Adaptive Probabilistic Routing Schemes for Real Time Traffic in High Speed Dynamic Networks

Abdelhamid Mellouk [(1)], Samia Larynouna[(2)], Saïd Hoceïni[(1)]

[(1)] ***LISSI/SCTIC, IUT Creteil/Vitry, University Paris XII, Vitry sur Seine, 94400, France***
[(2)] ***PGS « Network & Telecommunication », Science Computer Dept, Borj El Bahri, Algiers, Algeria.***

Summary
Routing is a relevant issue for maintaining good performance and successfully operating in a network. We focused in this paper on neuro-dynamic programming to construct dynamic state-dependent routing policies which offer several advantages, including a stochastic modelization of the environment, learning and evaluation are assumed to happen continually, multi-paths routing and minimizing state overhead. In this paper, we propose an approach based on adaptive algorithm for packet routing using reinforcement learning called K Shortest Paths Q Routing which gives much more better performances compared to standard shortest path, K-Shortest algorithm and Q-routing algorithms. To improve the distribution of the traffic on several paths, we integrate a probabilistic module in order to compute dynamically a probabilistic traffic distribution. This module takes into account the capacity of the queuing file in the router and the average packet delivery time. The performance of our algorithm is evaluated experimentally with OPNET simulator for different levels of traffic's load and compared to standard shortest path and Q-routing algorithms on large interconnected network. Our approach prove superior to a classical algorithms and is able to route efficiently in large networks even when critical aspects, such as the link broken network, are allowed to vary dynamically.

Key words:
Quality of Service based Routing, Multi Path Routing, Dynamic Network, High Irregular Traffi.

1. Introduction

Network, such as Internet, has become the most important communication infrastructure of today's human society. It enables the world-wide users (individual, group and organizational) to access and exchange remote information scattered over the world. Currently, due to the growing needs in telecommunications (VoD, Video-Conference, VoIP, etc.) and the diversity of transported flows, Internet network does not meet the requirements of the future integrated-service networks that carry multimedia data traffic with a high Quality of Service (QoS). The main drivers of this evolution are the continuous growth of the bandwidth requests, the promise of cost improvements and finally the possibility of increasing profits by offering new services. First, it does not support resource reservation which is primordial to guarantee an end-to-end Qos (bounded delay, bounded delay jitter, and/or bounded loss ratio). Second, data packets may be subjected to unpredictable delays and thus may arrive at their destination after the expiration time, which is undesirable for continuous real-time media. In this Context, for optimizing the financial investment on their networks, operators must use the same support for transporting all the flows. Therefore, it is necessary to develop a high quality control mechanism to check the network traffic load and ensure QoS requirements. It's clear that the integration of these QoS parameters increases the complexity of the used algorithms. Anyway, there will be QoS relevant technological challenges in the emerging hybrid networks which mixed several different types of networks (wireless, broadcast, mobile, fixed, etc.). Constraints imposed by QoS requirements, such as bandwidth, delay, or loss, are referred to as QoS constraints, and the associated routing is referred to as QoS routing which is a part of Constrained-Based Routing (CBR).

Interest in constrained-based routing has been steadily growing in the Networks, spurred by approaches like ATM PNNI, MPLS or GMPLS [1]. With MPLS, fixed length labels are attached to packets at an ingress router, and forwarding decisions are based on these labels in the interior routers of the label-switched path. MPLS Traffic Engineering allows overriding the default routing protocol (e.g., OSPF [2]), thus forwarding over paths not normally considered.

CBR approaches determines only a path, and does not reserve any resources on that path. A resource reservation protocol such as RSVP must be employed to reserve the required resources. Another Architecture proposed for providing Internet QoS is the Differentiated Services architecture. Diffserv scales well by pushing complexity to network domain boundaries.

In this paper, we focus our attention in some special kind of CBR called state dependent QoS Routing. Basically, these algorithms selects routes based on flow QoS requirements and network resource availability. A lot of study has been conducted for an alternative routing paradigm that would address the integration of dynamic criteria. The most popular formulation of the optimal distributed routing problem in a data network is based on a multicommodity flow optimization whereby a separable objective function is minimized with respect to the types of flow subject to multicommodity flow constraints [3]. However, due their complexity, increased processing burden, a few proposed routing schemes could been accepted for the internet like QOSPF (Quality Of Service Path First) [4].

