The Online Plancement Optimization Design for a Service Network in IP Networks

Phayung Meesad, Pongsarun Boonyopakorn
Faculty of Information Technology
King Mongkut's University of Technology North Bangkok
Bangkok, Thailand
{pym, krunb}@kmutnb.ac.th

ABSTRACT: *Service providers such as an Autonomous System (AS) are controlled by a network overlaid on top of the underlying network which aims to optimize a certain metric value, reduce cost, and increase the performance or efficiency of network traffic flow. Normally, the service gateways are placed on the boundaries of an AS for all inter-domain traffic to pass through to extra domain networks. In cases where the time constraint is crucial as in interactive traffic flow transmissions, the service gateways have more impact on the performance. So that end-to-end paths have to pass, at least, one of the service gateway locations to carry the traffic flows from source to the destination. Traffic Engineering requires efficient tools to optimize network performance and traffic delivery. This paper covers the canalization of performance problems found in IP networks, to maximize the admitted traffic flow transmission without delay violations. The solution approach can be divided into two sub-problems. The first applies the placement algorithms to determine the gateway locations; the second applies the selection algorithms to dynamically route flows via one of the pre-determined gateways in real-time. The results indicate that the placement locations will perform best when the path length to the gateway and the current residual bandwidth are both taken into consideration which has higher priority to the first one.*

Keywords: Traffic Engineering, QoS, Algorithm, GA, PSO

Received: 14 June 2012, Revised 10 August 2012, Accepted 26 August 2012

1. Introduction

The Internet today provides a collection of routers and links managed by Internet Service Providers (ISP). ISP consists of routing paths between any node pair (routers) that limits the throughput achievable between them. Traffic Engineering (TE) implements strategies for a good QoS achieve operational efficiencies and differentiate to their service offerings. TE is defined as that aspect of Internet network engineering dealing with the issue of performance evaluation and performance optimization of operational IP networks. The goal of performance optimization of operational IP networks is accomplished by routing traffic in a way to utilize network resources efficiently and reliably. TE has been used to imply many problems such as load balancing, constraint-based routing, multi-path routing, and fast re-routing. One research area is single-path routing which can be effectively used for maximum utilization of network resources. The various algorithms discussed give solutions for effectively calculating the single-path and way to minimize delay and increase throughput. Especially, these works can be applied to IP networks, then enhance network performance through traffic engineering to meet the QoS requirements. The service gateways have been placed on boundaries of an Autonomous System for all traffic in the inter-domain to pass through. However most ISP have more than one service gateway in the inter-domain AS, but it is possible to service many connections in only one service gateway. Placement service gateways are interesting in how to place the service gateway in the inter-domain to service traffic smoothly and guarantee that traffic can be transmitted from source to destination. To solve this problem most researches in traffic engineering are based on the placement problem. This problem comes from how to place the service gateway to minimize the average length between routes, so that each router must select a gateway with the shortest path route. To guarantee traffic that can be transmit we need to find a worst-case delay in the link capacity to guarantee link capacity of the router which should have reserved bandwidth to service traffic in the AS network, for example to find the bandwidth guarantee capacity link, the QoS is based on the delay guaranteed services. According to [3], an approach of placement has been provided in the way of dividing it into a sub-problem: to find the location gateway and select these service gateways. This paper focuses on selection and location of the service gateway path subjects to find the best location of a service gateway in the AS (offline). The paths that satisfy these constraints are called feasible paths. Since the optimal solution of this type of problem for multiple additive and independent metrics is NP-complete, usually heuristics or approximation algorithms are used. The approach that we have considered to solve the placement problem is an alternative approached which uses artificial intelligent such as GA and PSO as they are more efficient algorithms to locate the number of service gateways. We also improved some parts of these GA and PSO algorithms to increase traffic efficiency. In this case, the selection problem is defined as a Bandwidth Restricted Path (BRP) problem. The algorithms that solve the BRP problem use metric ordering of the Widest-Shortest Path (WSP), Shortest Path First (SPF), and Closest Gateway algorithms. These three algorithms specified for traffic engineering in the case of AS traffic flow have to traverse one or more services gateways before reaching their destinations.

The rest of the paper is organized as follows. In Section 2 we present related work. Section 3 presents the proposed algorithm to solve the problems addressed. In Section 4 we present extensive simulation results that demonstrate the significant performance gain achieved by our approach for real and synthetic topologies and various load settings. Finally, Section 5 concludes the paper.

2. Related Works

2.1 Placement and Selection Algorithms in IP Networks

Many papers are interested in finding the optimal placement of service gateways to increase traffic flow. This criterion is relevant for service providers that must provide guaranteed bandwidth of flows, which have strict QoS constraints. Traffic engineering approaches can also be applied to several other optimization metrics, such as minimizing the delay, minimizing the average packet loss, or minimizing the maximum load. Most papers based on performance average the length between routes [1] – [3]. In [1] studied the performance improvements as the number mirrors increases under different placement algorithm. Some papers covering placement focus on web server replicas [2], they developed placement algorithms for workload information, such as client latency and request rates, to make informed placement decisions. Another researcher such as [3], divided placement problems of IP network services into two sub-problems, find the best location service gateway and select the best service gateway of an AS network service. This algorithm model is used in this case study of traffic engineering.

The path computation algorithm is at the core of QoS routing strategies. Instead of using a shortest path algorithm based on statically configured metrics, as in traditional routing protocols, the algorithm must select several alternative paths that are able to satisfy a set of constraints regarding, for instance, end-to-end delay bounds and bandwidth requirements. In [4] proposed an alternative approach, called two-step restorable QoS routing. In the first step, an active path is found using the WSP routing. In the second step, the corresponding backup path is determined using one of the three variants of SWP routing: basic SWP and approximated SWP.

2.2 Genetic Algorithm Approach

A genetic algorithm approach when designing a network is one of the ultimate solutions because traditional heuristics have limited success. In [5], proposed a different existing optimization algorithm for resource placement in a content delivery network. It is confirmed that the best sub-optimal solution can be obtained by the greedy algorithm. In [6], focus is on the design of reliable computer network topologies. A generalized framework based on genetic algorithms is developed which is applicable to a wide range of network design problems. Researchers in [7], proposed a novel genetic algorithm to design a network with robust fitness function. They presented a strong fitness function developed to solve the network optimization problem which only reduces the number of generations but follows the concept of survival of the fittest. In [8], development of a genetic algorithms is developed which is applicable to a wide range of network design problems. Researchers in [11], proposed a novel genetic algorithm to design a network with robust fitness function. They presented a strong fitness function developed to solve the network optimization problem which only reduces the number of generations but follows the concept of survival of the fittest.

