Improvements to Tree-based GA Applications for QoS Routing

Vincenzo Maniscalco, Silvana Greco Polito, and Antonio Intagliata

Facoltá di Ingengeria, Architettura e delle Scienze Motorie
Libera Universitá degli Studi di Enna ”KORE”
Enna, Italy
{vincenzo.maniscalco,silvana.grecopolito, antonio.intagliata}@unikore.it

***Abstract*—Genetic Algorithms (GAs) are emerging as a promising instrument for quality of service (QoS) routing in Mobile Ad hoc Networks (MANETs). They implement an iterative process that can solve the NP search problem of routing with multiple QoS constraints. In each iteration new solutions are found through the mutation and crossover genetic operations. In this paper we focus on an existing GA tree-based application for QoS routing in MANETs which applies the genetic operations on a tree built from the network topology and uses fixed length chromosomes. The chromosome encodes junctions tree crossed by routes. This application suffers in convergence speed because its mutation operator does not allow deep exploration of the search space. The inefficiency of the adopted mutation operation is more evident in networks with big size and high network connectivity, i.e., networks with a larger search space. In this paper, we elaborate on the tree-based application with the main objective of improving its performance. We first introduce a criterion for junctions tree sorting based on their distance from the root. Later, we use chromosome properties due to the sorting criterion to design a sequential mutation technique with adaptive probability that allows faster convergence. We provide simulation results showing the effectiveness of the proposed enhancements while increasing the MANET size and connectivity.**

Keywords-Genetic Algorithm; QOS routing; junctions tree; MANET

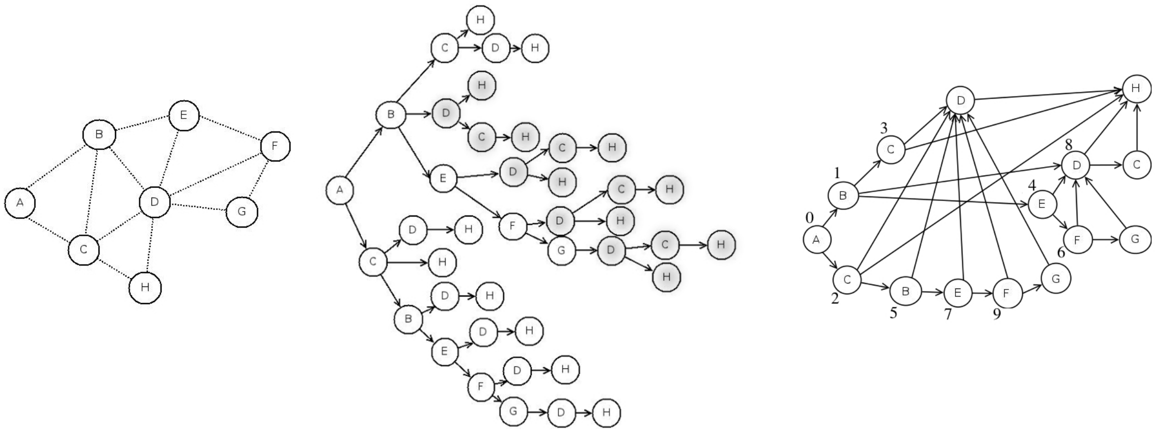
I. Introduction

MANETs are composed of wireless nodes that can move, join and leave the network dynamically. They can be used in different scenarios such as campus and disaster recovery areas for communication between students or emergency operators. They do not refer to a fixed infrastructure and need routing protocols that adapt to topology changes quickly. They also need routing algorithms that can provide quality of service (QoS) routes, i.e., routes satisfying QoS constraints such as delay or jitter posed by realtime applications. Therefore the need of QoS routing for MANETs is motivating research work on QoS protocols [1], metrics and algorithms [2]. Regarding QoS algorithms, the Literature shows an increasing interest around solutions based on Genetic Algorithms which can solve NP problems. GA routing algorithms, given network topology and QoS costs of network components, implement genetic operations to search the best route subject to multiple QoS constraints. This is known to be an NP problem. The Literature discusses two main GA approaches for QoS routing with fundamental differences in their encoding schema. An approach employs variable length chromosomes with genes representing the nodes along a route from source to destination [3-5]. The other models the set of routes from source to destination as a tree and implements a coding schema with fixed length chromosomes [6-8]. Genes of the chromosome encode the junctions tree. The main advantages of the tree-based approach are due to the tree structure that avoids loops generation during the GA process, and the fixed length of the chromosome that allows to use simpler models for genetic operations.

