dialogs", IEEE Computer Society, pp.44-52.

[10] Dariusz Barbucha, 2012, "Agent-based guided local search", ELSEVIER, Expert Systems with Applications 39 12032–12045, No.39, pp. 12032–12045.