Another kind of CBR which is interesting for us is those which take into account an accurate prediction of the network dynamics during propagation of the message through the network. This, however, is impossible unless the routing algorithm is capable of adapting to network state changes in almost real time. So, it is necessary to develop a new intelligent and adaptive routing algorithm. This problem is naturally formulated as a dynamic programming problem, which, however, is too complex to be solved exactly. In our approach, we use the methodology of reinforcement learning (RL) introduced by Sutton [5] to approximate the value function of dynamic programming. One of pioneering works related to this kind of approaches concerns Q-Routing algorithm [6] based on Q-learning technique. In this approach, each node makes its routing decision based on the local routing information, represented as a table of Q values which estimate the quality of the alternative routes. These values are updated each time the node sends a packet to one of its neighbors. However, when a Q value is not updated for a long time, it does not necessarily reflect the current state of the network and hence a routing decision based on such an unreliable Q value will not be accurate. The update rule in Q-Routing does not take into account the reliability of the estimated or updated Q value because it's depending on the traffic pattern, and load levels, only a few Q values are current while most of the Q values in the network are unreliable. For this purpose, other algorithms have been proposed like Confidence based Q-Routing (CQ-Routing) or Dual Reinforcement Q-Routing (DRQ-Routing) [7].

All these routing algorithms use a table to estimate Q values. However, the size of the table depends on the number of destination nodes existing in the network. Thus, this approach is not well suited when we are concerned with a state-space of high dimensionality. For this, we have developed an algorithm, called K Shortest Optimal Path Q Routing Algorithm (KORA) [8], which improves standard Q-Routing algorithm in term of average packet delivery time. It reduces the search space to K-Best no loop paths in terms of hops number

We give in section II a summarized view of our developed approach. Next, we present a probabilistic module which takes into account the capacity of the queuing file mixed with the average packet delivery time. Finally, performance of our probabilistic KORA is evaluated experimentally with OPNET simulator for different levels of traffic's load and compared to the version with no probabilistic distribution of multi path routing.

2. K Q-Routing Optimal Shortest Paths Approach.

In this section, we present an adaptive routing algorithm based on a multi-paths routing technique combined with the Q-Routing algorithm. This approach requires each router to maintain a link state database, which is essentially a map of the network topology. When a network link changes its state (i.e., goes up or down, or its utilization is increased or decreased), the network is flooded with a link state advertisement (LSA) message [9]. This message can be issued periodically or when the actual link state change exceeds a certain relative or absolute threshold [9]. Obviously, there is tradeoff between the frequency of state updates (the accuracy of the link state database) and the cost of performing those updates. In our model, the link state information is updated when the actual link state change. Once the link state database at each router updated, the router computes the K-Optimal optimal paths and determines the best one from Q-Routing algorithm.

2.1. Constructing K-Optimal Paths

Several papers discuss the algorithms for finding K-Optimal paths [10]. Our solution is based on a label setting algorithm (based on the Optimality Principle and being a generalization of Dijkstra's algorithm) [10]. The space complexity is $O(Km)$, where K is the number of paths and m is the number of edges. By using a pertinent data structure, the time complexity can be kept at the same level $O(Km)$ [10]. We modify the algorithm to find the K-Optimal non-loop paths [10].

2.2. Q-learning algorithm for routing

In our routing algorithm, the objective is to minimize the average packet delivery time. Consequently, the reinforcement signal which is chosen corresponds to the estimated time to transfer a packet to its destination. Typically, the packet delivery time includes three variables: the packet transmission time, the packet treatment time in the router and the latency in the waiting queue. In our case, the packet transmission time is not taken into account. In fact, this parameter can be neglected in comparison to the other ones and has no effect on the routing process.

The reinforcement signal T employed in the Q-learning algorithm can be defined as the minimum of the sum of the estimated $Q(y, x, d)$ sent by the router x neighbor of router y and the latency in waiting queue q_y corresponding to router y.

$$T = \min_{x \in \text{neighbor of } y} \{ q_y + Q(y,x,d) \} \quad \textbf{(1)}$$

Where $Q(s, y, d)$, denote the estimated time by the router s so that the packet p reaches its destination d through the router y. This parameter does not include the latency in the waiting queue of the router s. The packet is sent to the router y which determines the optimal path to send this packet [8].

