A Survey of Overlay Simulators

Abstract

Many peer-to-peer networks are overlay networks, because they run on the top of the Internet. It is infeasible to assess the capability and limitations of newly developed peer to peer protocols in a real environment. Many overlay peer to peer overlay simulators were developed for modeling overlay protocols. Efficiency of these simulators differ in various aspects such as scalability, configurability, usability, portability etc. Most of the P2P overlay simulators were developed by different groups for customized applications. Each of these simulators outstands in a specific area. In this paper, we present a survey on certain prominent overlay simulators that exist today. We have evaluated the simulators on a 11-point scale and classified them. These classifications help the users to easily identify a simulator suitable to their requirement.

Keywords: Overlay networks,peer to peer, simulator

1 Introduction

An overlay network is a logical network which is built on top of another network. Nodes in the overlay can be thought of as being connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network. It is widely used in the distributed environment. Overlay networks are large heterogeneous collection of nodes with complex interactions between them. During the simulation of heterogeneous and highly scalable networks, some practical considerations are being suppressed. Simulation with insufficient data of the system leads to inaccurate and ambiguous results[31]. It is not feasible to perform an experiment on overlay networks. Overlay simulators are widely used despite these limitations. Eventhough there exist many overlay simulators, a user of the distributed environment finds it difficult to select a particular simulator without an elaborate study. This paper helps the research community to find an appropriate simulator of their need.

In this paper, we have studied popular overlay simulators and identified properties which are common to all of them. We also identified the features which makes them unique from others. We have classified these simulators based on certain criteria.

This paper is organized as follows. Section 2 gives an evaluation of the simulators. Section 3 describes the evaluation metrics for the surveyed simulators. Section 4 outlines a classification of the overlay simulators based on some properties are shown. Section 5 concludes the paper.

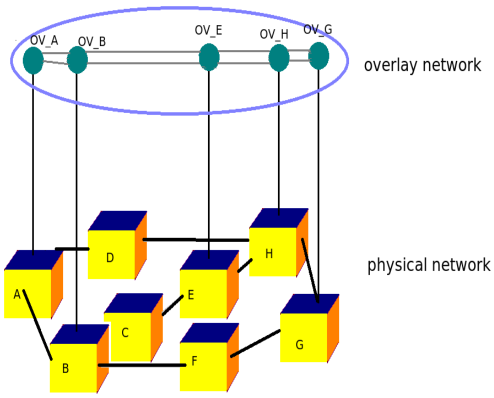

Figure 1: An Illustration of an overlay network

2 Evaluation Of Simulators

We present the highlights of popular overlay simulators that we have studied.

2.1 QualNet

QualNet[24] is a modeling tool for wireless and wired networks. It provides an environment for designing protocols, creating and animating experiments, and analyzing the results of those experiments. It is a simulator for large, heterogeneous networks and the distributed applications that execute on such networks. Qualnet is written in Java. Experiments can be configured at each layer and layer wise simulation can be performed. After simulation the results are stored in the files.

Discussion: Qualnet provides some tools, among them simulator is one tool. It is a discrete event simulator that can simulate large networks. It supports upto 50000 mobile nodes. Modeling a system and simulation can be done within one framework/environment. QualNet lacks predefined model constructs and parameter documentation. QualNet has time consuming modeling through missing modularity and reusability[20].

2.2 Peersim

Peersim[18] was partly developed in BISON project[1][7]. Peersim is written in java and is released under the GPL open source license. It is a single threaded simulator. Peersim is specialized for supporting the simulation of P2P protocols based on modular paradigm[19]. It supports features like decentralized control, high autonomy of participating nodes and heterogeneity in terms of shared resources like other network simulators.

Peersim can simulate a dynamic and extreme scalable system which are the striking requirements of a P2P simulator. Peersim can simulate both structured and unstructured overlays. Peersim supports both cycle and event based simulation models. The peersim simulator supports only console based operations.

Discussion: Peersim supports only console based input and output. There are less user manuals provided by the developers. There are no debugging or visualization tools provided. The degree of configurability of peersim is less.

2.3 DSSim

DSSim[23] was developed by HERMES team. DSSim is a single threaded discrete event simulator written in java. DSSim can measure bandwidth consumption in the simulated network. DSSim simulates the routing of messages in the physical network topology. It uses a hierarchical two level distance vector routing algorithm.

DSSim has *PluginIF* and *EnvironmentIF* as main interfaces.*PluginIF* enables DSSim to interact with the four plugins implemented into it. The *EnvironmentIF* interface enables a logical node to interact with the DSSim environment. *EnvironmentIF* monitors and handles messages and time aspects during the simulation run. DSSim outputs a trace file which contains statistics collected during simulation run that can be plotted as graphs. DSSim can also display logical and physical visualization of the currently running simulation.

