Pedestrian Crossing: The Long and Winding Road toward Fair Cross-comparison of ICN Quality

Michele Tortelli⋆, Dario Rossi†, Gennaro Boggia⋆, Luigi Alfredo Grieco⋆
⋆Politecnico di Bari, Bari, Italy - first.last@poliba.it
†Telecom ParisTech, Paris, France - first.last@telecom-paristech.fr

***Abstract*—While numerous Information Centric Networking (ICN) architectures have been proposed over the last years, the community has so far only timidly attempted at a quantitative assessment of the relative quality of service level that users are expected to enjoy in each of them.**

This paper starts a journey toward the cross comparison of ICN alternatives, making several contributions along this road. Specifically, a census of 20 ICN software tools reveals that about 10 are dedicated to a specific architecture, about half of which are simulators. Second, we survey ICN research papers using simulation to gather information concerning the used simulator, finding that a large fraction either uses custom proprietary and unavailable software, or even plainly fails to mention any information on this regard, which is deceiving. Third, we cross-compare some of the available simulators, finding that they achieve consistent results, which is instead encouraging. Fourth, we propose a methodology to increase and promote cross-comparison, which is within reach but requires community-wide agreement, promotion and enforcement.

I. Introduction

ICN attracted considerable attention in the last decade [10], with numerous architectural proposal emerging, featuring rather different designs in terms of naming, routing and deployment strategies. While these numerous proposals *enriched* the ICN research, the study of each architecture in isolation leads to an *impoverishment* of the overall knowledge of the ICN domain, as comprehensive cross comparison of architectural models within the ICN domain is still lacking.

While the survey and taxonomy presented in [10] constitutes a first attempt in this direction, it is however limited at a *qualitative* description and comparison of pros and cons of the different architectures, while a *quantitative* comparison is still, again, lacking.

In this position paper, we present a state of the ICN research along this perspective, proposing ways to let move forward from this empasse, which would lead to hopefully to a hearty and healty scientific debate in the ICN community.

Complementarily to the focus of [10], we conducted a preliminary survey of the software tools available for ICN performance evaluation, that we are unable to report for lack of space, but that is available for the interested reader at [1]. The survey includes all open source ICN software, including prototypes, emulation tools, simulators, etc.

As we find in [1], 9 of them pertain to a specific architecture, namely Content Centric Networking (CCN)[7], whereas the remaining 10 tools are spread over 5 architectures. Additionally, while tools in the ICN set are essentially *prototypes*, for

QSHINE 2014, August 18-20, RHODES, Greece

DOI 10.4108/icst.qshine.2014.256305

CCN, specifically, all tools, including prototypes, emulators, and simulators are represented. Finally, about half of the tools are CCN simulators, which thus represent the largest category of the ICN software spectrum.

From this preliminary analysis, it follows that whereas cross-comparison of ICN architectures would be desirable, it is however very complex, as it forcibly involves an experimental approach with instrumentation of several testbeds running multiple prototype implementations. Cross comparison of ICN, at this stage, is thus very costly both in terms of the physical infrastructure, as well as in terms of the human effort involved in setting up and running the experiments. Conversely, we argue that comparison of multiple strategies within a specific architecture, namely CCN, albeit having a more narrow focus, is both feasible and relevant. Feasibility follows from the availability of multiple tools, and especially simulators. Relevance follows from the significant and increasing attention that has been devoted to this specific architecture.

In this paper, we take the long and winding road that lead to a thorough cross comparison of architectures in the ICN domain. On the one hand, we drive through the very first mile of the road, that forcibly passes through the cross comparison of strategies and algorithms in the more focused CCN scope. On the other hand, we closely look at trajectories of fellow colleagues crossing this road, surveying good (and bad) publishing practices in the ICN community. During this trip, we also plan possibly useful directions that can lead the community up to the last mile, avoiding previous pitfalls in this insidious trip of fair assessment of the quality in ICN – so fasten your seat belt to drive along this winding road, and look carefully before crossing if you are a pedestrian.

II. Motivations

Despite this work relates to ICN, we argue that the scientific literature which is closest to our work is represented by effort applying the same *methodology* and spirit to other areas, whose findings consitute our main motivation. We are surely not the first authors, nor the first community, addressing the issue of fair comparison of scientific work. Yet, in the networking community, this seems to be a rather recurring, and unfortunately neverending, research issue. While researchers use a multiplicity of tools to conduct their own research, and should always retain free to select the best tool to fit their need, however this practice makes it harder to reproduce results in the community. This is pretty well known and has produced research on new techniques to make experimental research results reproducible (or confutable; see for instance [5] and references therein), as well as suggestions so as to incentivate reproducibility by conditionally accepting papers to top venues (e.g., see for instance the argument of runnable papers to enlarge the ACM SIGCOMM tent in [11]).