In this paper, we elaborate on the main existing tree-based GA application for QoS routing introduced by Barolli et al. in [6-8], and we propose enhancements that strongly improve its performance. We have observed that the convergence of the existing tree-based applications becomes slower while increasing network connectivity and network size as the mutation probability, which is fixed (it is equal to the inverse of the chromosome length), does not allow deep exploration of big search space. Available results [6] describe the algorithm scalability while increasing the number of nodes in the network, but how scalability is affected by the number of links per node is not discussed. In addition tree-based chromosomes have to be sorted properly to avoid generation of invalid routes with genetic operations. Although the authors [9] are aware of this problem they do not propose a sorting model.

To overcome the above discussed lacks, we (i) introduce a proper sorting model for genes within the chromosome, and (ii) propose a sequential mutation technique with adaptive probability. The sorting model makes junctions representation within the chromosome dependent on their depth, i.e., the distance from the source. It avoids generation of invalid routes from genetic operations. The mutation technique uses a mutation probability which adapts to any specific route: it is equal to the inverse of the junctions number crossed by the route. The adaptive probability allows deeper exploration of the search space and therefore faster convergence to better QoS routes. We carry out simulations demonstrating the strong effect of the mutation solution on the scalability of the GA application while increasing both network size and node connectivity. Study of protocols for acquisition and maintaining

ISBN: 978-1-61208-231-8



(a)

(b)

(c)

Fig. 1. (a) MANET example; (b) Tree network for source A and destination H; (c) Reduced tree network.

of topological information is out of the scope of this paper. Any protocol able to collect topology and QoS data can be used as the one discussed in [10]. According to this protocol the source sends route discovery requests in flooding toward the destination, which replays with multiple routes and per hop QoS parameters.

The remaining of the paper is organized as follows. In Section II we summarize the main aspects of the tree-based GA approach to which we refer. In Section III we introduce the criterion for junctions sorting and we describe the adaptive mutation method. Section IV provides simulation results showing the performance improvements due to the proposed enhancements while increasing network size and connectivity.

II. Review of GA and the tree-based application for QoS routing

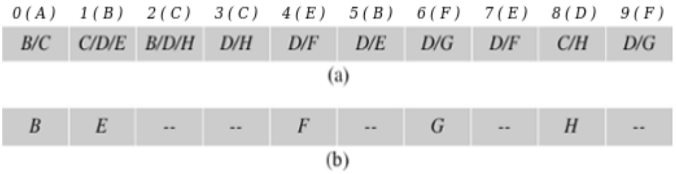
GA is a heuristic method that simulates the natural evolution process to solve optimization problems. In the following we first introduce the GA iterative process and later we review the existing tree-based GA application for QoS routing in MANETs.

A. The GA iterative process

GA employes iterative processes on a set of individuals representing a generation for each cycle [11]. Each individual is a candidate solution and is encoded with a chromosome composed of genes. The GA process includes two main stochastic genetic operations, i.e., crossover and mutation. The former is to transmit the genetic heredity from selected parents to next generation. The parents' selection process is based on a fitness function which depends on constraints of the specific problem to solve. For selected parents, the crossover operation exchanges parts of their chromosomes to generate new descendants. These descendants are submitted to the mutation operation that changes their genes for the new generation. Usually, the GA starts with an initial population chosen at random and the cycle repeats until the solution is found or termination criteria are satisfied. Bigger population sizes guarantee solutions with better quality. The number of evaluations of the fitness function, which increases with the number of iterations and the size of the initial population, measures the GA computational complexity. When applications have stringent time constraints, fast GA hardware implementations in FPGA can be used [12].

