Performance Measures of Swarm based Active Network for Multiclass Packet Routing-A Simulation study

Constandinos X. Mavromoustakis and Helen D. Karatza
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
E-mail: {cmavrom,karatza}@csd.auth.gr

KEYWORDS
Bandwidth Clustering, Embedded Packet based Swarms, Tunable Scheduling Scheme.

ABSTRACT

Pervasive computing environments demand reliability and multiclass QoS for end to end real time applications. The concept of active network has been recently adopted in many distributed environments. This paper describes and examines the combination of the active network concept with swarm based control method called Swarm-based Active Network scheme for optimized and tunable QoS priority routing. This hybrid scenario is using the bandwidth clustering approach in which packets are following the predetermined path allocated by swarm based packets (embedded swarms). This mechanism enables the adaptation of the system to new conditions, and does not require message bro-kers to fulfill multiclass QoS demands. Simulation results reveal the response in terms of the end-to-end delay, available bandwidth and packet loss of this scheme under different traffic measures.

1. INTRODUCTION

Active routing has become one of the most attractive methods in wired and wireless communication networks. By combining various hybrid schemes of different active agent-based methods it enhances the overall performance of the system offering end-to-end user reliability and integrity. The objective of the generic concept of active agent-based techniques is to achieve high resource utilization and to reduce user contention for network resources. As load is not uniformly distributed, network resources decrease in the form of efficiency, bandwidth, processing power and memory, leading network to an unpredictable behavior. It becomes evident that the mechanism used for load balanc-ing and resource allocation has to avoid overloaded nodes such that transmission delays (latencies) are minimized, whereas alternative routes should be activated whenever load conditions are drastically changing.

While there are often models and efficient algorithms facing allocation problems in deterministic systems (typically formulated as matching or flow optimization problems) solutions for the networks with stochastic components are not straightforward. Decentralized schemes [3, 6, 10] involve a number of controllers each of which is supervising a part of a network. This allows the central control to be applied locally to a part of a network enabling all “local” supervisors to cooperate deploying a decentralized exchange of information. Indeed many distributed systems are unreliable and subject to random failures of their components. Examples of such systems are power grids where the distribution ability of the network can be affected by demand overloads and other random events or various transportation/information networks subjected to congestions and intermittent failures. Thus optimally designed systems have to offer end user reliability and integrity by enabling equal share of network resources.

This work examines an active network technique, which combines a reactive and proactive behavior of message passing using the Split Agent Routing Technique (SART) [14, 15] based on swarms. Embedded swarm based packets [8] that are delay sensitive are marked as prioritized. In turn, agents recognize these packets as being a part of a packet. This hybrid method is called swarm-based message passing [15]. It provides a model for active and distributed network data flow organization and also continuous resource reservation on demand. SART technique is applied to an active network using smart/active swarm based packets and nodes. In turn the bandwidth clustering mechanism is activated for priority routing-as will be discussed later-in order to assign a certain bandwidth. A thor-ough study and discrimination is made for the class of service offered at any time in the network as well as for the QoS issue (particularly for delay sensitive packets where a slotted window tangles). We have considered a number of metrics that are associated with network performance and evaluation of the degree of distribution ability. Further-more measures for handling multimedia streaming (MMs) are presented with the relative trade-offs, based on the reliability and QoS offered by the proposed scheme.

The organization of the paper is as follows: In section 2 a description of the basic principles of the swarm-based scheme and adaptive resource allocation is presented

Proceedings 19th European Conference on Modelling and Simulation
Yuri Merkuryev, Richard Zobel, Eugène Kerckhoffs © ECMS, 2005
ISBN 1-84233-112-4 (Set) / ISBN 1-84233-113-2 (CD)

with extensive use of active multiclass bandwidth clus-tering method. Section 3 draws the simulation results and conclusions and suggestions for future research are sum-marized in section 4.

2. SWARM BASED ADAPTIVE RESOURCE ALLOCATION

Agent-based approach was first introduced and standardized by Appleby and Steward's mobile agent's algorithm [3]. Further studies [4, 6, 10] have shown that an ant-like mobile agent algorithm could be applied to a network with significant optimization of the QoS metrics of the network. In [10], Dorigo and Gambardella used the metaphor of trail laying by ants to certain combinatorial optimization problems [3, 4]. Several agent-oriented approaches [4, 6, 10, 14] have recently been proposed that appeal to principles extracted from Swarm Intelligence (SI) and aspire to solve routing problems to wired and wireless communication networks. On the other hand, within these methods there are some trade-offs that have to be taken into account. These trade-offs deal with network overall performance, such as generated overhead in message passing for agents communication, network utilization, simplicity for the implementation, etc.