It follows that part of the research effort has been focused on calibration and cross-validation of the tools we use in our daily activities. As simulation is a popular tool for scientific evaluation, cross-comparison of simulation tools has been done in several domains such as TCP/IP [16], [12], [3], Wireless Sensor Networks (WSN) [8], [15], Peer-2-peer (P2P) [9], etc.

Nevertheless, most of the work focus on the scalability assessment [8], [12], [16], while fewer address the accuracy and consistency of results across multiple tools. Those who are brave enough to address accuracy comparison often reveal large discrepancies in the results [3], [15], that could question entire lines of work. At the same time, in the networking community it is rare to have broad work aimed at validating or confuting previous research results, as it is more common in medical science and phyisics, since the accent is often put on the novelty of the proposal, rather than on the solidity and verifiability of the investigation.

This is exemplified by findings in [9], that considers 9 popular P2P simulators and surveys 280 papers on P2P topic; the discovery is that only about 20% of papers used one of these 9 popular simulators, wheres the vast majority of papers either claimed to use a custom proprietary simulator, or did not even made an explicit mention – implying that P2P simulation results were generally reported in the literature in a fashion that precludes any reproduction, validation or invalidation.

III. Pedestrian crossing

In similar spirit to [9] we start by overviewing papers in the ICN literature[1]. In paricular, of the 58 papers conducting an evaluation of ICN performance, we choose the 44 ones that specifically employ *simulation tools*. Clearly, with respect to [9] that overviews P2P research in (or just after) its climax, this work is still comparatively premature – as research in ICN is, though fast growing, still younger. As such, the amount of papers we overview is limited, but nevertheless allows to gather interesting insights – that are very similar to that in [9].

The pie chart in Fig. 1 reports the results of the survey, where the label *Custom* includes both papers in which the authors claimed to use their custom simulator, and papers in which generic tools, like Matlab, Omnet++, ns-2, ONE and BitTorrent simulators were mentioned without any indication about the required modification and/or without any reference to the used code. What emerges is that: (i) there is a variegated set of ICN simulation tools, of which the most popular is ndnSIM (18%); (ii) about 2/3 of presented results in the surveyed papers are not reproducible, because either the authors have used a custom simulator (i.e., 48%), or they have not even specified the tool used for the evaluation part of their proposal

[1]Namely, we include the following series of ICN workshop or conferences having an ICN session: ACM SIGCOMM ICN, IEEE INFOCOM NOMEN, ACM/IEEE NoM, IEEE ICNP, IFIP Networking, IEEE CCNC, CFI and NoF

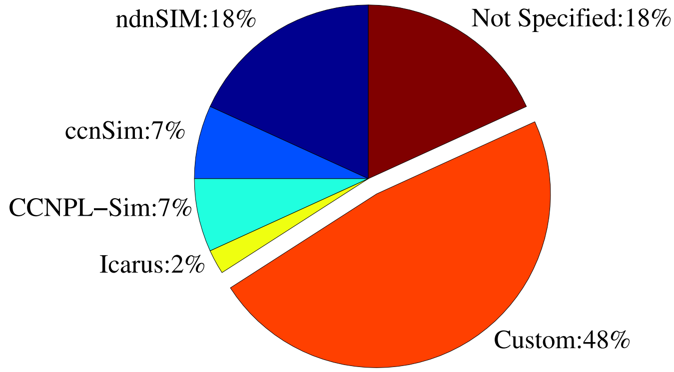

Fig. 1: ICN Simulation and Emulation tools used in the surveyed papers

(18%). While these numbers are slightly more encouraging than [9], we stress once more that the situation can evolve, and diverge, with more simulator being used without any clear indication of their soundness. In the remainder of this paper, we propose ways for ICN research to avoid being stuck in unpleasant scenarios like the one depicted in [9].

IV. The long and winding road

As previously outlined, about half of the availabile software tools for ICN pertains to CCN [7], and about half of the CCN tools are simulators. This is not surprising, as despite the importance of real experiments, simulations are still worth to be used because of their good compromise between cost and complexity. Still, even narrowing down our scope to the state of CCN simulators, we find that many algorithms that are proposed in the community are hard to compare, because they may not be available in the same simulator: yet, narrowing down the comparison to a very small set of naive approaches, would avoid the very same cross-comparison goal. We therefore propose a rigorous methodology to collect, at the same time, a consistent but broad performance evaluation of the ICN quality gathered with different approaches implemented in heterogeneous tools.

