“Running” ModelGraft to Evaluate Internet-scale ICN

Michele Tortelli†, Dario Rossi†, Emilio Leonardi⋆

†Telecom ParisTech, Paris, France - first.last@telecom-paristech.fr
⋆Politecnico di Torino, Torino, Italy - first.last@tlc.polito.it

ABSTRACT

The analysis of Internet-scale Information-centric networks, and of cache networks in general, poses scalability issues like CPU and memory requirements, which can not be easily targeted by neither state-of-the-art analytical models nor well designed event-driven simulators.

This demo focuses on showcasing performance of our new hybrid methodology, named ModelGraft, which we release as a simulation engine of the open-source ccnSim simulator: being able to seamlessly use a classic event-driven or the novel hybrid engine dramatically improves the flexibility and scalability of current simulative and analytical tools. In particular, ModelGraft combines elements and intuitions of stochastic analysis into a MonteCarlo simulative approach, offering a reduction of *over two orders of magnitude in both CPU time and memory occupancy*, with respect to the purely event-driven version of ccnSim, notably one of the most scalable simulators for Information-centric networks.

This demo consists in gamifying the aforementioned comparison: we represent ModelGraft vs event-driven simulation as two athlets *running a 100 meter competition using sprite-based animations.* Differences between the two approaches in terms of CPU time, memory occupancy, and results accuracy, are highlighted in the score-board.

CCS Concepts

•**Networks** → **Network performance analysis;** *Network simulations;* •**Computing methodologies** → *Modeling methodologies; Discrete-event simulation;*

Keywords

Information-Centric Networks; Internet-scale; Hybrid; Open-Source Software;

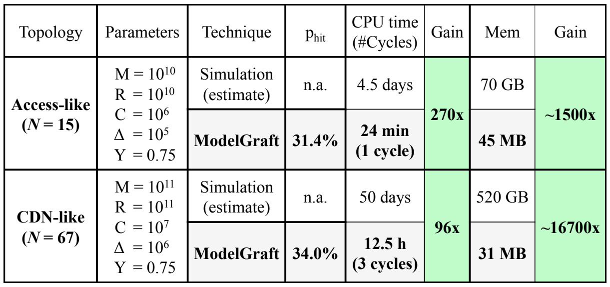
1. INTRODUCTION

For the last decade, research on Information Centric Networking (ICN) has proceeded in parallel with the content (re)volution of the Internet network, designing and evaluating alternative solutions to the current host-centric approach of the TCP/IP model. With the possibility for end-users' applications to directly access named contents, as opposite to addressable entities, and by introducing in-network transparent caching, ICN offers a better infrastructure to support mobile connectivity, flow/congestion control, and traffic reduction. Despite these fascinating expectations, advocated by numerous works in literature [1], and despite the availability of a fair number of tools [2], there is still considerable skepticism on ICN effectiveness, due to the different orders of magnitude that named contents have w.r.t. addressable hosts. Internet Service Providers (ISPs) require tools for an assessment of the expected return of their potential investment in ICN, before going to a fully-fledged ICN deployment, planning, and management.

From an ISP perspective, the above question mandates dealing with more than 10^{12} objects. Already enumerating these objects is a daunting task: uniquely distinguishing 10^{12} objects requires names that are at least 5 Bytes long, resulting in a requirement of 5 TBytes of RAM. It follows that simulating very large ICN networks, with large caches and huge catalogs, is not only unfeasible due to hardware limitations (e.g., memory bottleneck), but also due to extremely long computational times (e..g, CPU bottleneck) needed to simulate a sufficient number of statistically relevant requests [3]. Moreover, simply downscaling the scenario by jointly reducing catalog cardinality and cache size would introduce excessive distortion in the gathered Key Performance Indicators (KPIs) [3]. These unprecedented difficulties in assessing ICN benefits in realistic scenarios have, also, effects on the time to market for an ISP.

As a workaround to the aforementioned scalability limit, we propose a new hybrid technique, namely ModelGraft [3], which exploits a synergy between stochastic analysis of Least Recently Used (LRU) caches [4, 5] and MonteCarlo approaches based on Time-To-Live (TTL) caches [6]. Our method, implemented in the last version of the ccnSim simulator [7], leverages the intuition that the behavior of realistic networks of caches, regardless of their complexity, can be well represented by means of much simpler Time-To-Live (TTL)-based caches, where the eviction timer is set using the characteristic time T_C of Che's approximation [4]. The complexity is, also, reduced by downscaling the original catalog in a way that preserves its key statistical properties. Given that T_C in complex scenarios is not known a priory, our system uses a feedback loop to iteratively converge to the correct T_C values, even when the initial guess is hugely