Once the choice of the next router made, the router y puts the packet in the waiting queue, and sends back the value T as a reinforcement signal to the router s. It can therefore update its reinforcement function as:

$$\Delta Q(s,y,d)=\eta(\alpha+T - Q(s,y,d)) \quad \textbf{(2)}$$

So, the new estimation $Q'(s,y,d)$ can be written as follows (fig.1):

$$Q'(s,y,d) = Q(s,y,d)(1-\eta) + \eta(T+\alpha) \quad \textbf{(3)}$$

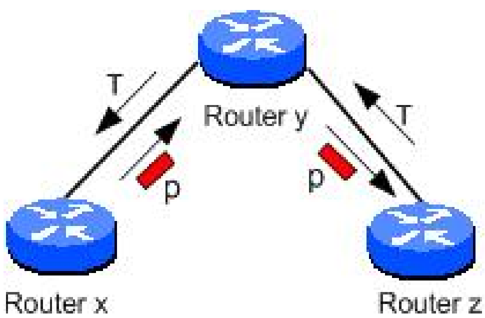

Fig. 1. Updating of the reinforcement signal

2.3. Implementation and Simulation results.

The adaptive routing algorithm is based on the Q-learning approach, the Q-function is approximated by a reinforcement learning based neural network (NN). In this approach, NN ensure the prediction of parameters depending on traffic variations. Compared to the approaches based on a Q-table, the Q-value is approximated by a reinforcement learning based neural network of a fixed size, allowing the learner to incorporate various parameters such as local queue size and time of delay, into its distance estimation. Indeed, a Neural Network (NN) allows the modelling of complex functions with a good precision along with a discriminating training and a taking into account of the context of the network. Moreover, it can be used to predict non-stationary or irregular traffics. In this approach, the objective is to minimize the average packet delivery time. Consequently, the reinforcement signal which is chosen corresponds to the estimated time to transfer a packet to its destination.

The input cells in NN used correspond to the destination and the waiting queue states. The outputs are the estimated packet transfer times passing through the neighbors of the considered router.

This approach offers advantages compared to standard Distance Vector (DV) routing policy and Q-routing algorithm, like the reduction of the memory space for the storage of secondary paths, and a reasonable computing time for alternative paths research. The Q-value is approximated by a reinforcement learning based neural network of a fixed size. Results given in figure 2, experimented on large interconnected network, shows better performances of the proposed algorithm comparatively to standard DV and Q-routing algorithms. In fact, at a high load level, the traffic is better distributed along the possible paths, avoiding the congestion of the network.

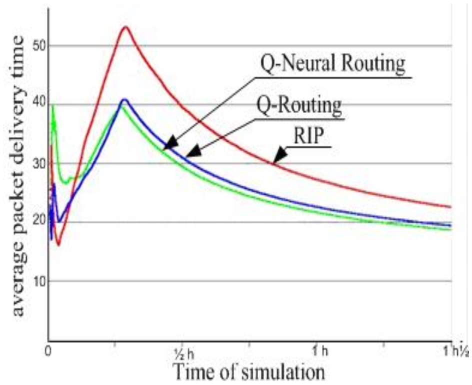

Fig. 2. Network with a high load

Our Q-Neural Routing explore all the network environment and do not take into account loop problem in a way leading to large time of convergence algorithm. It needs a rather large computational time and space memory. In the goal of reducing the complexity of this algorithm, computational time and space used memory, we improved the first version of our algorithm by proposing an hybrid approach combining neural networks and reducing the search space to K-Best no loop paths in terms of hops number reducing. This approach requires each router to maintain a link state database, which is essentially a map of the network topology. When a network link changes its state (i.e., goes up or down, or its utilization is increased or decreased), the network is flooded with a link state advertisement (LSA) message [9]. This message can be issued periodically or when the actual link state change exceeds a certain relative or absolute threshold. Obviously, there is tradeoff between the frequency of state updates (the accuracy of the link state database) and the cost of performing those updates. In this model, the link state information is updated when the actual link state change. Once the link state database at each router updated, the router computes the K Shortest optimal paths and determines the best one from Q-Routing algorithm. This solution is based on a label setting algorithm (based on the optimality principle and being a generalization of Dijkstra's algorithm).