A. Terms of comparison

Let for the sake of clarity identify with **CCN strategy** ($\mathcal{C}$) a selection of algorithms to be used in a CCN network (e.g., routing, forwarding, caching decisions and replacement policies) that are available in one or more **Simulators** ($\mathcal{S}$), and denote with **Network scenario** ($\mathcal{N}$) the set of exogenous factors (e.g., catalog skew, workload, network topology, number of simulated events, etc.) under which the strategy is simulated.

As a first step, we select a subset of simulators (namely ndnSIM, ccnSim and Icarus) that offer at least two different CCN strategies to compare (summarized in Tab. I). In particular, regarding the forwarding strategies, Shortest Path (SP) assumes that a control plane routing protocol (like Named-data Link State Routing Protocol (NLSR) [6]) proactively disseminates name-level reachability to build FIBs of

TABLE I: Considered CCN Strategies ($\mathcal{C}$) and Simulators ($\mathcal{S}$)

	Forwarding strategies	**Decision policy**	**Cache replacement**
ndnSIM	SP, SmartFld [17]	LCE,FIX [2]	LRU
ccnSim	SP, NRR [4]	LCE,FIX [2]	LRU
Icarus	SP	LCE,FIX [2], ProbCache [14]	LRU

nodes. Smart Flooding (SmartFld) [17] couples routing and forwarding as the path is reactively discovered on the data plane. Nearest Replica Routing (NRR) [4], instead, is an ideal strategy in which Interest packets are forwarded toward the nearest, possibly off path, content replica. We next consider, as cache admission policies, either Leave Copy Everywhere (LCE), where a copy of the retrieved content is systematically cached in every traversed node along the reverse path, FIX [2], where the content is cached with a fixed probability P, and ProbCache [14], where the probability depends on the distance that the content has traveled. Finally, we set caching replacement to the well known Least Recently Used (LRU).

As a networking scenario, we consider a 10x10 grid topology, where a single repository, randomly placed in each of the 10 simulated runs, stores the entire catalog of $M = 10^4$ contents. We randomly place 30 clients which issue requests for contents characterized by a Zipf's probability distribution, that is $P(X = i) = i^{-\alpha} / \sum_{j=1}^{M} j^{-\alpha}$, with an exponent $\alpha = 1$. The considered cache to catalog ratio is $C = 0.005$, and the number of simulated events is $R = 10^7$. To avoid any bias, all the simulations have been performed using the same machine (8-core Intel Xeon at 3.6 GHz, equipped with 32 GBytes of RAM, and with Ubuntu Linux 12.04) running ndnSIM v0.5.1, ccnSim v0.2, and Icarus v0.1.1.

We then proceed in two steps: we first verify consistency of the simulators when using the limited subset of common strategies in the same scenario; we then extend the comparison to a more rich set of strategies. In other words, we compare homogeneous and heterogeneous strategies implementing the same scenario over different simulators, that we treat separately in what follows. The overall goal of these two steps is to find a common baseline strategy worth to be implented in all simulators to simplify cross-comparison.

B. Homogeneous comparison

First, we select a subset of simulators ($\mathcal{S}$), where we can implement at least two different CCN strategies $c_1, c_2 \in \mathcal{C}$. In this first phase, we compare the performance that the same CCN strategy $c \in \mathcal{C}$ over identical networking scenario $n \in \mathcal{N}$, has in different simulators $s_1, s_2, s_3 \in \mathcal{S}$. In particular, from Tab. I, it emerges that SP with LCE and LRU (c_1), and SP with FIX and LRU (c_2) are implemented on all simulators.

More formally, denote performance of metric X gathered under CCN strategy c, on network scenario n over simulator s as $X^{c,n,s}$. The aim of this section is to compare $X^{c,n,s}$, $X^{c,n,s'}$, and $X^{c,n,s''}$ for different c. In particular, we consider the *hit probability* and *hit distance* (that relates to the quality of ICN performance) and *network load* (that relates to the cost for ISP, and is calculated by counting the number of generated *Interest* and *Data* packets in the whole network).

