Volume 4, Issue 6, June 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Load Balancing in Network to Handle Largest Influx of Traffic Using Ant Colony Optimization

Nitin Girme (Student)
Department of Computer Engineering, Dr. D.Y.Patil COE Ambi,
Pune, India

Abstract— This survey describes a method of achieving load balancing in telecommunications networks, excess data carrying nodes can become congested in network and causes to be chance of data lost. In addition to calls, the network also supports to population of simple mobile agents with their behaviours of modelled on the trail laying abilities of ants. The ants move across the network in between randomly chosen pairs of nodes; as they move they deposit simulated pheromones as a function of their distance from their own source node, and the congestion encountered on their root. They select their path at each intermediate node according the distribution of simulated pheromones at each node. The calls between nodes are routed as a function of the pheromone distributions at each intermediate node. The performance of the networks is measured by the average no of hops taken to complete the calls. In this report, the results of using the ant based control (ABC) are compared with those achieved by using fixed shortest path routes (dijkstra's algorithm) used in network management.

Keywords— Load Balancing, AntNet Algorithm, Ant Colony Optimization, Pheromone Table, Trail Laying, Optimal

I. INTRODUCTION

Load balancing is a technique to spread work between two or more computers, network links, CPU's, hard drives, or other resources, in order to get optimal maximize throughput, minimize response time and resource utiliza-tion. Load balancing can be useful when dealing with redundant communications (networks) links. Load balancing is basically the construction of call routing schemes which successfully distribute the changing load over the system and minimize lost calls. Absolutely it is possible to determine the shortest routes from every node to every other node in the network. In this way the average utilization of nodes will be minimized, but it is not necessarily the ideal way to avoid congestion of node, as this has to do with how the traffic on the network is distributed. This type distributed system controlling by means of a single central controller has several disadvantages. The controller usually needs to have current knowledge about the entire system, necessitating communication links of system from every part of the system to controller. Since technology growing faster, people are more interested to finding ways that connect each other faster than last one. However many technologies invented for this problem, some that are highly effective in individual cases. Ants first evolved 120 million years ago, they found more than 10,000 different species. They leaved in different colonies. For moving from one place to other ants having ability to create their own path called as "ant streets". Streets are in the form of long, bidirectional lanes of single pathways in which they navigate landscapes in order to reach a destination in optimal time. Ants drop the pheromone on the ant streets which guide them using a shortest path mechanism. This technique allows finding optimal minimum path. In the year 1990's Computer Scientist researched new routing algorithm in result Ant Colony Optimization, it minimize number of nodes that are useful to connect in order to complete network and minimizing load and increasing reliability. Marco Dorigo and Thomas Stutzle gave the design to implement ANTNet, it conclude how the algorithm perform and how it could be further implement [22].

II. REVIEW

A. Problems with Routing in the Internet

Ants have the ability to create long bi-directional paths between their sources to destination within optimal minimum time. There are several obstacles in the network, there need to change at any point in network, several network algorithm ants perform vital roles in such condition. Ants are able to change the path, that reason researchers are interested most. By applying these ability to routing algorithm internet can flow faster. Many times routing devices taken down due to many reasons, it is hard to new router or device to find a path but router which was previous in the path. ACO aimed is doing that one easily.

B. Packet Switching and Circuit Switching

As usual two technologies Circuit Switched and Packet Switched Network used for sending information towards the internet. Circuit switching is related to telephone lines, is follow telephone calling procedure and Packet Switching is related to packet, packet is sent from one computer to another by using several paths.

