

On Routing IP Traffic Using Single- and Multi-objective Genetic Optimization

Jules A. Baruani¹, Antoine B. Bagula^{1,2} & Americo Muchanga²

Abstract—We formulate the routing of traffic in emerging IP networks as both single and multi-objective optimization problems which are solved using *classical* and *hybrid* genetic optimization. We adopt the occurrence of links in the computed paths as a cost function used to achieve load balancing by reducing the interference among competing flows. We adapt the the Non-Sorting Genetic Algorithm(NSGA) to find a *Pareto front* for the multi-objective problem. This *Pareto front* expresses a set of solutions which represent plausible set of paths that minimize both the interference among competing flows and the queuing delay in the network. Using both multiplicative and additive composition rules, we compare the quality of the paths achieved by the different evolutionary optimization strategies.

I. INTRODUCTION

Evolutionary algorithms use concepts from real-world genetics to evolve solutions to problems. They are based on an evolutionary paradigm where each iteration of the algorithm transforms one population of individuals into a new generation, using some pre-determined fitness measure for an individual. In applications of evolutionary algorithms, potential solutions must be represented and encoded in terms of *genome*. Each problem generally has its own genome representation, and more than one representation could be used for a given problem. The fitness measure or *fitness function* determines how good the solution represented by some genome is. The appropriate fitness function is determined by the problem and genome representation. *Genetic Algorithms (GAs)* are based on a population selection where the evolution of one generation into another relies on the three main genetic operations: (1) *replacement* (2) *crossover* and (3) *mutation*. *Replacement* is a direct copying of a member of the current generation into the next generation. These probabilities are denoted P_c , P_r and P_m respectively, and $P_c + P_r + P_m = 1$. In general, $P_m \ll P_c$ and $P_c \approx P_r$. Candidates for the genetic operations are chosen randomly, but the selection is *fitness-proportionate* to ensure the survival of good solutions over generations. The conditions for the termination of the algorithm are problem-specific, although for practical reasons one often limits the number of

iterations. Evolutionary algorithms can find acceptably good solutions to problems by examining and manipulating a set of possible solutions from a set of designs but are not guaranteed to find the global solution to a problem. Evolutionary algorithms have been deployed in the literature to solve real-life problems using different strategies. The selection of the appropriate strategy to be used for a specific application is an important aspect upon which the usefulness of the evolutionary optimization process depends.

This paper addresses the problem of routing traffic in IP networks using evolutionary optimization. We formulate the routing in emerging IP networks as both single and multi-objective optimization problems which are solved using different genetic optimization strategies. These include strategies which use classical genetic algorithms referred to in this paper as *genetic optimization* strategies and strategies using hybrid genetic algorithms referred to as *memetic optimization* strategies. We adopt the occurrence of nodes in the computed paths as a cost function used in both the *genetic optimization* and *memetic optimization* to achieve load balancing by reducing the interference among competing flows. We adapt the Non-Sorting Genetic Algorithm(NSGA) proposed by [2] to find a *Pareto front* for the multi-objective problem. This *Pareto front* expresses a set of solutions which represent plausible set of paths that minimize both the interference among competing flows and the queuing delay in the network. We consider both multiplicative and additive composition rules for the cost function to assess the relevance of using these rules in emerging IP networks.

In the rest of this paper the following words are used interchangeably: source and origin, vertex and node, link and edge.

The reminder of this paper is organized as follows. Section II presents the single objective optimization model while section III describes the multi-objective optimization model. An application of the different optimization models to compute paths for the flows offered to a 23- and 31-node networks is presented in section IV. We draw some preliminary conclusions and present guidelines for future work in section V.