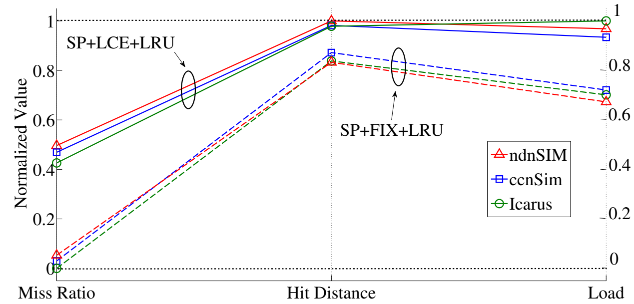

Fig. 2: Homogeneous comparison of the three CCN simulators through shared strategies, i.e., c_1=SP+LCE+LRU and c_2=SP+FIX+LRU.

Results are reported in Fig. 2, where we show multiple metrics X at the same time by using a parallel coordinates graph, where each metric is normalized to their respective observed maximum value $X^{c,n,s} / \max X^{c,n,s}$. To better separate the curves, we report the *miss probability* as $1 - hit^{c,n,s} / \max X^{c,n,s}$ so that a 0 miss corresponds to the highest *hit probability*: by doing so, the *lower the curve, the better and more desirable the strategy.*

What emerges from Fig. 2, is that all the simulators provide very consistent results, despite heterogeneous implementations, codebases and the use of different tracing systems. This represents a comforting result, considering the disagreement experienced in other networking areas [3], [15] and that could be tied to several reasons, such as e.g., bugs in the software implementation, poor entropy of the random number generator, etc. A case of results disagreement, in fact, would clearly have implied that the long and winding road to a fair and comprehensive cross-comparison in ICN is also very steep, and the trip would have likely have stopped here.

Instead, since results are in agreement across different simulators, this allows us to conclude that available open source CCN simulators yield comparable results on different but accurate implementations of the same strategy, and that scenarios are thus well calibrated across simulators.

C. Heterogeneous comparison

Given results of the homogeneous phase, we could expect that, provided that a strategy is correctly implemented over all simulators, it would again lead to similar results across simulators. This clearly would extend the boundaries of our comparison, allowing us to find additional and relevant candidates for the comparison.

More formally, the homogeneous comparison has shown that $X^{c,n,s} \approx X^{c,n,s'} \approx X^{c,n,s''}$ yield similar results for the strategy c across heterogeneous simulators s, s', and s''. We now argue that if a strategy c' is only implemented in s but not in s', provided that the implementation is correct, then we would have $X^{c',n,s} \approx X^{c',n,s'}$ also for heterogeneus strategies. If this holds, it follows that we can compare

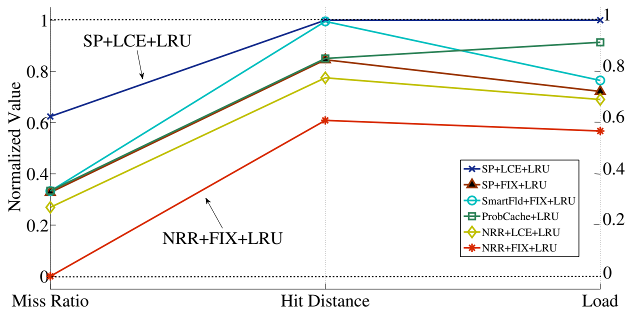

Fig. 3: Heterogeneous comparison of the three CCN simulators through different strategies.

performance of different strategies even though they are not implemented in all simulators: e.g., we could compare $X^{c,n,s}$ (or equivalently $\approx X^{c,n,s'}$) against $X^{c',n,s'}$ and $X^{c'',n,s''}$.

We have performed an exhaustive comparison of the possible combinations from Tab. I. The most interesting candidates are reported in Fig. 3, where curves related to homogeneous scenarios are drawn using results gathered from ndnSIM only, while the remaining ones are characteristic of each simulator. Again, recall that the lower the curve, the better the quality.

From Fig. 3, it emerges that *naive strategies* (e.g., SP+LCE+LRU), that are usually considered as terms of comparison significantly underestimate ICN quality and do not represent a useful baseline for comparison. Rather, novel proposals should aim at achieveing performance that are as close as possible to the quality achieveable through *ideal strategies* (e.g., NRR+FIX+LRU). Notice also that under this light, limitedly focusing on a single aspect (e.g., SP vs NRR as in [4]) could lead to underestimation of ICN quality are achievable by *jointly deploying multiple proposed strategies* (i.e., comparing NRR+LCE vs NRR+FIX).

V. Moving forward

As ICN research matures, evaluation and cross-comparison of the different proposals becomes an important, imperative, topic. We notice that, while the current situation is not as bad as for other communities, much road still has to be done: a significant part of research is still unverifiable, as it is carried out using proprietary or unknown tools.