C. Developing Ants to Travel the Network

In the real world ants are in search of foods, once they do they come back to the colonies. They can do this because at leaving their colony they trail their pheromone on the path and at the time of returning they easily come back to home. Pheromone is some kind of liquid that trails by ants on the path to identify the optimal path. This is what about ants, but scientist tried to follow the same thing artificially, they need to artificial pheromone trail for accomplish the task of optimal path. Greedy heuristic algorithm state that in network packet proceed down multiple paths and picks the next step based upon previous packet knowledge. Trail is updated whenever successful path is achieved. Tail designed by two types of ants, the regular ants and uniform ants. ACO have regular ants which transport data from source to destination in the most efficient manner and in optimal time. At every node in the path ants maintain routing table, which is initially setup by uniform ants and determine which next node to take as it progresses to the destination. This will make all the regular ants converge on to that route which is then taken by almost all of the packets. When this state is achieved the ants are said to be stable. This is the technique finds the fastest path through the topology. The regular ants do not use as much intelligence as the uniform ants, which are used to find the fastest routes through the network. Uniform ants are dispatched to travel the network and seed the probabilistic routing tables so that the worker ants can use their heuristics to choose the best path. These ants do not necessarily need a destination because they are just searching to find the fastest paths to different nodes. This is also important because the origin node may not have knowledge of the entire network layout. Uniform ants are also immune to the probabilities which have been previously set at each router to guide the regular ants; this makes them immune to the oscillation problems in the network. These ants use backwards reporting and once they reach a certain destination will report back to the previous node the cost that was encountered from that hop, the routers then update their probabilistic routing tables. These ants also use their own heuristics to determine which route to take next, which ensures that not all of the uniform ants take same path at a given node. Every path has the same probability of being taken. They also have an attribute associated with them called time to live, at every node that the worker ant passes a variable is increased by one and once this hits a certain limit the ant dies. This ensures that the ant will not be crawling around the network forever.

D. Nature of Ant

Behaviour of each ant is much unsophisticated insects. They having small amount of memory that appears to have a large random component. They acts as collective as well as manage to perform a variety of complicated tasks with great reliability and consistency [15].

E. Principles of Trail Laying

Depending on the species, ants may lay pheromone trails when travelling from the nest (source) to food (destination), or from food (destination) to the nest (source), or when travelling in either direction. Sometimes they follow these trails, which are a function of the trail strength. Ants drop pheromones as they walk by stopping briefly and touching their gaster, which laying the pheromone secreting gland, on the ground or any area. The ants have a decision to lay there pheromones when trailing around the ground. Pheromones evaporate and spread away, the strength of the trail when it is encountered by another ant is a function of the original strength, and the time from the trail was laid. They applied some principles for finding food for example [7]:

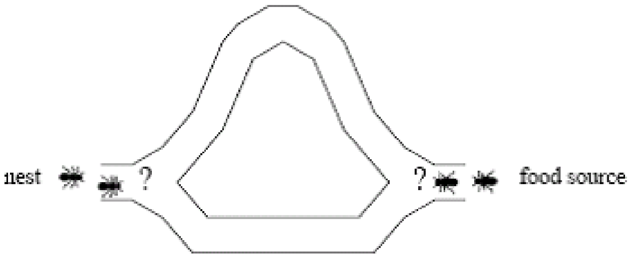


Fig.2.1 Ants have decision to make

In Figure 2.1, there are two possible routes between nest and food-source. Firstly, an ant arriving at a T-crossing (choice point), they makes a random decision with a probability of 0.5 to turn left or right. The path lines represent the pheromone trails. Ants chose the shorter branch have arrived at their destination (food source), while the ones that chose the longer branch are still on their way. For both branches ants initially select their way with a 0.5 probability, if there is no pheromone on the paths yet. If pheromone on the path, then there is higher probability of an ant for choosing the path with the higher pheromone concentration, after several moments later the simulation of ants are as follows:

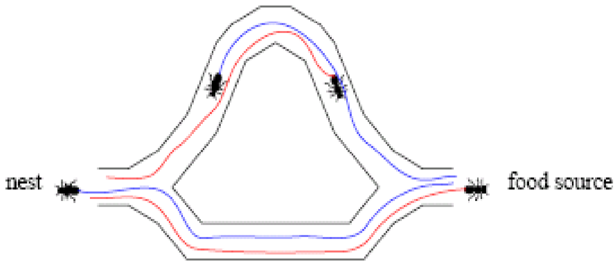


Fig. 2.2 Simulation several moments later

Although this is essentially self-organization rather than learning, ants have to cope with a phenomenon that looks very much like overtraining in reinforcement learning techniques. There are two main issues: the blocking problem and the shortcut problem (Sutton, 1990). The blocking problem occurs when a route previously found by the ants is no longer available. It can then take a relatively long time for the ants to find a new route. The shortcut problem occurs when a new, shorter route suddenly becomes available. In this case the new route will not easily be found, because the old trails are so strong that almost all the ants choose them.

III. PARAMETERS OF SYSTEM

In telecommunication network, the calls between two nodes are routed through number of intermediate hopes. Load balancing it is the process of constructing calls in network and distribute data in network without lost. Artificially we required Ant Based Control to managing load balance in net-work. Following are the parameters used to implement artificial ant in network.

