Network Management Game

Engin Arslan*
SUNY at Buffalo
enginars@buffalo.edu

Murat Yuksel
University of Nevada, Reno
yuksem@cse.unr.edu

Mehmet Hadi Gunes
University of Nevada, Reno
mgunes@cse.unr.edu

This article is an editorial note submitted to CCR. It has NOT been peer reviewed.
The authors take full responsibility for this article's technical content. Comments can be posted through CCR Online.

ABSTRACT

Management and automated configuration of large-scale networks is one of the crucial issues for Internet Service Providers (ISPs). Since wrong configurations may lead to loss of an enormous amount of customer traffic, highly experienced network administrators are typically the ones who are trusted for the management and configuration of a running ISP network. We frame the management and experimentation of a network as a "game" for training network administrators without having to risk the network operation. The interactive environment treats the trainee network administrators as players of a game and tests them with various network failures or dynamics.

Categories and Subject Descriptors

C.2.3 [**Network Operations**]: Network Management—*Network Monitoring*

General Terms

Management, Performance

Keywords

Network Game, Network Training

1. INTRODUCTION

Online management of a running large-scale network poses many challenges that have attracted significant research. Due to the significant costs and risks [1] of building and operating a large-scale ISP network, the challenges in managing the network resources are exacerbated [2] as the margin for error or inefficiency is slim [3, 4]. Today, vital and critical applications such as VoIP, IPTV and financial markets are converging onto the Internet infrastructure, and thus making the job of provisioning high-performance network services an even more important one.

From the technical side, emergence of various substrate networking technologies like 3G wireless and mesh networking is also complicating the management tasks to the extent that network operators give up on optimizing their networks' configuration and barely cope with handling default configuration settings of tremendous number of components involved.

Even though there have been several tools and outcomes [5] to automate the process of large-scale network management, network operators have found themselves more comfortable with trusting highly experienced well-trained human administrators. Highly experienced human administrators are of critical importance as they are typically the only ones who can quickly find the optimum (or close-to-optimum) response to a major failure, e.g., finding an optimum rerouting for a significant amount of traffic on a broken pipe [6].

However, the complexity of the management and configuration problem is increasing as new technologies emerges. Thus, tools to train administrators and achieve automated ways of managing a running network are vitally needed. In this paper, we propose the concept of "network management game" (NMG) that frames the problem of training network administrators in exploring what-if scenarios as a "game". *The NMG aims to establish a game-like environment for trainee administrators to experiment and play with the networks, without having to risk the large-scale network operation.*

Achieving higher utilizations via better load balancing (also known as traffic engineering) is one of the main management problems for network operators. Many algorithms and tools are developed to find optimal or close-to-optimal solutions for high network utilization; however, they may not be completely relied upon for handling extreme cases or large scale failures.

A common implication of load balancing for network administrators is to configure interior gateway protocol (IGP) link weights so that shortest paths give result to a well-balanced traffic load on network links. Several prior studies employed advanced optimization techniques to set the IGP link weights for a given topology and traffic matrix to improve various network performance metrics such as delay, throughput, or congestion [5, 7, 8, 9].

In this paper, we discuss application of our network management game framework to the problem of IGP link weight setting. The player/user[1] interfaces with the animator and inputs new IGP link weights as new configurations. The animator, then, conveys these new configurations to the simulation engine to produce real time results.

In Section 2, we provide an overview of current practices in training a network administrator and expansion of network infrastructure. In Section 3, we discuss the general framework of the network management game. In Section 4,

---

*Mr. Arslan was with the University of Nevada, Reno when this work was performed.

[1] The words "user" and "player" refer to each other throughout the paper.

we analyze the sample experimental setup of load balancing via IGP link weight adjustment. Finally, we conclude with future directions in Section 5.

2. BACKGROUND

Currently, most of the basic training for a network administrator is performed by means of well-defined certification procedures [10]. The administrators receive months of education and take exams to obtain these certifications to prove that they have the basic skills and knowledge about configuring and administering a network. However, custom skills related to maximizing performance of a particular operational network cannot be attained via generic certifications. Such custom skills require long hours of training in work environment where the certified trainee can learn what to do in real-time from peers with more experience on that particular network.

Customized training of network administrators involves what-if analysis [11], which is done by in-house tools according to operators we have contacted. What-if analysis is a brainstorming activity that uses extensive questioning to guess potential failures and issues in a system, and ensure that appropriate precautions are taken against those problems.

