

Router CPU Time Management in Cellular IP Networks using GA

Mohammad Anbar

Jawaharlal Nehru University
School of Computer and Systems
Sciences, Jawaharlal Nehru University,
New Delhi, India, 110067

Deo Prakash Vidyarthi

Jawaharlal Nehru University
School of Computer and Systems
Sciences, Jawaharlal Nehru University,
New Delhi, India, 110067

ABSTRACT

Providing good QoS in Cellular IP networks is one of the important issues in order to improve the performance of the Cellular IP network. Resource reservation is one of the methods used in achieving this goal and is proven to be effective. Resources of Cellular IP network that can be reserved are bandwidth, buffer and CPU cycles. Router CPU cycle is the time taken to process a packet before forwarding it to the next router (hop). This paper is proposing a model for CPU cycle optimization of routers for real-time flows in Cellular IP network. The model applies Genetic Algorithm as a soft computing tool to optimize the CPU cycles and reduces the flow processing time at each router in the route taken by a flow. Simulation experiments show the efficacy of the model.

Categories and Subject Descriptors

C.2.6 [Internetworking]: Routers; C.4 [PERFORMANCE OF SYSTEMS]: Modeling techniques; C.1.3 [Other Architecture Styles]: Cellular architecture (e.g. mobile).

General Terms

Algorithms, Management, Performance.

Keywords

Genetic Algorithms, Router CPU, Processing time, Packet Arrival Rate, Packet Processing Rate,

1. INTRODUCTION

As the All-IP networks in which IP is used as a transport protocol are evolving rapidly, real-time applications such as (video, data and voice) need to be supported. Supporting real-time applications in IP networks involves improving the parameters affecting the QoS level in such networks. QoS parameters, in general, include delay, bandwidth, buffer, packets processing time at routers, jitter, etc. Supporting the QoS in any network is the main concern for many researchers and substantive attempts have been made to improve QoS especially in All-IP networks.

Nonetheless, the problem still gives the possibility of further improvement in parameters meeting QoS.

Resource reservation is one of the methods proven to be successful in improving QoS. Resource reservation includes the most important resources in the network such as the bandwidth [1] and buffer [2]. Resource Reservation Protocol (RSVP) is a common protocol for resource reservation [3]. As RSVP depends on signaling (messaging) as the main principle, it may flood the

network with signals. This results in an extra load in the network and therefore wastage of the resources. Having such difficulties, RSVP, can be replaced by other solutions for resource reservation.

Soft computing can be of good help in optimization problems that are not having straightforward solutions [4]. Evolutionary Algorithms including (Genetic Algorithms, Particle Swarm Optimization, Support Vector Machine, etc.) have been applied very frequently for optimization problems. It is the powerful tool used quite often in resource reservation/ resource management also [5].

CPU router cycle is an important resource in communication networks, especially in Cellular IP networks, due to the wireless nature of these networks. CPU at router in Cellular IP networks takes some amount of time to process a packet. This time is calculated as the time delay during routing a flow of packets from a source to a destination. Any router in Cellular IP networks (including base stations as routers) can process a certain number of packets per second. As a rare resource [6], router's CPU cycle can be considered a precious resource to be reserved. When such a router's CPU is loaded according to its policy at admission control, it must consider real-time packets as higher priority packets than the non real-time packets. In other words the router CPU must be reserved for real-time packet processing. The aim of this work is reducing the processing time at each router on the flow path to the final destination.

The rest of this paper is organized as follows: Section 2 explains the routing operation in these networks. The proposed model using Genetic Algorithms has been explained in section 3. The algorithm used in the model is listed in section 4 step by step. Simulation experiments, to prove the performance of the proposed model, have been elaborated in section 5. Section 6 contains the conclusion drawn from the experiments.

2. ROUTING IN CELLULAR IP NETWORKS

Cellular IP network is divided into cells; these cells are controlled by the base stations considered as access points and wireless routers in Cellular IP networks. Gateway connects the network to the Internet. Cellular IP protocol is designed for mobility management at the micro level of mobility. Mobile hosts in Cellular IP networks implement Cellular IP protocol. Many operations take place in this network such as handoff, paging and routing. Structure of Cellular IP networks is shown in figure 1.