From fig. 3, in the case of continuous high load of simulated traffic on large interconnected network, the results obtained show that the routers implementing classical RIP ignore completely the increasing charge of the network. The packets continue to take the same path. In spite of the many packages taking secondary ways, K-Shortest Path Routing does not present better performances because it rests on a probabilistic method to distribute the load of the network, and not on the degradation of the times of routing. The K-Best Q-Routing presents the best performances than standard Q-Routing algorithms clearly. Indeed, after one period of adaptation, the times are lower than the times induced by the other methods. Thus, mean of average packet delivery time obtained by K-Best Q-routing algorithm is reduced by 25% compared to Q-routing algorithm.

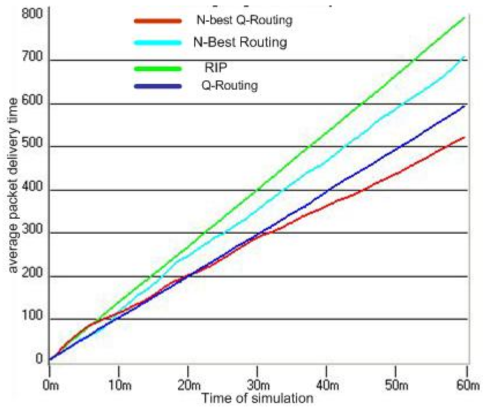

Fig. 3. Network with a continuous high load

These results confirm that Q-Routing algorithm has weak performances due to speed of adaptation of the routers. Moreover, this policy does not take into account the loop problem in way of destination. On the other hand, K-Best Q-routing algorithms explore only the K-Best paths.

3. Probabilistic Path Selection in Multi-Path Routing.

In order to distribute traffic on K Best Paths, we fixed a probability for all founded paths. This probability is computed by counting the number of times where a packet take this path, exactly like ant based routing process [11]. Thus, the shortest path will have a P_{max} probability and the (K-1) remaining paths probability computed in function of the number of times where the router chose this path. To force the router to take the alternative routes find in K Best paths and not only the best one, we added a uniform distributed random process in each router. This process chooses randomly a number between [0, 1]. Next, a router choose the path verifying the condition that it's probability is less than this random number. In this manner, the flow packets reach their destination with a time close to optimal, while ensuring a good exploration of the remaining paths.

However, random process is a rough strategy. It will not take into account the dynamic of the traffic. The value of P_{max} is fixed by a counting process. If this value is big, packets take this path mainly, and the exploration process did not take the other paths. The risk is that the router cannot be informed of a fall of traffic on one of these ways, and continuous to overload a way which is not optimal actually. If the value of P_{max} is small, we can note a fall of performances in term of delivery time: too many packets will be intend for exploration and will make increase the time of average delivery time.

It would be more desirable to compute the probability affected to each path automatically. Contrary to the previous algorithm, this one is based on two values: the packet delivery time computed by the Q learning process and the latency in a waited queue.

Let a router *n*, Let the path *i*, $i \in \{1..K\}$, *K* is the number of Best considered Path in KORA. Let $D_i\ (t)$ be the packet delivery time for path *i* at time *t*. It's a dynamic variable that depends on the path's load and estimated by our Q reinforcement learning process. Let $T_i\ (t)$ be the latency in queuing file associated to closest router *n'* in the direction of path *i* at time *t* (that is, the neighbour of router *n*).

We defined the probability $P_i^k\ (t)$ associated to path *i* in router *k* at time *t* by:

$$P_i^n(t) = \frac{\left[1 \div D_i(t)\right]^{\alpha} * \left[1 \div T_i^{k'}(t)\right]^{\beta}}{\sum_{i=1}^{K} \left[1 \div D_i(t)\right]^{\alpha} * \left[1 \div T_i^{k'}(t)\right]^{\beta}} \quad (5)$$

α and β are two tunable parameters that determine the respective influences of delay or waited queue.