A. Pheromone Table

In network routing tables are replaced by probabilities which are named as pheromone table. Every node has the pheromone table which records the every move towards the destination node [16].

B. Pheromone table update criteria

Ants move from one node to another according to probabilities of the destination node. Once they arrive at destination node they update the pheromone table. Here method used to update the table is quite simple, an ants arrives at node, they entries into the corresponding pheromone table according to the following formula [16].

$$P_{new} = \frac{P_{old} + \Delta P}{2 + \Delta P}$$

Here p is the new probability and Δp is the probability (or pheromone) increase. The other entries in the table of this node are decreased according to the formula.

$$P = \frac{P_{old}}{1 + \Delta P}$$

C. Call routing in network

To determine look for a call artificial ants look in pheromone table before moving from source node to destination. Routes are valid if destination is clear to reach the ant. If route is congested then newly arriving call pressure the load on exiting node, it means open the delay mechanism. The relationship between calls, nodes, pheromone table and ants as follows.

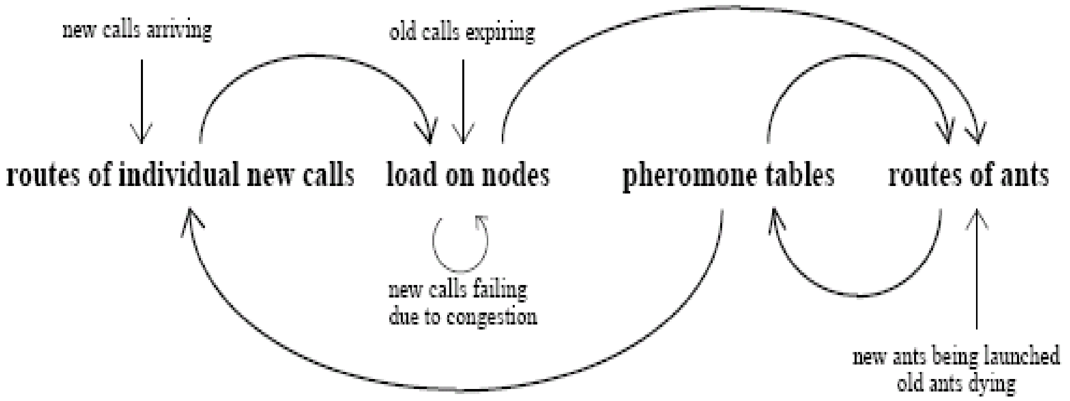


Fig. 3.1 Figure 3 Relationship between calls, node utilization, pheromone tables and ants. An arrow indicates the direction of influence [16].

IV. METHODOLOGY

A. ANT NET ALGORITHM

The AntNet first version, presented in 1997 [Dorigo et al 97], will be denominated AntNet1.0, and the second version published in 1998 [Dorigo et al 1998] will be called AntNet2.0.

Suppose a data network, with N nodes, where *s* denotes a generic source node, when it generates an agent or ant toward a destination *d*. Two types of ants are defined:

1. Forward Ant, denoted F_{s-}, which will travel from the source node to a destination .
2. Backward Ant, denoted B_s that will be generated by a forward ant F_s in the destination d, and it will come back to s following the same path traversed by F_s, with the purpose of using the information already picked up by F_s in order to update routing tables of the visited nodes

Every ant transports a stack $S_{s\to d}($ of data, where the k index refers to the ke-st visited node, in a journey, where $S_{s\to d}($ = s and $S_{s\to d}(\gamma$ = d, being m the amount o jumps performed by F_s for arriving to d. Let k be any network node; its routing table will have N entries, one for each possible destination. Let j be one entry of k routing table (a possible destination). Let be set of neighboring nodes of node k. Let be the probability with which an ant or data packet in k jumps to a node i, $i \in I$ when the destination is j $(j \neq$. Then, for each of the N entries in the node k routing table, it will be values of subject to the condition:

$$\sum_{i \in Nk} P_{ij} = 1; j = 1, \dots, N$$

The following lines show AntNet pseudo code, using the symbols and nomenclature already presented:
BEGIN
{Routing Tables Set-Up: For each node k the routing tables are initialized with a uniform distribution:

$$P_{ij} = \frac{1}{N_k}, \forall i \in N_k$$

DO always (in parallel)
{
STEP 1: In regular time intervals, each node s launches an $F_{s\to d}$ ant to *a* randomly chosen destination *d*.
/*During its trip to *d*, when $F_{s\to d}$ reach a node *k*, (*k*≠*d*), it does step 2*/
DO (in parallel, for each $F_{s\to d}$
{
STEP 2: F_s pushes in its stack $S_{s\to d}($ the node k identifier and the time elapsed between its launching from s to its arriving to k.
$F_{s\to d}$ selects the next node to visit in two possible ways:
(a) It draws between i nodes, i→N_k, where each node i has a P_{di} probability (in the k routing table) to be selected.
IF the node selected in (a) was already visited
(b) It draws again, but with the same probability for all neighbor nodes i, i→N_k. Fs→d jumps to chosen node.
IF the selected node was already visited
STEP 3: A cycle is detected and $F_{s\to d}$ pops from its stack all data related to the cycle nodes, since the optimal path must not have any cycle. $F_{s\to d}$ comes back to step 2 (a).
END IF
END IF
} WHILE jumping *node ≠ d*
STEP 4: $F_{s\to d}$ generates another ant, called backward ant $B_{s\to d}$. $F_{s\to d}$ transfers to $B_{s\to d}$ its stack $S_{s\to d}$ and then dies.
/*$B_{s\to d}$, will follow the same path used by $F_{s\to d}$, but in the opposing direction, that is, from d to s*/
DO (in parallel, for each $B_{s\to d}$ ant)
{
/*When $B_{s\to d}$ arrives from a node *f*, *f*∈N_k to a *k*, it does step 5*/
STEP 5: $B_{s\to d}$ updates the k routing table and list of trips, for the entries regarding to nodes k' between k and d inclusive, according to the data carried in $S_{s\to d}$ (k').
IF *k*≠s
$B_{s\to d}$ will jump from k to a node with identifier given by $S_{s\to d}$ $(k-1)$
END IF
} WHILE (*k*≠s)
}
}
END
The routing table and list of trips updating methods for k are described as follows:

1. The k routing table is updated for the entries corresponding to the nodes *k*' between *k* and d inclusive. For example, the updating approach for the d node, when $B_{s\to d}$ arrives to *k*, coming from *f*, $f \in N_k$ is explained, next:

A P_{df} probability associated with the node f when it wants to update the data corresponding to the d node is increased, according to:

$$P_{df} \leftarrow P_{df} + (1-\acute{r}).(1-P_{df})$$

where r´ is an dimensional measure, indicating how good (small) is the elapsed trip time T with regard to what has been observed on average until that instant. Experimentally, r' is expressed as:

$$\acute{r} = \begin{cases} \frac{T}{c\mu} & c \geq 1 \quad \frac{}{c\mu} < 1 \\ 1 & otherwise \end{cases}$$

where: µ is the arithmetic observed trip time T average. c is a scale factor experimentally chosen like 2 (Dorigo 1997). More details about r' and its significance can be found in (Dorigo 1997).

The other neighboring nodes $(j \neq f)$ P_{df} probabilities associated with node *k* are diminished, in order to satisfy equation (1), through the expression:

$$P_{df} \leftarrow P_{df} + (1-\acute{r}).P_{df} \qquad \forall j \in Nk, j \neq f$$

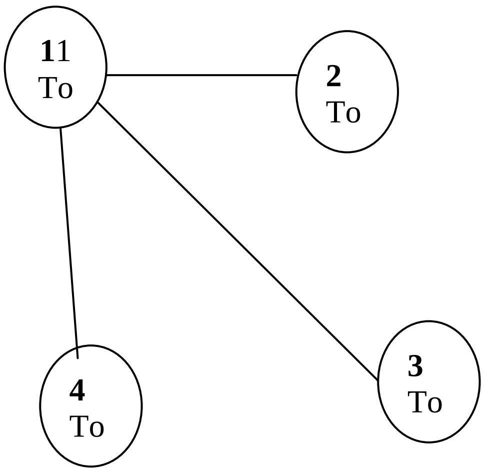
2. A list $trip_k$ μi σi^2, μ i, σi^2 of estimate arithmetic mean values μi and associated variances σi^2 for trip times from node k to all nodes i $(i \neq k)$ is also updated. This data structure represents a memory of the network state as seen by node k. The list trip is updated with information carried by $B_{s\to d}$ ants in their stack $S_{s\to d}$. For any node pair source-destination, µ after (n+1) samples (n>0) is calculated as follows:

$$\mu_{n-1} = \frac{n\mu_x + x_n + 1}{n+1}$$

Where: xn+1 trip time T sample n+1, µn arithmetic mean after n trip time samples [16].

