A Novel PSO Based Algorithm Approach for the cMTS to Improve QoS in Next Generation Networks

Gia Nhu Nguyen, Dac-Nhuong Le, Nguyen Dang Le, and Vinh Trong Le

Abstract—**In this paper, we propose an effective Particle Swarm Optimization (PSO) algorithm to solving the capacitated minimum spanning tree (cMTS) problem to improve Quality of Service (QoS) in Next Generation Network (NGN). To improving QoS of communication network with considering the network provisioning capability and dynamic environment, we formulate this problem with minimizing the communication cost (as a kind of performance measures for network's QoS). We calculate the fitness value of each scheme and update them step by step with the best method to quickly find good approximate solutions of cMST problems. Numerical experiments show that our algorithm proposed have achieved much better than recent researches.**

Index Terms—**Capacitated minimum spanning tree, communication network, quality of service, next generation network, particle swarm optimization.**

I. Introduction

In Next Generation Network (NGN), the backbone of the overall network architecture will be IP network, supporting different access network technologies such as wireless Local Area Network (WLAN), UMTS Terrestrial Radio Access Network (UTRAN), and WiMax. Moreover, this integrated wireless system, will have to handle diverse types of traffics: data traffics (e.g. web browsing, e-mail, ftp), voice traffic (e.g. voIP), and multimedia traffics (e.g. video conferencing, online TV, online games), etc...NGN will provide advanced services, such as Quality of Service (QoS) guarantees, to users and their applications.

However, current Internet routing protocols such as Open Shortest Path First (OSPF), Routing Information Protocol (RIP), and Border Gateway Protocol (BGP) are called "*best-effort*" routing protocols, which means it will try its best to forward user traffic, but can provide no guarantees regarding loss rate, bandwidth, delay, delay jitter, etc. It's intolerable for NGN services, for example video-conferencing and video on-demand, which require high bandwidth, low delay, and low delay jitter. And provide the different type of network services at the same time is very difficult. Thus, the study of Quality-of-Service (QoS) is very important nowadays [1]. To provide QoS in NGN, many techniques have been proposed and studied, including Integrated Services [2], Differential Services [3], MultiProtocol Label Switching (MPLS) [4], Traffic Engineering and QoS-based Routing [1]. And most problems can be formulated as the optimization models, such as the network reliability optimization model, shortest path routing model and constrained minimum spanning tree (MTS) model etc.

In [5], Lin Lin *et al.* focus on the network topological design for providing NGN's QoS. The authors formulated the problem as a extended capacitated minimum spanning tree (cMST) problem, which the objective is minimizing the communication cost (defined as a kind of performance measures for NGN's QoS) with considering the following constraint:

- Consider the capabilities of the network.
- Define different priority for different types of services.
- Dynamic environment.

As we know, this cMST is NP-hard problem. In addition, the complex structures, complex constraints of this problem to be handled simultaneously, which make the problem intractable to traditional approaches. There are many Evolutionary Algorithms (EAs) have been successfully applied to solve constrained spanning tree problems of the real-life instances; and also have been used extensively in a wide variety of communication network design problems. In [6], the author used traditional Prim's algorithm (without considering the capacity constraint) to solving MST. G. Raidl and B.A. Julstrom in [7] proposed Edge Sets: An Effective Evolutionary Coding of Spanning Trees. Zhou and Gen presented a note on genetic algorithm approach to the degree-constrained spanning tree problems in [8],[9],[22]. The authors in [5] proposed a PrimPred-based Evolution Algorithm to solving cMTS. They adopted PrimPred-based encoding, Prim-based crossover, LowestCost mutation, Immigration operators and a parameter auto-tuning strategy.

In the latest paper [10],[11], we have introduced two algorithms based on Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) to solving the optimal communication spanning tree (OCST) problem finds a spanning tree that connects all node satisfies their communication requirements for a minimum total cost.

In this paper, we proposed a new PSO algorithm approach to solving NGN's QoS problem. Numerical experiments with various scales of communication network problems show the effectiveness and the efficiency of our approach by comparing with the recent researches. The experimental results show that our proposed algorithms have achieved much better than previous algorithms.

The rest of this paper is organized as follows: Section II presents the problem formulation the capacitated minimal

Manuscript received November 9, 2012; revised January12, 2013.

Gia Nhu Nguyen is with Duy Tan University, Danang, Vietnam (e-mail: Nguyengianhu@duytan.edu.vn).