Discussion: DSSim offers good degree of configurability and scalability.

2.4 Trust Overlay Simulator [TOSim]

TOSim[33] is developed based on PeerSim[18]. TOSim simulates transport layer and the upperlayers of the OSI Reference Model[4]. TOSim simulates trustworthy behaviour in overlay networks. TOSim is highly modular and configurable.

TOSim supports dynamic networks with good memory management. The simulator considers the simulation of good nodes and malicious nodes. TOSim considers some good nodes as highly trusted nodes. It can simulate various threat models by setting appropriate parameters. The Simulation engine of TOSim adopts a time-stepped simulation model instead of an event-based model. TOSim computes the interaction between peers using a probabilistic content distribution model.

Discussion: TOSim simulator outstands many other simulators in its features like scalability.

2.5 PlanetSim

PlanetSim[6][22] is a discrete-event overlay network simulator. It is written in Java and is available under GPL license. PlanetSim has a layered modular architecture. It offers the users two main working levels of creating and testing new overlay algorithms or services on the top of existing overlays. The main aim of PlanetSim is to enable a smooth transition from simulation code to experimentation code running in the Internet. It provides wrapper code that takes care of network communication and permits us to run the same code in network test beds such as PlanetLab. PlanetSim ensures high degree of Portability. It supports structured overlays.

PlanetSim provides a Common API which provides routing and processing messages and accessing node routing state information. Failures are handled by replicating data in different nodes. It uses Java and UNIX shell for processing.

PlanetSim offers several output formats like Pajek or GML. These can be used for overlay visualization and analysis. Users can use these output system to generate their own graph formats.

Discussion: Planetsim does not mention about trust in overlay applications. The simulator offers less scalability. PlanetSim is designed to have good usability. It offers optimized code to enable scalable simulations in reasonable time. PlanetSim has in-built support for *Chord* and *Symphony*.

2.6 Narses

Narses [3][13]is a discrete-event simulator written in Java. It was released from Stanford University. Narses is a scalable, application-level, flow-based network simulator. Narses is hierarchically placed between packet level simulators and analytical models. Narses approximates physical,transport and network layers.

Narses does not have any in-built P2P protocols. It scales up to 600 nodes. Narses does not support distributed simulation. Narses reduces the simulation time by grouping packets into flows. It further reduces total number of events in the simulation and the memory constraints. Narses offers a range of models that trade between fast runtimes and accuracy. It has four underlying network topology models. They are Naive, NaiveTop, FairTopo and SafeFairTopo. Naive models a star topology where all nodes are connected to a hub. Naive model is fast and but inaccurate as it ignores the message contention. Each connection to the hub has a value for bandwidth and latency associated with it. NaiveTopo resembles Naive in that it is notch up to the star topology only. FairTopo assumes no bottleneck at the network topology. It considers the effects of competing TCP flows. It is slower and more accurate than the Naive models. SafeFairTopo dynamically checks that no link at the network core become a bottleneck.

Discussion: Narses assumes traffic in the network. It scales down to less number of nodes. There are no evidences of existing overlay networks modeled by Narses. Trust is not incorporated with the simulator.

2.7 3LS

3LS[3 Level Simulator] [30] [32] is a discrete simulator using a central step-clock. 3LS is written in Java. It can scale only a maximum of 1000 nodes. 3LS has a clearly defined Network model, Protocol model and User model. These three models allow communication between directly connected levels.

- Network model (Bottom Level- Describes distance between nodes)
- Protocol model (Middle Level -Describes the P2P protocol used)
- User model (Top Level- User input via GUI or file)

During the execution of the system, the events are displayed on the command prompt screen. On completion of simulation , all simulation data is saved into a file for future analysis. The data stored after the simulation is viewed using a visualization tool named AiSee[5] which renders the user an image of the graph. 3LS incorporates an implementation of Gnutella 0.4.

Discussion: 3LS has inefficient memory usage and mediocre scalability. Trust is not incorporated with the simulator.

2.8 Query Cycle Simulator

The Query-Cycle Simulator [10][17]is a standardized and extensible query cycle file-sharing P2P network simulator. During each cycle,active peer issues a query,waits for the list of incoming responses, and downloads one of the responses.

The simulator is modeled for real-world file-sharing P2P networks. Query Cycle simulator can model unstructured network involving both good and malicious peers. The simulator uses the EigenTrust[27] algorithm to assign peer trust values based on the success of a file upload. EigenTrust algorithm decreases the number of inauthentic files on the network. The Algorithm performs good degree of authentication even under a variety of conditions where malicious peers cooperate in an attempt to sabotage the system.