B. The existing tree-based method for QoS routing

In the tree-based genetic algorithm for QoS routing proposed by Barolli et al. [6-9], the network (Figure1.a) is modelled as a tree having the source node as root and the destination node as leafs (Figure 1.b). The chromosome structure is expressed by the tree junctions which leads to fixed chromosome length. To compact the chromosome length a tree reduction algorithm has been proposed [9]. Tree reduction is carried out merging nodes having the same identifier (ID) and the same sub-routes. In the example of Figure 1.b the four junctions D become a single junction in the reduced tree (Figure1.c). A number is assigned to each junction which is its locus in the chromosome. Each gene can assume the value of one of the adjacent junctions. Genes may be active or inactive: a gene is active if its corresponding junction is in the route. Gene values are assigned only to active genes. In Figure 2 the gene encoding and an encoded route are shown.



0(A)	1(B)	2(C)	3(C)	4(E)	5(B)	6(F)	7(E)	8(D)	9(F)
B/C	C/D/E	B/D/H	D/H	D/F	D/E	D/G	D/F	C/H	D/G

(a)

B	E	--	--	F	--	G	--	H	--

(b)

Fig. 2. (a) Gene coding; (b) Chromosome encoding the route $A-B-E-F-G-D-H$.

Genetic operations must be carried out between active genes. In this manner, the crossover operation interchanges sub-routes in the tree network from the junction that corresponds to the crossover point, while the mutation operation

ISBN: 978-1-61208-231-8

generates a new suitable sub-route from the junction that corresponds to the mutation point.

Regarding the fitness function, delay time (DT) and transmission success rate (TSR) are used to express the QoS of a route as in the following equation where N is the number of links in a route:

$$T = \frac{\sum_{i=1}^{N} DT_i}{\prod_{i=1}^{N} TSR_i} \qquad (1)$$

The rank selection method is used along with single point crossover, and elitism to maintain the best solution in the next generation. The mutation technique is not discussed, the mutation probability is equal to the inverse of the chromosome length.

III. Enhancements to the tree-based GA application for QoS routing

The enhancements to the tree-based application are (a) a proper junction sorting criterion, and (b) a novel mutation technique with adaptive probability. The junction sorting criterion leads to a chromosome structure that depends on the junctions depth. This feature is used to get always valid routes from genetic operation. We also leverage on it to build the mutation technique.

A. Junction sorting

Junction sorting is based on the distance from the junction to the root, which is called junction depth. In the following we first introduce the sorting criterion and later on the main chromosome properties derived from it.

1) Depth-based sorting criterion: It consists of the following three steps.

Step 1: Build a junction tree from the tree network with junctions sorted according to their distance from the route. The sorted junctions tree for the network of Figure 1 is shown in Figure 3. In this example, the root junction A has depth equal to 0, the junctions B and C have depth 1 and so on.

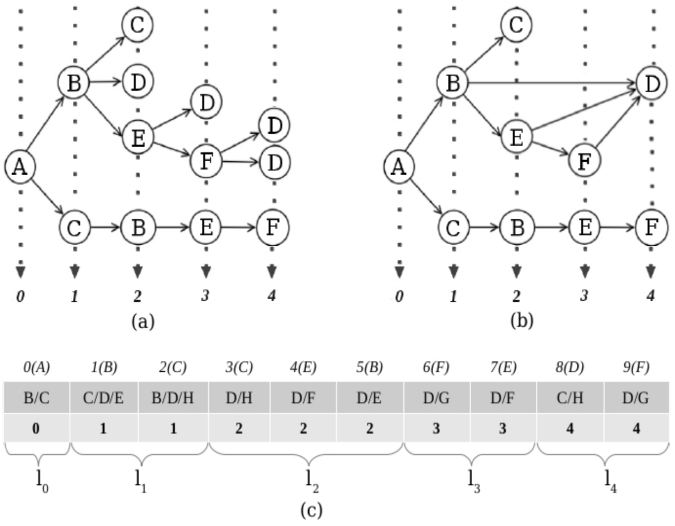
Step 2: Merge junctions for tree reduction and assign the highest depth to them. For the example of Figure 3 this means assigning depth 4 to the junction D.

