PViz: Visualising P2P Multi-Agent Simulations (Demonstration)

Dimitris Giouroukis[1] Nikos Platis[1] Christos Tryfonopoulos[1,2]
[1]University of the Peloponnese, Greece
[2]Max-Planck Institute for Informatics, Germany
{dgiouroukis, nplatis, trifon}@uop.gr

ABSTRACT

Simulators are an inseparable part of the design and evaluation of distributed multi-agent protocols. In this work we put forward PViz, a novel visualisation tool built for one of the most prominent, publicly available P2P simulators, PeerSim. Our tool provides network visualisation for different overlays, interaction with the simulation through scenario re-playability and stepwise execution, and intuitive visualisation features such as panning and zooming in/out of the network, color-coding of nodes, and event alerting.

Categories and Subject Descriptors

I.6.8 [**Computing Methodologies**]: Simulation and Modeling—*Visual*

General Terms

Experientation, Measurement

Keywords

peer-to-peer, multi-agent, simulation visualisation, PeerSim

1. INTRODUCTION

Over the past few decades, multi-agent systems, followed later by the peer-to-peer (P2P) paradigm, have revolutionalised system design by shifting the attention from monolithically designed systems to approaches consisting of a collection of simple interacting components. The flexible system architecture offered by these approaches, together with the low maintenance and administration costs and the self-organising properties inherent in the architectural design, have intrigued both academia and industry into contributing a vast amount of research results and deployed prototypes. Following the P2P explosion era, multi-agent applications that scale to millions of users and resources have been developed in domains like file-sharing [1], voice and video distribution [7], and distributed search engines [2, 17]. Lately, P2P concepts are also being introduced in the –clearly different– cloud paradigm [10, 21, 13, 23] in an effort to exploit the benefits of both technologies, and in the context of distributed online social networking [14, 16, 9], aiming at solving content ownership and scalability issues.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

Concept and approach. Gaining insight into the designed distributed protocols and assessing their efficiency and effectiveness in a real environment is, especially in the early stages of development, not feasible. For this reason, researchers and developers typically resort to *simulations*, carried out on a home-brewed or publicly available simulator, in order to evaluate the large-scale properties of their design. *PeerSim* [18] is one of the most versatile and popular simulators for P2P systems, offering both cycle-based and event driven simulations, various peer churn models including peer joins, leaves, and failures, extensive action logging, and a scalable design that allows the deployment of very large simulations. For these reasons, PeerSim has been used to experimentally evaluate the P2P protocols of over 250 scholarly papers in the area of P2P multi-agent systems, and is now part of the course material of many advanced undergraduate and postgraduate courses [4, 3, 6, 5].

In this work we propose a *visualisation tool*, coined PViz, that allows the graphical representation of the peer network and the execution process of a subset of protocols supported by PeerSim (i.e., unstructured as in [12], semi-structured as in [19, 20], and the fundamental structured protocols Chord [22] and PGrid [8]), and supports many useful operations such as *zooming in* on a specific area of the network, *node monitoring/highlighting*, single- or multi-step *re-execution* of the simulation, and appropriate *visual representation* of the overlay network (e.g., spring model for unstructured networks or circular placement for Chord overlays).

To the best of our knowledge, PViz is the first effort to provide a visualisation component for a publicly available –and widely adopted– general-purpose simulation environment. Our approach leverages and extends the usefulness of PeerSim by providing means to graphically represent and interact with the supported protocols and user-defined setups.

Only a couple of approaches in the literature [11, 15] share similar goals with PViz. [11] is a research report outlining a visualisation tool for the Chord DHT; however the tool is tied to a home-brewed implementation of a Chord simulator, which makes it unusable for anyone but the authors, and the visualisation provided is specific to the Chord overlay and does not support other types of networks. In a different path, [15] proposes VizPub, a simulation visualisation tool that (i) focuses on the visualisation of publish/subscribe-style data dissemination and is thus unusable for the majority of multi-agent simulations (e.g., simulations for complex-adaptive systems or emergent properties), and (ii) emphasises performance analysis through the visualisation of metrics such as message load and the creation of related graphs.

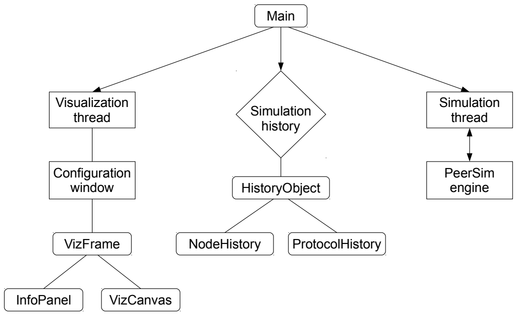

Figure 1: PViz architecture.

