Multiple Ant Colonies Optimization for Load Balancing in Distributed Systems

Al-Dahoud Ali and Mohamed A. Belal
Faculty of Science and Information Technology
Al-Zaytoonah University
Jordan
{aldahoud, dr.belal}@alzaytoonah.edu.jo,
Tel:00962795323223

Abstract

Ant colony optimization (ACO) has proved its success as a meta-heuristic optimization in several network applications such as routing and load balancing. In this paper, a proposed ACO algorithm for load balancing in distributed systems will be presented. This algorithm is fully distributed in which information is dynamically updated at each ant movement. Multiple colonies paradigm will be adopted such that each node will send a colored colony throughout the network. Using colored ant colony helps in preventing ants of the same nest from following the same route, and hence enforcing them to be distributed all over the nodes in the network. Each ant acts like a mobile agent that carries newly updated load balancing information to the next visited node. Finally, the proposed algorithm will be compared with the standard work-stealing algorithm.

1. Introduction

Load balancing attempts to improve the performance of a distributed system by using the processing power of the entire system to smooth out periods of high congestion at individual nodes [10,11], this is done by transferring some of the workload of heavily loaded nodes to other nodes for processing. Decisions on how to balance loads among the nodes are either static [1, 2, 5, 7] or dynamic [3, 4, 6, 8, 9]. A static decision is independent of the current system state. Although static load balancing is simple and easy to analyze with queuing models, but its potential benefit is limited, since it does not adapt itself to time-varying system state [4]. On the other hand, a dynamic decision is depending on the system state at the time of the decision. When a dynamic load balancing is used, an over loaded node can transfer its jobs to other nodes using the information on the current system state. The dynamic policy is inherently more complex than any static policy because it requires that each node must to know the states of the other nodes. The load balancing algorithms are further divided into several clusters according to the amount of information required for them [6].

On the other hand, artificial swarm intelligence, in particular Ant Colony Optimization (ACO), is a relatively new computational and behavioral paradigm for solving optimization and combinatory problems; it is based on the principles that control the behavior of natural systems. In such a simulation model, many distributed agents evolve and interact with each other in order to reach a global goal, such as ant colonies and bird flocks. This approach emphasizes the distributed structure of the problem, direct or indirect interactions among relatively simple agents, and also among the agents and their environment.

The application of swarm intelligence to networks problems arises when a group of autonomous programs (agents) are working together. This is referred to be "*Ant Colony Optimization*" ACO or multi-agent system. Each individual or program or autonomous module can be represented as an agent, these multi-agents can be used for network applications such as finding the shortest path, routing, load balancing, and management, and so on.

Some related works to using ACO in load balancing is to the use of multi-agents system, two algorithms has been proposed by [21]: The first one is based on round trip routing agents that update the routing tables by backtracking their way after having reached the destination. The second one relies on forward agents that update the routing tables directly as they move toward their destination.

On the other hand, Salehi [22] presents an echo system of intelligent, autonomous and cooperative ants. The ants in this environment can procreate and also may commit suicide depending on existing conditions. A new concept called Ant level load balancing is presented for improving the performance of the mechanism.

Sim [23] has presented a *Multiple Ant Colony Optimization (MACO)* approach for load balancing in circuit-switched networks. MACO uses multiple ant colonies to search for alternatives to an optimal path. One of the impetuses of MACO is to optimize the performance of a congested network by routing calls via several alternatives paths to prevent possible congestion along an optimal path. In MACO, each group of mobile agents corresponds to a colony of ants, and the routing table of each group corresponds to a pheromone table of each colony [24]. By adopting the MACO approach, it may be possible to reduce the likelihood that all mobile agents establish connections using only the optimal path [24]. The advantage of using MACO in circuit-switched routing is that it is more likely to establish connections through multiple paths to help balance the load but does not increase the routing overhead [24].

Bulancea [25] used mobile agents to encapsulate tasks, in distributed memory message-passing computers. In order to increase system flexibility and to balance an arbitrary graph topology computing environment, in the proposed model, the agents have the possibility to explore, learn and share information about the load. The proposed model was compared with deterministic DASUD and evolutionary algorithms. The allocation on physical processors of the proposed agent algorithm was generally better than that given by the DASUD algorithm when the task's time requirement is increased.

In this paper, a fully-distributed, multiple colonies ACO algorithm is introduced. The load balancing strategy is time dependent and follows the natural dynamics of the pheromone in real life. At a specified time period, each node will act like a nest and sends number of ants (the number of ants depends on the loading status of each node; the overloaded and underloaded nodes will send more ants). Each ant will travel a tour (the tour length depends on the size of the system and the loading status of the node).