Query Cycle Simulator models both content behavior and Peer behaviors. Content distribution incorporates the minimum number of categories per peer, total number of such peers and how they are distributed. Peer behavior classifies the peer as Good Peer, Highly Trusted Peer and Malicious Peer.

Query Cycle Simulator uses a GUI to accept the input parameters. The output displays the simulation view. The peers and download history can also be obtained along the output. Animation of peer downloads is also incorporated with the Query Cycle Simulator. Query cycle simulator can be used to demonstrate adaptive topologies algorithm.

Discussion: Query Cycle Simulator can model a system with good degee of system stability. The degree of configuration is less.

2.9 NeuroGrid

NeuroGrid [9][26] is a Java-based overlay simulator specialized in simulation of search protocols for file-sharing systems such as Gnutella, Freenet and other NeuroGrid systems. The NeuroGrid simulator is a single-threaded event simulator and uses a configuration file to define the protocol to simulate and the network properties. These properties include the number of nodes and queries to execute. Statistics including the number of successful queries and messages parsed can be saved into files for later analysis.

Discussion: It is designed for neural network related applications. With NeuroGrid the information updating speed is slow, the result of query is uncertain and the size of route table is interrelated with the number of files and the number of nodes in the network. It optimizes search process through the construction of associated rule.

2.10 P2PSim

P2Psim[11] is a C++ based overlay simulator designed for the evaluation of P2P protocols. P2PSim is a multi-threaded, discrete event simulator to evaluate, investigate, and explore peer-to-peer protocols. It supports several protocols such as Chord, Accordion, Koorde, Kelips, Tapestry, and Kademlia. These implementations are specific to P2PSim and do not model all features of protocols. Output statistics are stored in log file and can be used for further research. It is a discrete event simulator which uses only command line interface.

Discussion: P2PSim runs only on UNIX like operating systems. Its scalability is very less compared to other simulators. It fulfills very basic functionalities.

2.11 Overlay Weaver

OverlayWeaver [12][15] is a peer-to-peer overlay construction toolkit written in Java which can be used for easy development and testing of new overlay protocols and applications. The toolkit contains a Distributed Environment Emulator which invokes and hosts multiple instances of Java applications on a single computer. OverlayWeaver is event driven simulator which is having both graphical and command line modes of operation. Comparison of performance can be done with various kind of algorithms such as chord, tapestry. It has animation tools incorporated.

Discussion: Simulations have to be run in real-time. OverlayWeaver has no statistical output. This limits the use of the simulator as an overlay network simulator.

Parameters	Peersim	TOSim	DSSim	QualNet	PlanetSim	NeuroGrid	P2Psim	Overlay Weaver	Narses	Query Cycle Simulator	3LS
Scalability	10^6	10^6	10^5	5×10^4	2^{15}	3×10^5	3×10^3	4×10^3	600	N/A	1000
Configurability	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Simulation Model	Cycle Driven Event Driven	Cycle Driven	Event Driven	Event Driven	Event Driven	Event Driven	Event Driven	Event Driven	Event Driven	Cycle Driven	Event Driven
Protocols to Experiment	P2P protocols	P2P protocols	P2P protocols	All protocols	P2P protocols	P2P protocols	P2P protocols	Application layer protocols & P2P	P2P protocols	P2P protocols	P2P protocols
Interfaces	Node CDProtocol Linkable Control	N/A	Plugin IF EnvironmentIF	Node Link Channel Network	Node Link	Interface for database access	No interface for application layers	Interface for Emulator	NetworkStub ForNetwork ForClock Clockstub	QueryCycleSimulatorImpl QueryCycleSimulatorRun	N/A
Output Generated	Console	Console	Trace File	Trace File	GML,Pajek	Trace file	Log files	Console	Trace File	Trace File	Trace File
Network Representation	Array Of Nodes	Array Of Nodes	Transit Stub	Grid of nodes	Ring of Nodes	Graph of Nodes	Ring of nodes	Ring of nodes	Array Of Nodes	Graph of Nodes	
Modes of Operation	Console	Console	GUI	GUI /Console	GUI	GUI /Console	Console	GUI /Console	Console	GUI	GUI
Tools Provided	No	No	No	Analyzer Designer Tracer Animator Importer	No	No	No	Yes	DistSim	No	Aisee
Language	Java	Java	Java	Java	Java	Java	C++	Java	Java	Java	Java
License	LGPL	N/A	GPL	QualNet License	LGPL	GPL	GPL	Apache	GPL- like	Apache	GPL
Remarks	Peersim is a simulation test bed on epidemic protocols.The degree of configurability is average.	ToSim has an average degree configurability	DSSim offers a good degree of configurability.	Modeling and simulation can be done at each layer, degree of configurability is high	Modeling the protocols of different layers is easy.	It is applicable to neural networks and knowledge theory area	It works only on UNIX operating systems	It provides emulators which are helpful to create many host application on one system	Flow based simulator with good simulation speed.	Graceful support to make the fault tolerant system .	Simulating different layers has good degree of configurability.