II. SINGLE OBJECTIVE OPTIMIZATION

We consider a single optimization model where a physical network is viewed as a graph $G = (N, L)$ where N is the number of nodes and L is the number of links. S is a set of ordered pairs consisting of origin o and destination d , that is

$$S = \{(o, d) | o \text{ is the origin node and } d \text{ is the destination node}\}$$

1. School of Mathematical Sciences, Computer Science Section, University of Stellenbosch, 7600 Stellenbosch, South Africa. Email: abaruani, bagula@cs.sun.ac.za. Tel: +27 21 8084232 Fax: +27 21 8084416

2. Department of Electronics, Computer and Software Systems, Telecommunication Systems Laboratory, Royal Institute of Technology (KTH), Isafjordsgatan 39, SE-164 40 Kista, Sweden. Email: abbagula@kth.se, americo@imit.kth.se Tel: +46 8 790 42 84 Fax: +46 8 751 17 93

This work is supported by grant numbers 2047362 and 2677 from the South African National Research Foundation, Siemens Telecommunications and Telkom SA Limited.

In most routing schemes traffic is routed along shortest paths. As a result these paths become overloaded thus causing a bottleneck for the network. It is therefore reasonable to select a set of paths which may not be shortest but lightly loaded. This is achieved by assigning high cost to all chromosomes containing links that appear in several paths.

Our aim is to find a set of paths by minimizing a penalty function expressing the frequency of occurrence of each link in the entire sets of paths found between all source destination pairs. A similar goal consists of finding a set of link weights to be used as routing metrics (cost metrics) when finding the least cost paths.

A. Genetic optimization

In a genetic optimization setting, the optimum paths are obtained from a chromosome with the least cost. For example if a chromosome consists of paths whose links do not appear in any other paths, this will be the optimum chromosome. Let a chromosome be presented as $H = P_{ij}, \dots, P_{od}$ where P_{od} represent paths chosen randomly from the set of all paths found between the source o and the destination d . We define $E(P_{od})$ as the set of links in the path P_{od} . Now, let $K^* = \bigcup_{\forall P_{od} \in H} E(P_{od})$. A cost c_l is then assigned to each link $l \in K^*$ where c_l is the number of paths in H having l as an edge.

Problem formulation

The problem can now be formulated as an optimization problem as follows:

$$\text{Minimize } G(H) = \sum_{\forall l \in K^*} N^{c_l}, \quad (1)$$

where N is the number of nodes in the network. As expressed by equation (1), the penalty function to be minimized is expressed as a power function known to be computationally expensive. However this function has been selected to differentiate low and high cost paths by setting a big gap between them. Note that a logarithmic transform of this function may lead to an additive composition rule expressed by

$$\text{Minimize } G(H) = \sum_{\forall l \in K^*} c_l \log N. \quad (2)$$

The mathematical formulation above is derived from [1] and illustrated by Figure 1. The objective function to be minimized is presented as follows. Each chromosome is associated with a cost which measures its fitness. A cost refers to a value that the function takes when evaluated in each chromosome and the fitness of a chromosome is a measure of this value. The smaller the value the higher the fitness will be and vice versa.

The genetic algorithm

In this section we present the genetic algorithm that we have used in order to solve the problem in (1). We also present its features as well as issues pertaining to its implementation.

Initialization of population The daunting task in any genetic algorithm is the encoding of the chromosome to represent the relevant problem. In this case each chromosome

is represented by genes (set of one or more bits) which point to actual paths in the look-up table. Figure (1) displays an example of a typical chromosome. Genes of a chromosome are generated randomly with each decimal value of a gene representing the index of the path in a particular source destination pair. In each source destination pair a path is selected randomly and its index is converted to binary and so represents an actual path in the look-up table. A candidate chromosome therefore contains genes, each one of them pointing to a path in all source destination. Therefore, from a single chromosome one gets plausible paths from all source destination pairs in the network.

Crossover and reproduction Crossover is performed according to a certain probability. Unlike other typical crossover where two chromosome exchange one or more bits randomly, in this case exchange of genes is also random but in addition the exchange is done in a way that the identity of a path is kept entirely. This is to avoid a gene pointing to a non-existent path index. A Roulette wheel selection is then applied to reproduce chromosomes for the next generation. This selection method is fitness proportionate, that is to say fitter individuals (with least cost) have a great chance of being selected and passed onto the next generation while weaker ones (with high cost) are less likely to be selected but left out and discarded.

