Continuously Running Genetic Algorithm for Real-Time Networking Device Optimization

Amit Mandelbaum
Nvidia Networking
Jerusalem, Israel
amitma@nvidia.com

Doron Haritan
Nvidia Networking
Jerusalem, Israel
doronh@nvidia.com

Natali Shechtman
Nvidia Networking
Yokneam, Israel
natalis@nvidia.com

ABSTRACT

- Networking devices operated in ultra-scale data centers must run perfectly and in real-time.
- The optimal configuration depends on the network topology and traffic patterns.
- It is impossible to specify a single configuration that fits all scenarios, and manual tuning is required to optimize the device's performance.
- Zero Touch Tuning (ZTT): a continuously running Genetic Algorithm is designed for online, automatic tuning of the networking device parameters.
- ZTT was tested in a diversity of real-world traffic scenarios, and it was able to obtain a significant performance boost over static configurations.

INTRODUCTION

- In modern data centers, communication between servers is crucial to the operation, and as a consequence, the entire data center performance is highly determined by the performance of its networking devices. [1]
- In this work, we present the Zero Touch Tuning (ZTT) algorithm which is designed to enable continuous operation, rapid reaction, and a low computational cost, which are all characteristics required for running on a Networking Device.

PROPOSED METHODS

- ZTT is a modified version of Generic Algorithm (GA).
- GAs are a heuristic solution search or optimization technique, motivated by the Darwinian principle of evolution through (genetic) selection and survival of the fittest. [2, 3, 4]
- A GA search consists of four phases: initial population generation, selection, crossover, and mutation.
- Modifications to the GA that are done to make ZTT continuously running and to increase its speed:
  - **Removal of the stopping criteria:** Since GA does not run continuously due to stopping criteria. One option to fix it is to use some mechanism that automatically restarts the algorithm. [5] Given the high frequency of changes in traffic, the mechanism will not work.
  - **Continuous selection phase:** GA selection takes place after the crossover and mutation phases. For continuous evaluation at each creation of an individual, the fitness score is compared to the score of the best performing individual to determine the lowest scoring individual.
  - **Refresh of best individual's score:** We update the fitness score of the best individuals before the crossover phase and the mutation phase. However, the traffic pattern might have changed since we last measured the best individual's score. This is done to ensure the comparison is done based on the current traffic pattern.

RESULTS

- We evaluate ZTT in two different scenarios.
- Independent Tests :
  - We examine how ZTT performs on static traffic.
  - In Table1 we see all three algorithms outperform the default static configuration.

Table 1: Results of Independent Tests

Alg.	Full Minute			Last 20 Seconds		
	Rx BW	Tx BW	Drops	Rx BW	Tx BW	Drops
ZTT	0.92±0.10%	4.95±0.23%	20.39±0.89%	0.96±0.16%	4.98±0.05%	22.17±1.58%
SA	0.85±0.12%	4.48±0.48%	17.95±3.52%	0.91±0.15%	4.66±0.63%	19.41±3.28%
RRS	0.81±0.03%	3.69±0.38%	15.79±1.26%	0.81±0.04%	3.71±0.42%	17.18±1.86%

Table 1 legend. Tx BW and Rx BW denote transmit and receive bandwidth respectively. Drops denote packet drop rate. Results represent the average and standard deviation of the percentage of improvement of the default configuration across all tests.

- Continuous Tests:
  - We check all three algorithms with and without the refresh mechanism.

Table 2: Results of Continuous Tests – 30 seconds

Alg.	Without Refresh			With Refresh		
	Rx BW	Tx BW	Drops	Rx BW	Tx BW	Drops
ZTT	0.83±0.33%	4.37±0.22%	20.48±2.76%	0.89±0.16%	4.48±0.82%	21.48±2.67%
SA	0.64±0.28%	3.05±1.31%	16.93±6.57%	0.74±0.3%	4.10±0.62%	18.35±8.7%
RRS	0.64±0.16%	2.55±0.2%	12.56±4.09%	0.66±0.06%	3.99±0.22%	15.81±2.42%

Table 2 legend. Notations and results format are the same as Table 1.

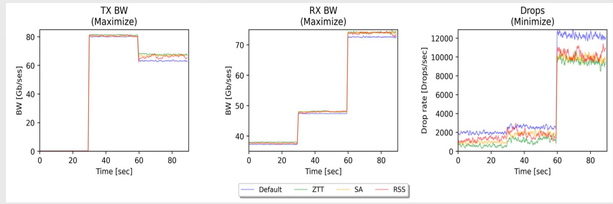

Figure 1: Example of a single run of 3 continuous tests. Performance is shown for the default static configuration. Rx/Tx BW denotes receive and transmit bandwidth respectively. Drops denote packet drops per second. Each test is running for 30 seconds and we can see that traffic patterns change significantly between tests.

CONCLUSION

- In this paper we presented ZTT, an online parameter optimization algorithm that automatically tunes the configuration registers of a Network Device for maximum performance in real-time.
- We showed that several modifications to a simple Generic Algorithm allow the algorithm to work fast, in a continuous manner, and fulfill the requirements needed for an algorithm to run on Network Devices.

REFERFENCES

[1] ChristoforosKachrisandIoannisTomkos.2012. Asurveyonopticalinterconnects for data centers. IEEE Communications Surveys & Tutorials 14, 4 (2012), 1021– 1036.

[2] SS Chaudhry* and W Luo. 2005. Application of genetic algorithms in production and operations management: a review. International Journal of Production Research 43, 19 (2005), 4083–4101.

[3] Mantas Paulinas and Andrius Usinskas. 2007. A survey of genetic algorithms applications for image enhancement and segmentation. Information Technology and control 36, 3 (2007).

[4] Mandavilli Srinivas and Lalit M Patnaik. 1994. Genetic algorithms: A survey. computer 27, 6 (1994), 17–26.

[5] ThomasHeinze,LarsRoediger,AndreasMeister ,YuanzhenJi,ZbigniewJerzak, and Christof Fetzer. 2015. Online parameter optimization for elastic data stream processing. In Proceedings of the Sixth ACM Symposium on Cloud Computing. 276–287.