4. Probabilistic Simulation results.

To show the efficiency and evaluate the performances of the new probabilistic module integrated in our approach, an implementation has been performed on OPNET software of MIL3 Company.

The proposed approach has been compared to that based on our earlier version of KORA and tested on a variety of network topologies. Performances of the algorithm are evaluated in terms of average packet delivery time.

After choosing a topology of network, three kinds of traffic have been studied: low load traffic, continuous high load and several peaks of high load traffic. In the first, a low rate flow is sent only by one source node to a destination node, for example in fig. 4, only node *noeud101* send packets to node *noeud100*. From the previous case, we have created conditions of congestion of the network. Thus, a high rate flow is generated by more than one pair of nodes, for example in fig. 4, node *noeud100* and node *noeud103* send packets simultaneously to node *noeud101* and node *noeud102*. The third one consists to create successive peaks of high load of traffic between nodes.

In order to show the efficiency of our probabilistic module integrated in KORA, all network's topology used in this study contains two parts: the left and the right parts. We begin with a few possible ways to route the flow packets between the two parts of the network. After, we increased gradually the number of these ways in order to multiply possibilities of routing decisions.

4.1. Simple topology with a few ways between left and right parts.

The topology of the network employed here for simulations, includes 40 interconnected routers with 4 possible ways to route the packets between the left part and right part of the network, as shown in figure 4.

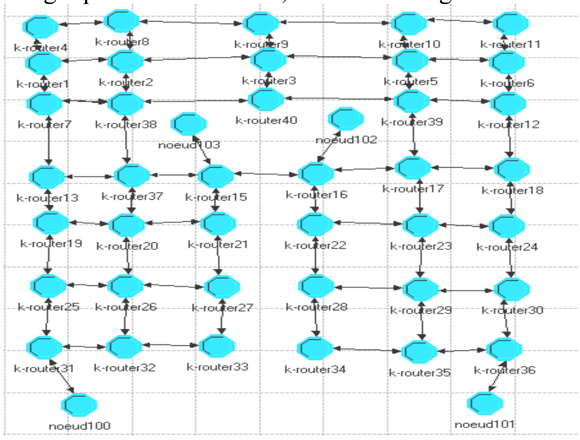

Fig. 4. Network topology with few ways

Performances of algorithms are evaluated in terms of average delivery time. Fig. 5 (a, b, c) illustrates the obtained results in the case of low load, heavy load and peak of traffic for three hours. From figure5a and 5b, one can see, that the two versions of our approach (noted in the graph "KshortestPathQRouting" wich include probabilistic module vs the older KORA version called "old_KshortestPathQRouting" with no computing of path's probabilities) exhibit same performances with a little bit more for the new version of our algorithm. In the case of a peak of high load (fig.5c), one can note that the probabilistic module gives best performances and permit to routers to adapt their decision very fast.

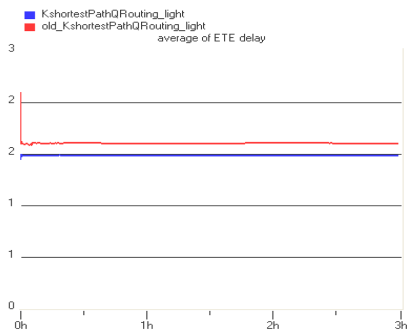

Fig. 5a. Network with a low load of traffic

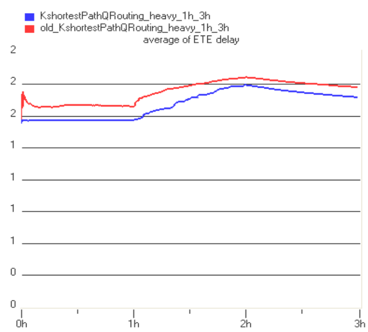

Fig. 5b. Network with a high load of traffic

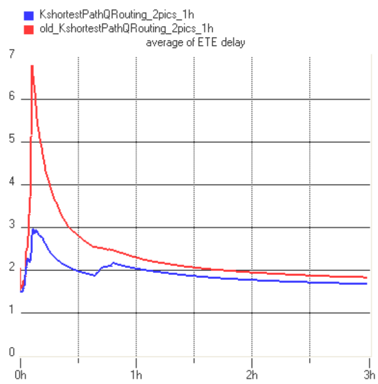

Fig. 5c. Network with a peak of high load of traffic

4.2. Complex topology with great number of ways between left and right part network.

The probabilistic version is apparent only if one has several combinations. In order to test the contribution of this module, we decided to work on networks integrating of many ways between the two principal parts of the network. For example, one of our simulated networks is presented in figure 6. It includes 84 interconnected routers with 12 possible ways to route the packets between the left part and right part of the network.