Agent based network routing could be biologically inspired and based on insect colonies which exhibit a simple behavior for their communication and living. Real ants have similar behavior with agents. They are repre-sented in the network as artificial agents that bias the network collecting useful information for the whole environ-ment through their hormones called pheromones. In previous researches [8, 14, 15] a hybrid (proactive and reactive) agent behavior is developed in which ants are adapting their communicational behavior to network circumstances, simultaneously splitting themselves for passing information to neighbor nodes. In [15] the SART technique was used for path marking on demand and capacity reservation, and was shown that this method efficiently marks the path and reserves the capacity required for offering optimized QoS service metrics to end users.

One of the major challenges of the service/resource allocation problem is to find algorithms that are reactive and deliver reliable high quality solutions. The nature of the environment and the underlying infrastructure with the consolidated use of end-to-end paradigm have focused the research on transport level as the proper place to address congestion issues. Sources usually react by reducing their flow. Indeed, this is a response to congestion, but it has the unavoidable side effect of reducing the throughput. Figure 1 shows that the flow control method/scheme is closely related to routing schemes used in the network. From a source S to a destination D, the used routing scheme affects the underlying discrete flow control-for any given time t-which results fluctuation in throughput response of the system.

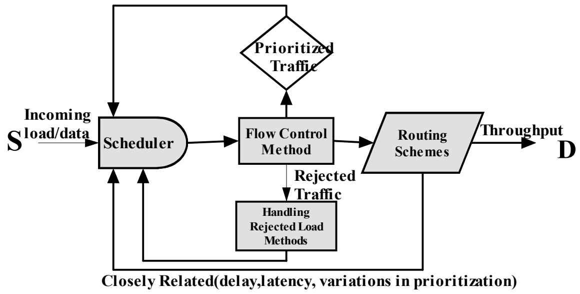


Figure 1: Relation between traffic-incoming load, routing schemes, flow control and throughput.

2.1 Embedded swarm based active packets for resource allocation

2.1.1 Active network approach

In an active network, packets are more than just passive data. They are programs executed by the nodes that are passing the packets along or embedded in packets that influence control decisions. This facility allows packets to make on-the-fly decisions about how to route themselves. Tennenhouse et al [2] provides a survey of this work. Active network's approach is motivated by both lead user applications, which perform user-driven computation at nodes within the network.

Current resource discovery approaches for offering optimized QoS have been designed primarily for small networks or for networks where dynamic updates are not common or frequent. Hence, they do not scale well when the number of resources grows, and updates are common. Some of the most popular service discovery systems are Jini from Sun Microsystems, SLP [5], INS (Intentional Naming System) [11], and SDS [9]. In those systems, ser-vices advertise their descriptions to resolvers-sometimes to intermediate nodes- which act as resource directories. In turn resolvers allocate the dedicated streaming packets to a link where available resources are adequate to efficiently forward streams from a node to another.

This paper proposes an active network strategy for optimized QoS provision and adaptive resource alloca-tion based on embedded swarm agents. This research is using the SART technique for interaction and updating in-formation between packets and nodes. Resource capacities are provided locally to where demands occur, avoiding cross-network traffic. This active technique is proved to be very adaptive and can rapidly respond to changes in the environment while significantly reduces the generated network overhead.

2.1.2 Swarm based active network

Active packets and nodes can perform different computations reactively. The encapsulation abilities of such a net-work enable advanced mechanisms for end to end communication to scarce or sensitive resources. Traditional packet headers are replaced with control programs in order to make on the fly decisions.

The swarm based system approach associates pheromone trails to features of the solutions of a combinatorial prob-lem, which can be seen as a kind of adaptive memory of the previous solutions. In our implementation each packet-agent launches in the network and influences the pheromone table [10, 14] by increasing or reducing the entry for the proper destination using antipheromone [14]. The pheromone quantity represents the available end-to-end resources. The pheromone table at each node with neighbors can be measured as:

$R_i = \left[r^i_{1,m}\right]_{n-1,k(i)}$,*n-1* destinations and *k* next nodes(1)