Application scenarios. As a typical application scenario for PVIZ, consider a researcher or a student in the area of distributed multi-agent systems who wants to use a simulator to test his newly designed protocol. She resorts to PeerSim and implements her protocols, only to find out that things are not working as they should be. In such a case, a tool that allows her to visualise the different steps of the protocol, replay certain parts of the simulation, and examine the overlay either at per node or at a global level, would be a valuable resource, beyond anything supported by current simulation environments. In another application scenario, an academic teacher would resort to such a tool to explain tricky distributed protocols in an interactive way, encourage student involvement in the learning process, and visualise how protocol changes affect the overall system behaviour.

2. OVERVIEW OF PVIZ

Architecture. Implementing a visualisation tool on top of PeerSim was not trivial, since PeerSim simulations run uninterrupted for a specified number of steps; this is in sharp contrast with the needs of a visualisation tool that requires stepwise, forward and (especially) backward, playback of the simulation.

The solution that we adopted was to allow the simulation to complete, and record, at each step, the state of the overlay; therefore, our visualisation tool runs on top of the simulation *history* (Fig. 1) and can traverse it according to the wishes of the end user.

In order to facilitate the visualisation, we record for every node at each step additional network data to that maintained by PeerSim, in a `HistoryNode` structure. Furthermore, we maintain the *difference* between the successive steps of the simulation history, which greatly simplifies the visualisation, especially for large networks.

Visualisation. PVIZ is built upon the Piccolo2D graphics library, which possesses several capabilities useful for the task, especially zooming and node selection, while being well integrated with Java Swing.

The basic functionality offered by PVIZ consists of:

- The ability to replay the simulation at will. Events (steps) of the simulation are numbered in sequence and the user is able to go to a specific event in order to observe the overlay; she is also able to easily move, either one or several steps at a time, forward or backward through the simulation history.
- The ability to select a node of the overlay in order to observe its connections. By default, only the connections of the selected node are drawn in order to unclutter the display. Connections are depicted using appropriate methods relevant to each visualised protocol (e.g., Bezier curves for Chord fingers as shown in Fig. 2). Additionally the user is able to select adjacent nodes easily, using the keyboard, so as to gain better insight into the structure of the network.
- The ability to pan and zoom in or out on the network visualisation intuitively using the mouse; during this interaction any selected nodes are retained. Therefore the user is able both to have a general overview of the network structure and to focus on interesting areas or locate important nodes accurately, especially in dense networks.
- Nodes are colour-coded depending on the protocol and user preferences, and additional data about them is displayed on the application window. Also information about the changes of the event being displayed are given in textual form (e.g., "Node added").

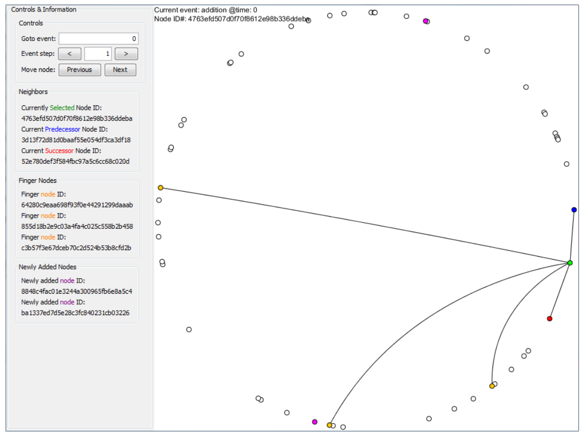

Figure 2: Visualising a Chord protocol simulation.

Further Work. We are currently working on enriching PVIZ with semantic zooming, comprehensive event timelines, and more structured overlay protocols.

More information about the project may be found at `www.uop.gr/~nplatis/pviz/`.

3. DEMONSTRATION PLAN

In our demonstration, we will present a full-fledged usage scenario for the PVIZ prototype and invite attendees for a hands-on experience. Attendees will be assisted to setup a PeerSim simulation for their network of preference among the ones supported by PVIZ and decide the appropriate parameters (e.g., number of nodes, size of routing table). Subsequently they will execute the simulation and launch the PVIZ tool to visualise the different simulation steps.

They will be shown how to monitor the life of a node through the visualisation environment, and discover its neighbouring nodes by simply clicking on the node at hand. They will also be able to use the pan and zoom feature to better examine a cluster of closely grouped nodes, and select some of them for further examination through mouse or keyboard interactions. Subsequently, they will be prompted to explicitly jump to a simulation event, move back and forth in the simulation (both one and multiple steps at a time), and examine how node churn affects the structure and connectivity of the overlay. They will also have the opportunity to visualise the routing table of nodes and watch it get modified due to peer churn. Finally, they will be guided through other intuitive features of the visualisation, like node color-coding and textual messages/alerts, and shown how to modify them at their preference.