2.3 Particle Swarm Optimization Approach

Many researches have used PSO to optimize solutions for their problems. In [12], proposed a particle swarm optimization algorithm based on fuzzy logic to solve the single source Shortest Path Problem (SPP). The method takes advantage of an efficient encoding mechanism in the PSO so as to include the parameters of the path graph in the representation itself. A particle encoding-decoding scheme has been devised for particle-representation of the SPP parameters, which is free of the previous randomized path construction methods in computational problems, for example the SPP. The search capability of PSO is diversified by hybridizing the PSO with fuzzy logic. The local optimums will not be the point of convergence for the particles and the global optimum will be found in a shorter period of time if the PSO is correctly modified using fuzzy logic rules. Additionally, in order to enhance the search efficiency, the inertia weight whose right values can prevent the search from falling in the trap of local optima, is determined using fuzzy rules. In [13], they applied Bio-inspired and evolutionary optimization methods to design fuzzy logic controllers (FLC) to minimize the steady state error of linear systems. They tested the optimal FLC obtained by the genetic algorithms and applied the PSO to linear systems using benchmark plants. The bio-inspired evolutionary methods are used to find the parameters of each membership function of FLC to obtain the optimal controller. Researchers in [14], described the application of bio-inspired methods as a design to optimize fuzzy logic controllers using genetic algorithms and particle swarm optimization. They also presented results of a genetic algorithm and PSO applied to two linear systems, using two different levels of complexity. The main result showed that the FLC obtained by GA and PSO are stable at less than 10 seconds. On the other hand, the FLCs obtained by PSO are better than the FLCs obtained by GA, because the PSO uses less time consumption when processing and achieves lower overshot, the plots of the results clearly show this difference. Referring to [15], analysis of the simulation results of 3 evolutionary methods qualified them for use in our research, in this case the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and PSO+GA. The conclusion is that the optimization of these 5 mathematical functions, in all cases one can say, that the 3 proposed methods should work correctly and can be applied successfully to this type of problem. The advantage of this method is that it incorporates fuzzy logic to improve the optimization of results when the number of variables is incremented. In [16], presented a hybrid genetic algorithm which combined the particle swarm optimization technique in order to improve the search efficiency of a classical genetic algorithm. There algorithm gave a new crossover operation and a mutation strategy based on the idea of particle swarm optimization. The experiment results showed that there algorithm obtained better results than other competitive algorithms when dealing with average convergence generation and global convergence probability. In [17], presents the application of particle swarm optimization (PSO) based on search algorithms to solve single source Shortest Path Problems (SPP) commonly encountered in graph theories. A new particle encoding/decoding scheme has been devised to represent the SPP parameters as a particle. In order to enhance the search capability of PSO, a selective local search mechanism and periodic velocity re-initialization of particles have also been incorporated.

3. The Proposed of Algorithm to Placement and Selected Service Network

We are interested in finding the optimal placement of service gateways where the optimal shortest paths of each gateway location are calculated by various algorithms. This should search for the optimization of placement in each gateway location following the optimal results. When we maximize the volume of admitted traffic, it raises the upper levels of load imposed on each link. To avoid having to forward states from one service gateway to another, and to avoid packet reordering or route oscillation, we require that the same gateway serve each flow for the entire flow duration. This approach can also be applied to several other optimization metrics, such as minimizing the delay, minimizing the average packet loss, or minimizing the maximum load. This section describes the following placement algorithms, namely, Probabilistic approximation, Hybrid GA, and Hybrid PSO. These three algorithms not only require the knowledge of network topology but also need knowledge of long-term traffic distribution, i.e., the traffic demand matrix.

***Definition* 1:** Let $H \subseteq V$. An instance of 1-hub shortestpath routing is H-limited for a given set of flows F, if only nodes from H serve as hubs for the flows in F. A route in an H-limited 1-hub shortest-path routing is denoted a 1-hub(H) route. The routing domain is represented by a directed graph, $G = (V, E)$, where V is a set of routers and E is a set of directed links. A bidirectional link is represented by two counter edges. Let $u(e)\ e \forall E$ be the bandwidth capacity of link e. Let $F \subseteq V \times V$ be a set of flows and $t(f), f \in F$ be the bandwidth demand of flow f. Let $k \leq |V|$ be the maximum number of service gateways that can be deployed. Each such gateway can be deployed on one of the network routers. The load of a link is the total traffic the link carries divided by its capacity.

The *placement problem* is defined as follows. Find a set $H \subseteq V$, where $|H|$ d $\leq k$, for which there exists an "*H limited*1-*hub shortest-path routing*" of $F' \subseteq F$, such that the load imposed on each link does not exceed 1, and the total bandwidth of *F*' is maximized.

The *selection problem* is defined as follows. Let $H \subseteq V$ be the set of (already placed) service gateways. For each flow $f \in F$, *f* is either assigned a service gateway $h \in H$ or rejected, such that the sum of the bandwidth demands of admitted flows is maximized, and the load imposed on every link does not exceed the threshold *L*. These two problems are NP-complete.

3.1 Offline and Online Placement Problems in Traffic Engineering

The difference between Online and Offline algorithms are the areas of traffic flow which passes through nodes in the AS. When source (ingress node) sends traffic flows to a destination (egress node), source nodes then search for feasible shortest paths to send the traffic flows, then the gateway nodes are selected by offline algorithms. Many researchers proposed several techniques to optimize traffic flows in the AS, we will discussed these techniques next. Online algorithms perform by searching for the optimal path to reach destination nodes between egresses of the AS. There are many problems in networking which apply this technique such as Facility Location problems, k-Center problems, and the Greedy Heuristic algorithm.

3.2 Probabilistic Approximation

The probabilistic approximation algorithm proposed in [3] uses the knowledge of traffic demand matrix to determine the set of gateways that maximizes the total amount of carried traffic. Defined below are the following variables:

X_{if} = 1 if node *i* is assigned as a gateway for flow *f*, and 0 otherwise

X_f = 1 if flow *f* is admitted, and 0 otherwise

h_i = 1 if node *i* is used as a gateway for some flow, and 0 otherwise

T_{fe} = Bandwidth demand of flow *f*

Z_{if} = 1 if link *e* is on the shortest-path of flow *f* to or from node *i*, and 0 otherwise

Then, the placement problem is formulated as follows:

Maximize: $\sum^{f \in F} T_f \,.\, Z_{if}$
Subject to

a) $\sum_i x_{if} \geq X_f \,, \forall_f \in F$

b) $\sum_i x_{if} Z_f^e T_f \leq u(e), \forall_e \in E$

c) $x_{if} \leq h_i \forall_i \in V, \; \forall_f \in E$

d) $\sum_i h_i \leq k_s$

e) $x_f \in \{0, 1\}, X_f \in \{0, 1\}, h_i \in \{0, 1\}, \forall_i \in \{V\}, \forall_f \in \{F\}$

Solving the linear relaxation of the program will yield the non-integer solution of x_{if} and X_f. The solution to the original integer problem is then obtained by rounding x_{if} and X_f to 0 or 1.