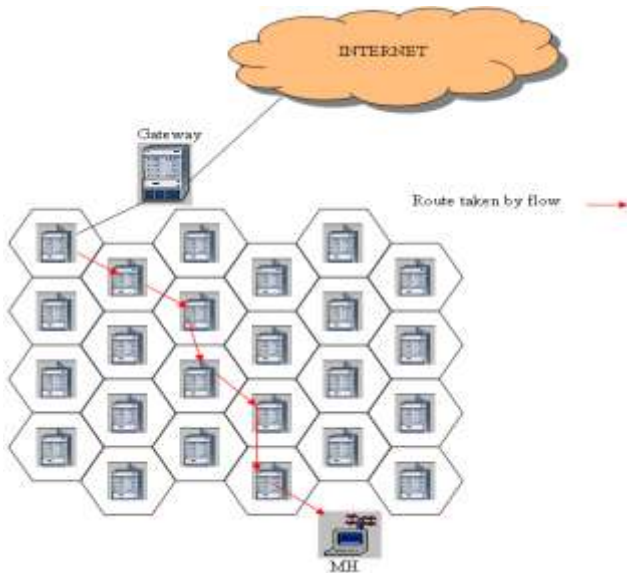


Figure 1. Cellular IP Network

Routing in Cellular IP networks is one of the important operations by which the packets of a flow are forwarded from source to destination. Two types of routing are usually used in communication systems; end-to-end routing and hop-by-hop routing. In general, Hop-by-hop routing is used in mobile communications [7] especially in Cellular IP [8]. Hop-by-hop routing may help in reducing energy consumption and packet processing time [9]. The main principle of hop-by-hop routing is that the packets are forwarded (routed) according to an independent decision taken at the router (base station in case of Cellular IP networks) based on the destination addresses for the incoming flow. Each base station, in Cellular IP networks, maintains a routing cache. Two types of information are stored in the routing cache; the source IP address and the previous neighbor from which the packet reached the current base station. Hop-by-hop routing differs from other types of routing by way of routing tables. In hop-by-hop routing one packet carrying the path to the destination information is enough as entry to the routing table and no need for carrying the full header to the destination by the following packets [7]. Definitely, the route information must be updated through the data packets being sent. As long as the mobile host is sending packets through this route regularly, the routing cache will keep valid routing information. It is to be observed that route in Cellular IP network stays valid for a specific period of time known as route-time-out [8].

Each router in Cellular IP network has a packet processing capacity i.e. each router has a policy to accept or reject a flow if the load exceeds its capacity. Any router can not accept any flow if it is already loaded by the maximum allowed limit as per its policy (capacity) [6]. According to the queuing system theory, a flow must reserve enough router CPU cycles in order to avoid a long queue delay. Real-time flows, as delay sensitive flows, must be provided with the required CPU cycles to ensure that they are processed in a minimum time at the router.

Therefore; admission control, subjected to the wireless router capacity (policy), and minimizing the packets processing time at

the wireless routers in Cellular IP networks can ensure better QoS in this network in sense of time delay for real-time flows.

3. ROUTER'S CPU TIME OPTIMIZATION USING GA

Genetic Algorithm is a computerized search and optimization technique based on the mechanics of natural genetics and natural selection. In general, it is not good at taking larger, potentially huge search space and navigating them looking for optimal solutions which may take huge amount of time. GA is a technique that is applied in such cases to produce sub-optimal results in reasonable amount of time. GA has many good features such as broad applicability, ease of use, and global perspective; therefore GA has been applied to various search and optimization problems in the recent past. Because of their population based approach, GA has also been extended to solve other search and optimization problems such as multi-objective problems and scheduling problems [10]. Population in GA consists of number of individuals which is a solution to the given problem. The individual solution is also called chromosome [11] and consists of many genes, as shown in Figure 2, depending on the type of the problem being addressed.