Step 3: Encodes the junctions within the chromosome according to their depth in the reduced junction tree.

This criterion guarantees that genetic operations always result in valid routes. How invalid routes can be created when junctions are not sorted properly is shown in the example of Figure 4. In this example, the junction D is located in the 5th gene instead of the 8th. If we perform crossover between routes A-B-E-D-C-H and A-B-D-H with crossover point D, an invalid descendent is generated as the 5th gene is inactive. If the mutation operator does not change the values of this chromosome the decoding operation does not return a route.

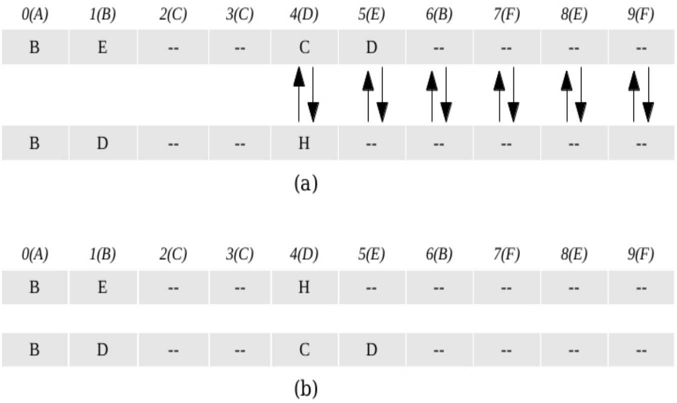
2) Chromosome properties due to sorting: The junction sorting criterion also leads to the following chromosome encoding properties:

- At most one gene encoding junctions with the same depth may be active.



0(A)	1(B)	2(C)	3(C)	4(E)	5(B)	6(F)	7(E)	8(D)	9(F)
B/C	C/D/E	B/D/H	D/H	D/F	D/E	D/G	D/F	C/H	D/G
0	1	1	2	2	2	3	3	4	4
l_0	l_1		l_2			l_3		l_4	

Fig. 3. (a) Junctions tree; (b) Reduced junctions tree; (c) Chromosome structure sorted by depth



(a)

0(A)	1(B)	2(C)	3(C)	4(D)	5(E)	6(B)	7(F)	8(E)	9(F)
B	E	--	--	C	D	--	--	--	--
B	D	--	--	H	--	--	--	--	--

(b)

0(A)	1(B)	2(C)	3(C)	4(D)	5(E)	6(B)	7(F)	8(E)	9(F)
B	E	--	--	H	--	--	--	--	--
B	D	--	--	C	D	--	--	--	--

Fig. 4. (a) Single-point crossover between chromosomes sorted improperly; (b) Invalid descendants.

- If two junctions with different depths are directly connected, all the genes encoding junctions between them are inactive.
- If a gene encodes a junction reaching the destination, then all the following genes are inactive.

B. Sequential mutation technique with adaptive probability

The encoding chromosome properties due to junction sorting described in the previous section let us design a sequential mutation technique with adaptive probability. This is equal to the inverse of active genes number for each chromosome. The mutation details are the following:

(1) Apply the mutation probability to each active gene of the chromosome sequentially until a mutation point is found or the chromosome is ended;

(2) If the mutation point has been found, take into account the set of adjacent nodes, excluding the current one, and select one of them with equal probability. Deactivate the following genes in the chromosome.

(3) Search the chosen node in the reduced tree network;
  - If this node is a junction, select one of the adjacent nodes with equal probability for the gene of the junction and return to step (3);
  - If this node is not a junction and it is not the destination, take into account the next element in the tree network and return to step (3);
  - If this node is the destination, the algorithm ends.