Artificial ants are biasing the network by generating at every simulation time step ant-packets destined to every node randomly. In the network ants are walking according to probabilities assigned in pheromone tables and they are visiting one node at every time step. In this way ants increase the entry in the pheromone table corresponding to the node from which they came from:

$$\mathrm{P} = \frac{P_{old} + \Delta P}{1 + \Delta P} \tag{2}$$

where ΔP is the quantity of pheromone increased and P_{old} is the previous entry. The other entries in the table of this node are decreased accordingly following the formula:

$$P = \frac{P_{old}}{1 + \Delta P} \tag{3}$$

Routing tables contain a two-way pheromone table parameters (bi-directional links with different capacity) which are maintained in each node, and are expressed as:

$$P^{i_t \mapsto n_t}_{k(i)} \text{ and } P^{n_t \mapsto i_t}_{k(i)} \tag{4}$$

where $k(i)$ are the next nodes[1] for N_i , *n* is one of the *n-1* possible destinations and n_t is the possible next node at a certain time step.

All probabilities are thresholded [14] between $\frac{1}{(number\_of\_neighbors)^2}$ and 0.75 in order to prevent the pheromone saturation state.

The route, where intermediate nodes have large pheromone quantities, is selected as the best-chosen path. If the destination can be reached on a hop-by-hop look-up table method, the route is valid otherwise the packet is blocked [14] and lost. Routing table entries updates are measured according to the following:

$$r^i_{i-1,s}(t+1) = \frac{r^i_{i-1,s}(t) + \delta r}{1 + \delta r} \tag{5}$$

[1] Nodes can be both endpoints (can be source and destinations) and switches (can perform routing functions).

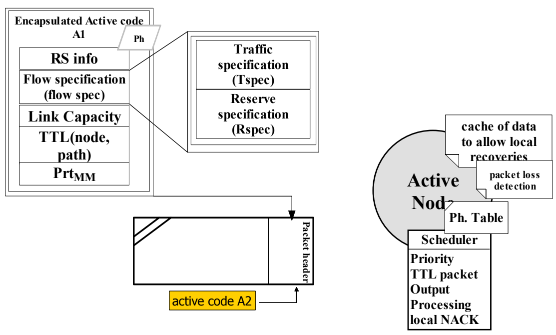
where δr is the step size parameter and *s* is the source node. Similarly for all neighbors to $i, r^i_n(t)$ is found that:

$$r^i_{n,s}(t+1) = \frac{r^i_{n,s}(t)}{1 + \delta r}, n \neq i-1 \tag{6}$$

In this way smart data packets and ants have an interaction in the means that ants affect the routing tables while data packets influence the service rate of the traffic on nodes, which affects the ants with the delay mechanism. This swarm/agent-oriented approach encompasses the generic agent based concept which enables agents to move around the network, gathering information about the topology of the system and the traffic at a discrete time.

Active swarm based packets are used in a network for internal communication and auto-configuration between *packets* and *nodes*. Roughly speaking active packets are self-contained piece of software that has the properties of autonomy and interaction. Packets make on the fly decisions about how to route themselves and learn (gather information) from their passing-by environment. Figure 2 shows the structure of swarm based packet's and node's structure.

A packet consists of a "flow spec" field and a "filter spec" field as in the well known Resource reSerVation Protocol (RSVP) [13]. Packets 'carry' traffic spec and path information from a source to destination and reserve information during their journey to destination. The RS info field of the encapsulated active code is the main swarm based active field in the packet's header. RS info field, in cooperation with other fields in both node and packet, can expose the determined quantity of pheromone for the selected source and destination. This pheromone quantity is based on the available end-to-end resources [8, 15]. The flow spec specifies the desired QoS for the packet. The filter spec defines the set of data packets to receive the QoS defined by the flow spec. The entire service class in packet header (both A1 and active code A2) defines the desired QoS and beyond others, it describes the type of the data flow (priority degree of MM or don't care packets).



Packet structure **Node structure**

Figure 2: Active swarm based packet's and node's structure.

Active packet's communication, on a continuous basis, enables adaptation of the system to new conditions. This method is not error prone and enables additional information to be passed to neighboring nodes embodied in transmitted packets. This communication between node and packet is opaque to other network layers.