Mutation The mutation operation is carried with a small probability of 0.01; this is in hope that it will strengthen weak individuals. This operation is also aimed at introducing diversity in the population by bringing in features that are not presently in the parent chromosomes. Offspring obtained from crossover are modified by randomly changing some parts of their chromosome bearing in mind that the gene should point to a candidate path in that respective source destination pair.

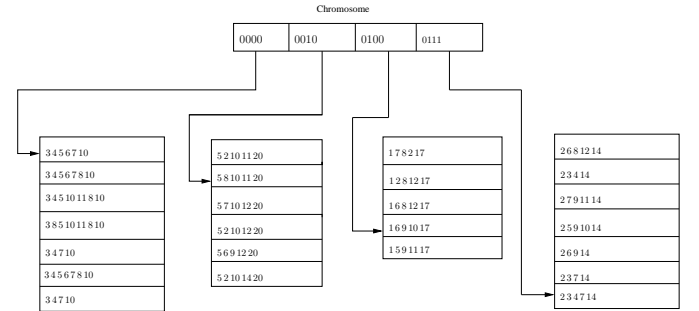


Fig. 1. Typical chromosome: each gene encodes a path corresponding to one source destination pair

B. Memetic Optimization

Let the following functions in equations 3 and 4 define the cost functions of a certain network. We will refer to equations 3 and 4 as power and additive cost functions respectively.

$$\frac{f_l^\alpha}{(F_l - f_l)^{(1-\alpha)}} \quad (3)$$

$$\alpha \times \ln(f_l) + (\alpha - 1) \times \ln(F_l - f_l) + (1 - \alpha) \times \ln(F_l) \quad (4)$$

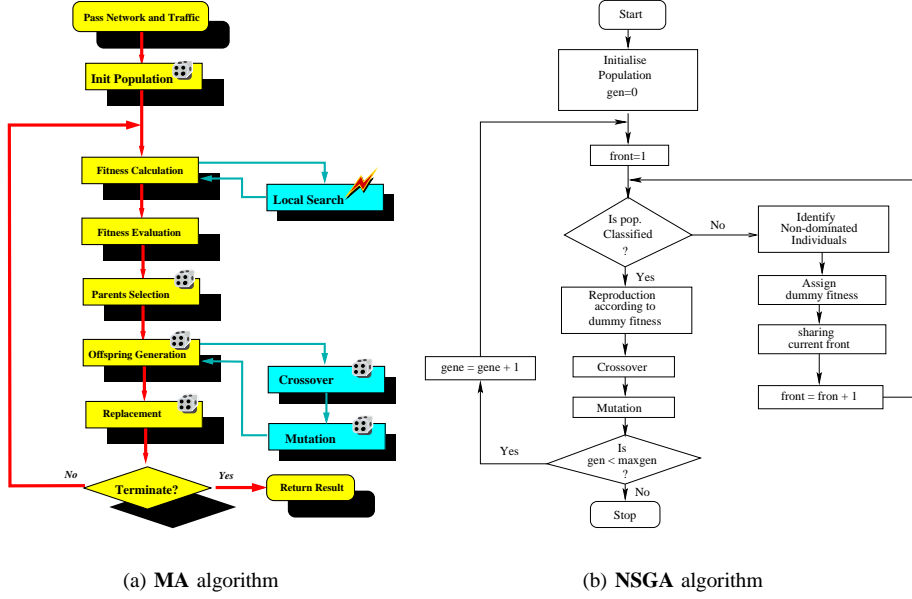


Fig. 2. “Genetic algorithms”

where f_l is the total flow of the link l , F_l is the upper bound of the set of total flows of all links in the network and $0 < \alpha < 1$.

Problem formulation

The main objective is to find a set of link weights that will result into finding optimal paths based on the OSPF protocol. This is achieved by minimizing the functions in equation 3, 4.

The memetic algorithm

Memetic Algorithms (MAs) belong to a class of evolutionary algorithms referred to in this paper as **HybridGA**. They use a hybrid optimization model where local search is used to complement the global search implemented by classical **GAs** to improve the genetic individuals fitness through hill-climbing. Local search is used to speed up the convergence process towards an optimal solution while avoiding premature convergence at the same time. We consider a MA algorithm illustrated by Figure 2(a) where the fitness of each chromosome is evaluated using either of the functions in equations 3 or 4 depending on whether a multiplicative or additive composition rule is used.