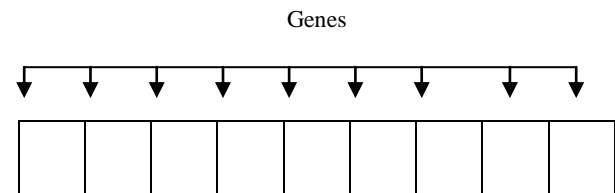


Figure 2. Chromosome structure

Data in the chromosome can be represented either in binary or real values. A pseudo-code of the simple GA is as follows.

```
GA () {
    Create a random population of any size;
    Evaluate the fitness function for each individual in the
    population;
    For number of generations
    {
        Select parents for reproduction;
        Perform crossover ();
        Perform mutation ();
        Evaluate population;
    }
}
```

Some of the functions used in the GA are as follows.

Selection: Through selection operation, good produced offspring (solutions) are selected depending on their fitness value obtained using fitness function for producing more offspring.

Cross-over: In this operation the parts of the two chromosomes are swapped to produce new offspring. Cross-over can be either one site cross-over or multiple sites cross-over. Cross-over operation is done according to a cross-over probability.

Mutation: In mutation the parent chromosome is changed by mutating one gene or more for yielding a new offspring [12]. Mutation operation is done according to mutation probability which is often low.

Packet processing at a router starts when a packet reaches to this router. The time taken by the router to complete processing a packet is T_i . Therefore the time taken to process a flow consists of (N) packets is $(N * T_i)$. Total time taken to process a complete flow in the route, having M routers, is:

$$T = \sum_{i=1}^M N * T_i \quad \dots\dots\dots (1)$$

Few assumptions have been considered in this model listed as below.

Flow is studied in this model rather than considering a packet; therefore the flow is assumed to be having N packets.

Flows, studied in this model, are considered to be real-time flows and for this reason the procedure of admission control is not discussed because they all are considered to be accepted flows by the routers. The main concern of this model is the processing time at the router for each flow.

Each wireless router (base station) in the Cellular IP network has its own capacity (packet processing rate referred as μ_i). Packet arrival rate at each router is different and is referred as λ_i .

Queue is built up until the processing time of each flow consisting of (N) packets is completed.

Based on the queuing system theory the time taken to process a packet by a router [13] is:

$$T_i = \frac{1}{\mu_i} \times \frac{1}{1 - \frac{\lambda_i}{\mu_i}} \quad \dots\dots\dots (2)$$

The model considers routing operation in Cellular IP networks. In figure 1 an example of routing operation is depicted, where a flow (group of packets) passes its way (route) from the moment it enters the network through the Gateway which connects the network to Internet until it reaches the final destination. Each base station forwards the flow based on independent decision, the main principle of Hop-by-Hop routing as discussed earlier. At each router (base station) CPU takes time to process the packets and forward them to the next hop. This model is reduces the total processing time taken on this specific route to process the flow. To achieve this objective the model applies Genetic Algorithm. Each router will be represented by its CPU processing time because this is the parameter to be optimized in this model. Two main parameters that affect the processing time at each router (evident from equation 2) are processing rate of the routers and packet arrival rate. Due to these two parameters, it results in

different processing times at the routers. Therefore, the total time taken to process a flow on a route is different for every new flow. To optimize the processing time at the routers of the flow becomes an optimization problem which is desired to be as low as possible. GA is applied as one of the powerful tested optimization tools to solve such optimization problems. GA, in this model, minimizes the processing time taken to process a flow that passes through number of routers in the way.

Data values used in the gene representation for the chromosome is real and is as follows.

T1	T2	T3	TM
----	----	----	-------	-------	-------	----

Figure 3. Chromosome structure used in this model

Here T_i , as mentioned before, is the processing time (router's CPU time) for a flow at each router in the route. Fitness function used to evaluate the solutions in this model is T, as specified in equation (1).

4. ALGORITHM USED FOR THE MODEL

The algorithm for the router's CPU time minimization using GA is listed as below.

- 1) Enter the number of routers (hops).
- 2) Enter the number of packets in the current flow.
- 3) Consider that packet arrival rate and packet processing rate values in different ranges of values for each experiment.
- 4) Generate the initial population of individuals (solutions), as shown in figure 3, by randomly generating packet arrival rate and packet processing rate for each router within the given range.
- 5) Compute the fitness values (processing time at each router) for the generated individuals (solutions) according to equation (2).
- 6) Compute the total processing time on the route using equation (1).
- 7) Sort the individuals according to their fitness values in ascending manner. Select half of the best individuals.
- 8) Perform crossover.
- 9) Perform mutation.
- 10) Evaluate the new generated offspring using equations (2).
- 11) Sort the new population in ascending manner. Select half of the best individuals.
- 12) Compute the total processing time taken to process the current flow on the route using equation (1).
- 13) Output the value of new total processing time for the current generation.
- 14) Repeat the steps from 8 until 13 for specified number of generations.

