2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI) Guadalajara, Jalisco, Mexico, November 7-9, 2018

# Traffic Engineering in SDN with Cultural Algorithms

Thyago de Amorim Monteiro
*Polytechnic School of Pernambuco*
*University of Pernambuco (UPE)*
Recife, Brazil
tam@ecomp.poli.br

Edson Queiroz de Albuquerque
*Polytechnic School of Pernambuco*
*University of Pernambuco (UPE)*
Recife, Brazil
edison@ecomp.poli.br

Andson M. Balieiro
*Department of Computing (DC)*
*Federal Rural University of Pernambuco (UFRPE)*
Recife, Brazil
andson.balieiro@ufrpe.br

***Abstract*—The huge Internet traffic growth and the applications heterogeneity have requested great resilience from the communication networks and made the provision of Quality of Service (QoS) to the applications challenging, where the packet routing presents a fundamental role. Currently, the Shortest Path First (SPF) algorithm predominates the routes computation in the networks and it composes the Open Shortest Path First (OSPF) protocol. Although several improvements have been proposed, they are not able to meet the current demands stemming from the transmission of super-connected global events and the live view culture, which have generated data traffic beyond the OSPF forwarding capability. In this regard, this work explores traffic engineering in a software-defined network (SDN) through a hybrid approach based on genetic and cultural algorithms.**

***Index Terms*—routing, load-balancing, genetic algorithm, cultural algorithm**

## I. Introduction

The Open Shortest Path First (OSPF) is the most widespread link state and Interior Gateway Protocol(IGP) in the world [1]. It routes the packets by using the best paths, i.e. those with the lowest costs. The path cost is obtained by summing the costs of each link between the routing nodes (routers). Due to the increase of the traffic forwarding and the increase number of routers in the networks, several changes were incorporated into OSPF, such as Equal Cost Multi Path (ECMP) and Incremental Shortest Path First (ISPF). The former allows OSPF to perform load balancing [2] in a network with homogeneous path costs and, thus, distribute traffic across the network, maximizing its utilization. The latter provides faster protocol converging during link oscillations, because it generates Shortest Path Tree (SPT) updates [3], without needing all the routers from an IGP recalculate their own SPTs. However, these improvements are no longer able to meet the current demands of managing an IGP without the direct intervention of a network administrators team. The diversity of applications in super-connected and global events, and the live-view culture makes dynamic routing challenging and brings the need for ostensible and costly monitoring teams, proportional to the network size.

The network programmability, separation of the control plane from the data plane and the global view of the network provided by the software-defined networks (SDN) paradigm allow the development of new solutions for dynamic packet routing, and less costly and more efficient network management [4]. An example of cost reduction and operational efficiency with the use of SDN is Google's case [4]. In this respect, this work explores the dynamic routing problem in software-defined networks (SDNs) through an evolutionary and self-adaptive approach, based on genetic (GA) and cultural algorithms (CAs). Although the GAs have been widely adopted in the packet routing [5] [6] [7] [8], even in SDN networks [9], our proposal differs from the others, since it combines AG with ACs [10] in order to achieve better load balacing in the network. Through the influence and beliefs functions, cultural algorithms act directly on the evolution (mutation) of the GA population (candidate solutions). The routing convergence time in large networks is a challenge faced by both OSPF and GAs, efficient load balancing on the network, maximizing its utilization and minimizing operational costs are benefits envisioned by our proposal. This work is organized as follows. Section II decribes the proposed approache and its components. Section III presents the evaluation envirorment and obtained results. Section IV concludes this paper.

## II. Proposed Approach

### A. Software Defined Networks

SDN is an emerging architecture for computer networks that allows the dissociation between control plane (responsible for network intelligence) and data plane (responsible for forwarding packets in the network), enabling unprecedented network programmability [11]. It provides a global view of the network, with the control logic placed in a central element, denoted controller, which enables a better management of the routing elements [12]. The communication between data plane and controller is performed via Southbound Application Program Interface (API), where the OpenFlow [13] is the most used protocol. Figure 1 illustrates the comparison between traditional architectures and SDN with the proposed approach (CA and GA).