V. EXPERIMENTAL RESULT

To begin with, each possible path has an even likelihood of being chosen. An ant is placed on a network of 4 nodes, with the source node 1 and destination node 2. A chance mechanism is invoked and a path is chosen.



Next node	% chance
2	33.33333
3	33.33333
4	33.33333

Table 5.1 Pheromone table for node 1

Fig. 5.1 The network graph

In this case, node 2 has been selected [Fig. 5.1] and the ant arrives at its source destination.

The ant then moves and updates the pheromone tables for the visited nodes with a higher (and more mathematically biased) value. This would be calculated for Figure 5.1 and Table 5.1 in the following way:

Node 2 was the final destination

It took 1 hop to get to its destination

Divide 1 (hop) by 100: 100%
Add 100 to the probability value of node 2 (currently 33.3333): 133.3333
Add the values of the other nodes to 133.3333 (133.3333 + 33.3333 + 33.3333): 200 (approximately)
Calculate the ratio: ratio = 100/200 0.5
Set the probability of the node to its current value multiplied by the ratio
Node 2: 133.3333 * ratio (0.5) = 66.6666%
Node 3: 33.3333 * ratio (0.5) = 16.6666%
Node 4: 33.3333 * ratio (0.5) = 16.6666%
Node 2 (66.6666%) + Node 3 (16.6666%) + Node 4 (16.6666%) = 99.9999%

The system isn't 100% accurate as the total will never add up to exactly 100%, but it will be close enough to allow accuracy within the level required [16].

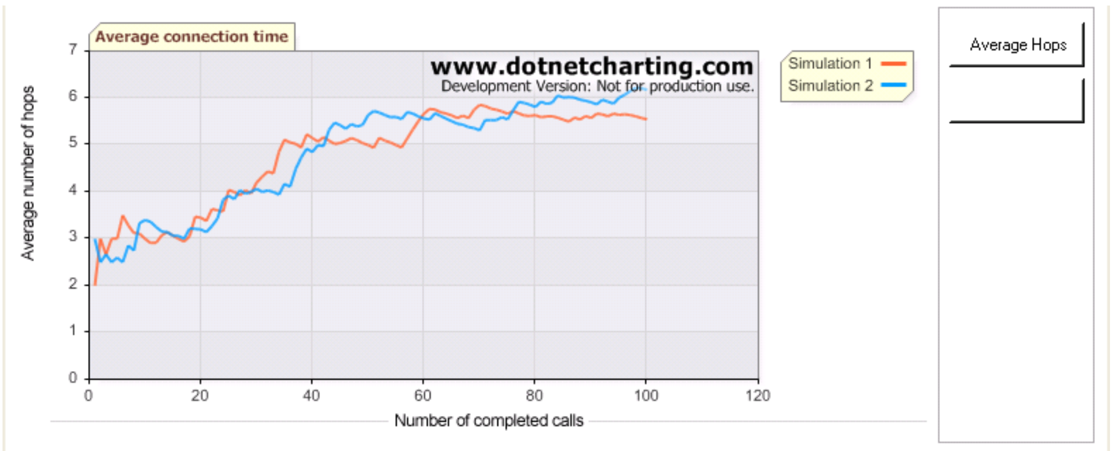


Fig. 5.2 for 100 calls AntNet required 5.5 hopes (Orange Color) and Dijkstra required 6.18 hopes (Blue Color).

VI. CONCLUSIONS

As we know the current telecommunication networks suffer from network issues such as network congestion which causes failing of calls in network. For that purpose required routing algorithm as well load sharing algorithm to minimize the effect of congestion. In this survey we have simulated ant colony algorithm for load balancing in Network to handle largest influx of traffic. Average no of nodes used for complete a fixed number of calls in the network. We found that average no of hopes are less in case of ant colony optimization.