When a flow consisting of N packets enters the network through the gateway routed to a correspondent node in the network, it has to pass through many routers on hop-by-hop basis until it reaches to the destination. The algorithm considers the number of routers which a flow will pass through is known and therefore entering number of routers (hops) will be done at the beginning as an input. The experiment using this algorithm is repeated for number of routers 8, 12, 16 and 20 every time. As each flow has different number of packets, the algorithm asks for number of packets per

flow as an input also. The initial population is formed (figure 3) by generating the values of packet arrival rate and packet processing rate for each router in four different ranges for each experiment. This part of algorithm is done because the values of packet arrival rate and packet processing rate are different and the algorithm has to cover most of the possible values. As the objective of the algorithm is to minimize the processing time at the routers, the fitness function considered is the processing time at each router and the total processing time at the entire route consisting of M routers. Once this fitness value is calculated, the individuals are sorted according to their fitness values and half of the best solutions are crossed over in order to generate new offspring which probably have better fitness values (processing time) than the old population. The new generated offspring are also scored against the fitness function and the ones with better values will survive for further crossover. Mutation is done once every five generations in order to avoid local minima. This procedure is repeated number of times till the result converges.

5. SIMULATION EXPERIMENTS

Experiments are conducted with different values of packet processing rates and different values of packet arrival rates with the objective of minimizing the total processing time taken to process a flow consisting of N packets. The experiments have been divided according to the values of packet processing rates and packet arrival rates as follows.

5.1 First set of experiment considers various processing rate of the routers and packet arrival rate at the router. Various graphs below depict the performance for different number of packets per flow with varying number of routers. For all the experiment in this set the input are as follows. Packet arrival rate is between 500 to 1000 packets/sec., Packet processing rate is between 1000 to 1500 packets/sec, and numbers of routers are 8, 12, 16, and 20 for four experiments.

5.1.1 Number of packets per flow is 200

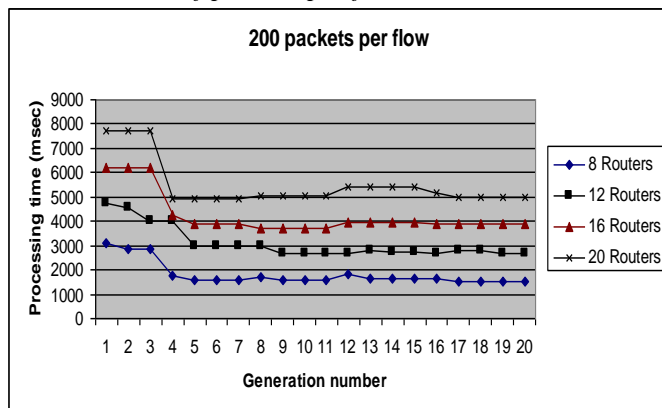


Figure 4. 200 packets per flow

The graph in figure 4 shows good improvement in flow processing time. Total processing time, when number of routers

are 20, before applying GA is almost 8 msec. which drops to 5 msec. after applying GA.

5.1.2 Number of packets per flow is 400

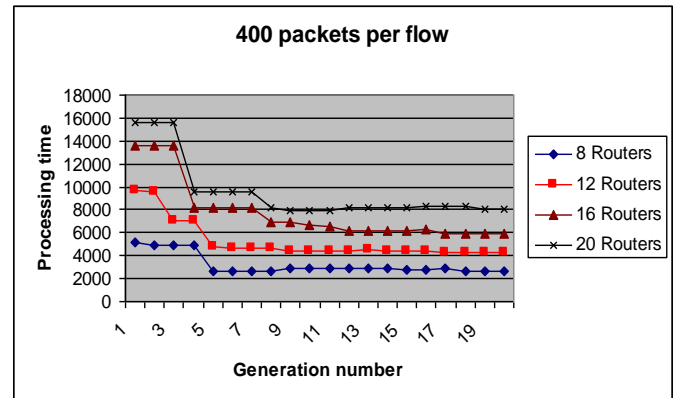


Figure 5. 400 packets per flow

Increasing number of packets per flow leads to increase in the processing time. There is no constant ratio for the reduction in packet processing time as in GA new generated offspring may be better than the parents or it might be worse. For example, the reduction found in figure 4 (for 20 routers) is from 8 msec to 5 msec while the reduction in figure 5 (for 20 routers) is from 16 msec to 8 msec.