III. MULTI-OBJECTIVE OPTIMIZATION

In the multi-objective case, in addition to the objective in equation 1 we add the delay as another objective and try to minimize the two objectives simultaneously. The link delay is taken as in the case of an MM1 queue and it is given by $\frac{f_l}{C_l - f_l}$, where f_l is the total traffic carried by link l and C_l the capacity of the link l .

A. Problem formulation

Formally the multi-objective problem is presented as follows:

$$\text{Minimize } \begin{cases} F(H) = \frac{f_l}{C_l - f_l} \\ G(H) = \sum_{\forall l \in K^*} N^{c_l} \end{cases} \quad (5)$$

To solve the above multi-objective optimization problem we have used a slight modification of the Non-dominated Sorting Genetic Algorithm (NSGA) proposed by Srinivas and Deb [2].

B. Algorithmic solution

NSGA classifies individuals into layers. A stochastic remainder proportionate selection is used before the normal selection takes place. Using Pareto ranking the entire population is ranked on the basis of domination. Non-dominated individuals are given a dummy fitness proportional to the population size and later classified into one category. To maintain diversity in the population, classified individuals are shared in decision space with their dummy fitness values. All classified individuals are then removed from the population and another layer of non-dominated individuals is considered. This process is carried-out repeatedly until all individuals in the population are classified. Figure 2 (b) (taken from Srinivas and Deb [2]) shows the general flowchart of the NSGA algorithm. The following are a few modifications we have made to in NSGA: note that they are mostly based on the encoding of the chromosome.

- **population encoding**: although original NSGA solves problem encoded in both real and binary variable, in our case we first encode all chromosomes as binary variable and the length of a chromosome is not entered as an input since it is

TABLE I
TABLE SHOWING STATISTICS FOR BOTH NETWORKS

Network & function	Parameter	Strong correspo.	Weak correspo.	HybridGA routes	GA routes
Africa(power function)	$\alpha = 0.25$	36 %	0%	32%	32 %
	$\alpha = 0.5$	36%	0 %	32%	32%
	$\alpha = 0.75$	36%	0%	32%	32%
Africa(additive function)	$\alpha = 0.25$	36 %	0%	32%	32 %
	$\alpha = 0.5$	34%	0 %	33%	33%
	$\alpha = 0.75$	36%	0%	32%	32%
USA(power function)	$\alpha = 0.25$	60 %	0%	20%	20 %
	$\alpha = 0.5$	53%	0 %	23%	23%
	$\alpha = 0.75$	60%	0%	20%	20%
USA(additive function)	$\alpha = 0.25$	60 %	0%	20%	20 %
	$\alpha = 0.5$	60%	0 %	20%	20%
	$\alpha = 0.75$	60%	0%	20%	20%

TABLE II

TABLE SHOWING STATISTICS FOR BOTH NETWORKS, DISPLAYING THE MAX AND AVERAGE LENGTHS AND STANDARD DEVIATION

Network	Functions & algorithm	parameter &	Max. length	av. length	stdDev
African	HybridGA(Power)	$\alpha = 0.25$	8	3	2.18
		$\alpha = 0.50$	8	3	2.18
		$\alpha = 0.75$	8	3	2.18
	Hybrid (Additive)	$\alpha = 0.25$	9	4	2.10
		$\alpha = 0.50$	9	4	2.05
		$\alpha = 0.75$	8	3	2.18
	GA		15	5	3.98
USA	Hybrid (Power)	$\alpha = 0.25$	7	4	1.93
		$\alpha = 0.50$	7	4	2.14
		$\alpha = 0.75$	7	4	1.93
	Hybrid (Additive)	$\alpha = 0.25$	7	4	1.93
		$\alpha = 0.50$	7	4	1.93
		$\alpha = 0.75$	7	4	1.93
		GA		15	7

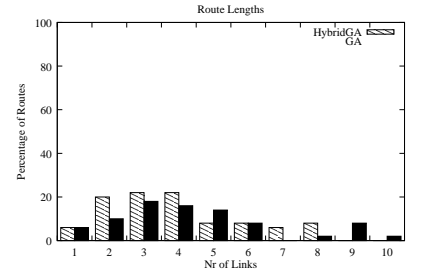
not known before hand because it depends on the size of the network in question.