What-if analysis for a network typically involves a comparison of the network's current performance and "would-be" performance under certain scenarios such as some particular link/node failures, traffic spikes or a new pipe installation between two points. Network managers regularly use such analysis to quantify robustness or riskiness of the network at hand, in order to answer questions like "Is my network robust enough for potential failures?" or "Where should I put more capacity in my network?". Investment decisions, planning and dimensioning on a network are heavily guided by technical what-if analysis.

Businesses may use the scenario manager tool of Excel to explore different scenarios such as the decision making process in e-commerce [12]. Although what-if analysis has high impact within business intelligence platforms, its usage is extended for several purposes such as hazard analysis [13], index selection for relational databases [14], and multi-tier systems [15]. For instance, in [16], authors developed a tool (named, WISE) that predicts how a deployment of a new server to an existing CDN affects service response time. They use machine learning techniques to process old dataset and discover the dependencies among system variables. Then, using these dependencies and new dataset which is representative of new deployment, WISE predicts how response time could be affected when deployment changes are done.

Furthermore, various tools have been developed to guide investments and determine how to improve network performance [11] with minimal investments. Though existing what-if analysis tools are pretty successful in helping a network administrator and strategic director make an informed decision about future investments, they cannot train for dynamic events such as demand spikes or failures. A key difference in our approach is the capability of simulating the interactivity and dynamism that might take place in an operational network.

The concept of using a virtualized game-like environment for training has been done in the past [17, 18]. It has especially been used for cases where experimentation with the real system is too costly or risky. Military training involves a lot of such practices, e.g. pilot training [19], commander training [20]. Financial investment training [21] is another venue where a game-like environment can be used to train before deploying money on stock market. To our knowledge, use of game-like environments has not been adapted to train network administrators.

3. NETWORK MANAGEMENT GAME (NMG) FRAMEWORK

The NMG framework has two parts as shown in Figure 1. On the backend, we use a simulation engine to imitate a real network. We interface an animator graphical user interface (GUI) to the backend simulation engine to visualize simulation events and to provide interactivity to the player. The player makes changes on the GUI, which are taken to the simulation engine on real-time.

In order to synchronize the GUI and the simulation engine, we need to establish a two-way message passing interface between them. Briefly, once the engine receives the initial configuration file, it starts the simulation and generates the event traces. Via the first communication interface, we will transfer these traces to GUI, which will then animate them to the user. When the user makes a change on a particular link's weight, the engine is informed about it through the second interface. Once the engine receives the changes, it recalculates the routes based on these changes and carries on to generate more traces.

3.1 The Animator

We need to design a GUI which animates the condition of the simulation on real time and provides interactivity via changing link weights. As the simulation engine runs, event records are generated and sent to the GUI. Then, the GUI processes these events to visualize them.

The player's goal in NMG is to maximize the network's throughput by manipulating link weights. To help user in tracking the ongoing condition of the simulated network, the links are colored to represent instant utilization as in Figure 2.

We can increase or decrease the width of the links based on their bandwidth to make it easier for the user to realize the links with larger bandwidth. When the user moves the mouse on a link, the traffic flows on that link can be observed as well as its current utilization and bandwidth. The links

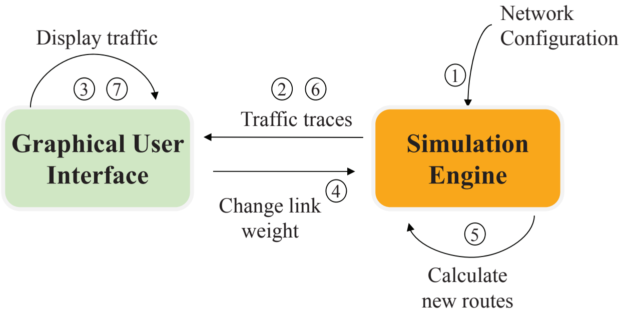

Figure 1: Block diagram of Network Management Game (NMG) components. The sequence (4)-(7) repeats whenever the player makes changes.