3.3 Hybrid Genetic Algorithm

Genetic algorithms are a type of machine learning algorithm. It employs chromosomes in a population to stand for the possible solutions to the problem. As time evolves, these chromosomes are given different chances to reproduce according to their fitness. In the reproduction one chromosome can crossover. After many generations one expects the optimum chromosomes or solutions will emerge from the population. A genetic algorithm and tabu search approach have a number of significant differences, when designing a network this is one of the ultimate solutions because traditional heuristics have limited success.

3.3.1 Objective

Given a network of *N* nodes, each has a value *k*, the *k* value is cost of the shortest path where each node reaches other nodes in the network topology. This can be computed as follows

$$f_x = \sum_{i=0}^{N} C_i \tag{1}$$

where x is the node in the network topology and c is the shortest path cost of each node. We summarize the minimum path length to pick the set of node in network topology to be a service gateway where the group has optimal solutions.

3.3.2 Chromosome Structures

All nodes connected in the network will maintain the link to other nodes. Each link and node is analogous to the chromosome in the genetic algorithm. In this case we considered *N* nodes in a network topology, so that we have *n* genes for each chromosome. The structure of chromosome is considered to be a group as follows

Chromosome_0	11011011010111000100101111… 01001
Chromosome_1	00111010000010001101100010… 11010
Chromosome_2	00111010110110101001101010… 10000
Chromosome_N	01111011100111010101011001… 11001

Figure 1. Binary matrix representation of solution

where element $a_i = 1$ if the node *i* is optimal path length in the network topology, otherwise 0. A set of solutions is randomly produced initially. By defining the neighborhood of a solution as a set of solutions having one-bit different from the original one, the optimal path length of a group of each existing solution are computed.

3.3.3 Crossover

Crossover points are randomly selected and a matching section is specified for swapping the gene of the parents. Beyond the crossover, we again use the cross point index of 20%. This will make our new varieties of generation chromosomes.

3.3.4 Sorting

We re-order the positions of the chromosomes where optimal solution is computed by fitness.

3.3.5 Mutation

The operational rates for mutations are set to 0.025 and two randomly chosen genes are then swapped.

3.3.6 Fitness

In order to calculate the summary of each path length, we need to transform the order chromosome from binary to an integer matrix. The Greedy algorithm (Figure 3) is applied to achieve this task. Path length will be put into a node according to their orders in the order chromosome. We then achieve an optimal group set for service gateways in the network. Figure 2 shows the chromosome conversed to an integer. The number of shortest path cost has been calculated by the greedy algorithm.

Chromosome	0	1	2	3	4	…	N
C_i	C_0	C_1	C_2	C_3	C_4	…	C_N

Figure 2. Order chromosome representations of genes

Each chromosome contains two strings of the same length. The first one is an ordered integer string, where each integer value represents a node ID. The second string is an ordinary integer string, where each integer value represents the number of distance of the node as calculated via greedy. We also apply the Tabu Search (TS) algorithm to solve the optimization of shortest path in an IP network which nodes have chosen to be a service gateways, the basic idea is picking a feasible solution GA as we mention above. The parameters are shown below.

The Greedy algorithm is considered to be a good candidate in solving the traffic engineering problem. This algorithm places nodes on the network iteratively in a greedy fashion. First it exhaustively checks each node in G where G to be a set of gateway,

Population size	5,000
Number of generations	100
Crossover rate	Random between 0.2-0.8
Mutation rate	0.025

Table 1. Parameters of Hybrid GA

to determine the node that best satisfies the optimization condition for a given nodes set. Let $\ell = 0$, after assigning the first gateway to this node, the algorithm looks for an appropriate location for the next gateway, etc. until all |G| gateways are placed. For general ℓ, the algorithm allows for ℓ step(s) backtracking: it checks all the possible combinations of removing ℓ of the already placed gateways and replacing them with new gateways. That is, ℓ number of the already placed gateways can be moved around to optimize the gain. Summarizes of the algorithm follows:

```
if (|M| ≤ $\ell$)
    Choose among all sets M’ with |M’| = |M|
        the set M” with minimal O(M”, p)
    return set M”
end
Set M’ to be an arbitrary set of size $\ell$
while (|M’| < |M|)
    Among all sets X of $\ell$ elements in M’
        and among all sets Y of $\ell + 1$ elements
        in V – M’ + X, choose the sets X, Y
        with minimal O(M’ – X + Y, p)
    M’ M’ – X + Y
end
return set M’
```

Figure 3. Greedy Algorithm

We apply TS algorithm to solve the optimization of shortest path in IP network which nodes have chosen to be a service gateways, the basic idea is picking a feasible solution GA as we mention above. We then, select the best result. In order to design a TS algorithm, the definitions of three problem elements are as follows [18] [19].

- Initial solution, this can be found in the best optimized values from the GA.
- Cost function to evaluate the solutions generated by the algorithm.
- Perturbation procedure to generate new solutions from the current one.

3.4 Particle Swarm Optimization (PSO)

The first one is the best solution called P_b. The second one is tracked by the particle swarm optimizer to find the best value. Any other particles in the population left over are called P_g. Finally, each particle updates its velocity and positions using the following equations:

$$v(k+1) = w \,.\, v(k) + c_1 \,.\, rand() \,.\, (p_b - p(k)) + c_2 \,.\, (g_b - p(k)) \tag{2}$$

$$p(k+1) = p(k) + v(k+1) \tag{3}$$

where v is the particle’s velocity, p is the current particle’s position, *rand*() is a random number between [0, 1], c_1 and c_2 are learning factors. The inertia weight w is employed to control the impact of the previous history of velocities on the current velocity, thus to influence the tradeoff between global and local exploration abilities of the flying points. In stochastic traffic networks, each particle is defined as a sequence of vertexes that represent a valid path, the fitness function is the cost of the path according to the cost of the edges. The approach of this paper is to adapt PSO and FLC to search for the optimized location in an IP network to maximize the admitted traffic load for a service gateway and guarantee the load imposed on each link which can then admit new traffic flow. Firstly, look at the method of encoding with GA-based methods [12].