We point out that mutation is always applied once a mutation point is found and the route is always valid. If D is the highest depth level, the mutation probability range can be expressed as follows:

$$\frac{1}{D} \leq p_m \leq 1 \qquad (2)$$

In other words, the mutation probability is maximum when source and destination nodes are directly connected, while it is minimum when the route that connect them crosses one junction per each depth.

IV. Performance study and discussion

This performance study aims to show the effects of adaptive mutation probability on the tree-based application while increasing the network size and connectivity. For this purpose, we have implemented the tree-based GA application with both the fixed mutation probability introduced in [6-9] and the adaptive mutation probability proposed in this paper. For both the implementations we have used the junction sorting criterion and the mutation technique proposed in Section III-A and Section III-B, respectively, as any description of them is missing for the existing application. Therefore, the two implementations have the same computational cost. The implementations are made with initial population generated randomly, linear ranking selection method, single point crossover and elitism. The fitness function is the one in eq (1) with DT and TSR selected randomly. Each simulation runs 1000 times and provides the rank, i.e., the position of the solutions sorted according to their fitness, as output. Simulations are carried out with MATLAB.

In the following, we will refer to the two implementations as fixed-GA and adaptive-GA, respectively.

A. Scalability while increasing the network connectivity

Figures 5 and 6 show the rank achieved by the GA application with both fixed and adaptive mutation probability on a sparse and dense network. The sparse and dense networks have 3 and 5 links per node on average, respectively. These figures also report the size of the search space (routes) and the chromosome length. Population sizes are selected equal to 20 and 50 in the sparse and dense networks, respectively.

In the sparse network (see Figure 5), the application with adaptive mutation probability convergences around the 10th generation, while the applications with fixed mutation probability converges on the 20th generation. Therefore, the adaptive mutation probability cuts by half the computation complexity, which is equal to (population size)*(iterations number). The performance improvements are stronger in the dense network as shown in Figure 6. In this scenario the application with adaptive mutation probability convergences around the 50th generation, while the native application is still far to converge. This because dense network does have big chromosome length L and the fixed mutation probability equal to $1/L$ is too small for a deep exploration of the search space.