Dac-Nhuong Le is with Faculty of Information Technology, Haiphong University, Vietnam (e-mail: Nhuongld@ hus.edu.vn).

Nguyen Dang Le is with Faculty of Trainning, Haiphong University, Vietnam (e-mail: Nguyenld@ hus.edu.vn).

Vinh Trong Le is with Faculty of Mathematics, Mechanics and Informatics, Hanoi University of Science, Vietnam National University (e-mail: Vinhlt@vnu.edu.vn).

DOI: 10.7763/IJFCC.2013.V2.196

spanning tree. Section III summarized the several kinds of classification of encoding methods for developing an EA to network design problems. Section IV presents our new algorithm for the capacitated minimal spanning tree based on PSO algorithm. Section V presents our simulation and analysis results, and finally, Section VI concludes the paper.

TABLE I: THE NETWORK DATA SETS OF 12 NODES AND 40 EDGES

k	Edge (i, j)	Weight w_{ij}
1	(1, 2)	35
2	(1, 3)	23
3	(1, 4)	26
4	(1, 5)	29
5	(1, 6)	52
6	(2, 3)	34
7	(2, 4)	23
8	(2, 5)	68
9	(2, 6)	42
10	(3, 4)	23
11	(3, 7)	51
12	(3, 8)	23
13	(3, 9)	64
14	(3, 10)	28
15	(4, 5)	54
16	(4, 7)	24
17	(4, 8)	47
19	(4, 10)	24
20	(5, 6)	56
21	(5,7)	26
22	(5, 8)	35
23	(5, 9)	63
24	(5, 10)	23
25	(6, 7)	27
26	(6, 8)	29
27	(6, 9)	65
28	(6, 10)	24
29	(7, 8)	38
30	(7, 11)	52
31	(7, 12)	41
32	(8, 9)	62
33	(8, 11)	26
34	(8, 12)	30
35	(9, 10)	47
36	(9, 11)	68
37	(9, 12)	33
38	(10, 11)	42
39	(10, 12)	26
40	(11, 12)	51

II. PROBLEM FORMULATION

Following [5], The communication network is modeled using an edge-weighted undirected graph $G=(V, E, Q, U)$ with n nodes and m edges.

Fig. 1 presents a simple network with 12 nodes and 40 edges.

The network data sets of 12 nodes and 40 edges defined in Table I below:

The capacitated minimal spanning tree problem can be defined as follows:

Indices

- $i, j, k=1, 2, \ldots, n$, is index of node
- $l=1..L$ is index of service type

Parameters

- $n=|V|$ is number of nodes
- $m=|E|$ is number of edges
- $q_{st} \in Q$ is requirement of type l form source node s to sink node t.
- $u_{ij} \in U$ is capacity of edge (i, j)
- $w_l \in W$ is weight (*priority*) of type l communication service
- $d_{ij} \in D$ is delay of edge (i, j) (or defined as a kind of performance measures for NGN's QoS), where

$$d_{ij} = \sum_l w_l G\left(q_{ij}^l - u_{ij}\right) \tag{1}$$

- $G\left(q_{ij}^l - u_{ij}\right)$ is function for delay definition of service type l

Decision variables

- y_{ij}: the amount of requirement through arc(i, j)
- x_{ij}:: 0-1 decision variable

Mathematically, the problem is reformulated as a capacitated Minimal Spanning Tree (cMST) model is as follows:

$$\min f(x) = \sum_{(i,j)\in E} \left(\sum_{l=1}^{L} w_l \times \Gamma\left(\min\left\{0, \left|y_{ij} - u_{ij}\right|\right\}\right)\right) \tag{2}$$

Subject to:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} = n-1 \tag{3}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \le |S|-1 \text{ for any set } S \text{ of nodes} \tag{4}$$

$$\sum_{j=1}^{n} y_{ij} - \sum_{k=1}^{n} y_{ki} = \begin{cases} q_{ij}^l, & if\ i=s \\ 0, & if\ i' \in V-\{s,t\} \\ -q_{ij}^l, & if\ i=t \end{cases} \tag{5}$$