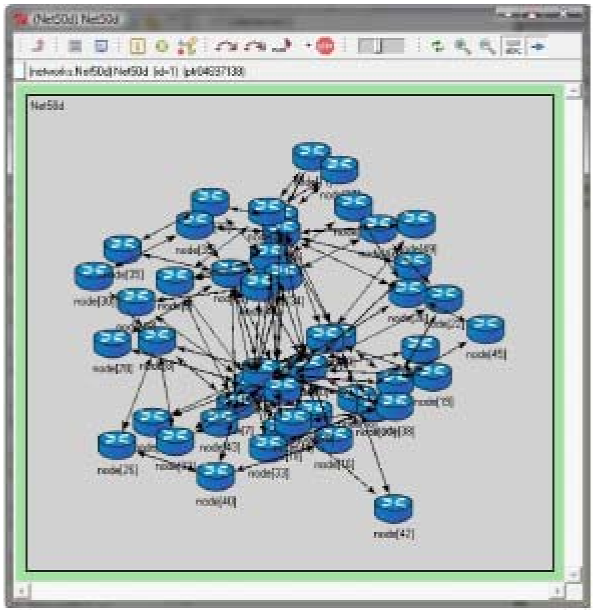

Figure 4. Random network topology 50-nodes

In figure 4, a 50-node random network is depicted showing the shortest path problem solution method. The described encoding scheme is shown in figure 2 C_0, C_1,…,C_n are the priorities of the nodes 1, 2, …, 5 respectively where the node priorities are computed by the greedy algorithm.

Fuzzy Logic Control is a process of formulating the mapping from a given input to an output using fuzzy logic. There are two types of fuzzy logic control, Mamdani-type and Sugeno-type. They are similar to Mamdani-type and Sugeno-type which are used with three membership functions for each input, "*Negative, Zero, and Positive*" [13], and one output defined with constant values. The FLC membership functions are shown in Figure 5.

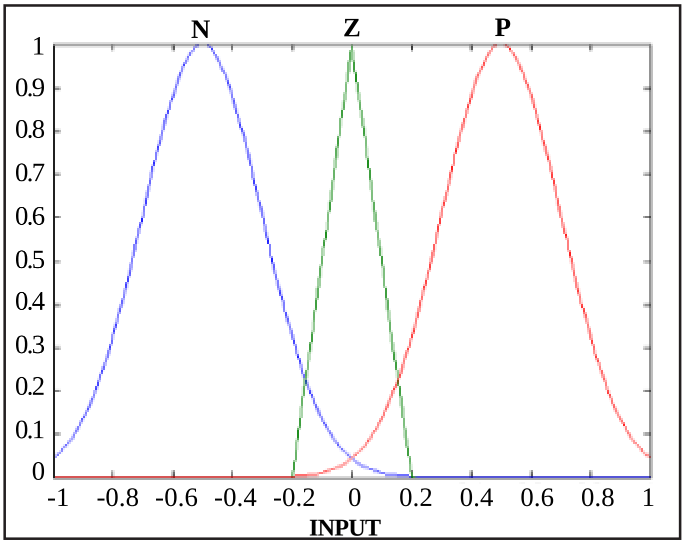

Figure 5. Gaussian and Triangular membership functions

	Negative	**Zero**	**Positive**
Negative	Negative	Negative	Zero
Zero	Negative	Zero	Positive
Positive	Zero	Positive	Positive

Table 2. Shows Fuzzy Rules Of The Sugeno-Type Flc, [13]

The Fuzzy-based method used in this paper was proposed by [12], a variable P (the priorities set value of group optimal nodes) which indicates the fitness level of a particle at each iteration. P_{min} is the best fitness then P_{max} is the worst fitness. In [12], the normalized value P in the interval [0, 1] is obtained using the three variables: P, d_1, and d_2 where d_1 and d_2 represent distance between current positions of particle which are local best and global best respectively. Moreover, output of the fuzzy sets are also applied as w in (2).

$$P = \frac{P - P_{min}}{P_{max} - P_{min}} \tag{4}$$

	Inputs			**Output**
Rule	***d*1**	***d*2**	***P***	***w***
1	Min	Min	Min	Min
2	Min	Min	Mid	Max
3	Min	Mid	Mid	Max
4	Max	Max	Max	Max

Table 3. Shows The Number Of Rules Used In This System

The description of table 3 is as follows [13]

- **Rule 1:** if d_1, d_2, and P are min, the particle is close to the optimum and the fitness is acceptable; therefore w is given a low value so that the search continues around the global optimum.
- **Rule 2:** if d_1 and d_2 are min, but P is mid, it means that the particle is close to the optimum, but the fitness is not acceptable (local optimum); therefore w is given a max value to increase the particle's velocity and change its position.
- **Rule 3:** if d_1 is min, d_2 is mid, and P is mid, it means that the particle is close to the local optimum but not close to the global optimum; therefore w is given a max value to increase the particle's velocity and change its position
- **Rule 4:** if both d_1 and d_2 are max; the particle's velocity must increase; therefore w is given max value.

Once the gateway locations are determined from the placement algorithm, the placement algorithm dynamically routes (micro) flows through one of the gateways from source to destination. An arriving flow will be rejected if the path chosen by the algorithm has not enough residual link capacities to support the flow. The following selection algorithms, namely, Widest-Shortest Path (WSP), Shortest Path First (SPF), and Closest Gateway algorithms are used to select the service gateway from the previous solution. The inertia weight w when updating can be seen in the flow chart below.