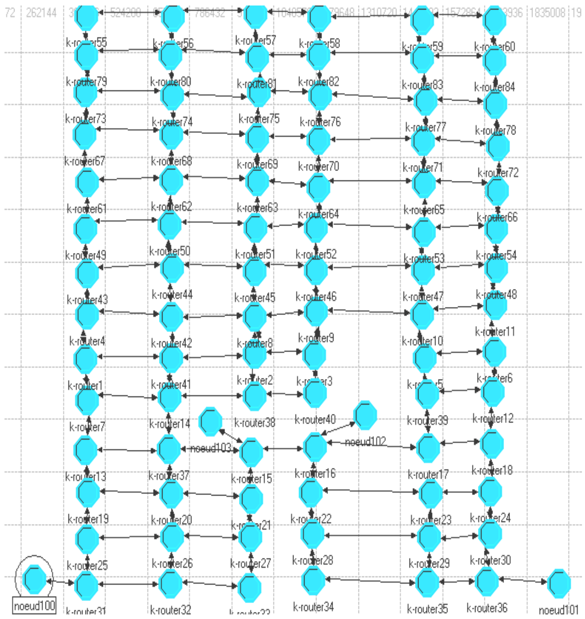

Fig. 6. Network topology with a great number of ways

In the case of a low load (fig 7.a), one can note that after a small period of initialization, performances of probabilistic module are approximately the same as those obtained with old KORA routing policy. Fig. 7.b. illustrates the average packet delivery time obtained when a congestion of the network is generated during 60 minutes.

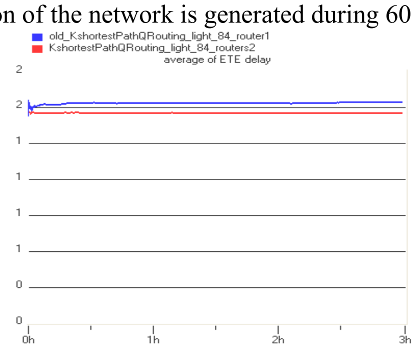

Fig. 7a. Network with a low load of traffic

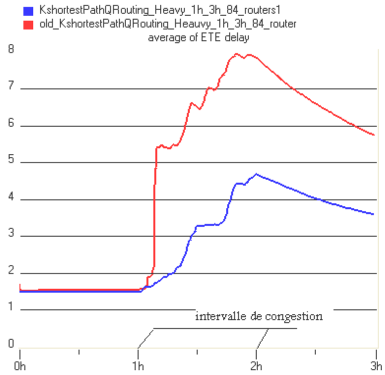

Fig. 7b. Network with a high load of traffic

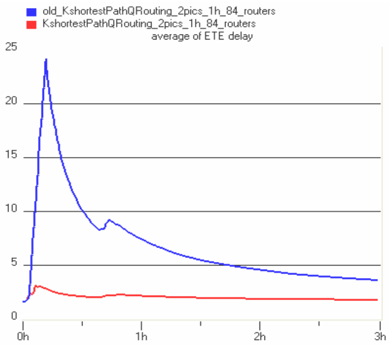

Fig. 7c. Network with a peak of high load of traffiic

Thus, in the case where the number of packets is more important, the version with probabilistic module gives better results compared to older KORA algorithm. For example, after 2 hours of simulation, the new version exhibits a performance of 25% higher than old version. Indeed, the utilization of waiting queue state of the neighboring routers in the decision of routing and average delivery time, allows anticipation of routers congestion. Results obtained in fig7.c confirm that the probabilistic module permit to KORA to give best performances. Thus, mean of average packet delivery time obtained by the new version is reduced by 8% compared to old version. Parameters adaptation of our routing algorithm take into account the high variations of traffic.