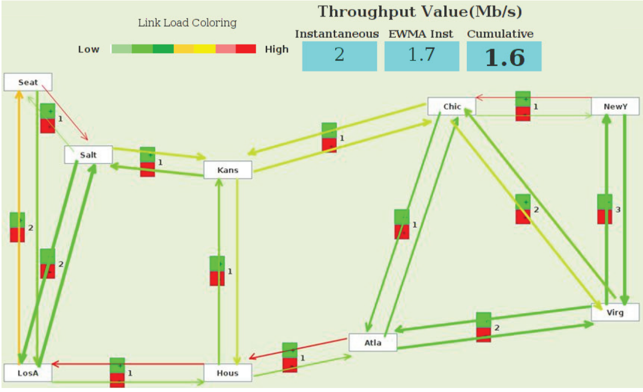

Figure 2: Animation on Internet2 backbone.

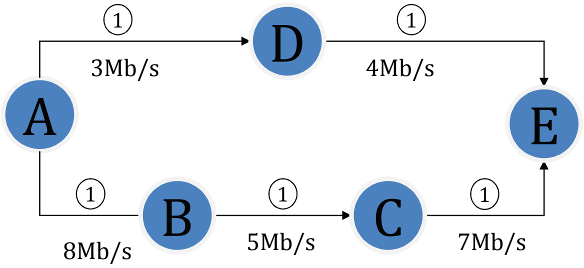

Figure 3: IGP path selection example. Initial link weights are same. Traffic flows from A to E.

are colored with respect to their current utilization where links with no traffic on it are green colored whereas completely utilized links are red colored. For instance, in the Figure 2 there are four links that are congested as a result of four bottleneck links for four flows.

We may also monitor cumulative and instantaneous network throughput to evaluate the player's performance. The cumulative throughput, τ_{cum}, is mainly used to see how the player performed in the overall test whereas the instantaneous throughput, τ_{ins}, is used for measuring how well the player responds to changes in the network such as failures or demand spikes. We calculate the throughput values as follows:

$$\tau_{cum} = \frac{\sum_{i=0}^{t_{total}} P_s(i)}{t_{total}}$$

$$\tau_{ins} = \frac{\sum_{i=t_1}^{t_2} P_s(i)}{t_2 - t_1}$$

where t is the current time in seconds from the start of the game and $P(t_1, t_2)$ is the total size of packets successfully transferred to destination(s) within the time period $[t_1, t_2]$. While all transmitted packets are taken into the consideration for τ_{cum}, τ_{ins} considers only the packets transmitted in $[t_1, t_2]$.

3.2 Interaction between Simulation Engine and Animator

A crucial challenge in designing the NGM framework is to synchronize the simulation engine and the GUI. Normally, the simulation engine will be able to simulate a given network configuration very fast; however, such speedy simulation is unrealistically fast for the user as he/she cannot keep track of the simulation.

To obtain a more realistic and interactive environment, we need to reduce the speed of the simulation engine. The engine will wait for a fixed amount of time before processing the next event. We may tune the wait time to so that the engine runs concurrently with the GUI. When the player changes a link weight, the change is sent to the engine and new routing paths are calculated. Thus, the simulation resumes with new network configurations.

4. SAMPLE EXPERIMENTAL SETUP

The end goal of our Network Management Game (NMG) framework is to train people in network management and administration. To observe potential benefits of NMG in training, we have considered a sequence of test scenarios on a difficult network management problem, i.e., intra-domain traffic engineering.

4.1 Game Goal

The Interior Gateway Protocol (IGP) is an intra-domain routing protocol which uses link-state costs (a.k.a. link weights) to determine end-to-end shortest paths. A typical ISP network management problem is to tune the IGP link weights so that end-to-end shortest paths change, and thus the load on individual links are balanced.

Figure 3 illustrates a simple test scenario where link weights cause traffic to shift from one end-to-end path to another. When the traffic from A to E is considered, the traffic will follow the path A-D-E by default because its total weight is less than that of the alternative path A-B-C-E. However, available bandwidth on the path A-B-C-E is larger than A-D-E path in this scenario. Thus, the player is expected to increase the total weight of the default path such that the traffic flows through the higher bandwidth path.

An intuitive way of tuning the IGP link weights for load balancing is to increase the link weights on the links that are highly utilized so that flows are diverted to low utilization links. However, association of link weights to observed link loads automatically is known to cause instability in routing, and hence, is avoided in practice [6, 22].

The goal of a player is to maximize the aggregate network throughput by manipulating the IGP link weights. The player is given a network topology/scenario with an initial configuration and allowed to increment or decrement each bi-directional links weight by clicking a red or green button on the corresponding link as in Figure 2. As the game progresses, the player can change the link weights as many time as he/she desires to adjust route of flows.