2.1.3 Active resource reservation mechanism using bandwidth clustering method

The bandwidth clustering approach is based on the available bandwidth on each data link in the network. Due to the unpredictable incoming traffic measurements that occur at any time in the network, the capacity of each channel (bandwidth) is reduced progressively with an increased flow of packets. Bandwidth clustering method is based on the idea of clustering nodes using different levels of bandwidth [8]. The clustering idea is based on whether the remaining capacity could efficiently be reserved by different streams to better utilize the path. This method takes place after the SART algorithm is applied to the network to bias the paths and overcome the transient state [14]. Overloaded paths are also clustered but the remaining capacity will mark the cluster as a lower bandwidth cluster [8]. The bandwidth clustering method enables the manipulation of different paths that offer different levels of bandwidth based entirely on the information collected from swarm based packets. Furthermore the bandwidth clustering method is associated with a cooperative learning and active environment, producing a robust and decentralized way for adapting link's changing capacities quickly.

Figure 3(i) illustrates the typical decision selections of a proper node i to destination D. In figure 3(i) node *i* has four different options for the D destination. Path options a and b on one hand are using different intermediate nodes (j and k respectively) and different clusters leading to D destination (C1 and C2 respectively). On the other hand c and d are using a combination of clusters C1, C2, C3 using intermediate nodes l and m respectively. Figure 3(ii) shows a topology and the regions arising when clustering with respect to several levels of bandwidth. Having as source node A and a destination node D, agents have already marked the path [15] and allocated the remaining capacity to paths. The same time a cluster is being constructed at each path having the available bandwidth of the channel with the lowest free remaining capacity (1).

$$Max\left(C_{A\mapsto D}\right) = BW \tag{7}$$

In (7) BW is the available bandwidth, which is determined by the lowest free remaining capacity in the path from A to D (bottleneck concept).

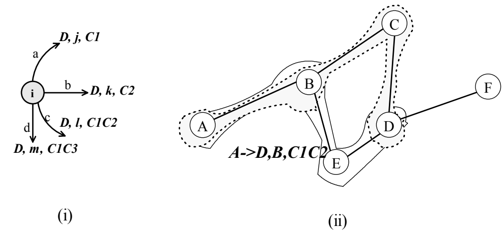


Figure 3(i) and (ii): A topology and the regions arising when clustering with respect to several levels of bandwidth. Color regions represent nodes connected by links with higher free bandwidth (solid lines). Dashed lines indicate more congested links.

After split agents measure the path from a source to destination, the clustering bandwidth mechanism is activated for each node estimating each time step the differences in their links. In figure 3 (*i, ii*) all nodes are informed and the clusters are constructed according to capacity reservation information spread by agents onto nodes. Thus a packet has to decide whether Cluster 1 or Cluster 2 should be chosen for reaching destination. These decisions are made according to information of remaining bandwidth which was obtained in the previous time step. In that step packets passed and updated the information as expressed in (5) and (6). Although routing is performed through the swarm based method, the bandwidth reservation method occurs simultaneously with routing decision. Bandwidth reservation method is based on the priority degree of each packet. Swarm based routing decisions affect the bandwidth clustering method, which in turn affects bandwidth reservation.

To define the accurate resource allocation in terms of bandwidth the updated entries of the pheromone table of node *n* should satisfy the following:

$$p_b = \sum_i r^i_{n,s} = 1 = full\_BW \tag{8}$$

Then decision of which path should be followed can be measured as follows:

$$Max\left(C_{A_{C1}\mapsto D_{C1}}\right) > Max\left(C_{A_{C2}\mapsto D_{C2}}\right) \tag{9}$$

for which the ideal path ensuring packet transmission with adequate capacity is $\left(C_{A_{C1}\mapsto D_{C1}}\right)$. This clustering scheme will enable the reduction of lookup tables and generated overhead, thereupon additional queries to neighbors are avoided.

2.1.4 Constructed clusters' recreation frequency and triggering

A critical issue arises as to when or whether the clusters should remain unchanged, as well as the duration that each cluster handles (remains in active period). After consecutive network flows it is shown that network is biased overcoming the transient state. Thus paths are marked [15] and ready for the creation of each cluster to host a bounded capacity. The creation criteria are entirely based on path marking of split agents where the successfully marked path is chosen to be set as a cluster. Beyond the path marking shown in [15] we used cellular automata (CA) to overcome the pathlock [14]. For instance node N_A, uses CA simply to exchange information with the neighboring nodes examining[2] whether its neighbors have at least a single link leading to the proper destination (that has higher remaining capacity). If these links of N_A's neighbor are not empty N_A remains 'alive' and the cluster is not destroyed. Otherwise the cluster is obliterated.