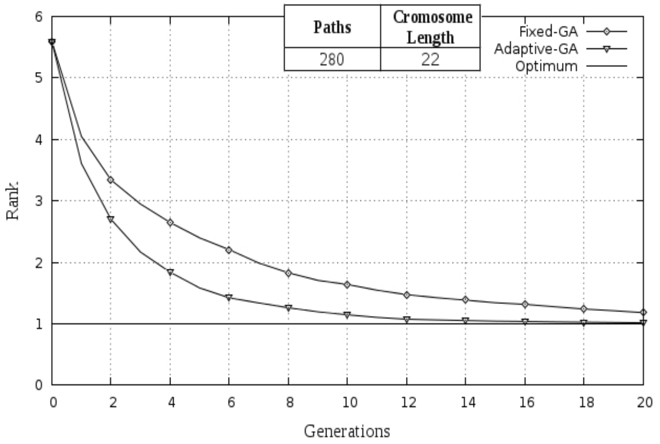


Fig. 5. Comparison between fixed-GA and adaptive-GA in sparse networks

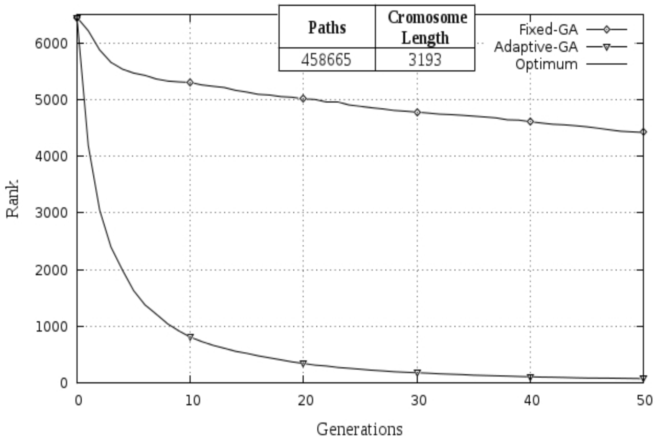


Fig. 6. Comparison between fixed-GA and adaptive-GA in dense networks

Figures 7 and 8 show the convergence curves of the GA application with adaptive mutation for different population sizes. We have used them to set the population size of the previous simulations. The population size of 20 guarantees convergence to the optimum solution on the 20th generation in the sparse network. The one of 50 for the dense network guarantees convergence to a suboptimal solution on the 50th generation. We recall that (see Section II-A) the population size should be selected according to the desired trade-off between computation complexity and quality of solution, which depends on the specific application and the computation features of the mobile nodes. The total computation cost and delay of any tree-based GA application also depends on the tree generation and reduction procedure which increases with

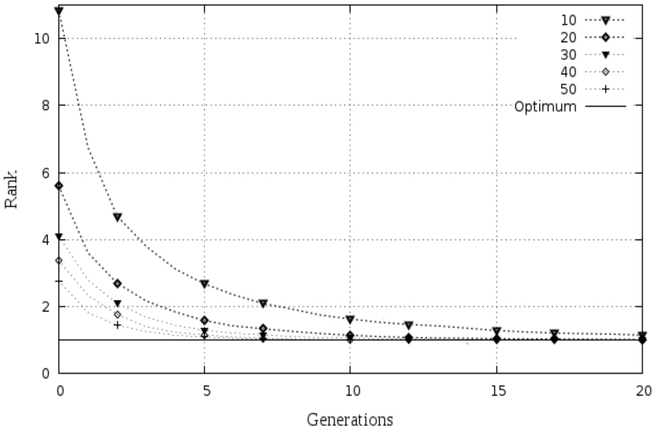


Fig. 7. Performance for different population sizes in sparse networks

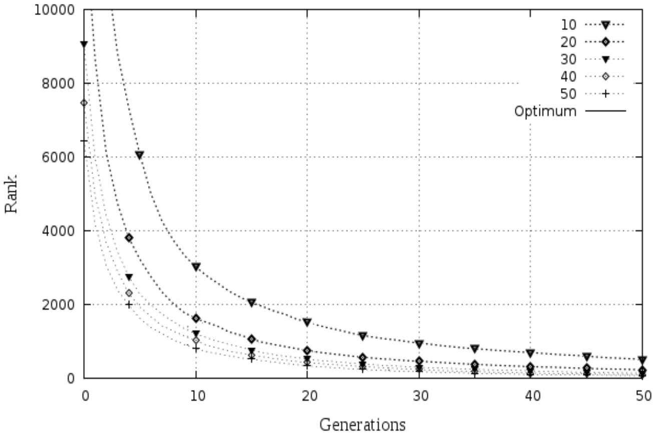


Fig. 8. Performance for different population sizes in dense networks

the solutions space. Note that it is around 458.000 for the dense network with 5 links per node. MANET scenarios with high mobility speed poses constraints on the delays for route discovery, therefore the applicability of the algorithm in high connected networks depends on the mobility speed.

B. Scalability while increasing the network size

The effects of network size on the GA application convergence is shown in Figures 9, 10 and 11 with networks of 20, 30 and 40 nodes, respectively. The connectivity density is of 3 links per node on average.Simulations are made with population size equal to the number of nodes. QoS DT/TSR parameters are selected and network topologies are generated randomly per each of the 1000 simulations. Given the population size, the total number of generations is chosen according to the convergence of the faster application, which is always the one with adaptive mutation probability.