3.5 Widest-Shortest Path (WSP)

The Widest-Shortest Path algorithms will select the shortest path that is a feasible path according to the bandwidth constraint of flows. The main metric considered in WSP algorithms is the number of hops, and the second metric is available bandwidth. The algorithm proceeds as follows. In the first stage all existing shortest paths between each source and all destinations in the network are computed. In the second stage, bandwidth is used to break ties among paths that have the same number of hops,

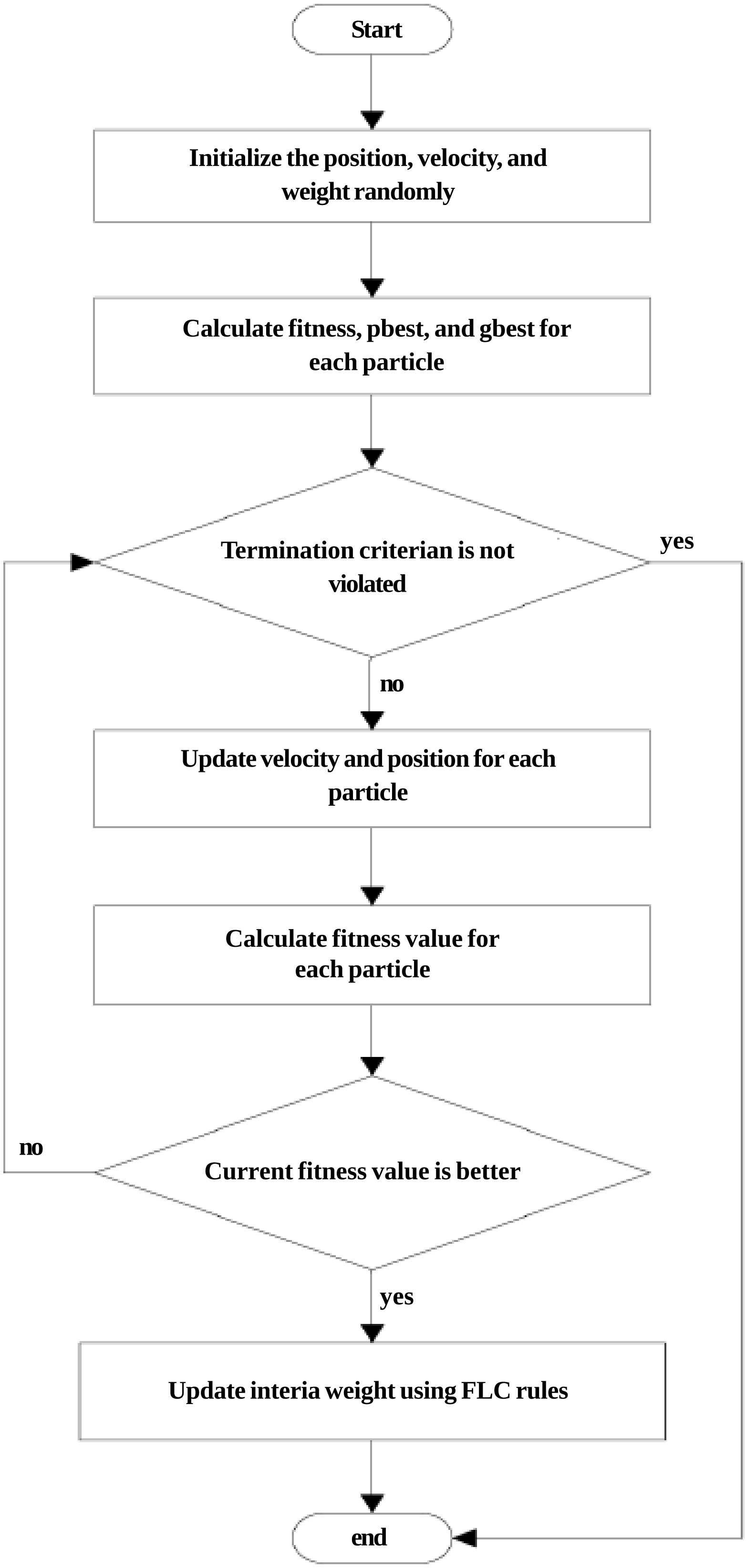

Figure 6. A flow chart of algorithm steps

and it selects the path that has the highest amount of available bandwidth. WSP can be computed by modified versions of Bellman-Ford or Dijkstra algorithms [4].

3.6 Shortest Path First (SPF)

A flow is routed via a gateway with least-hop path between source and destination [3].

3.7 Closest Gateway algorithms

A flow is routed to a gateway close (least-hop) to the source. Note that both shortest path and closet-gateway algorithms do not consider the current link loads. However, the shortest path algorithm tends to consume less network resources [3].

4. Simulation Results

This section presents simulation results for combinations of selection algorithms and placement algorithms in the previous sections. The performance metric is the total number of admitted traffic flows. The simulation is carried out in OMNET++[1]. The network topology is a 50-node router-level AS topology, as shown in Figure 4, which is generated from the Barabasi Albert model by using the PSGen simulator[2], this model suggests two possible causes for the emergence of a power law in the frequency of out degrees in network topologies: incremental growth and preferential connectivity. Incremental growth refers to growing networks that are formed by the continual addition of new nodes, and thus the gradual increase in the size of the network. Preferential connectivity refers to the tendency of a new node to connect to existing nodes that are highly connected or popular.

RouterBarabasi Albert interconnects the nodes according to the incremental growth approach. When a node *i* joins the network, the probability that it connects to a node *j* already belonging to the network is given by:

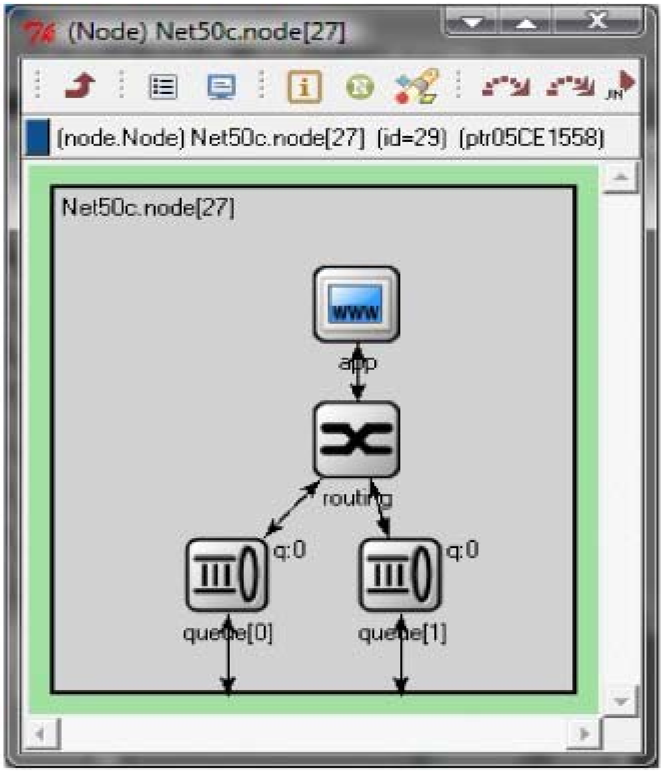

Figure 7. Node component designs

---

[1] http://www.omnetpp.org

[2] http://www1.inf.tu-dresden.de/~rf913578/psgen.html

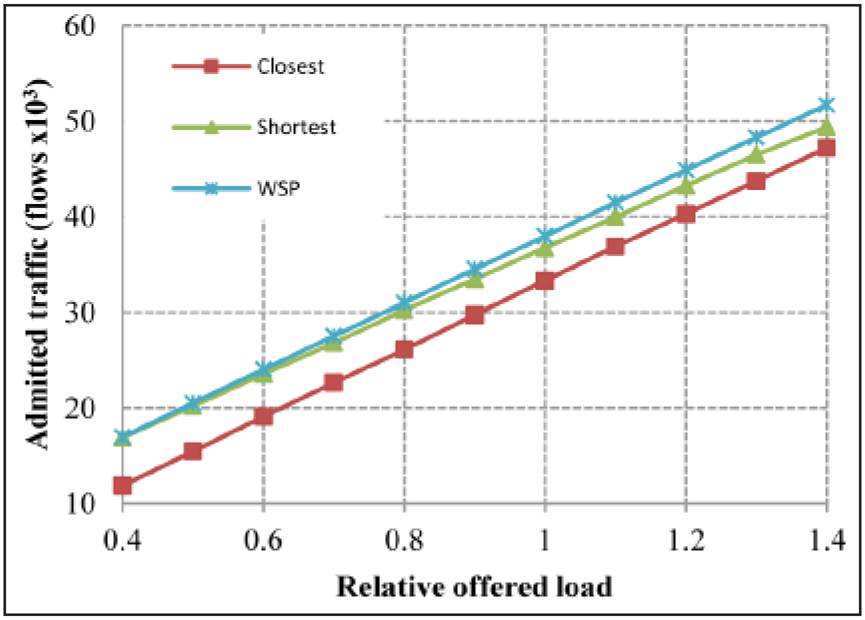

Figure 8. Admitted traffic under Probabilistic Approximation algorithm

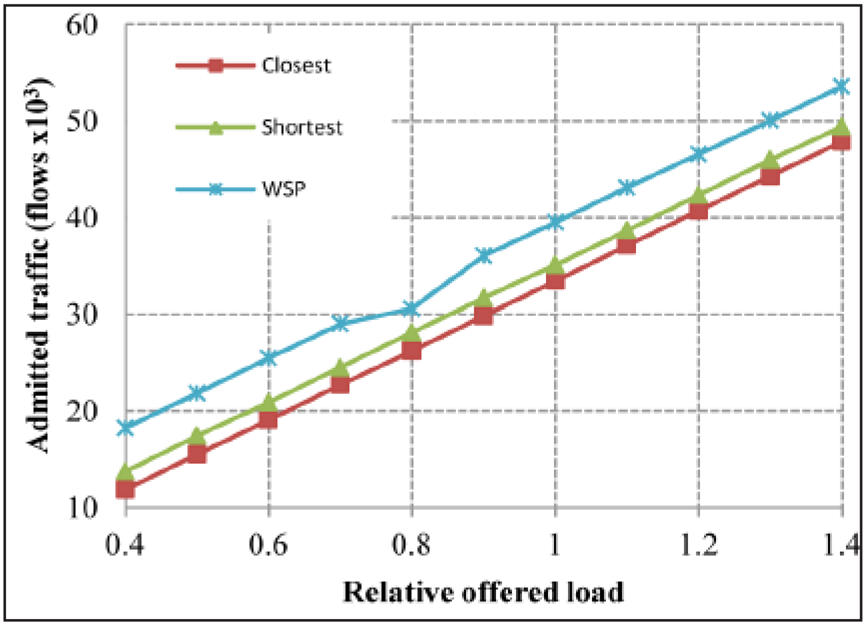

Figure 9. Admitted traffic under Hybrid GA algorithm

$$P(i,j) = \frac{d_j}{\sum_{k \in V} d_k} \tag{5}$$

where d_j is the degree of the target node, V is the set of nodes that have joined the network and $\sum_{k \in V} d_k$ is the sum of outdegrees of all nodes that previously joined the network. Ten out of 50 nodes are to be chosen as the service gateway. The link capacity is drawn from a discrete uniform distribution in the range 100 - 500 Mbps. The traffic demand matrix is synthetic, whereby the average traffic rate of the aggregated flow between each node pair is symmetric and uniformly distributed between 1 and 5 Mbps. In the simulation, traffic flows are generated between all the node pairs according to the Poisson process with rate

Offered Load Parameter	
Link Capacity	Discrete Uniform (100-500Mbps)
Average Traffic Demand Between Node Pair	Continuous Uniform (1-5Mbps)
Average Flow holding Time (1/mu)	300 second
Average Flow Bandwidth	100 kbps
Scale of lambda's	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0

Table 4. Offered Load Parameters

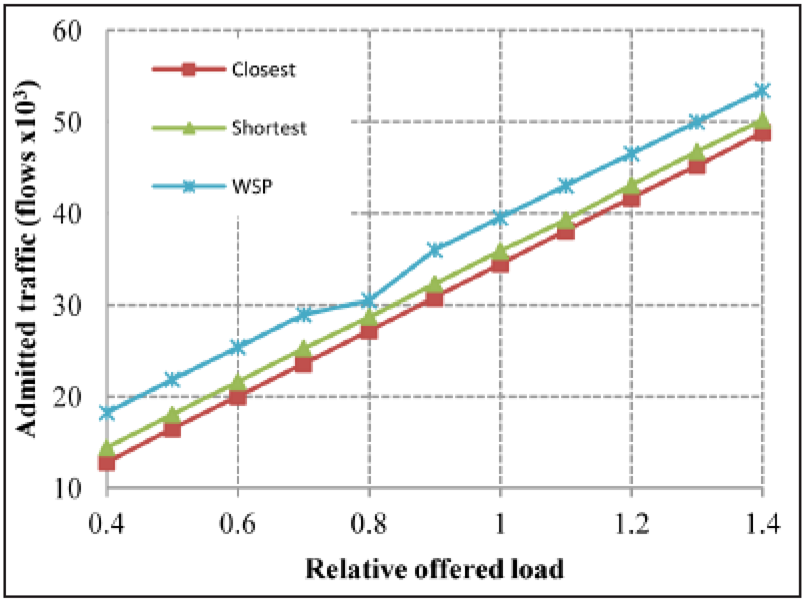

Figure 10. Admitted traffic under Hybrid PSO algorithm

λ_{ij}, and each flow has an exponentially distributed holding time with mean $1/\mu_{ij}$ and requires a constant bandwidth of *r* kbps also the holding time for each flow are 300 seconds. The admitted traffic demand can be calculated as follows.

$$Traffic\ demand = (\lambda/\mu).Traffic\ flow\ bandwidth \qquad (6)$$

The communications between OMNeT++ modules are made via message exchange. The simulation area in figure 4 consists of 50-nodes. Each node is a compound as shown in figure 7 which importantly encapsulates the following modules:

- Application Module
- Routing Module
- Queue Module

From the results of figure 8 - 10, we see that the gateway nodes with Probabilistic, Hybrid GA, and Hybrid PSO are similar node because this scheme considers the link bandwidth and path length where connected to each node. Note, this may result in a bottleneck when many traffic flows arrive at the gateway node. The results from the selection algorithms are then combined with the placement algorithms. Firstly, we cover the selection algorithms under the probability approximation algorithm, we then look at the results with other placements.