5.1.3 Number of packets per flow is 600

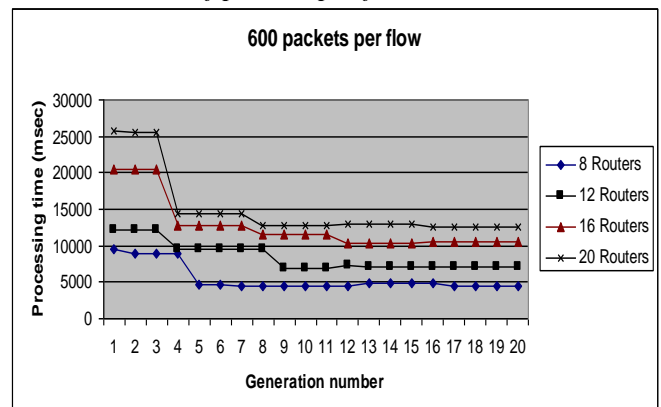


Figure 6. 600 packets per flow

The effect of packet arrival rate and packet processing rate is shown as follows. In figure 6, the packet arrival rate is randomly generated in the range of values 500 up to 1000 packets/sec and the packet processing rate is varied in the range of values 1000 up to 1500 packets/sec. The range of packet processing rate should be higher than packet arrival rate; otherwise the flow will be rejected.

5.1.4 Number of packets per flow is 800

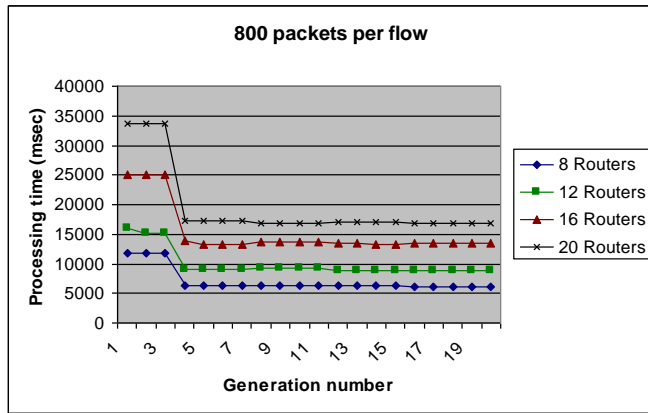


Figure 7. 800 packets per flow

Processing time increases when number of routers on the route taken by a flow increases, where processing time when number of routers 8 is less than the processing time when number of routers is 20 as clear for figures 4, 5, 6 and 7.

5.1.5 The graph below shows the final results of four experiments (5.1.1-5.1.4) for various flows.

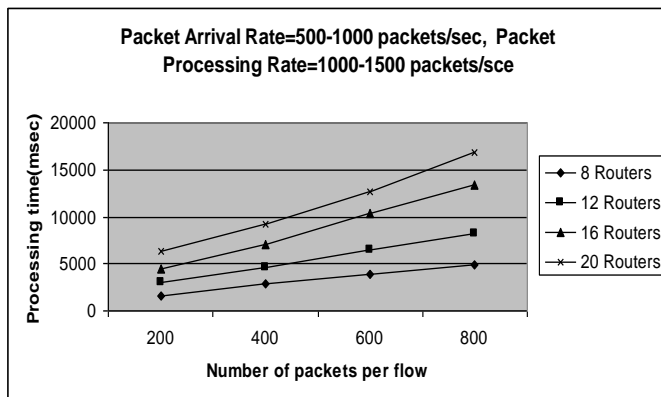


Figure 8. Flow processing time at the first range of values

This figure contains the end values the model obtains after applying GA for the first range of values for packet arrival rate and packet processing rate. It is clear how packet processing time is affected by number of packets per flow and by number of routers (hops) on the route to the final destination.