4.2 Game Stages

Though it is possible to set up an experimental process, a simple one would be a four-stage process to observe efficacy of NMG in training network management skills:

1. Tutorial
2. Testing (Before Training)
3. Training
4. Testing (After Training)

We first give a tutorial to the player about the environment so that the player knows what the game is about and how the GUI works. Then, we test the player with a relatively complex network scenario to observe how he/she performs before the training. Next, we train the player with a sequence of scenarios. Finally, after the training stage is over, we expose the player to a test scenario, which is of similar difficulty to the test scenario before the training stage. By comparing the player's performance before and after the training stage, it is possible to make an observation on the effectiveness of the training.

One should use different training and test scenarios. The training scenarios can be different from each other in terms of complexity and number of traffic flows. The purpose of the training scenarios is to train the users with simple networks.

When designing the testing stages (Before and After Training), we should produce scenarios that are noticeably more difficult than the training scenarios. This reduces the dependence of training to the specific scenarios and reveals how much the player learned the skill rather than the specific scenarios or networks.

5. SUMMARY AND FUTURE WORK

In this work, we introduced a game-like environment, i.e., the Network Management Game (NMG) framework, to train people in terms of network management skills. This position paper is a first step in exploring the concept of using games to train network administrators.

Understanding if NMG can help improve reflexes of the players (e.g., the response time of players to dynamic events like link failures) is a critical future work. For example, it will be interesting to see how well players can get trained against demand spikes occurring due to temporary changes to the traffic matrix of the network. Such demand spikes are one of the typical situations where highly experienced administrators (i.e., the target community of NMG) are needed.

Furthermore, using metrics other than throughput, such as delay or congestion, for evaluating the performance of players is also a worthy future work. In this paper, we thought that maximizing a value would give a better "game-like" feeling to the player than minimizing a value. Since both delay and congestion would require the player to minimize a value, we picked the throughput as the metric for the game. However, other metrics may be of more relevance for certain situations. For example, quality-of-service (QoS) metrics such as delay and loss are highly relevant to the practice of network operation, particularly when some customers are promised higher priority service.

Several open questions arise when one is to extend NMG to more realistic settings and larger networks. Such extensions involve non-trivial steps such as visualization of larger topologies and more traffic flows. Alternative approaches include (a) *large screens* to visualize larger networks (e.g., NYC subway [23, 24]), (b) *distributed visualization* of a network divided into multiple zones each being handled by a different player on a separate screen with backend communication among zones (e.g., air traffic control [25], power grid [26, 27]), and (c) *multi-level visualization* with zooming capabilities [28].

Another challenge is the visualization and handling of many traffic flows by a single player. Clearly, aggregation of multiple flows into one will be necessary so that the player can plausibly relate to the visualized traffic. The visualization granularity can range from link-level utilizations (i.e. complete ignorance of flows) to separate colored line for each flow. Selection of the right visualization approach will depend on the size of the network at hand and the particular management problem being tackled. For instance, the first approach will probably suffice to address the visualization of the IGP link weight setting on the inter-PoP topology of a large ISP backbone. Regardless, the specifics of the user interface and the visualization granularity will have to be tuned for the particular cases.

Finally, it is certainly possible to extend the NMG concept to design dimensions with more fun. For instance, one may design a multi-player strategy game involving multiple network managers each trying to grow their networks while competing with other neighbor networks – similar to the competition among ISPs on the Internet. Further, simplified versions of NMG can be considered for mobile handhelds where the reflex of the player to various network dynamics is of interest.

Acknowledgments

This work is supported in part by the NSF awards 0721600 and 0721609.

6. REFERENCES

[1] P. Srinagesh, *Internet economics.* MIT Press, 1997, ch. Internet cost structures and interconnection agreements, pp. 121–154.

[2] Z. Kerravala, "As the value of enterprise networks escalates, so does the need for configuration management," Enterprise Computing and Networking, The Yankee Group, January 2004.

[3] W. Enck, T. Moyer, P. McDaniel, S. Sen, P. Sebos, S. Spoerel, A. Greenberg, Y.-W. E. Sung, S. Rao, and W. Aiello, "Configuration management at massive scale: System design and experience," *IEEE JSAC*, vol. 27, no. 3, pp. 323–335, April 2009.