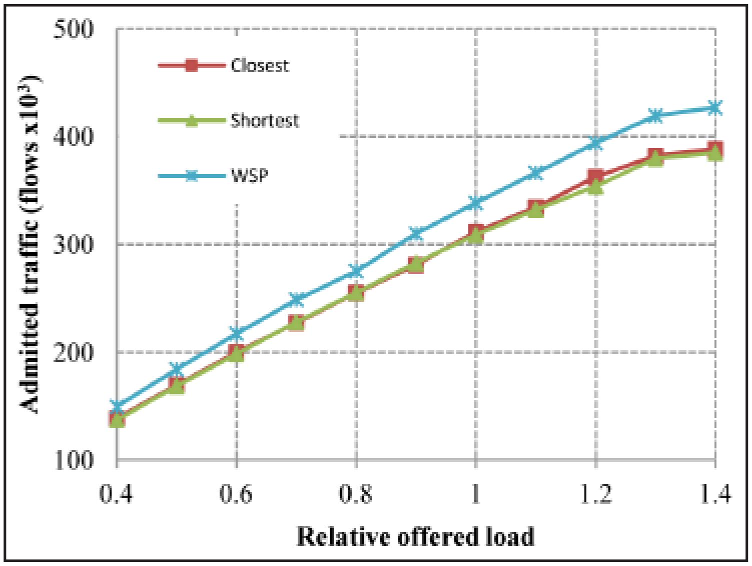

Figure 11. Admitted traffic under Probabilistic Approximation algorithm and 5% offered load

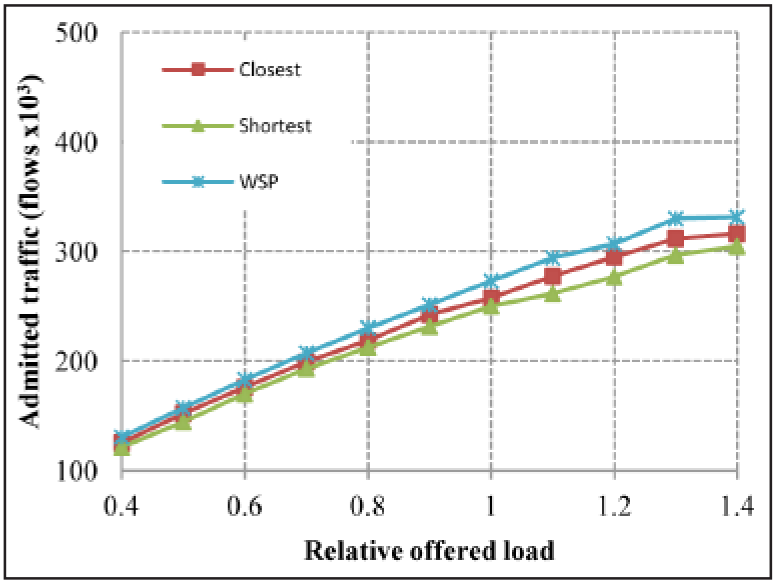

Figure 12. Admitted traffic under Hybrid GA and 5% offered load

From the results one can see that the closest gateway algorithm performs the worst out of all three algorithms followed by the shortest algorithm which was similar in performance. The reason for this is that both schemes select only one gateway for each source destination pair. This leads to poor load balancing as clearly seen in the above graphs. The strongest algorithm from the three tested was WSP. Figure 11 – 13 shows the effect of the traffic load via adding more offered load by 5% to the network. The result shows that the admitted traffic increased the Probabilistic Approximation serving traffic flow more than the others.

Next, figure 14 shows the results of the shortest path in the hold network where the shortest path algorithm worked at the start after that, WSP reduced the shortest path by dynamic change when the simulation was running because the WSP tries to balance the path in the network.