The issue of when the path should change is critical and has to be taken into account. Triggering with real traffic (swarm based active packets) is an action in order to set different paths in combination with the co-operative agents and CA comparison in each node. Each time a packet is transmitted from a source to a destination, information regarding the cluster of which the packet itself is a part of, is exchanged between the packet and the nodes. This mechanism occurs in a dynamic way (pheromones extraction) exchanging information about the available bandwidth at each time step. Each cluster has a threshold S_{cap}, and provides an active environment with a proper QoS. Some thresholds for bandwidth should be chosen so that nodes could be dynamically clustered in areas called "blocking areas". Every node in the path is grouped and belongs to the blocking area labeled with *X*Mb/s, if at least one route with as much free bandwidth between nodes exists. Adaptivity to the different traffic flows which results a significant variation in bandwidth is performed using SART. SART simply enables the wider/distributed view of path availability in order to host traffic based on bandwidth availability. It is undoubtedly true that if the threshold S_{cap}, where $S_{cap} < link\_available\_BW$, is overtaken, meaning that the cluster has reached or passed the overloaded value, then once again the cluster is obliterated and recreation procedure takes place, with agents' contribution and their path marking.

2.2 Adaptive resource allocation with reliable traffic flows in constructed clusters

When transmitting data traffic, high reliability is a parameter of main importance; thus extensive retransmission and rate control schemes may be used. On the other hand, when transmitting continuous media (on demand Multimedia streaming) the requirement of on time delivery must be balanced against that of transmission reliability. In our approach we tried to balance both concepts. A freshness degree evaluation and the reliability of links are substantial metrics to ensure reliability for data traffic flow.

Each node measures the number of links at the beginning of each time step τ and the number of its broken links τ_{Tot} (if any). It becomes evident that the rate of broken links is equal to: $\frac{\tau_{Tot}}{\tau_b}$, where τ_b is the number of node's broken links. One way to evaluate reliability is by using the following notation:

$$\left(1-\frac{\tau_{Tot}}{\tau_b}\right)^{\mathrm{T}_i \cdot N} \qquad (10)$$

where T is the time steps that have passed since the creation of the cluster(i), and N is the number of nodes in the cluster.

According to (11) we have measured the total path reliability notation and the link survivability factor as follows:

$$\mathrm{Re}\,liability=\left(1-\frac{\tau}{\tau_b}\right)^{\mathrm{T}_i \cdot N+\sum_{i=1}^{hops}\frac{i-1}{Rf}}, \text{ number of hops} >1 \qquad (11)$$

where *Rf* is the link capacity refreshment factor and *i* is the number of links in the path. In this way we know the reliability degree of each link where we can evaluate the total reliability by using equation (11) above.

2.3 Network generated overhead and Quality of Service (QoS)

Many applications are sensitive to the effects of delay, delay variation (jitter) and packet loss. As known, resource and bandwidth reservation generated overhead is the number of control packets that are sent relative to the data packets. In the described hybrid scenario using bandwidth clustering, packets are delivered from one node to another following the predetermined path allocated by agents. The swarm-based scenario does not use specific control packets like other routing schemes since ants pursue the control by being a part of the packets (smart-active packets). Therefore embedded swarm agents map and control the traffic at any time during transmission in the network, and consequently overhead is potentially reduced.

Roughly speaking the large number of control packets introduces more chances of packet collision, longer delay, and more packets dropped resulting in insufficient QoS. The above scheme with embedded swarms allows scalability, and efficient usage of network bandwidth. Additionally no message brokers are needed to fulfill QoS demands issued by applications. As known accurate network state information is very expensive to maintain and monitor.

Another issue that has to be taken into account for resource overhead reduction is that swarm based scheme enables each node to view its absolute neighbor(s), thereupon additional queries to neighbors are avoided. It must be stated out that the inexistence of generated overhead is a result of the non-transmitted routing tables values or other information blocks to neighbors or to all nodes of the network. In other words this scheme significantly reduces the cost of QoS

[2] Examination is performed within routing tables.

prioritized routing. But in essence it increases the efficiency with the no need for packet tunneling and the 'useless' need to process packets at layers other than the network layer.