[4] D. Oppenheimer, A. Ganapathi, and D. A. Patterson, "Why do Internet services fail, and what can be done about it?" in *Proc. of USENIX USITS*, 2003.

[5] T. Ye, H. T. Kaur, S. Kalyanaraman, and M. Yuksel, "Large-scale network parameter configuration using an on-line simulation framework," *IEEE/ACM Trans. Netw.*, vol. 16, pp. 777–790, August 2008.

[6] E. J. Anderson and T. E. Anderson, "On the stability of adaptive routing in the presence of congestion control," in *Proc. of IEEE INFOCOM*, 2003, pp. 948–958.

[7] B. Gonen, M. Yuksel, and S. Louis, "Probabilistic trans-algorithmic search for automated network management and configuration," in *Proc. of IEEE GLOBECOM MENS Workshop*, Miami, FL, 2010.

[8] B. Fortz, "Internet traffic engineering by optimizing OSPF weights," in *Proc. of IEEE INFOCOM*, 2000, pp. 519–528.

[9] L. S. Buriol, M. G. C. Resende, C. C. Ribeiro, and M. Thorup, "A hybrid genetic algorithm for the weight setting problem in OSPF/IS-IS routing," *Journal of Combinatorial Optimization*, vol. 6, pp. 299–333, 2003.

[10] "Cisco Certifications," http://www.cisco.com/web/learning/index.html.

[11] *Mate*, http://www.cariden.com/products.

[12] H. K. Bhargava, R. Krishnan, and R. Müller, "Electronic commerce in decision technologies: A business cycle analysis," *Int. J. of Electronic Commerce*, vol. 1, pp. 109–127, June 1997.

[13] P. Baybutt, "Major hazards analysis an improved process hazard analysis method," *Process Safety Progress*, vol. 22(1), pp. 21–26, 2003.

[14] S. Chaudhuri and V. Narasayya, "Autoadmin what-if index analysis utility," in *Proc. of ACM SIGMOD*, 1998, pp. 367–378.

[15] S. Chen, K. R. Joshi, M. A. Hiltunen, W. H. Sanders, and R. D. Schlichting, "Link gradients: Predicting the impact of network latency on multitier applications," in *Proc. of IEEE INFOCOM*, 2009, pp. 2258–2266.

[16] M. Tariq, A. Zeitoun, V. Valancius, N. Feamster, and M. Ammar, "Answering what-if deployment and configuration questions with wise," in *Proc. of the ACM SIGCOMM*, 2008, pp. 99–110.

[17] R. E. Chatham, "Games for training," *Communications of the ACM*, vol. 50, pp. 36–43, 2007.

[18] M. N. Nicolescu, A. Olenderski, R. Leigh, S. J. Louis, S. Dascalu, C. Miles, J. C. Quiroz, and R. Aleson, "A training simulation system with realistic autonomous ship control," *Computational Intelligence*, vol. 23, no. 4, pp. 497–519, 2007.

[19] "Wikipedia: Flight Simulator," http://en.wikipedia.org/wiki/Flight_simulator.

[20] "Serious Games by BreakAway," http://www.breakawaygames.com.

[21] "The Stock Market Game," http://www.stockmarketgame.org.

[22] Z. Wang and J. Crowcroft, "Analysis of shortest-path routing algorithms in a dynamic network environment," *ACM CCR*, vol. 22, pp. 63–71, 1992.

[23] "Visualization of NYC Subway Rides," http://data.fabernovel.com/nyc-subway/.

[24] "Musical Visualization of NYC Daily Schedule," http://mta.me.

[25] R. Azuma, I. Neely, H., M. Daily, and R. Geiss, "Visualization tools for free flight air-traffic management," *IEEE Computer Graphics and Applications*, vol. 20, no. 5, pp. 32–36, 2000.

[26] F. Wu, K. Moslehi, and A. Bose, "Power system control centers: Past, present, and future," *Proc. of the IEEE*, vol. 93, no. 11, pp. 1890–1908, 2005.

[27] T. Overbye and J. Weber, "Visualizing the electric grid," *IEEE Spectrum*, vol. 38, no. 2, pp. 52–58, 2001.

[28] D. Shelley and M. H. Gunes, "Inner sphere network visualization," *Computer Networks*, vol. 56, no. 3, pp. 1016–1028, February 2012.