REFERENCES

[1] [1] B. Barán, M Almirón, E. Chaparro. ‘Ant Distributed System for Solving the Traveling Salesman Problem’. pp. 779-789. Vol. 215 th Informatic Latinoamerican Conference-CLEI,. Paraguay (1999).
[2] [2] S. Bak, J. Cobb, E. Leiss. ‘Load Balancing Routing via Randomization’. pp. 999-1010. Vol. 215 th Informatic Latino american Conference-CLEI.Paraguay (1999).
[3] [3] M. Dorigo, V. Maniezzo, A. Colorni. ‘The Ant System: Optimization by a colony of cooperating agents’. pp. 1-13. Vol. 26-Part B. IEEE Transactions on Systems, Man, and Cybernetics,. Number 1 (1996).
[4] [4] M. Dorigo, G. Di Caro. ‘AntNet. Distributed Stigmergetic Control for Communications Networks’. pp. 317-365 Journal of Artificial Intelligence Research. Number 9 (1998).
[5] [5] [Schoonderwoerd et al 97] R. Schoonderwoerd, O. Holland, J. Bruten. ‘Ant-like agents for load balancing in telecommunications networks’. Hewlett-Packard Laboratories, Bristol-England (1997).
[6] [6] Beckers, R., Holland, O.E., & Deneubourg, J.L. (1994). From local actions to global tasks: Stigmergy and Collective Robotics. In R.A. Brooks, & P. Maes (Eds.) Artificial Life IV, Cambridge, MIT Press. p. 181-189.
[7] [7] Beckers, R., Deneubourg, J.L., & Goss, S. (1992).Trails and U-turns in the Selection of a Path by the Ant Lasius Niger. In J. theor. Biol. 159, 397-415.
[8] [8] Stickland, T.R., Tofts, C.M.N., & Franks, N.R. (1992). A path choice algorithm for ants. In Naturwissenschaften 79, 567-572.
[9] [9] Goss. S., Beckers, R., Deneubourg, J.L., Aron, S., & Pasteels, J.M. (1990). How trail-laying and trail following can solve foraging problems for ant colonies. In R.N. Hughes (Ed.) NATO AS1 Series, Vol. G20 Behavioral mechanisms of food selection, Springer Verlag.

[10] [10] Deneubourg, J.L., Goss, S., Franks, N., Sendova-Franks, A., Detrain C., & Chrétien, L. (1990). The dynamics of collective sorting robot-like ants and ant-like robots. In J.-A. Meyer & S. Wilson (Eds.), From Animals to Animats: Proceedings of the first international conference on simulation of adaptive behavior. Cambridge, MIT Press.

[11] [11] Beckers, Deneubourg & Goss, 1992; Deneubourg & Goss, 1989; Goss et.al., 1990; Franks, 1989.

[12] [12] Beckers, Holland & Deneubourg, 1994; Russell 1995; Deveza et.al., 1994.

[13] [13] A. Shankar, C. Alaettinoglu, I. Matta. 'Performance Comparison of Routing Protocols using MaRS. Distance Vector versus Link-State'. Technical Report, Maryland-USA (1992)

[14] [14] Modeling the collective building of complex architectures in social insects with lattice swarms, Theraulaz and Bonabeau, Journal of Theoretical Biology, 1995.

[15] [15] Bert Hölldobler, E.O. Wilson: "Journey to the Ants: A Story of Scientific Exploration, 1994, ISBN 0-674-48525-4.

[16] [16] http://ethesis.nitrkl.ac.in/1128/1/thesis.pdf

[17] [17] Grid Load Balancing Using Ant Colony Optimization Nasir, H.J.A. ; Ku-Mahamud, K.R. Computer and Network Technology (ICCNT), 2010 Second International Conference on Digital Object Identifier: 10.1109/ICCNT.2010.10 Publication Year: 2010 , Page(s): 207 – 211

[18] [18] Load Balancing of Nodes in Cloud Using Ant Colony Optimization Nishant, K. ; Sharma, P. ; Krishna, V. ; Gupta, C. ; Singh, K.P. ; Nitin, N. ; Rastogi, R. Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on Digital Object Identifier: 10.1109/UKSim.2012.11

[19] [19] Load Balancing in Non-dedicated Grids Using Ant Colony Optimization Yixiong Chen Semantics, Knowledge and Grid, 2008. SKG '08. Fourth International Conference on Digital Object Identifier: 10.1109/SKG.2008.41

[20] [20] A Survey of Ant Colony Optimization-Based Approaches to Routing in Computer Networks Janacik, P. ; Orfanus, D. ; Wilke, A. Intelligent Systems Modelling & Simulation (ISMS), 2013 4th International Conference on Digital Object Identifier: 10.1109/ISMS.2013.20 Publication Year: 2013 , Page(s): 427 – 432