3. SIMULATION RESULTS AND DISCUSSION

To demonstrate the design methodologies discussed in this paper, exhaustive simulations were performed to a partially meshed 100-node network. Nodes capacity has been chosen to be relatively high on each node's buffer (680kb). Network's performance is examined through a number of various metrics that characterize the efficiency of the proposed scheme. In the implementation-simulation of this work we used our own libraries implemented in C programming language (based in C/Objective C). In the implementation of the bandwidth clustering scheme different traffic input streams have been tested[3,4]. The evaluation took place for different levels of prioritization (marked 1-5(highest)) and 'don't care' packets (no priority at all). The network traffic is modeled by generating constant bit rate (CBR) flows. Each source node transmits one 512-bytes (~4Kbits-light traffic) packet. In the described scenario link capacity is 6Mbps (bi-directional) each. Packet's requests are routed only once using the saved values on nodes.

In figure 4 the average packet delay with the number of injected packets in the network is illustrated. As the number of packets increases the average packet delay increases slightly. After consecutive simulations for this metric it has been shown that when the number of injected packets reaches 550-600, the average packet delay increases dramatically. This occurs due to capacity limitations that bind each node. Swarm based active scheme shows that in order to reject the incoming packet, data flow scheduler and flow specification fields are responsible to cache locally the packet to node (figure 2). This results in significant delay but negligible compared with the case of packet loss shown in figure 5.

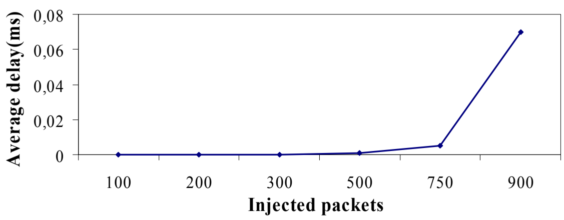


Figure 4: Average packet delay versus the number of injected packets in the network.

[3] Streams included Light Traffic(CBR randomly selected ~4Kbits) and Heavy Traffic(4·Light Traffic)

[4] Multimedia traffic (MM) is prioritized and delay sensitive.

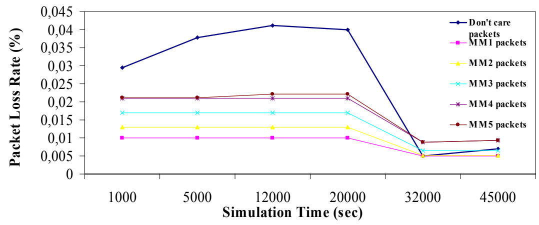


Figure 5: Packet Loss rate for different grade of prioritization (1-5) of MM packets and non prioritized packets (don't care).

Figure 5 illustrates packet loss rate for different grade of prioritization of MM packets as well as for "don't care" packets. "Don't care" packets have the highest packet loss rate because no prioritization takes place using the swarm based active scheme. Prioritized packets in real time could be video streams where packet loss and delay could be disastrous for the offered QoS to end users. From figure 5 we can discriminate that the packet loss rate for any of the MM 1-5 prioritized packets is relatively low compared with that of "don't care" packets. Additionally the prioritized packets MM-4 and MM-5 are behaving almost the same for packets loss rate.

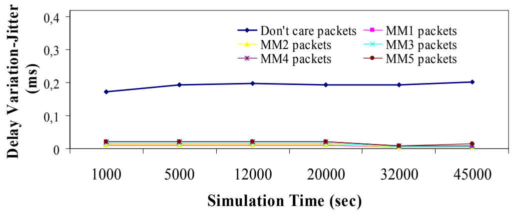


Figure 6: Delay variations (jitter) for different grade of prioritization of MM packets and non prioritized packets (don't care).

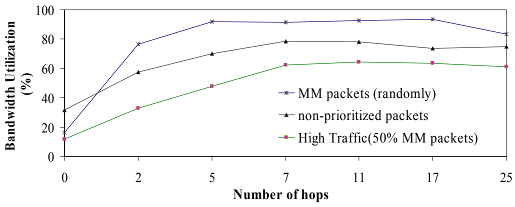


Figure 7: Bandwidth utilization (%) for different traffic flows versus the number of hops.

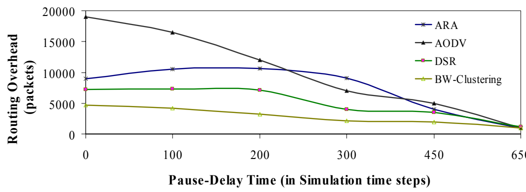


Figure 8: Comparison of different protocols for the generated routing overhead packets.