2. Ant Colony Optimization

Social insects, such as ants, show an intelligent group-behavior that characterizes the whole colony; examples of that emergent behavior include foraging and nest building. This collective behavior can be viewed as powerful problem-solving systems. Starting from simple interacting agents with rules of interaction among individuals and between individuals and the environment, the ant colony can provide intelligence far away from any individual capability. Properties associated with their group behavior, such as self-organization, flexibility and robustness, can be shown as characteristics that should exist in complex system for control, optimization, and problem solving techniques.

One of the problems that attract ethologists was to understand how almost blind ants could cooperate in order to find the shortest path to the food source. It was found that they communicate by placing pheromone trails, a chemical substance that attracts other ants, along with their movements towards the source of food [12]. An ant will prefer with high probability to follow the path having more pheromone, thus enforcing the trail with its own pheromone. This collective behavior is called *autocatalytic* behavior, in which it is defined to be a positive feedback process in which a process reinforces itself to enhance its performance, this feedback enforces the process towards a rapid convergence towards the final solution [12].

Generally, Ant Colony Optimization (ACO) is a general-purpose heuristic algorithm, which can be used to solve different combinatorial optimization problems [12]. In ACO, the search activities are distributed over artificial ants, which mimic the behavior of real ants. The advantages of that system are positive feed-back, distributed computation, and the use of a constructive greedy heuristic. Positive feed-back refers to the ability to rapid discovery of good solutions. The ACO is also a population-based approach in which parallization can easily be achieved [12].

The ants do not forage only for finding food, rather they forage in order to wander, search, return to home, attract to a target, trace pheromone, and carry food [13,14]. The pheromones also evaporate over time. As a consequence, the pheromone will become less detectable after a while, and the longer trails will be less attractive to other ants [15].

Ants can construct the shortest path from their nest to the source of food, through the use of pheromone trails. The ant leaves some quantities of pheromone on the ground while it walks. The next one will sense it, and based on a probability proportional to the amount of pheromone, it will choose its path [16].However, artificial ants have some major differences with real ones: First, ants have a memory. Second, ants are not completely blind, finaly, the time of artificial ants is discrete [17].

The collective activities of social insects are self-organizing, meaning that complex group behavior emerges from the simple interactions of individuals. The results of self-organization are of global nature, but come basically from local information and interactions [15]. The interaction within a society of insects can take one of the two forms: direct and indirect. Direct interactions can take the form of bodily contact, visual contact, and food exchange. Indirect interaction is also important, it can occur when agents exchange information through the environment in which they exist. Thus the storage of information occurs at the colony level as well as individual level. This cooperation through modification is called stigmergy [17].

In ACO, the ants are adaptive, i.e. if the environment changes, the ants will look for a better solution. The ACO is suitable to discrete optimization problems [16]. The main characteristics of AC system are positive feedback, distributed computation, and a constructive greedy heuristic [12].

The characteristics of the swarm intelligence model of ACO are the new added concepts of self-organization and stigmergy. In a distributed system such as Ant System, communication between agents is of the great importance. The form of communication is indirect. This communication can be viewed as a space deformation of the system in which ants reside [17]. The behavior of the whole ant colony is highly structured. Interactions are based on very simple flows of information [18].

The pheromone that each ant lays attracts the following ants so that they will likely search in the same region of the search space. In general, it is assumed that pheromone evaluation is done locally by the ants. Merkle [19] proposed to extend the local view of the ants by a look-forward strategy.

3. The proposed strategy

Our model is fully distributed, i.e. each node nodes behaves independently as well as each ant or agent, this mean that each node or ant is autonomous. Figure 1; represent the table attached to each node or ant.

Node ID	Loading Status	Pheromone

Table (1): The attached information to each node or ant.

In our model, each node contains a table that includes information about other nodes in the system. At the initial state, the table entries are Null.

In each ant tour, the ant will carry the updated information about all nodes that the ant has been passed through. Upon arrival of the ant at each node, the following actions will be done:

1- If the node does not have the information contained in the ant table, these information will be passed to the node table as it is.

2- If the node contains information that does not exist in the ant's table, the ant table will be updated.

3- If both of them share the same information, the newly updated one will replace the other.