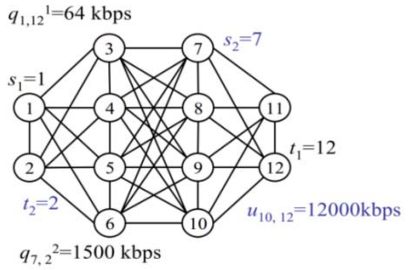
$$\forall (s,t) \in \text{source node and sink node of } q_{ij}^l, \forall l \in L$$

$$y_{ij} \ge 0, \forall i, j = 1..n \tag{6}$$

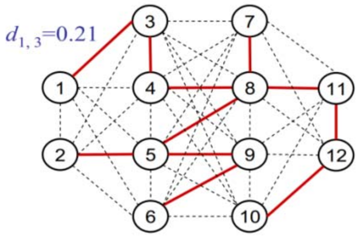
$$x_{ij} \in \{0,1\}, \forall i, j = 1..n \tag{7}$$

where,

- The constraint (3) is a cardinality constraint implying that we choose exactly *n-1* edges.
- The packing constraint (4) implies that the set of chosen edges contain no cycles (if the chosen solution contained a cycle, and S were the set of nodes on a chosen cycle, the solution would violate this constraint)
- The constraint (5) implies a flow conservation law communication requirement on the is depended observed at each of the nodes other than sort.
- The constraint (7) implies the 0-1 variable x_{ij} indicates whether we select edge (i, j) as part of the chosen spanning tree (note that if $y_{ij} > 0$ then $x_{ij}=1$, else $x_{ij}=0$).



(a) Simple example of network



(b) A minimum spanning tree

Fig. 1. A simple network with 12 nodes and 40 edges

III. Related works

How to encode a spanning tree T in a graph G is critical for developing an EA to network design problems, it is not easy to find out a nature representation. Because, designing an appropriate encoding method so as to build an effective EA.

In this section, we summarized the several kinds of classification of encoding methods as follows:

- *Characteristic Vectors-based Encoding:* used by Davis et al. (1993) [12], Bean (1994) [13], Piggott and Suraweera (1995) [14].
- *Edge-based Encoding:* used by Knowles and Corne (2000) [15], Raidl (2000) [16],[17], Chou et al.(2001) [8], Raidl and Julstrom (2003) [7].
- *Node-based Encoding:* is discussed by Cayley (1889) [18], Zhou and Gen (1997, 1999, 2000) [19-21].

In [8], Chou et al predecessor-based encoding generates some chromosomes that are illegal (i.e., not a spanning tree). Combining the simple random initialization, most of the chromosomes will be illegal due to three reasons: missing node i, self-loop, or cycles. The complex repair process will be used at each generation (computational cost), and after repairing, the offspring of the crossover and mutation are difficult to represent solutions that combine substructures of their parental solutions (worst heritability and locality).

Lin Lin and Mitsuo Gen in [5], proposed a PrimPred-based encoding, improved predecessor-based encoding. The initialization depends on an underlying random spanning-tree algorithm. The detailed procedure of this PrimPred-based encoding and decoding is introduced in [22].

IV. Particle Swarm Optimization for the cMST

A. Particle Swarm Optimization

Particle swarm optimization (PSO) is a stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy [23],[24], inspired by social behavior of bird flocking or fish schooling. It shares many similarities with other evolutionary computation techniques such as genetic algorithms (GA). The algorithm is initialized with a population of random solutions and searches for optima by updating generations. However, unlike the GA, the PSO algorithm has no evolution operators such as the crossover and the mutation operator [25], [26], [27].

In the PSO algorithm, the potential solutions, called particles, fly through the problem space by following the current optimum particle. By observing bird flocking or fish schooling, we found that their searching progress has three important properties. First, each particle tries to move away from its neighbors if they are too close. Second, each particle steers towards the average heading of its neighbors. And the third, each particle tries to go towards the average position of its neighbors. Kennedy and Eberhart generalized these properties to be an optimization technique as below.

Consider the optimization problem P. First, we randomly initiate a set of feasible solutions; each of single solution is a *"bird"* in search space and called *"particle"*. All of particles have *fitness values* which are evaluated by the *fitness function* to be optimized, and have *velocities* which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles. The better solutions are found by updating particle's *position*. In iterations, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called *pbest*. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called *gbest*. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called *lbest* [28]-[30].