5.2 In this experiment, the packet arrival and packet processing rates have been changed. Packet arrival rate is between 1000 to 1500 packet/sec, Packet processing rate is between 1500 to 2000 packets/sec, Number of routers are 8, 12, 16, 20 for four experiments and again the number of packets per flow are 200, 400, 600, 800 for four experiments. In the graph only the final results are shown.

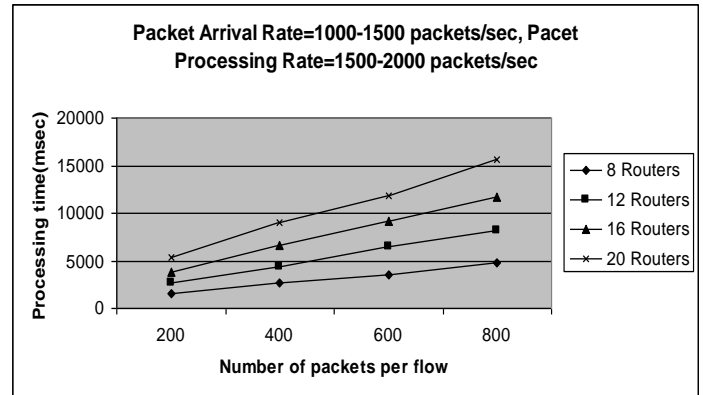


Figure 9. Flow processing time at the second range of values

5.3 Again, a different packet arrival and packet processing rates have been considered.

Packet arrival rate is between 1500 to 2000 packet/sec, Packet processing rate is between 2000 to 2500 packets/sec, Number of routers are 8, 12, 16, 20 for four experiments, Number of packets per flow are 200, 400, 600, 800 for four experiments. The final results are shown in figure 10.

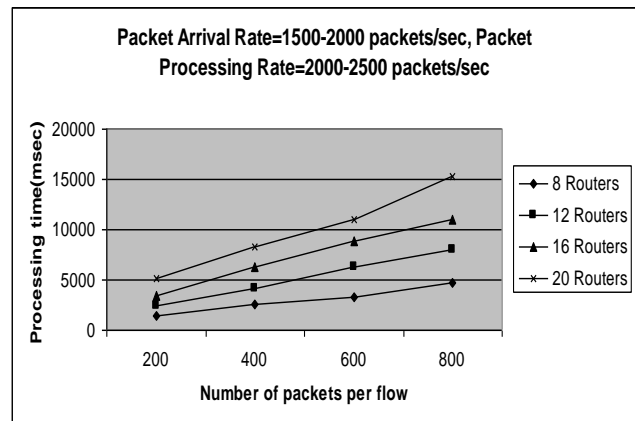


Figure 10. Flow processing time for the third range of values

5.4 Other values of packet arrival rate and packet processing rates have been experimented with Packet arrival rate is between 2000 to 2500 packet/sec, Packet processing rate is between 2500 to 3000 packets/sec, and numbers of routers are 8, 12, 16, 20 for four experiments, Number of packets per flow: 200, 400, 600, 800 for four experiments.

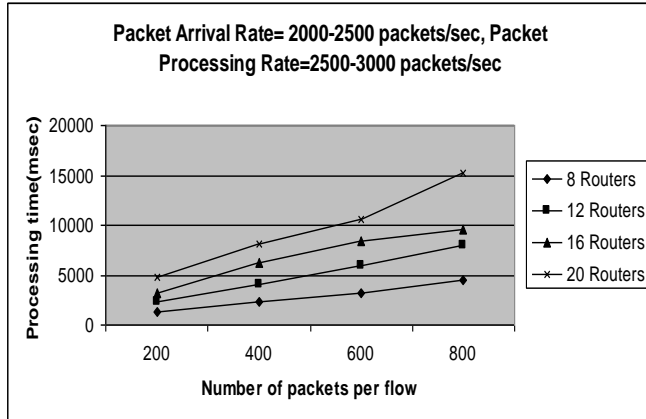


Figure 11. Flow processing time for the fourth range of values

Experiments in figures 8, 9, 10 and 11 are repeated for number of times for the following reason.