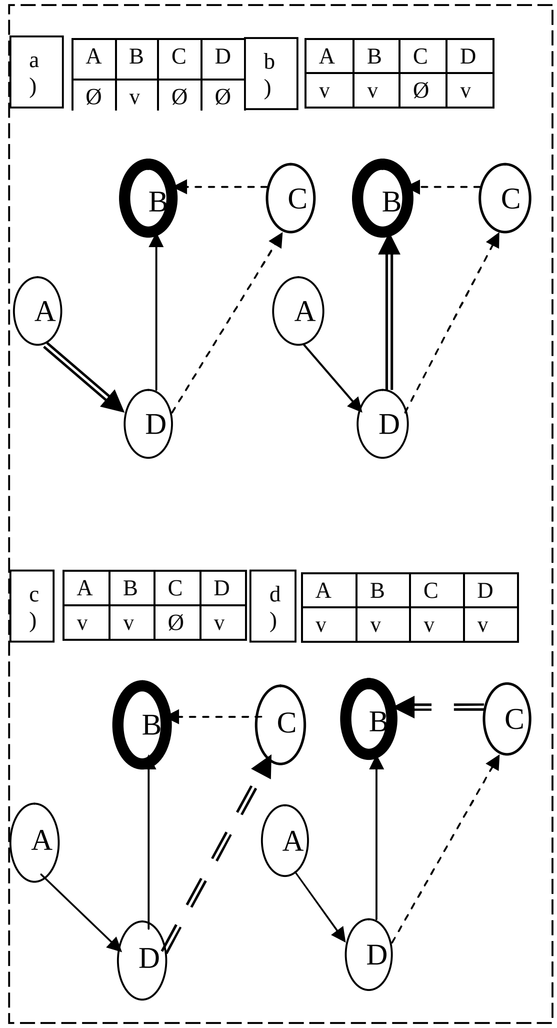

Figure (1): Example of the operations of the model
a) An ant moves from A to D b) The ant moves from D to B
c) Another ant moves from D to C d) The ant moves from C to B

4. Simulations and results

The proposed strategy was simulated and tested, the number of nodes in the distributed system was assumed to be 30, an ant is assumed to travel from one node to another in 1 time step, each task is assumed to take 40 steps. In order to emphasis the efficiency of the proposed algorithm, we consider the case when the distributed system is very irregular; it is assumed that node number 1 is busy with 60 tasks, and the other nodes are idle; figure (2) shows the efficiency of both the work-stealing approach and ant-colony approach. It is clear that the efficiency of the ant-colony approach comes from the ability to distribute loading information through all the nodes, the tour of ants was randomly chosen, however the cleverness of the ants to carry the new loading-status to each nodes increases the chance of each node to quickly find a good food source or a busy node.

Figure (2): The number of busy nodes with respect to time-steps

Figure (3) studies the effect of enlarging the number of nodes against the number of steps needed to increase the efficiency of the distributed up to 50%. It is shown that as the number of nodes becomes larger, the time-steps needed for the work-stealing approach increases dramatically.

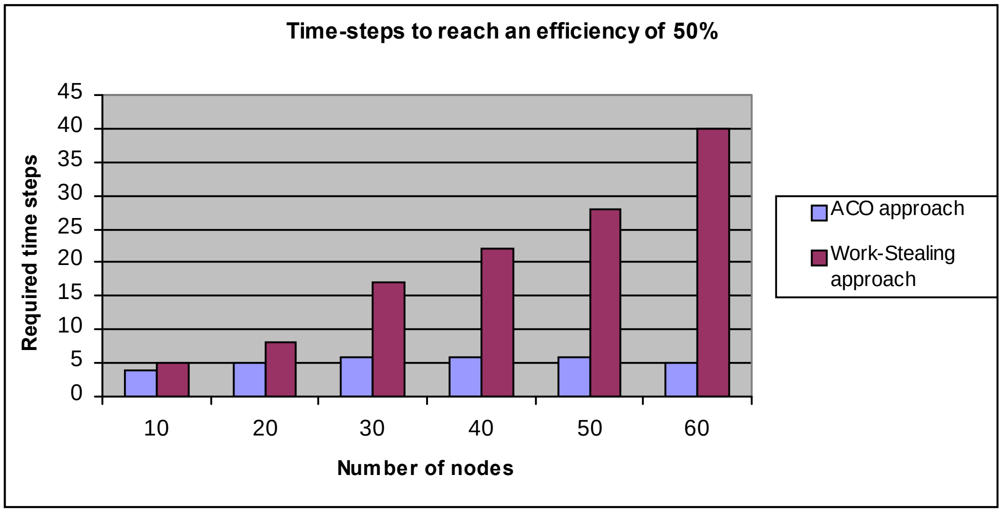

Figure (3): The required time steps needed to reach an efficiency of 50% versus number of nodes

Conclusions

In this paper, a new approach, for using multiple ant colonies in distributed load balancing, was proposed. One feature of this approach is the use of multiple nests, or ant colonies in the search process, this helps in raising the rate of information exchange all over the nodes in the system. Results have shown the efficiency of the proposed model compared with the standard work-stealing algorithm.

Finally, this approach gives an example that highlights the importance of the swarm system in the decision-making process in general, where each agent can play a small role and the global behavior could be robust and reliable.