While the scientific community should retain freedom to use custom tool, it would be good practice (i) to make the source code available (for verifiability), an even better practice would be (ii) to systematically release runnable scenarios metadata (for replicability). Additionally, (iii) the implementation of an agreed minimum set of algorithms is a needed step to promote cross-comparison, and (iv) a cross-calibration on some reference scenario and algorithm should be considered as a mandatory step for any new simulator.

We finally argue that new proposals should be contrasted to performance of *ideal* schemes, rather than being benchmarked against *naive* schemes, as it is usually the case nowadays. For caching studies, that are abundant in the literature, we individuate such comparison point as NRR+FIX+LRU, a combination of probabilistic metachaching (where new content does not systematically yield multiple unnecessary evictions) with ideal routing schemes [4] (where the closest, possibly off-path, cached copy can be accessed through an oracle). While NRR is not a practical scheme, as it requires instantaneous knowledge of the status of all caches in the network, it is however worth considering as: (i) it provides an ideal performance upper-bound, (ii) does not requires settings/tuning, (iii) has been proposed by researcher outside the ICN domain.

Clearly, identification of even better strategies than NRR+FIX+LRU, and comparison against NRR+FIX+LRU in the meanwhile is a community-wide process. This requires effort from developers, rigour from users, promotion from standard bodies and enforcement by peers. Promotion of this approach by normalization bodies, such as IRTF ICNRG that is already tackling the issue of a definition of baseline scenarios [13], is an important step. To facilitate adoption in the community, increasing the availability of the promoted scenarios and of the baseline comparison strategies, in open source simulators would surely constitute a beneficial step. Evaluation on agreed scenarios and strategies should be considered as *necessary*, though not sufficient, steps. In other words, acceptance of papers in ICN venues should require a *minimum* amount of verifiable information (e.g., show the results of the custom simulator on a standard scenario), or the use of a tool validated by the community (e.g., by work like ours). Finally, promoting cross-comparison in academic venues could be achieved through challanges, such as the Sort Benchmark in the SIGMOD community (http://opendatachallenge.org/). All these steps are necessary to move forward in this journey.

References

[1] http://telematics.poliba.it/images/file/tortelli/icn-2014.pdf.

[2] S. Arianfar and P. Nikander. Packet-level Caching for Information-centric Networking. In *ACM SIGCOMM, ReArch Workshop*, 2010.

[3] S. Jansen and A. McGregor. Performance, validation and testing with the network simulation cradle. In *IEEE MASCOTS*, 2006.

[4] S. Fayazbakhsh et al. Less Pain, Most of the Gain: Incrementally Deployable ICN. *ACM SIGCOMM*, 2013.

[5] N. Handigol et al. Reproducible network experiments using container-based emulation. In *ACM CoNEXT*, Dec. 2012.

[6] A. Hoque et al. NLSR: Named-data Link State Routing Protocol. In *ACM SIGCOMM ICN Workshop*, 2013.

[7] V. Jacobson et al. Networking Named Content. In *ACM CoNEXT*, 2009.

[8] J. Lessmann et al. Comparative study of wireless network simulators. In *Proc. of IEEE ICN*, Apr. 2008.

[9] S. Naicken et al. The state of peer-to-peer simulators and simulations. *SIGCOMM Comput. Commun. Rev.*, 37(2):95–98, Mar. 2007.

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

[11] N. McKeown. Enlarging the sigcomm tent. *SIGCOMM Comput. Commun. Rev.*, 44(1):63–65, Dec 2014.

[12] D. Nicol. Scalability of Network Simulators Revisited. In *CNDS*, 2003.

[13] K. Pentikousis et al. Information-centric Networking: Baseline Scenarios. Internet Draft, Mar. 2014. Expires: Sep. 3, 2014.

[14] I. Psaras, W. Chai, and G. Pavlou. Probabilistic in-network caching for information-centric networks. In *ACM ICN Workshop*, 2012.

[15] A. Stetsko, M. Stehlik, and V. Matyas. Calibrating And Comparing Simulators for Wireless Sensor Networks. In *IEEE MASS*, 2011.

[16] E. Weingartner, H. vom Lehn, and K. Wehrle. A performance comparison of recent network simulators. In *IEEE ICC*, 2009.

[17] C. Yi, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang. Adaptive forwarding in named data networking. *SIGCOMM Comput. Commun. Rev.*, 42(3):62–67, Jun. 2012.