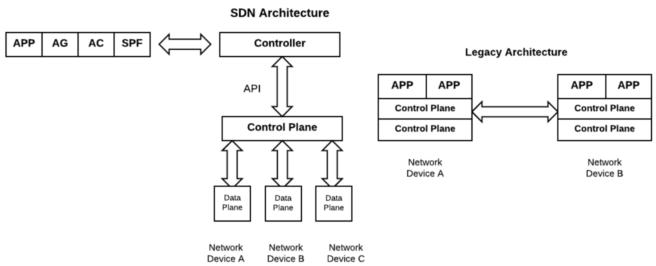


Fig. 1. SDN x Traditional Architecture

With SDN, new solutions for traffic engineering and routing in computer networks have been proposed by academia and industry. The integration between SDN paradigm and computational intelligence techniques brings more adaptive behavior to the network active elements, and provides new mechanisms to deal with problems from traditional networks and those that arise in SDN. [13] [14].

### B. Genetic Algorithm

GA is an optimization algorithm based on Darwin's theory of evolution. It is used to solve complex combinatorial problems by evolving a set of potential solutions (population) for number of iterations (generations). Each candidate solution is called chromosome. Eventually, through the genetic operators (selection, crossing and mutation), an optimal / suboptimal solution can be found with little human intervention [17]. In the proposed approach, the individual or chromosome represents a set of paths, one for each request (see Figure 2), and each individual is evaluated through a fitness function (see Section Fitness Function) that takes into acccount the total cost of the genes (accumulation of latency, occupation of the links, and size of the path) that makes up the individual.

### C. Fitness Function

The fitness Function reads the chromosome and evaluates the population based on the amount of costs to find the minimal cost path, defined as follows:

$$f_i = 1/\sum_{j=1}^{li-1} C_{gi(j),gi(j+1)}$$

Were $f_{\mathrm{i}}$, represents the fitness value of each chromosome ${}_{\mathrm{i}}$th , $l_{\mathrm{i}}$ is the length of the ${}_{\mathrm{i}}$th chromosome, $g_{\mathrm{i}}$(j) represents the gene (node) of the ${}_{\mathrm{j}}$th locus in the ${}_{\mathrm{i}}$th chromosome, and C is the link cost between nodes.

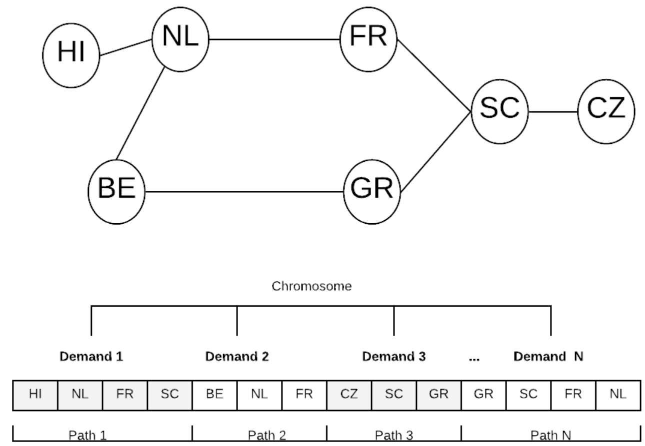


Fig. 2. Chromosome Structure and routing path

### D. Genetic Operators

For the selection operator, which simulates the natural selection mechanism of biological species and chooses the individuals for the next GA phase, we adopted the roulette wheel. It is based on the individual's fitness value in which individuals with greater fitness are more likely to survive for the next generation [17].

The crossover operation is not adopted in our approach, because the heterogeneity in terms of path size demands chromossomes with different sizes in the same population, which would bring more complexity to our solution.

The mutation operator randomly chooses a gene to change, and generates a partial route from that gene to the final destination described in the demand. To this purpose, the topology information is used. After the operation, the individuals are evaluated again in order to find more promising routes (less busy). GAs have been used to define packet routes in computer networks [6] [7] [8], mainly when the search space is large and the Djikstra algorithm becomes slow [15]. Different from other works, our proposal adopts CA in the evolution of the GA individuals in order to increase its efficiency. Next section 2.3 presents the CA.

### E. Cultural Algorithm

The Cultural Algorithm (CA), proposed by [10], is based on the principles of human social evolution. Unlike GA that does not use information from the problem domain during the search process, which may decrease its efficiency, the CA aims to increase the solution quality by inserting prior information about the problem domain, and thus motivate the algorithm to avoid unwanted regions. Figure 3 illustrates the CA operation.