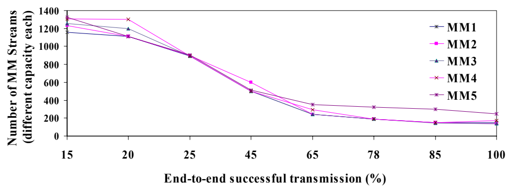


Figure 9: Service tunability for different priority degrees.

In figure 6 the delay variations (or jitter) for different grade of prioritization of MM packets are shown. Once again the swarm based active scheme enables active prioritization of MM packets which cause significant reduction in the jitter parameter. Jitter is almost the same for low prioritization MM packets (MM-1) and for high prioritization (MM-4, MM-5). Active swarm based scheme prove to be robust in delays where sensitive prioritized packets are scheduled to be transmitted in bounded end-to-end delay.

Figure 7 shows the percentage of bandwidth utilization as the number of hops increases, for MM streams traffic flow versus non-prioritized packets. In figure 7, MM bursty traffic utilizes the available bandwidth reaching □91% of the total available. “Don’t care” packets can not utilize more than 67%. This is caused by the high prioritization of MM packets conflicting with “don’t care packets”. MM packets block “don’t care packets” to reach their destination and balk them from utilizing any link.

In figure 8 we compared the routing overhead generated for 3 different schemes: the generic Ant-Colony-Based Routing Algorithm (ARA), generic Distance-Vector Protocol (DV) and Dynamic Source Routing (DSR). The large number of control packets introduced by other methods enables packet collisions and longer delays which results in insufficient QoS. Significantly less routing overhead is generated in DSR and bandwidth clustering schemes, while the other methods generated doubtless higher overhead packets.

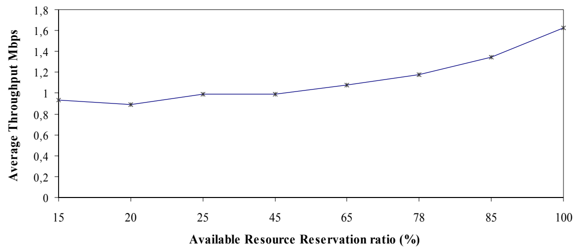


Figure 10: Average throughput as a function of the available resource reservation rate.

Figure 9 shows the service tunability for different priority degrees of MM packets. Each prioritized MM packet has different degree of prioritization to reach a destination. However MM-5 packets behave almost the same as MM-1 packets. This means that the swarm based active scheme enables localized prioritizations of packets to nodes. However nodes are then responsible for scheduling and co-responsible with packets for flow control. Thus when packets are marked as prioritized, swarm based active method “distributes” this prioritization along paths and influences other packets to follow the same path. This is why MM-5 packets behave almost the same as MM-1 pack-ets, while a further examination could be implemented in our future research for large scale networks where flow rerouting could take place on demand.

Figure 10 illustrates the average throughput response as a function of the available resource reservation rate. When all available resources are reserved, the average throughput response reaches the 1.52Mbps. On the other hand when the available resources are reserved in 15% the average throughput reaches 0.93Mbps. Swarm based active packets are shown to behave better in terms of the average throughput response for high percentage of reserved available resources.

4. CONCLUSIONS AND FURTHER RESEARCH

This work presents a novel method for active swarm based scheme for self-configuration suited for optimized QoS priority routing and the overhead reduction issue. Resources are allocated by using the bandwidth clustering method which is associated with a cooperative learning environment producing a decentralized way capable of adapting quickly to changing capacities. This scheme is applied in a swarm based active network environment where active packets continuously communicate with active nodes by using the SART. This mechanism enables the adaptation of the system to new conditions (bandwidth reservation/capacity allocation), as well as additional information to be passed to neighboring nodes for which information is embodied in transmitted packets. Paths are clustered with re-spect to different levels of bandwidth in order to enable capacity allocation and bandwidth reservation on demand, for any requested traffic. Simulation results reveal that this scheme offers path reliability and enable tunable control in data traffic flow while the same time significantly reduces the generated overhead in the network. Active band-width clustering scenario has shown that it can successfully perform optimized bandwidth utilization (avoiding satu-rated routing or pathlock) and based on the service provided, it successfully offers end-to-end reliability.

Multiclass routing is more than simply putting together routing algorithms designed for individual traffic classes. Scalable interclass resource distribution is essential to achieving high network throughput. Thus this research could be extended for network scalability examination using variants of hybrid-agent based schemes and applied to a Mobile Ad-hoc Network (MANET) where no infrastructure exists. Embedded agents could be entirely responsible for adapting -on demand- the proper resource allocation to the type of service (i.e. voice, multimedia, secure applica-tions for untrustworthy users). Furthermore a scope of interest would be the exploration of different ways in the ma-nipulation of the pheromone quantity.