Equation (2) can be modified to be in the form

$$T_i = \frac{1}{\mu_i - \lambda_i} \dots\dots\dots (3)$$

Equation (3) shows that the difference between packet processing rate and packet arrival rate determines the processing time. Due to the randomness in generating the values of packet processing rate and packet arrival rate the mentioned difference might be small yielding big processing time and might be big yielding small processing time; therefore in order to show both the cases the experiment is repeated for many times and the results indicate the decrease in processing time when both packet processing time and packet arrival rate is increased. This increment is shown in figures 8, 9, 10 and 11. For instance the processing time for flow when number of routers is 20 and the range of packet arrival rate is between 500 and 1000 packet/sec, packet processing rate is between 1000 and 1500 packets/sec is around 16 msec while the value for the same number of routers and for the range packet arrival rate between 2000 and 2500 packet/sec, packet processing rate between 2500 and 3000 packet/sec is close to 15 msec.

Performing the experiments on various ranges of values for packet processing rate and packet arrival rate reflects the effect of these two values on flow processing time on a route.

5.5 Another set of experiment has been conducted to observe the processing time for various range of packet processing rate and packet arrival rate with different number of packets but fixed number of routers. The figures below depict the result.

5.5.1 Processing time changes with different number of packets in the flow for four ranges of values for packet arrival rate and packet processing rate. Number of routers in the experiment is 20, number of packets: 200, 400, 600, 800 packets per flow.

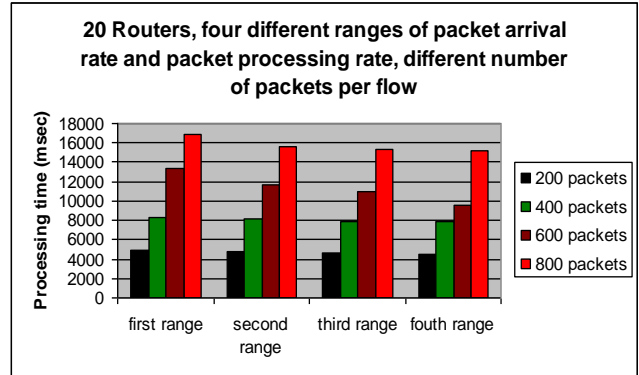


Figure 12. Packet processing time with different values of packet arrival rate and packet processing rate

5.5.2 Same experiment as in (5.5.1) but the representation is different. It is to study the effect of various ranges of values on processing time of the routers.

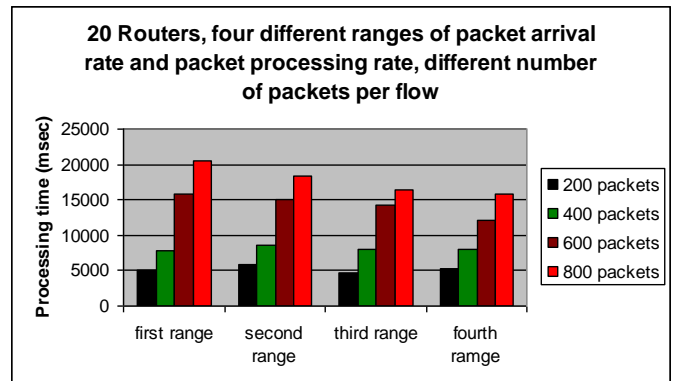


Figure 13. Packet processing time with different values of packet arrival rate and packet processing rate

In order to show the general case in which the randomness in generating values of packet arrival rate and packet processing rate, figure 13 shows that not each and every time the difference between packet arrival rate and packet processing rate is bigger than the previous difference and therefore, the value of processing time is not smaller. This case is clear in the first range of packet arrival rate and packet processing rate where the values of processing time are not decreasing proportionally with increasing packet processing rate and packet arrival rate.

Obtained optimal solutions (processing time) from figure 12 seem to be better than the optimal values obtained from figure 13. Physically, if these values are desired then there is an overhead. This overhead is represented by the time taken in the network to repeat the procedure in order to get these better optimal values.