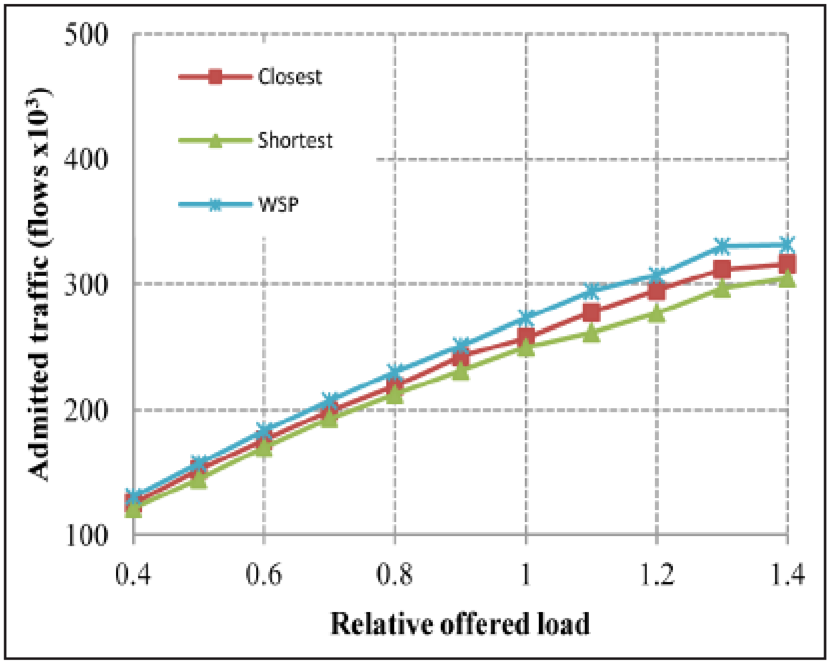

Figure 13. Admitted traffic under Hybrid PSO and 5% offered load

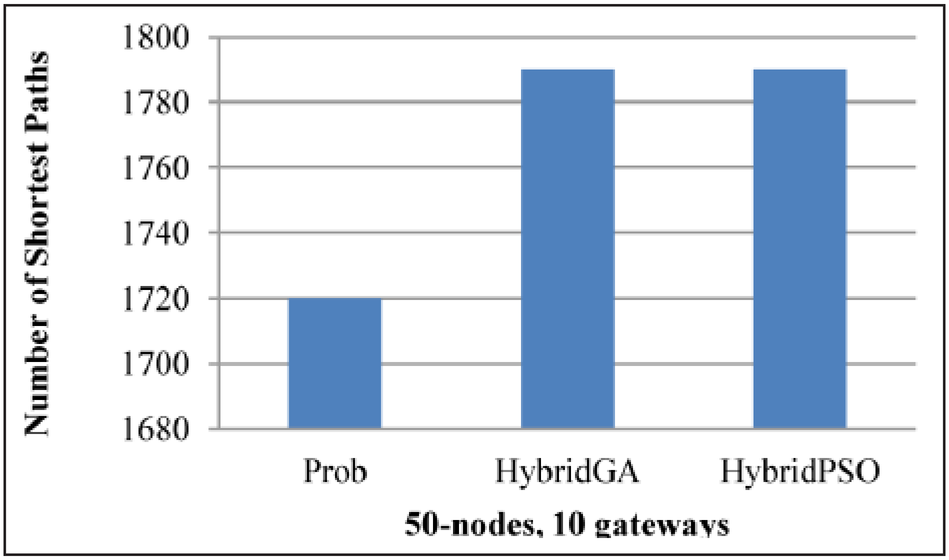

Figure 14. The shortest paths with ten-out gateways and 50-nodes topology

5. Conclusions

We present comparative performance evaluation among various placement and selection algorithms to solve the service gateway problem. Extensive simulation studies have been carried out. It has been discovered that the knowledge of long term distribution can be helpful in determining the gateway locations. However, when it comes to selecting the gateway to service flows, the current network condition must be taken into account. We have shown that even simple approaches like selecting a shortest path with largest residual bandwidth or a path with the largest residual bandwidth provides satisfactory performance. The use of probabilistic approaches to select a gateway and accept flows may lead to a performance penalty if the corresponding probabilities are derived from the static information.

References

[1] Jamin, S., et. al. (2001). Constrained mirror placement on the Internet, *In*: Proceedings of IEEE INFOCOM, p. 31–40.

[2] Qiu, L., Padmanabhan, V., Voelker, G. (2001). On the placement of web server replicas, *In*: Proceedings of IEEE INFOCOM, 3, 1587–1596.

[3] Cohen, R., Nakibly, G. (2007). A Traffic Engineering Approach for Placement and Selection of Network Services, *In*: Proceedings of IEEE INFOCOM, p. 1793–1801.

[4] Ji Li, Yeung K. L. (2005). A Novel Two-Step Approach to Restorable Dynamic QoS Routing, *Journal of Lightwave Technology*, 23 (11).

[5] Yin, J. J., Tang, W. K.S. (2005). Resource Placement in Content Delivery Networks Using Hybrid Genetic Algorithm, 3rd IEEE International Conference on Industrial Informatics (INDIN), p.281-285.

[6] Kumar, A., Pathak, R. M., Gupta, M. C., Gupta, Y. P. (1993). Genetic Algorithm Based Approach for Designing Computer Network Topology, *In*: Proceedings of the 993 ACM conference on Computer science, p.358-365.

[7] Kumar, A., Jani, N. N. (2010). A Novel Genetic Algorithm Approach for Network Design with Robust Fitness Function, *International Journal of Computer Theory and Engineering*, 2 (3) 459-465, June.

[8] Nagib, G., Wahied, G. (2010). Network Routing Protocol Using Genetic Algorithms, *International Journal of Electical & Computer Sciences* IJECS-IJENS. 10 (2) 40-44, March.

[9] Yin, J. J., Tang, W. K.S. (2005). Resource Placement in Content Delivery Networks Using Hybrid Genetic Algorithm, 3rd IEEE International Conference on Industrial Informatics (INDIN), p.281-285.

[10] Kumar, A., Pathak, R. M., Gupta, M. C., Gupta, Y. P. (1993). Genetic Algorithm Based Approach for Designing Computer Network Topology, *In*: Proceedings of the 993 ACM conference on Computer science, p.358-365.

[11] Kumar, A., Jani, N. N. (2010). A Novel Genetic Algorithm Approach for Network Design with Robust Fitness Function, *International Journal of Computer Theory and Engineering*, 2 (3) 459-465, June.

[12] Ghanizadeh, A., Sinaie, S., Abarghouei, A. A., Shamsuddin, S. M. (2010). A Fuzzy-Particle Swarm Optimization Based Algorithm for Solving Shortest Path Problem, *Computer Engineering and Technology* (ICCET), 6, 404-408, April.

[13] Martinez, R., Rodriguez, A., Castillo, O., Aguilar, L.T. Fuzzy Logic Controllers Optimization Using Genetic Algoritms and Particle Swarm Optimization, *In*: Proceedings of the 9th Mexican International Conference on Artificial Intelligence, p. 475-486.

[14] Antonio, R. M., Castillo, O., Aguilar L.T. (2010). Type-2 Fuzzy Logic Controllers Optimization using Genetic Algoritms and Particle Swarm Optimization, *IEEE International Conference on Granular Computing*, p. 724-727.

[15] Valdez, F., Melin, P., Castillo, O. (2009). Evolutionary Method Combining Particle Swarm Optimization and Genetic Algorithms using Fuzzy Logic for Decision Making, Fuzzy-IEEE, p. 2114-2119.

[16] Zhang, G., Dou, M., Wang, S. (2009). Hybrid Genetic Algorithm with Particle Swarm Optimization Technique, *International Conference on Computational Intelligence and Security*, p. 103-106.

[17] Mohemmed, A. W., Sahoo, N. C. (2007). Particle Swarm Optimization Combined with Local Search and Velocity Re-initialization for Shortest Path Computation in Networks, *In*: Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS2007).

[18] Josue, K., Nicolas, P., Maurice, G., Emmanuel, D. Routing Foreseeable Lightpath Demands Using a Tabu Search Meta-heuristic, *Global Telecommunications Conference* (GLOBECOM)' 2 (3) 28 3-2807, November 22.

[19] Anant, O., Wichai, S., Boonruang, M., Thanatchai, K. (2009). Tabu Search Approach to Solve Routing Issues in *Communication Networks,World Academy of Science, Engineering and Technology* 53, 1174-1177.