The results in Figure 9 show that, although the implementation with adaptive mutation probability has higher convergence speed, the behavior of the other implementation is similar as it was expected from the results in Figure 5. Improvements on performance increase in the bigger networks. In

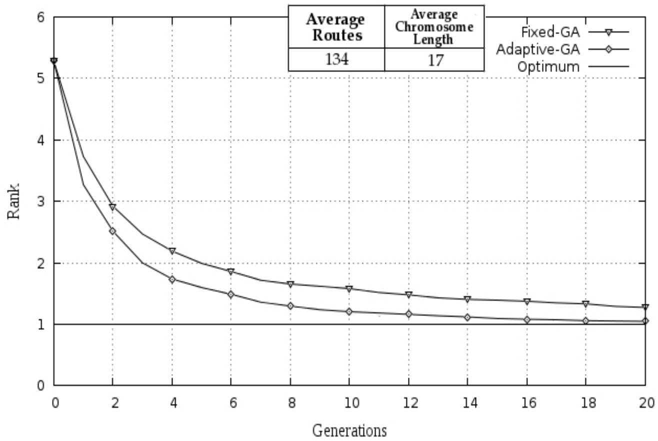


Fig. 9. Comparison between fixed-GA and adaptive-GA for a network with 20 nodes

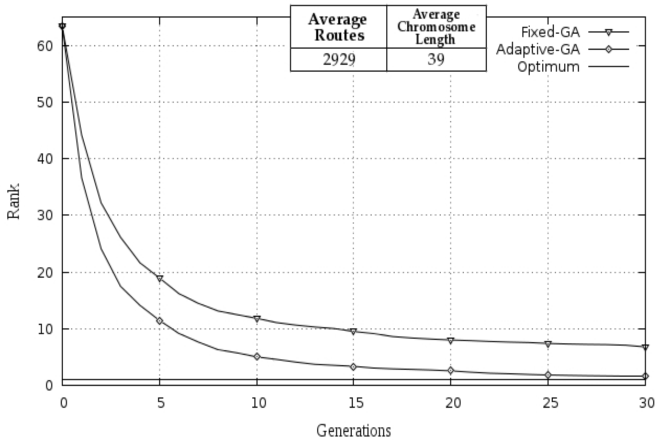


Fig. 10. Comparison between fixed-GA and adaptive-GA for a network with 30 nodes

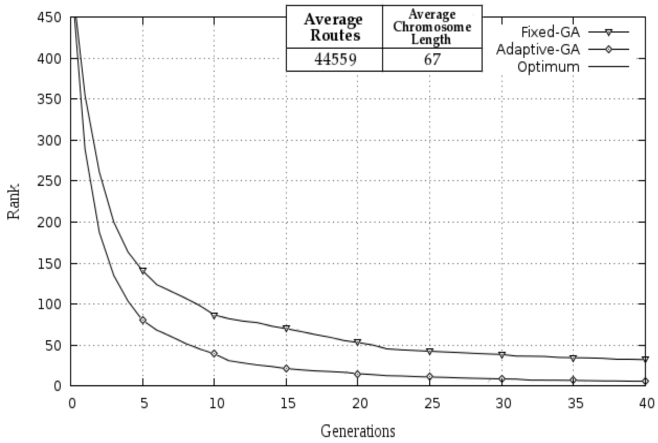


Fig. 11. Comparison between fixed-GA and adaptive-GA for a network with 40 nodes

ISBN: 978-1-61208-231-8

these networks, while the applications with adaptive mutation probability converge to the optimum or a close to optimum solution, the GAs with fixed probability provide low quality solutions (see Figure 10 and 11). The reason is the same pointed out for high connected networks: bigger networks have larger search space and longer chromosome and then a fixed mutation probability does not allow deep exploration of the search space.

Note that the number of routes does not increases linearly with the number of nodes. When the search space explodes the computational cost of the tree-generation process may be prohibitive. In these networks the GA application can be used with clustering architectures to work on smaller search spaces. This is similar to what done by the authors of [13]. They divide the network in zone and apply a GA-based algorithm with variable length chromosomes for inter-zone routing. Objective of our future study is verifying how to apply the GA tree-based approach on clustering architectures.

C. Statistics for the simulation results

As statistics for the simulation results we consider the standard deviation. We report in Table I rank and standard deviation for the first results of Figure 5. Note that the standard deviation decreases while increasing the generation number. This is also true for all the other simulations.