- **The maximum and minimum values** of the functions to be optimized are also not known a priori, they are therefore not entered as inputs.

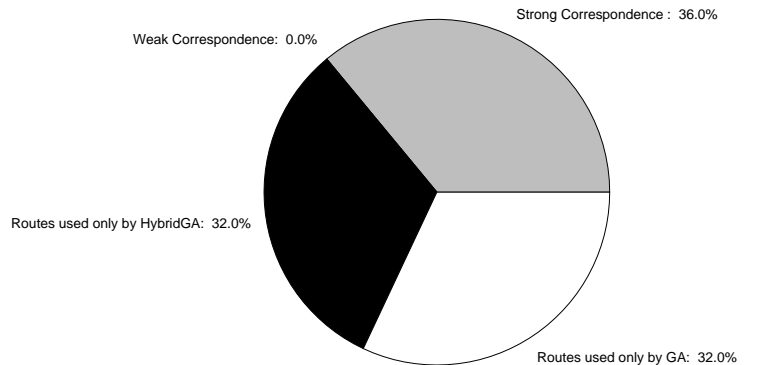
Objectives functions are not entered as row inputs but these are calculated based on the occurrence of an edge in a set of paths to be found at run time.

- **More program features** have been embedded in NSGA. For example modules that read a network topology and demands, those that find all paths leading from each source to destination and those that evaluate the fitness of each chromosome which is later evaluated as the objective function value.

- **Finally the best population** is represented in both real (representing Pareto points) and binary (representing chromosome with acceptable routing paths).



(a) Percentage of route lengths



(b) Routes correspondence

Fig. 3. Routes lengths (a) and routes correspondence (b)

IV. SIMULATION RESULTS

A. Single Objective

We ran both the genetic and memetic algorithms with the following parameters:

Initial population : 100

Number of generations :100

Probability of crossover: 0.9

Probability of mutation :0.01

To solve the problem we simulated these algorithms on two test networks. The African network (see Figure 4(b)) and the

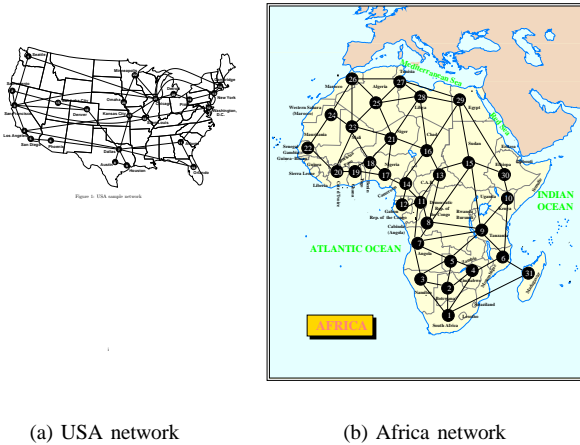


Fig. 4. “The USA (a) and African (b) networks”

USA network (see Figure 4(a)). For simulation purposes we have considered 31 nodes and 48 links for the African network and the USA network has 38 links, 23 nodes. The results found by our genetic algorithm have been compared with those found by the HybridGA that uses cost function described in equations (3) and (4) and then routes the traffic based on the OSPF protocol.

The statistics of paths found by both algorithms on the African and the USA network is displayed in table I and II.

From these tables we can see that the statistics of both networks do not change with parameter α except for $\alpha = 0.5$. This means that choosing α above or below half will not affect the quality of paths to route on.