ACM 978-1-4503-2138-9



Topology	Parameters	Technique	p_{hit}	CPU time (#Cycles)	Gain	Mem	Gain
Access-like (N = 15)	M = 10^{10} R = 10^{10} C = 10^{6} Δ = 10^{5} Y = 0.75	Simulation (estimate)	n.a.	4.5 days	**270x**	70 GB	**~1500x**
		ModelGraft	**31.4%**	**24 min (1 cycle)**		**45 MB**	
CDN-like (N = 67)	M = 10^{11} R = 10^{11} C = 10^{7} Δ = 10^{6} Y = 0.75	Simulation (estimate)	n.a.	50 days	**96x**	520 GB	**~16700x**
		ModelGraft	**34.0%**	**12.5 h (3 cycles)**		**31 MB**	

(a)

(b)

Figure 1: Demo highlights: (a) ModelGraft results and projected gains vs event-driven simulation in Internet-like scenarios, (b) Demo graphics (©Konami) example.

different from the true value. ModelGraft allows to reduce CPU time and memory usage by over two orders of magnitude, while limiting accuracy loss to less than 2%, with respect to classic event-driven simulation.

2. DEMO HIGHLIGHTS

The goal of this demo is to let users "gamifying" their experience in evaluating the performance of Internet-scale cache networks; indeed, for certain scenarios, ModelGraft ends its execution within few tens of seconds, thus obtaining results even on the fly. By providing a predefined set of different scenarios, obtained by varying catalog cardinality, cache size, cache decision policy, and so on, users can interactively select the one of interest and assist to the "competition" against ModelGraft and classic event-driven simulation. At the end of the competition, they can easily infer the winner by looking either at the finish line, or at the score-board, where relevant KPIs will be shown.

Evaluation Tools, Scenarios, and KPIs. The presented demo challenges ModelGraft performance against the event-driven version of ccnSim, which is already considered as one of the most scalable ICN simulators [2]. ModelGraft, as part itself of an integrated system released with the last open source version of ccnSim, can be seamlessly selected as an evaluation tool among others, like event-driven simulation or analytical models [7].

The demo offers the possibility to choose between different scenarios, which differentiate themselves according to catalog cardinality, topology, cache decision policy, and so on. The two evaluation tools compete against each other on the base of two KPIs: CPU *time*, which is the overall time needed to complete the simulation, and *memory occupancy* (i.e., the total allocated memory required to execute simulations). Regardless of the winner, a third KPI is monitored in order to check the fairness of the competition, that is the *accuracy*; in particular, the accuracy loss of ModelGraft with respect to simulation is shown (i.e., $|p_{hit}^{Simulation} - p_{hit}^{ModelGraft}|$). Tab. 1(a) reports an example of ModelGraft projected gains for Internet-like scenarios; in this case, KPIs related to the event-driven simulator are only forecasted by fitting a model against 50 different scenarios [3] due to CPU and memory limits.

Visualization. The aforementioned comparison is presented in the form of an animated sprint race, as reported in Fig. 1(b), where sprite-based animations are realized in pure HTML and JavaScript. ModelGraft and event-driven simulator are represeted as two athletes characters, with a distinctive vintage look that recall console based games of the 80s. Users first select details of the scenario (e.g., catalog, network size, etc.) on which the competition will be based. When the competition starts, each character runs at a speed that depends on the execution speed of the corresponding simulation engine running in the background – i.e., the faster the tool, the faster the character will run. Throughout the race, execution time, memory occupancy, and accuracy loss are shown on the scoreboard.

Acknowledgments

This work has been carried out at LINCS (http://www.lincs.fr). This work benefited from support of NewNet@Paris, Cisco's Chair "NETWORKS FOR THE FUTURE" at Telecom ParisTech (http://newnet.telecom-paristech.fr). Any opinion, findings or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of partners of the Chair.

3. REFERENCES

[1] G. Xylomenos et al. A survey of information-centric networking research. *Communication Surveys and Tutorials, IEEE*, 16(2), 2014.

[2] M. Tortelli, D. Rossi, et al. ICN software tools: survey and cross-comparison. *Elsevier Simulation Modelling Practice and Theory (SIMPAT)*, 63:23 , 2016.

[3] M. Tortelli, D. Rossi, et al. ModelGraft: Accurate, Scalable, and Flexible Performance Evaluation of General Cache Networks. In *Proc. of International Teletraffic Congress (ITC)*. 2016.

[4] H. Che, Y. Tung, et al. Hierarchical web caching systems: Modeling, design and experimental results. *IEEE JSAC*, 20(7):1305, 2002.

[5] V. Martina, M. Garetto, et al. A unified approach to the performance analysis of caching systems. In *Proc. of IEEE INFOCOM*. 2014.

[6] N. Fofack, P. Nain, et al. Performance evaluation of hierarchical TTL-based cache networks. *Elsevier Computer Networks*, 65:212 , 2014.

[7] http://perso.telecom-paristech.fr/~drossi/ccnSim.