After finding the two best values, the particle updates its velocity and positions with following equation: (8) (which use global best *gbest*) or (9) (which use local best *lbest*) and (10).

$$\begin{aligned} v[] = v[] + c_1 \times rand() \times (pbest[] - present[]) + \\ + c_2 \times rand() \times (pbest[] - present[]) \end{aligned} \tag{8}$$

$$\begin{aligned} v[] = v[] + c_1 \times rand() \times (pbest[] - present[]) + \\ + c_2 \times rand() \times (lbest[] - present[]) \end{aligned} \tag{9}$$

$$present[] = present[] + v[] \tag{10}$$

In those above equation, *rand* () is a random number between 0 and 1; c_1 and c_2 are cognitive parameter and social parameter respectively.

The stop condition mentioned in the above algorithm can be the maximum number of interaction is not reached or the minimum error criteria are not attained.

B. Solving the cMST based on PSO

In this subsection, we present application of PSO technique for the cMST problem. Our new algorithm is described as follows. We consider that configurations in the algorithm are sets of n nodes.

1) Represent and decode a particle: The encoding of the particle x configuration is matrix by means, say $x = \{x_{ij}\}_{n \times n}$

where $x_{ij} \in \{0,1\}$, $\forall i, j = 1..n$.

2) Initiate population: We use fully random initialization in order to initialize the particle population satisfied constraints (3) and (4).

We generate to express an element of matrix $y = \{y_{ij}\}_{n \times n}$ where $y_{ij} \geq 0$, $\forall i, j = 1..n$ and computed by the formula (5).

3) Fitness function: The cost function for the particle x is computed by the formula (2).

4) Stop condition: The stop condition used in this paper is defined as the maximum number of interaction N_{gen} (N_{gen} is also a designated parameter).

```
PARTICLE SWARM OPTIMIZATION ALGORITHM
{
 FOR each particle
   Initialize particle
 ENDFOR
 DO
   FOR each particle
       Calculate fitness value
       IF the fitness value is better than the
          best fitness value (pBest) in history
          Set current value as the new pBest
       ENDIF
   ENDFOR
   Choose the particle with the best fitness value
   of all the particles as the gBest (or Choose the
   particle with the best fitness value of all the
   neighbors particles as the lBest)
   FOR each particle
     Calculate particle velocity according to (8) or
     (9))
     Update particle position according to (10)
   ENDFOR
 WHILE (STOP CONDITION IS TRUE)}
```

Fig. 2. Particle swarm optimization algorithm

V. Experiments and Results

A. The problems tackled

For the experiments, we have tackled several cMST instances of different difficulty levels defined as follows:

We use the 3 complete network structures have 20 nodes (n=20) with 3 kinds of service:

- Type 1: Cable television.
- Type 2: IP phone.
- Type 3: Data

The weight (priority) of these 3 types respectively:

- w_1=0.60
- w_2=0.30
- w_3=0.10.

The capacity of each edge (i, j) are represented as random variables depend on the uniform distribution:

$$runif(m, 100, 1000)$$

The 20 time-period requirements of different service types from node s to node t are represented as random variables depend on the distribution functions:

- Type 1: exponential distribution:

$$r*exp(|Q|, 0.03)$$

- Type 2: lognormal distribution:

$$0.1*r*lnorm(|Q|, 0.1, 0.1)$$

- Type 3: normal distribution:

$$r*norm(|Q|,0.01, 0.001)),$$

where $|Q|$=100.

B. PSO algorithm specifications

In our experiments, we have already defined parameters for the PSO algorithm shown in Table II.

TABLE II: THE PSO ALGORITHM SPECIFICATIONS

Population size	P = 1000
Maximum number of interaction	N_{gen} = 500
Cognitive parameter	c_1 = 1
Social parameter	c_2 = 1
Update population according to	Formula (6) and (7)
Number of neighbor	K = 3

C. Numerical Results

In the experiment, our proposed PSO is compared with PrimPred-based EA [5] various evolutionary algorithms Edge-based EA [10], Prufer number-based GA [7] and traditional Prim's algorithm (without considering the capacity constraint) [6].

The objective function is total time average delay of our algorithms has achieved a much better performance than other algorithms. The experimental results show in Figure 3.