The CA uses a dual inheritance system, maintaining two search spaces. The first is the population space, modeled

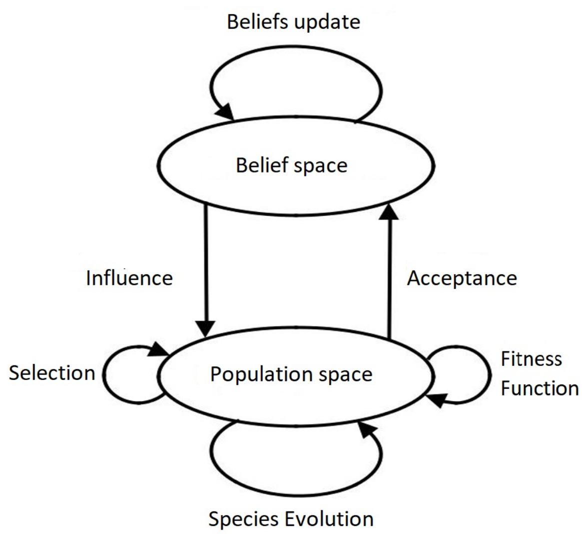


Fig. 3. Cultural chromosome operation

with the GA in our proposal. The second is the belief space, which contains the behavioral traits of the population (prior information). These spaces evolve, influencing each other in the following manner. One selects the individuals that will alter the beliefs and the other determines how the beliefs will modify the behavior of the population. An acceptance function is used to determine which individuals from the population space will influence beliefs. The experience of the accepted individuals will be used to update the beliefs that, in turn, will be used to influence the evolution of the population, simulating cultural evolution. GA genetic operators (population space) will use beliefs to control changes in individuals.

The belief space contains two types of knowledge, situational and normative. The first stores the most suitable solutions for each interaction, while the second one provides the behavior patterns of the individuals, used as guides in the adjustment process. In our approach, normative knowledge will be composed of information such as interconnection utilization (occupation), nodes and links availabilities, and latency. The influence on individuals takes place during the mutation process.

The acceptance function can be modeled using static methods (eg: ranking of individuals), traditional evolutionary algorithms (eg: tournament, elitism, roulette, etc.) or dynamic (number of selected individuals varies over the generations). In our proposal, a benchmarking is carried out to define the best approach to be taken in the problem in question. In terms of influence function, which applies the new culture in the AG population, it will determine the mutation step and the moviment direction.

### *F. Execution Flow*

The execution flow of our proposal is illustrated in Figure 4. Given the requested flow demands, the initial population of the GA is generated randomly with feasible paths. After that, individuals are evaluated by the fitness function, considering the latency, occupation of the links (load) and size of the path (hop count). Following this, the individuals are selected through the roulette wheel operator, and, thus, go to the mutation process, where the interaction between AG and CA happens through the influence function. The CA manipulates some characteristics in order to generate better individuals and avoid local minimums. Simultaneously, verification of network changes (e.g.link failures) are performed and belief space is updated. Afterwards, a new population is obtained, the belief space is updated by the acceptance function and the stop criterion of the algorithm is verified (number of generations). In our approach, we defined 30 cycles. If it has been reached, the best individual of the population is chosen as the solution. Then the controller sends the flows to the switches (updating the switch tables with the routes defined by the algorithm). Otherwise, the process repeats from the fitness evaluation stage.

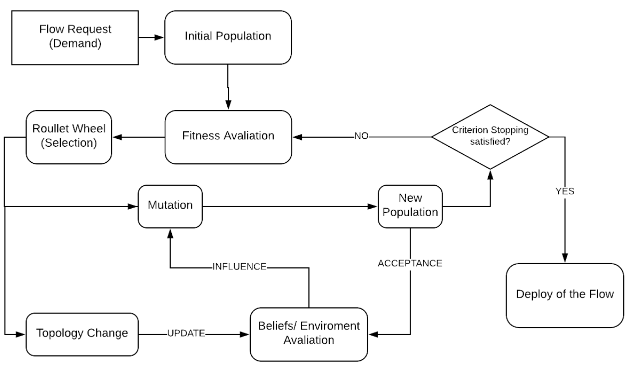


Fig. 4. Cultural Route Path Workflow

## III. Evaluation Enviroment

The proposal is being developed using the NetworkX library [18], which provides the use of data structures for the manipulation of the GML file with network topologies, and reproduces the OSPF protocol. It will be evaluated in comparison to the OSPF and CA, using a simulated environment and the GEANT2 network topology [19], which has 39 nodes, and 61 one-way interconnections (see Fig. 5). The demands will be defined by the following elements: origin, destination and requested bandwidth (in Mbps). Such elements will be defined via probability distributions (eg, uniform and exponential). Metrics such as convergence time, number of saturated bonds and duration of saturation are envisioned to be adopted in the analysis of the approaches.