A snapshot of the visualisation for a Chord overlay is shown in Fig. 2, were we have focused on key information about a given node. Based on our preferences, the currently selected node is coloured green, its successor red, its predecessor blue, and its fingers yellow. The unique identifiers of all these nodes are setup to appear on the information area at the left-hand side of the window so as not to clutter the display of the network itself. Furthermore, nodes that were added during the current event are also chosen to appear in purple. Stepwise execution of the simulation is, at the current snapshot, set at executing one event at a time.

REFERENCES

[1] BitTorrent webpage. `http://www.bittorrent.com`. Accessed: 2015-01-13.

[2] FAROO webpage. `http://www.faroo.com`. Accessed: 2015-01-13.

[3] IT Architecture for Networking Services Course at Telecom SudParis, France. `http://www-inf.it-sudparis.eu/COURS/ASR/`. Accessed: 2015-01-13.

[4] Open Distributed Processing Course at University of Oslo, Norway. `http://www.uio.no/studier/emner/matnat/ifi/INF5040/`. Accessed: 2015-01-13.

[5] Peer-to-Peer and Agent-Based Computing Course at University of Aberdeen, UK. `http://homepages.abdn.ac.uk/w.w.vasconcelos/pages/teaching/CS4027/`. Accessed: 2015-01-13.

[6] Peer-to-Peer Systems Course at University of Pisa, Italy. `https://elearning.di.unipi.it/moodle/course/view.php?id=41`. Accessed: 2015-01-13.

[7] Skype webpage. `http://www.skype.com`. Accessed: 2015-01-13.

[8] K. Aberer. P-Grid: A Self-Organizing Access Structure for P2P Information Systems. In *CoopIS*, 2001.

[9] K. Graffi, C. Gross, P. Mukherjee, A. Kovacevic, and R. Steinmetz. LifeSocial.KOM: A P2P-Based Platform for Secure Online Social Networks. In *P2P*, 2010.

[10] A. Hameurlain, F. Hussain, F. Morvan, and A. M. Tjoa, editors. *Data Management in Cloud, Grid and P2P Systems*. Globe Conference proceedings, Springer, 2012.

[11] F. Holmgren. Towards a Visualization Tool for Peer-to-peer Overlay Networks. Technical report, The Swedish Institute of Computer Science (SICS), 2005.

[12] M. Jelasity, R. Guerraoui, A.-M. Kermarrec, and M. van Steen. The Peer Sampling Service: Experimental Evaluation of Unstructured Gossip-Based Implementations. In *Middleware*, 2004.

[13] H. Kavalionak and A. Montresor. P2P and Cloud: A Marriage of Convenience for Replica Management. In *IWSOS*, 2012.

[14] D. Kontominas, P. Raftopoulou, C. Tryfonopoulos, and E. G. Petrakis. DS4: A Distributed Social and Semantic Search System. In *ECIR*, 2013.

[15] N. Korsveien, V. Setty, and R. Vitenberg. VizPub: Visualizing the Performance of Overlay-Based Pub/Sub Systems. In *DEBS*, 2014.

[16] A. Loupasakis, N. Ntarmos, and P. Triantafillou. eXO: Decentralized Autonomous Scalable Social Networking. In *CIDR*, 2011.

[17] D. Lütjohann, A. Shah, P. Christen, F. Richter, K. Knese, and U. Liebel. Sciencenettowards a global search and share engine for all scientific knowledge. *Bioinformatics*, 27(12), 2011.

[18] A. Montresor and M. Jelasity. PeerSim: A scalable P2P simulator. In *P2P*, 2009.

[19] Y. Petrakis and E. Pitoura. On Constructing Small Worlds in Unstructured Peer-to-Peer Systems. In *P2P&DB*, 2004.

[20] P. Raftopoulou, E. Petrakis, and C. Tryfonopoulos. Rewiring Strategies for Semantic Overlay Networks. *DAPD*, 26(2-3), 2009.

[21] V. Sinha, A. Gupta, and G. Kohli. Comparative Study of P2P and Cloud Computing Paradigm Usage in Research Purposes. In *CNC*, 2011.

[22] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a Scalable Peer-to-Peer Lookup Protocol for Internet Applications. *IEEE/ACM TON*, 2003.

[23] I. Trajkovska, J. Salvachua Rodriguez, and A. Mozo Velasco. A novel P2P and cloud computing hybrid architecture for multimedia streaming with QoS cost functions. In *ACM Multimedia*, 2010.