5. REFERENCES

[1] Chavez, A. Moukas, P. Maes, Challenger: A Multi-agent System for Distributed Resource Allocation, Proceedings of the First International Conference on Autonomous Agents, 1997 pp.323–331.

[2] D. Tennenhouse, J. Smith, W. Sincoskie, D. Wetherall, and G. Minden, A Survey of Active Network Research. *IEEE Communications Magazine*, 35(1):80-86, January 1997.

[3] S. Appleby, S. Stewart, Mobile software agents for control in telecommunications networks, *BT Technology Journal,* vol. 12, 1994, pp. 104-113.

[4] M. Dorigo, V. Maniezzo, A. Colorni, The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*, vol. 26, No. 1, 1996, pp. 29–41.

[5] E. Guttman, C. Perkins, Service Location Protocol, Version2. RFC2608. http: //www.ietf.org/rfc/rfc2608.txt (1999)

[6] R. Schoonderwoerd, O. Holland, J. Bruten, L. Rothkrantz, Ant-based Load Balancing in Telecommunications Networks, *Adaptive Behavior*, vol. 5, 1997, pp. 169-207.

[7] S. McCanne and S. Floyd, NS network simulator version 2.1b8, June 2001, http://www-mash.cs.berkeley.edu/ns/ .

[8] C. X. Mavromoustakis, H. D. Karatza, Bandwidth Clustering for Reliable and Prioritized Network Routing using Split Agent-based method. To appear in the Proceedings of International Workshop on Assurance in Distributed Systems and Networks (ADSN 2005) in conjunction with the 24th International Conference on Distributed Computing Systems Workshops (ICDCS 2005) IEEE Computer Society on Distributed Processing, Columbus, Ohio, USA, June 6, 2005.

[9] S. Czerwinski, B. Zhao, T. Hodes, A. Joseph, R. Katz, An Architecture for a Secure Service Discovery Service, International Conference on Mobile Computing and Networking, Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, 1999, Seattle, Washington, United States, pp. 24–35.

[10] M. Dorigo, L. M. Gambardella, Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computation, (1997) 1:53-66.

[11] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, J. Lilley, The design and implementation of an Intentional Naming System(INS). Proceedings of the seventeenth ACM symposium on Operating systems principles, Charleston, South Carolina, United States, pp: 186-201, 1999.

[12] S. Roy, B. Lesieutre, G. Verghese, Resource Allocation in Networks: A Case Study of the Influence Model, Proceedings of the 35th Hawaii International Conference on System Sciences-2002:pp. 220-228.

[13] . L.Zhang S. Berson S. Herzog S. Jamin, Resource Reservation Setup Protocol, official charter maintained by the IETF, 1997-Resource ReSerVation Protocol-Version 1 Functional Specification. RfC: 2205.

[14] C. X. Mavromoustakis, H. D. Karatza, Split agent-based routing in interconnected networks (SART). International Journal of Communication Systems (IJCS), Wiley, 2004; 17:pp.303-320.

[15] C. X. Mavromoustakis, H. D. Karatza, An on demand path marking and capacity reservation method using Split Agent. Proceedings of Workshop on Challenges of Large Applications in Distributed Environments (CLADE), in conjunction with the thirteenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-13), Honolulu, Hawaii, 07 June, 2004, pp.128-138.

CONSTANDINOS X. MAVROMOUSTAKIS *received a five-year dipl.Eng in Electronic and Computer Engineering from Technical University of Crete and MSc in Telecommunications from University College of London, UK. He is currently a PhD student at Aristotle University of Thessaloniki, Greece, at the department of Informatics. His research interests are in the areas of Multiagent systems, Swarm Intelligence, Epidemic Model for Information Diffusion in wireless networks, Computer network modelling and performance evaluation. His email and web address are <cmavrom@csd.auth.gr> and <agent.csd.auth.gr/~cmavrom>.*

HELEN D. KARATZA *is an Associate Professor in the Department of Informatics at the Aristotle University of Thessaloniki, Greece. Her research interests include Computer Systems Performance Evaluation, Multiprocessor Scheduling, Mobile Agents, Distributed Systems and Simulation. Her email and web address are <karatza@csd.auth.gr> and <agent.csd.auth.gr/~karatza>.*