In column 3 of table I we observe that 60% of routes in the USA network have strong correspondence meaning that 60% of routes of the two algorithms being compared are exactly the same. Column 4 of table I reveals 0% of weak correspondence. This implies that routes found by both algorithms are either the same or completely different. Lastly Column 5 and 6 show the percentage of routes that are exclusively used by each algorithm. Figure 3(b) shows the percentage of routes used by both algorithms (for $\alpha = 0.25$ power function). Looking at table II, we see that GA mostly uses longer paths (at least one more node) than HybridGA. This tells us that GA will not only choose shortest paths at times but mostly avoid congested links as a result of paths it chooses to route on. And in all cases, GA has a larger standard deviation, meaning that most of its paths have lengths that vary greatly from the average length. Figure 3(a) shows the percentage of routes lengths (also for $\alpha = 0.25$ power function)

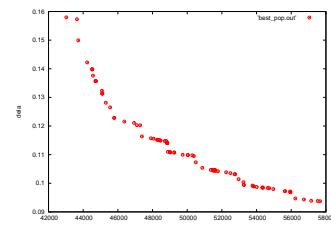
B. Multi-Objective

Like any other evolutionary multi-objective optimization algorithm, there is no optimum solution. The term optimum then will mean finding a set of point(s) which will give all objective functions acceptable values to the decision maker. This set of points is referred to as the set of Pareto points. In case of NSGA we have run the simulation on African

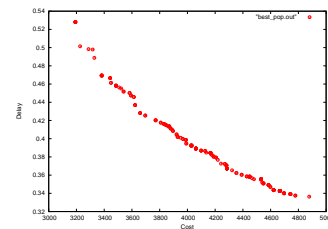
network where we minimized functions in equation 5. We ran the simulation with the following parameters:

Initial population : 200, number of generations :100, probability of crossover: 0.8, probability of mutation :0.01, distribution index for real variable SBX crossover: 5, distribution index for real variable polynomial mutation: 10.

The optimum paths can now be chosen by hand from the set of Pareto points which represent a chromosome fitness. Having a chromosome at hand we can get the paths from its genes that point to actual paths of each source destination pair. Figure 5(a) and 5(b) show the Pareto front obtained from both the African and the USA networks respectively. The simulation was run for several different parameters and it showed that more Pareto points are obtained as the number of generations start reaching 100. However increasing the number of generations to a larger number does not have any positive impact that is worth mentioning here.



(a) Pareto front of the AFRICAN network



(b) Pareto front of the USA network

Fig. 5. Pareto front of the African (a) and USA (b) networks

V. CONCLUSION

We have presented in this paper genetic optimization strategies used to solve single- and multi-objective optimization problems in IP networks. We compared the classical and hybrid genetic algorithms in terms of quality of the paths computed and found that (1) on average the two strategies find up to 60% strongly corresponding paths and (2) the hybrid algorithm finds shorter paths compared to the classical genetic algorithm. The experimental results obtained from the multi-objective solution revealed that the number of Pareto points increase with the number of generations.

The genetic optimization algorithms presented in this paper can be deployed in emerging optical networks to achieve Routing and Wavelength (RWA). The Wavelength assignment problem can be solved using graph coloring techniques. This has been reserved for future research work. Path comparison raises the issue of the interplay between single-and multi-objective optimization to evaluate the similarity between the set of paths found by a single objective optimization model and the set of paths represented by some of the Pareto points. The mechanism used to extract the path sets from a Pareto front has been explained in this paper but the comparison between paths has been reserved for future research work.

REFERENCES

- [1] Nilanhan Benerjee, Vaibhav Mehta and Sugam Pandey, "A Genetic Algorithm Approach for Solving the Routing Wavelength Assignment Problem in WDM Networks", *Indian Institute of Technology*, 2004.
- [2] N. Srivas and Kalyanmoy Deb. "Multiobjective optimization Using Non-dominated Sorting in Genetic Algorithms ", *Evolutionary computing*, 2(3):221-248, Fall 1994.
- [3] C.A Brackett. "Dense Wavelength division multiplexing networks: Principles and applications", *IEE J. Select. Areas Commun.*, vol. 8,pp. 948-964, Aug. 1990.
- [4] B. Mukherjee. "Optical communication Networks", McGraw-Hill, 1997.
- [5] Byrav Ramamuethy and Biswanath Mukherjee. "Wavelength Conversion in WDM Networks", *IEE J. Select. Areas Commun.*, vol. 16,pp. 1061-1073, Sept. 1998.