5. Conclusion

In this paper, our proposed adaptive routing approach based on a multi-paths routing technique combined with the Q-Routing algorithm called K Shortest Path Q Routing Algorithm is tested for improving distribution of traffic on K-Best paths. Because random process used in traffic distribution is a rough strategy, we also tested a new module to calculate automatically a probability associated at each path. This one is based on two values: the packet delivery time computed by the Q learning process and the latency in a waited queue. The learning algorithm is based on find K-Best paths in term of hops router and the minimization of the average packet delivery time on these paths. Simulation results on high interconnected network and dynamic ling changes shows better performances of the proposed algorithm comparatively to standard KORA System. Finally, our work in progress concerns the metric use in finding K-Best optimal paths (residual bandwidth, loss ratio, waiting queue state …) and take into account other parameters like the information type of each packet (voice, video, data) in path selection.

6. References

[1] W. Stallings, "MPLS", Internet Protocol Journal, Vol. 4, n° 3, September 2001.

[2] J. Moy, "OSPF Version 2", RFC2328, IETF, 1998.

[3] A.E. Ozdaglar, D. P. Bertsekas "Optimal Solution of Integer Multicommodity Flow Problem with Application in Optical Networks", Proc. Of Symposium on Global Optimisation, June 2003.

[4] Z. Wang and J. Crowcroft, "QoS Routing for Supporting Resource Reservation". In IEEE Journal on Selected Areas in Communications, September 1996.

[5] R.S. Sutton and A. G. Barto, "Reinforcement Learning" MIT Press, 1997.

[6] J. A. Boyan and M. L. Littman, "Packet Routing in Dynamically Changing Networks: A Reinforcement Learning Approach." In Cowan, Tesauro and Alspector (eds), Advances in Neural Information Processing Systems 6, 1994.

[7] S. Kumar and R. Miikkualainen, " Confidence-based Q-routing: an on-queue adaptive routing algorithm" In Proceedings of Neural Networks in Engineering, 1998.

[8] S. Hoceini, A. Mellouk, Y. Amirat, "K-Shortest Paths Q-Routing: A New QoS Routing Algorithm in Telecommunication Networks" in Lecture Notes in Computer Science, LNCS 3421, Networking - ICN: 4th IEEE International Conference on Networking, ICN April 2005, Pascal Lorenz and Petre Dini (Eds.), Springer-Verlag GmbH, ISBN: 3-540-25338-6, Volume 3421, 2005.

[9] J. Yanxia, N. Ioanis, and G. Pawel, " Multiple path QoS Routing" Proc. Int. Conf. Communications (ICC2001), pp. 2583–2587, June 2001.

[10] S. Hoceini, "Technique of reinforcement learning for an adaptive routing in telecommunication networks with irregular traffic" Phd thesis, A. Mellouk Supervisor's, University of Paris XII-Val de Marne, 2004.

[11] D. Subramanian, P. Druschel, and J. Chen. "Ants and reinforcement learning: A case study in routing in dynamic networks". In Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, volume 2, pages 832-839, 1997.

Abdelhamid Mellouk received his MSc degree from University Es-Senia Oran Algeria in 1989, his PhD degree from University of Paris Sud Orsay, France in 1994. He is a Head of Telecommunication and Network's (TN) Department since 1999 and Vice President of French National Assembly of Heads of TN Depts since 2002. His main research interests include: Multimedia and High Speed Communications, Wireless and Mobile communications, Routing in dynamic traffic network, Quality of service, Traffic engineering and shaping, Networks policy-based management and Real Time Application. He is Senior Member IEEE, a member of IEEE Computer Society and IEEE Communications Society and member of the Internet Society. He is member of international scientific committees of more than 15 international 2005 & 2006 conferences.

Said Hoceini received his MSc degree from University Houari Boumediene Algiers Algeria, his PHD degree from University of Paris XII, France in 2004. His research interests include dynamic Routing, Quality of Service and security vulnerability assessment.
Tel: + 33 (0)1 41 80 73 74 E-mail: hoceini@univ-paris12.fr