6. CONCLUSIONS

Packet processing time has an important effect in Cellular IP networks. Time delay occurs due to processing at the router CPU can affect the QoS in the network. Propagation delay is a fixed delay and it is subjected to the link between two routers in Cellular IP network. Processing time is considered itself where the propagation delay is not a part of its value. Processing time in

this experiment means the time taken by a router CPU to forward all the packets in an accepted flow based on admission control discussed earlier. If transmission time is concerned then propagation delay may be added to the total value of processing, transmission and propagation delay, at that time scheduling algorithms must be considered as well.

The proposed model applies Genetic Algorithm as one of the soft computing tools to reduce the packet processing time at the routers. The model assumes that the flows are admitted by the routers as they are real-time flows. Experiments are conducted for different number of packets per flow and for different number of routers (hops). Time reduction is clear from simulation experiments.

7. REFERENCES

- [1] Anbar, M. and Vidyarthi, D.P. 2009. On Demand Bandwidth Reservation for Real-Time Traffic in Cellular IP Network using Particle Swarm Optimization. *International journal of business data communications and networking (IJBDCN)*. 5, 3 (Jul. 2009), 53-66.
- [2] Anbar, M. and Vidyarthi, D.P. 2009. Buffer Management in Cellular IP Network using PSO. *International Journal of Mobile Computing and Multimedia Communications (IJMCMC)*. 1, 3 (Jul. 2009), 78-93.
- [3] Komolafe, O. and Sventek, J. 2005. RSVP performance evaluation using multi-objective evolutionary optimisation. In *Proceedings of IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies (Miami, FL, USA, March 13- 17)*. INFOCOM 2005. IEEE Press, Piscataway, NJ, USA, 2447- 2457. DOI= 10.1109/INFCOM.2005.1498530.
- [4] Huang, C. J, Chuang, Y.T., Lai, W. K., Sun, Y. H., and Guan, C. T. 2007. Adaptive resource reservation schemes for proportional DiffServ enabled fourth-generation mobile communications system. *Computer Communications Journal*. 30, 7 (May 2007), 1613-1623. DOI= 10.1016/j.comcom.2007.01.015.
- [5] Valdez, F., Melin, P. and Mendoza, O. 2008. , A new evolutionary method with fuzzy logic for combining Particle Swarm Optimization and Genetic Algorithms: The case of neural networks optimization. In *Proceedings of IEEE International Joint Conference on Neural Networks (Hong Kong, China, June 1-8, 2008)*. IJCNN 2008. IEEE Press, USA, 1536-1543. DOI= 10.1109/IJCNN.2008.4634000.
- [6] Tanenbaum, A.S. 2004. *Computer Networks*. Fourth Edition, Pearson Education (Indian Branch), New Delhi, India.
- [7] Heimlicher, S. and Karaliopoulos, M. 2007. End-to-end vs. hop-by-hop transport under intermittent connectivity. In *Proceedings of the 1st international conference on Autonomic computing and communication systems (Rome, Italy, Oct. 28-30, 2007)*. Autonomics 2009, ICST Press, Brussels, Belgium, Article No. 20.
- [8] Campbell, A.T., Gomez, J., Kim, S., Valko, A.G., Chieh-Yih, W. and Turanyi, Z.R. 2000. Design, implementation, and evaluation of cellular IP. *IEEE personal communications*, 7, 4 (Aug. 2000), 42-49. DOI= 10.1109/98.863995.
- [9] Sobrinho, J. L. 2002. Algebra and algorithms for QoS path computation and hop-by-hop routing in the internet. *IEEE/ACM Transactions on Networking (TON)*, 10, 4 (Aug. 2002) 541 – 550. DOI= 10.1109/TNET.2002.801397.
- [10] Goldberg D.E. 2005. *Genetic Algorithms in search, optimization, and Machine Learning*. Pearson, Upper Saddle River.
- [11] Khanbary, L.M.O. and Vidyarthi, D.P. 2008. A GA-based effective fault-tolerant model for channel allocation in mobile computing. *IEEE Transactions on Vehicular Technology*, 57, 3(May 2008), 1823-1833. DOI=10.1109/TVT.2007.907311.
- [12] Michalewicz, Z. 1995. *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd revised and extended edition Springer-Verlag.
- [13] Robertazzi, T. G. 2002. *Computer Networks and Systems, Queuing theory and performance evaluation*. Third Edition. Springer-Verlag, Heidelberg, Germany.