TABLE I
STANDARD DEVIATIONS FOR RESULTS IN FIGURE 5

Generations	Fixed-GA		Adaptive-GA	
	Rank	Standard Deviation	Rank	Standard Deviation
12	1,54	1,19	1,10	0,40
14	1,42	1,03	1,06	0,32
16	1,34	0,92	1,04	0,26
18	1,27	0,84	1,03	0,24
20	1,21	0,74	1,02	0,19

V. CONCLUSION AND FUTURE WORK

In this paper, we have proposed enhancements to the existing tree-based GA application for QoS routing in MANETs. We have discussed how the fixed mutation probability adopted for this application does not allow a deep exploration of the search space and therefore leads to slow convergence or convergence to solutions with low QoS. We have introduced a criterion for junction sorting based on the junction depth and we have proposed a sequential mutation technique with adaptive probability. The adaptive probability of the mutation technique allows a deeper exploration of the search space, which leads to faster convergence and better QoS solutions for given generations. We have shown with simulation results that the adaptive mutation probability strongly improve the performance of the tree-based GA application, particularly in big or highly connected networks. Future purposes will be about the application of the GA tree-based approach to clustering architectures.

REFERENCES

[1] S. Mittal, P. Kaur, *Performance Comparison of AODV, DSR and ZRP Routing Protocols in MANET'S*, International Conference on Advances in Computing, Control, and Telecommunication Technologies, Trivandrum, India, December 2009.

[2] L. Hanzo, R. Tafazolly, *A Survey of QoS Routing Solutions for Mobile Ad hoc Networks*, IEEE Communications Surveys, Vol.9, N.2, pp. 50-69, 2009.

[3] Chang Wook Ahn, R.S. Ramakrishna, *A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations*, IEEE Transactions on Evolutionary Computing, Vol. 6, pp. 566 - 579, December 2002.

[4] S. Yang, H. Cheng, F. Wang, *A genetic Algorithms With Immigrants and Memory Schemes for Dynamic Shortest Path Routing Problems in Mobile Ad Hoc Networks*, IEEE Transactions on Systems, MAN and Cynernetics-Part C: Applications and Review, Vol. 40, N.1, pp.52-63, January 2010.

[5] S. Yussof, O. H. See, *A Robust GA-based QoS Routing Algorithm for Solving Multi-constrained Path Problem*, Journal of Computers, Vol. 5, N.9, September 2010.

[6] A. Barolli et al., *Application of Genetic Algorithms for Qos Routing in Mobile Ad-Hoc Networks: A Survey*, International Conference on Broadband, Wireless Computing, Communication and Applications, Fukuoka, November 2010.

[7] L. Barolli, A. Koyama, N. Shiratori, *A QoS Routing Method for Ad-Hoc Networks Based on Genetic Algorithm*, 14th International Workshop on Database and Expert Systems Applications (DEXA'03), Prague, Czech Republic, September 2003.

[8] A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa,*Application of GA and Multi-objective Optimization for QoS Routing in Ad-Hoc Networks*,14th International Conference on Network-Based Information Systems (NBiS), Tirana, September 2011.

[9] L. Barolli, A. Koyama, T. Suganuma, N. Shiratori, *A Genetic Algorithm Based QoS Routing Method for Multimedia Communications Over High-Speed Networks*, Information Processing Society of Japan (IPSJ), Vol. 44, No. 3, pp. 544-552, March 2003.

[10] J. Abdulla, D. Parish, *Effect of Mobility on the Performance of GA-based QoS Routing in Mobile Ad Hoc Networks*, International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia, November 2007.

[11] D. E. Golberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Book, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA 1989.

[12] M. Vavouras, *High-speed FPGA-based implementations of a Genetic Algorithm*, International Symposium on Systems, Architectures, Modelling, and Simulations, Samos, Greece, July 2009.

[13] P. Sateesh Kumar, S. Ramachamndram, *Scalability of Network Size on Genetic Zone Routing Protocol for MANETs*, International Conference on Advanced Computer Theory and Engineering, Phuket, Thailand, December 2008.