Since the proposal is under development, the first results are encouraging. Fig. 6 shows that our proposal (with cultural algorithm) provides better load distribution in the network, achieving lower values for average link use than the Dijkstra (SPF) and pure GA approaches.

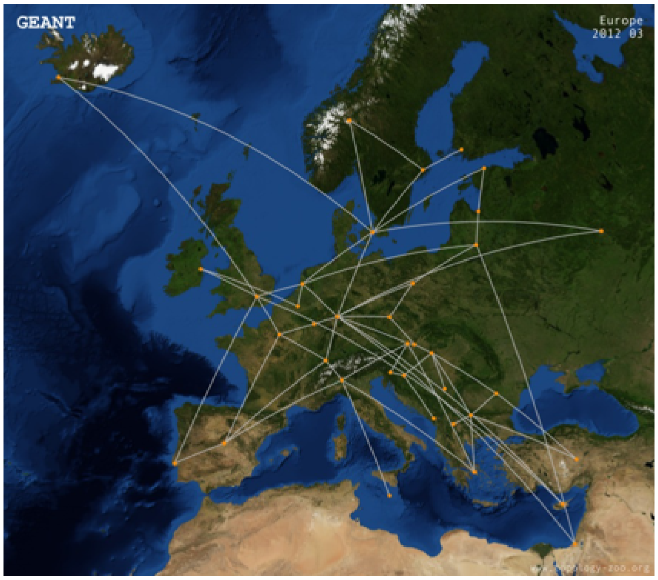


Fig. 5. GEANT Topology

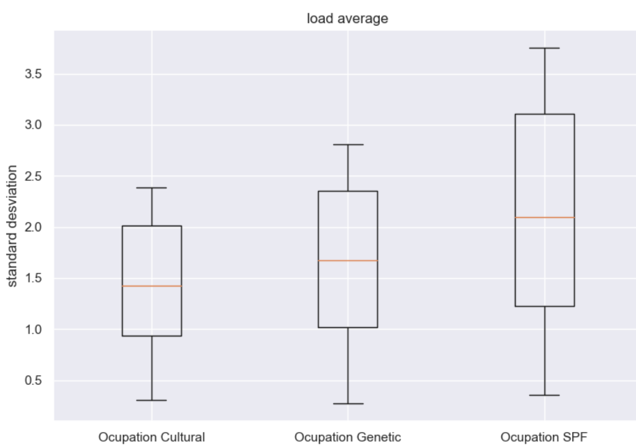


Fig. 6. Load average with Cultural, Genetic and SPF

The Fig. 7 the evolution of the fitness function (defined in Section II.C) across the simulation execution cycles considering three approaches (Cultural, pure GA and SPF). The execution cycles ranges from the initial occupation state of the network to heavy load, with the arrival of new flows to be routed. Since all the algorithms have the same population and are subject to the same routing demands, we observed that the Cultural Algorithm has more expressive results regarding the efficiency when compared to the Djikstra, making fine adjustment of the objective function.

## Conclusion

This paper presented a Cultural Algorithm as routing strategy in SDN. Our proposal adopts the mutation operation working on alternative paths and a influence function that

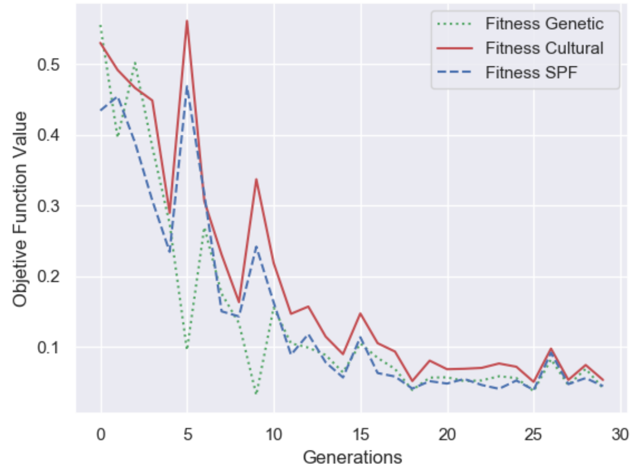


Fig. 7. Objective function with Dijkstra, genetic and cultural algorithms

modifies the network links with high usage to adjust the fitness of the individuals, distribute the traffic over the topology and reduce the network saturation period.