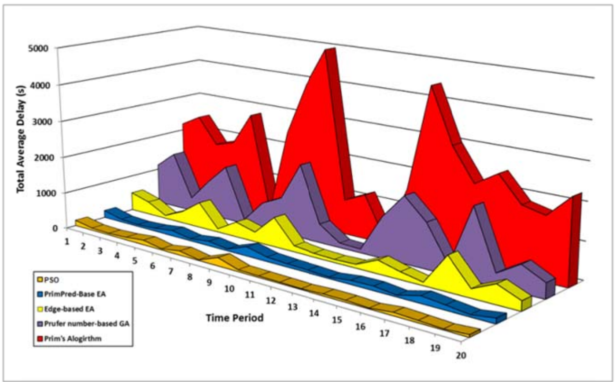


Fig. 3. Comparisons total time average delay results of PSO, primpred-based EA, Edge-based EA, prufer number-based GA and prim's algorithms

VI. Conclusion and Future Work

In this paper, we have proposed an effective PSO algorithm for improvement of Quality of Service in Next Generation Network. We have formulated this problem as an extended capacitated minimum spanning tree (cMST) problem with considering capabilities of the network, different priority for different types of services and dynamic environment. In our algorithm, the objective functions are determined by the total average time delay based pheromone matrix of ants satisfies capacity constraints to find good approximate solutions. Numerical experiments with various scales of communication network problems show the effectiveness and the efficiency show that our proposed algorithm is much better than the recent researches.

Optimizing quality of service in Next Generation Network with considering capabilities, different types of services, profit, coverage area and throughput maximization in dynamic environment is our next research goal.

References

[1] W. Sun. (1999). QoS/Policy/Constraint Based Routing. [Online]. Available: http://www.cse.wustl.edu/~jain/cis788-99/qos_routing/index.html

[2] R. Braden, D. Clark, and S. Shenker, *Integrated Services in the Internet Architecture: an Overview*, 1994.

[3] S. Blake. Architecture for Differentiated Service. [Online]. Available: http://ftp.isi.edu/in-notes/rfc2475.txt

[4] R. Callon. An frramework for Multiprotocol Label Switching. [Online]. Available: http:// draft-ietf-mpls-framework-05.txt

[5] L. Lin and Mitsuo Gen, *Auto-tuning Evolutionary Algorithm for a Capacitated QoS Model in Communication Network*, 2008.

[6] T. C. Hu, "Optimum Communication Spanning Trees," *SIAM Journal of Computing*, vol. 3, no. 3, pp. 188–195, 1974.

[7] G. Raidl and B. A. Julstrom, An Effective Evolutionary Coding of Spanning Trees, *IEEE Trans, on Evolutionary Computing*, vol. 7, no. 3, pp. 225-239, 2003.

[8] H. Chou, G. Premkumar, and C. Chu, "Genetic algorithms for communication network design – an empirical study of the factors that influence performance," *IEEE Trans, on Evolutionary Computing*, vol. 5, no. 3, pp.236-249, 2001.

[9] M. Gen, G. Zhou, and M. Takayama, "Matrix-based genetic algorithm approach on bicriteria minimum spanning tree problem with interval coefficients," *J.of Japan Society for Fuzzy Theory and Systems*, vol. 10, no. 6, pp. 643-656, 2000.

[10] A. T. Hoang, V. T. Le, and N. G. Nguyen, "A Novel Particle Swarm Optimization- based Algorithm for the Optimal Communication Spanning Tree problem," in *Proceedings of The 2010 International Conference on Communication Software and Network (ICCSN 2010)*, pp. 232-236, 2010.

[11] D. N. Le, N. G. Nguyen, and N. D. Le, "A Novel Ant Colony Optimization-based Algorithm for the OptimalCommunication Spanning Tree problem," in *Proc. of 5th International Conference on Computer Science and Information Technology (ICCSIT 2012*), December 29-30, 2012, Hongkong.

[12] L. Davis, D. Orvosh, A. Cox, and Y. Qiu, "A genetic algorithm for survivable network design," in *Proc. 5th Int. Conf. Genetic Algorithms*, pp. 408-415, 1993

[13] J. C. Bean, "Genetic algorithms and random keys for sequencing and optimization," *ORSA J. Comput*., vol. 6, no. 2, pp. 154-160, 1994

[14] P. Piggott and F. Suraweera, "Encoding graphs for genetic algorithms: An investigation using the minimum spanning tree problem," In X. Yao, editor, *Progress in Evolutionary Computation*, LNAI 956, pp. 305–314. Springer, Berlin, 1995

[15] J. Knowles and D. Corne, "A new evolutionary approach to the degree-constrained minimum spanning tree problem," *IEEE Trans, on Evolutionary Computing*, vol. 4, no. 2, pp. 125-134, 2000.