References

[1] GURURAJ S. RAO, HAROLD S. STONE and T.C. HU", Assignment of Tasks in A Distributed Processor System With Limited Memory", IEEE Trans. on Computers, Vol. C-28, No. 4, April 1979

[2] ASSER N. TANTAWI and DON TOWSLEY, " Optimal Static Load Balancing in Distributed Computer Systems", Journal of the Association for Computing Machinery, Vol. 32, No. 2 , April 1985, pp. 445-465

[3] LIONEL M. NI, CHONG-WEI, XU and THOMAS B. GENDREAU " A Distributed Drafting Algorithm for Load Balancing", IEEE Trans. on Software Engineering, Vol. SE-11, No. 10, October 1985.

[4] DEREK L. EAGER and EDWARD D. LAZOWSKA and JOHN ZAHORJAN, " Adaptive Load Sharing in Homogeneous Distributed Systems", IEEE Trans. on Software Engineering, Vol. SE- 12, No. 5, May 1986.

[5] SHLOMIT S PINTER and YARON WOLTSTAHL, " On Mapping Processes to Processor in Distributed Systems", International Journal of Parallel Programming, Vol. 16, No. 1, 1987.

[6] KANG G. SHIN and YI-CHIEN CHANG, "Load Sharing in Distributed Real- Time Systems with State-Change Broadcasts", IEEE Trans. on Computers, Vol. 38 , No. 8, August 1989.

[7] MING-SYAN CHEN and KAND G. SHIN," Subcube Allocation and Task Migration in Hypercube Multiprocessor", IEEE Trans. on Computers, Vol. 39, No. 9, September 1990.

[8] JIAN XU and KAI HWANG," Heuristic Methods for Dynamic Load Balancing in A Message-Passing Multicomputer", Journal of Parallel and Distributed Computing 18, 1-13(1993).

[9] MOURAD KARA, "Using Dynamic Load Balancing in Distributed Information systems", University of Leeds - School of Computer Studies Research Report Series, Report 94.18, May 1994.

[10] SONGNIAN ZHOU, DOMENICO FERRARI, " An Empirical investigation of Load Indices for Load Balancing Applications", Performance '87, Proc. Of the 12th IFIP WG7.3, International Symposium on Computer Performance Modeling Measurement and Evaluation, Brussels, Belgium, pp. 515-528, 1987.

[11] SONGNIAN ZHOU, DOMENICO FERRARI, " An Experimental Study of Load Balancing Performance", Tech. Rept, No. UCB/CSD 87/336, January 1987.

[12] *Dorigo, M.; Maniezzo, V.; Colorni, A.;* "_Ant system: optimization by a colony of cooperating agents", IEEE Transactions on Systems Man and Cybernetics, Part B,, Volume: 26 Issue: 1 , Feb. 1996.

[13] *Dorigo, M.; Di Caro, G.;* "_Ant colony optimization: a new meta-heuristic", Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99. , Volume: 2 , 6-9 July 1999 -1477 Vol. 2.

[14] Dorigo, M., "Swarm Intelligence, Ant Algorithms, and Ant Colony Optimization", Adaptive Computation and Simulation: Swarm Intelligence, Budapest 2001

[15] Peter Tarasewich , Patrick R. McMullen , "Swarm Intelligence: Powers in Numbers", Communications of the ACM August 2002 , Volume 45 Issue 8.

[16] http://www.nero.uni-onn.de/lehreWS02/skript_al_pfeifer_chap4.pdf

[17] Krohn, J. "Ant Algorithms and the Swarm Intelligence Model of Problem Solving". In Proceedings of UMM Computer Science Discipline Seminar Conference, Morris: University of Minesota, 2001

[18] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed optimization by ant colonies", *Proc. First European Conference on Artificial Life*, Paris, France, pp. 134-142, 1992.

[19] Merkle, D.; Middendorf, M.; Schmeck, H. Proceedings of the Twentysixth Annual Conference of the IEEE International Conference on Industrial Electronics, Control and Instrumentation IECON-2000.

[20] Martin Middendorf, Frank Reischle, Hartmut Schmeck: Information Exchange in Multi Colony Ant Algorithms. IPDPS Workshops 2000: 645-652.

[21] M. Heusse, ENST de Bretagne, Brest, France, "A New Distributed and Adaptive Approach to Routing and Load Balancing in Dynamic Communication Networks"

[22]M.A. Salehi and H. Deldari (Iran)," Grid Load Balancing using an Echo System of Intelligent Ants",

Proceeding (517) Parallel and Distributed Computing and Networks - 2006.

[23] Kwang Mong Sim, and Weng Hong Sun, " Multiple Ant Colony Optimization for Load Balancing", Department of Information Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong.,

[24] Kwang Mong Sim and Weng Hong Sun, *Member*, *IEEE*," Ant Colony Optimization for Routing and Load - Balancing: Survey and New Directions", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS, VOL. 33, NO. 5, SEPTEMBER 2003.

[25]Catalin Bulancea, Ben Paechter, Adam Carter, "Using ACO Metaheuristics on Load Balancing Algorithm"