The first obtained results are encouraging. We hope that, through few adjustments, the fitness function can achieve better. Future works include the extensive evaluation of the current proposal and the development of other approaches to solve the dynamic routing problem and achieve more efficient management of the SDN network and reduction of cost operational.To this end, the association between evolutionary computing and deep-learning is an envisioned direction.

## References

[1] Fortz, B. and Thorup, M., 2002. Optimizing OSPF/IS-IS weights in a changing world. IEEE journal on selected areas in communications, 20(4), pp.756-767.

[2] Mehta, A., Bhuktar, A., Santaluri, K. and Karale, S., 2012. A Resolution To The Equal-Cost Multipath (ECMP) Failure In The OSPF Not-So-Stubby-Areas (NSSA). University of Colorado, Boulder, 4.

[3] OSPF Incremental SPF, "https://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/ospfispf.html?dtid=osscdc000283",access in 2018 July.

[4] S. Jain et al., B4: Experience with a globally-deployed software defined wan, in Proc. ACM SIGCOMM Conf., 2013, pp. 314.

[5] Brander AW, Sinclair MC. A comparative study of k-shortest path algorithms. In: Proc. 11th UK performance engineering workshop for computer and telecommunications Systems; September 1995.

[6] Gonen, B., (2006). Genetic Algorithm finding the shortest path in Networks. Reno: University of Nevada. https://www.scribd.com/document/200889781/Gem-3983, access in march/2018.

[7] S. Behzadi, A. Alesheikh,"Developing a Genetic Algorithm for Solving Shortest Path Problem",WSEAS International Conference on Urban Planning and Transportation, Heraklion, Crete Island, Greece, July 22-24, 2008, http://www.wseas.us/e-library/conferences/2008/crete/upt/upt04.pdf, acesso em 2018 mar.

[8] Kumar, R. and Kumar, M., 2010. Exploring genetic algorithm for shortest path optimization in data networks. Global Journal of Computer Science and Technology.

[9] Yu, YunShuai, and ChihHeng Ke. "Genetic algorithmbased routing method for enhanced video delivery over software defined networks." International Journal of Communication Systems 31.1 (2018).

[10] Reynolds, R. G. An introduction to cultural algorithms. In: SINGAPORE. Proceedings of the third annual conference on evolutionary programming. [S.l.], 1994. v. 131139.
[11] Haleplidis, Evangelos et al. RFC 7426. Software-defined networking (sdn): Layers and architecture terminology. Internet Engineering Task Force. 2015.
[12] Latah, M. and Toker, L., (2016). Application of Artificial Intelligence to Software Defined Networking: A Survey. Indian Journal of Science and Technology, 9(44).
[13] Kreutz, Diego et al. Software-defined networking: A comprehensive survey. Proceedings of the IEEE, v. 103, n. 1, p. 14-76, 2015
[14] Latah, M. and Toker, L., (2016). Application of Artificial Intelligence to Software Defined Networking: A Survey. Indian Journal of Science and Technology, 9(44).
[15] Mittal, P. and Singh, Y., 2016. Development of intelligent transportation system for improving average moving and waiting time with artificial intelligence. Indian Journal of Science and Technology, 9(3).
[16] Latah, M. and Toker, L., (2016). Application of Artificial Intelligence to Software Defined Networking: A Survey. Indian Journal of Science and Technology, 9(44).
[17] Linden, R. Algoritmos Genticos. Rio de Janeiro, Brasil: Ciłncia Moderna, 2012.
[18] NetworkX,https://networkx.github.io/, access em 2018 mar.
[19] Geant. GANT topology map - a multi-gigabit pan-european data communications network, reserved specifically for research and education use. 2012. [Online]. access march/2018. Available: http://www.topology-zoo.org/files/Geant2012.gml