[16] G. Rail, "A weighted coding in a genetic algorithm for the degree-constrained minimum spanning tree problem," in *Proc .of SAC*, vol. 1, pp. 440-445, 2000.

[17] G. Raidl and C. Drexe, "A predecessor coding in an evolutionary algorithm for the capacitated minimum spanning tree problem," in *Proc. GECCO*, pp. 309-316, 2000.

[18] A. Cayley, "A theorem on trees," *Quart. J. Math*, vol. 23, 1889, pp. 376–378.

[19] G. Zho and M. Gen, "A note on genetic algorithm approach to the degree-constrained spanning tree problems," *Networks*, vol. 30, pp. 105-109, 1997

[20] G. Zho and M. Gen, "Genetic algorithm approach on multi-criteria minimum spanning tree problem," *E. J. of Operational Research*, vol. 114, no. 1, pp. 141-152, 1999.

[21] G. Zho and M. Gen, *Approach to degree-contrainted minimum spanning tree problem using genetic algorithm*, in M. Gen and R. Chen, *Genetic algorithms & Engineering Design*, New York: John Wiley, 2000

[22] L. Lin and M. Gen, "Node-based genetic alogrithm for communication spanning tree problem," *IEICE Transactions on Communications*, vol. E89-B, no. 4, pp.1091-1098, April 2006.

[23] "James Kennedy and Russell Eberhart, Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, pp.1942–1948, Piscataway, NJ, USA, 1995.

[24] J. Kennedy and R. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publisher Inc, 2001

[25] B. A. Julstrom, "The blob code is competitive with edge-sets in genetic algorithms for the minimum routing cost spanning tree problem," in Beyer, Hans-Georg et al. (Ed.), *Proceedings of the Genetic and Evolutionary Computation Conference 2005*, New York, pp. 585–590, ACM Press, 2005.

[26] J. Gottlieb, B. A. Julstrom, G. R. Raidl, and F. Rothlauf, "Pr¨ufer numbers: A poor representation of spanning trees for evolutionary search," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pp. 343–350. 2001

[27] F. Rothlauf, J. Gerstacker, and A. Heinzl, "On the optimal communication spanning tree problem," Technical Report 15/2003, Department of Information Systems, University of Mannheim, 2003.

[28] D. N. Le, N. G. Nguyen, and V. T. Le, "A Novel PSO-Based Algorithm for the Optimal Location of Controllers in Wireless Networks," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 12, no. 08, pp. 23-27, August 30, 2012, Korea.

[29] D. N. Le, "Optimizing Resource Allocation to Support QoS Requirements in Next Generation Networks using ACO Algorithm," *International Journal of Computer Science and Information Technology & Security (IJCSITS)*, vol. 2, no. 5, pp. 931-938, 2012.

[30] D. N. Le and N. G. Nguyen, "A New Evolutionary Approach for Gateway Placement in Wireless Mesh Networks," *International Journal of Computer Networks and Wireless Communications (IJCNWC)*, vol. 2, no. 5, pp.550-555, 2012.

Gia Nhu Nguyen received the BSc degree in computer science and the MSc degree in information technology from Dannang University, Vietnam, in 1998 and 2006, respectively. He currently works in Duy Tan University, Danang, Vietnam.

His research interests include algorithm theory, network and wireless security.

Dac-Nhuong Le received the BSc degree in computer science and the MSc degree in information technology from College of Technology, Vietnam National University, Vietnam, in 2005 and 2009, respectively. He is a lecturer at the Faculty of information technology in Haiphong University, Vietnam. He is currently a Ph.D student at Hanoi University of Science, Vietnam National University. His research interests include algorithm theory, computer network and networks security.

Nguyen Dang Le received the BSc degree in computer science and the MSc degree in information technology from College of technology, Vietnam National University in Hanoi, Vietnam, in 1996 and 2005, respectively. He currently works in Haiphong University, Vietnam. His research interests include algorithm theory, network and wireless security.

Vinh Trong Le received the MSc degree in information technology from Faculty of Mathematics, Mechanics and Informatics, Hanoi University of Science, Vietnam National University in 1997, PhD degrees in computer science from Japan Advanced Institute of Science and Technology in 2006, respectively. He is currently Associate Professor at the Faculty of Mathematics, Mechanics and Informatics, Hanoi University of Science, Vietnam National University. His research interests include algorithm theory, network and wireless security.