Table No:1 Summary Of Survey Of Overlay Simulators

3 Metrics Used For Evaluation Of Simulators

This survey was conducted to estimate the caliberity of the simulators to tune user needs. This paper surveys simulators based on the following criteria.Table1 summarizes the findings of our survey.

3.1 Scalability:

Most of the overlay simulators support good scalability but they differ in the efficient use of resources while maintaining support for scalability.

3.2 Configurability:

Configurability of the simulation system indicates how much the system can be customized. The configurability aims to control the behavior of the components. The degree of configurability indicates the ability to add components with ease. The higher the degree of configurability,the higher the ease with which the system can be customized.

3.3 Simulation Architecture:

Simulation architecture indicates how the objects refer to the modes of operation and design of the simulator. The two simulation architecture in use today are Event driven model and Cycle based model. In Cycle based engine, a change in the state of the system occurs at every clock cycle. In Event driven model, a change in the state occurs only on the occurrence of an event.

3.4 Protocols Supported:

A generic network simulator can support almost all the existing protocols. Specific simulators like the overlay simulators simulate only a limited number of overlay protocols.

3.5 Application Interfaces:

Interfaces decouple the simulator from the applications. Interfaces allow the simulator to internally modify without affecting the way outside entities interact with it. Interfaces provide multiple abstractions.

3.6 Output generated:

The output of the simulations can be stored in files. Some simulators do not support the output to be directed onto the tracefile. Such simulators support console based output. The output from the trace file or console can be plotted using an xplot.

3.7 Network representation:

Input to the simulator represented as a graph. The graph is represented using a suitable data structure mentioned by the simulator.

3.8 Types of networks that can be simulated:

A general purpose simulator can support all kinds of a network.

3.9 Modes Of User Interaction:

Simulator provides two modes. Those are console based and graphical user mode. Some simulators provide both options. Graphical User based modes are user friendly.

3.10 Tools provided:

Tools provided along the simulators can be software libraries for simulation of discrete event systems or continuous simulation or visualization tools or Bondgraphs. Visualization tools provide a visual block diagram or they can model animation. Bond Graphs like symbols2000, provides an object oriented modeling tool using bond graphs, block diagrams to model dynamic systems.

3.11 Simulation speed:

The simulation speed indicates how fast the simulation run completes. Maintaining scalability along with memory constraints can affect the simulation speed.

3.12 Fault tolerance:

Fault tolerance mainly depends upon the overlay topology constructed. Fault tolerance of each simulator are dealt with nodes/link/channel arrangement.

3.13 Usability:

Usability indicates how easily the simulator can be learned. This includes documentations of the source provided. Documentation of the configured simulators are available.

4 Classification Of Simulators

The set of simulators that we studied have many common properties. Metrics used for evaluation of simulators are the basis for categorizing the simulators. We have identified three important parameters on which we classify the overlay simulators. They are

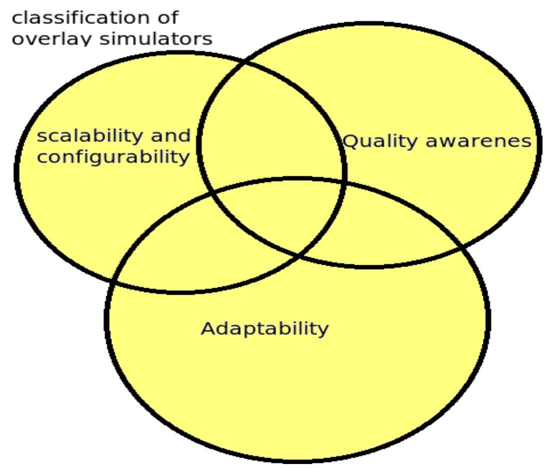

Figure 2: Classification Of Overlay Simulators

1. Scalability and Configurability
2. QoS Support
3. Adaptability

Section 4.1 deals in detail about the classification based on scalability and configurability. A few overlay simulators provide quality in addition to the prime requirement. In section 4.2, classification is based on the Quality of service supported by the overlay simulators is discussed. Section 4.3 deals with the simulators that support adaptability to any newly deviced routing protocols.

4.1 Classification based on Scalability and Configurability

Scalability is the maximum number of nodes that can be simulated on a host by the overlay simulator. Overlay simulators which can simulate more than 10^4 nodes are termed as *Large Scale* simulators, the one which can model between 10^3 and 10^4 are *Medium Scale* and the

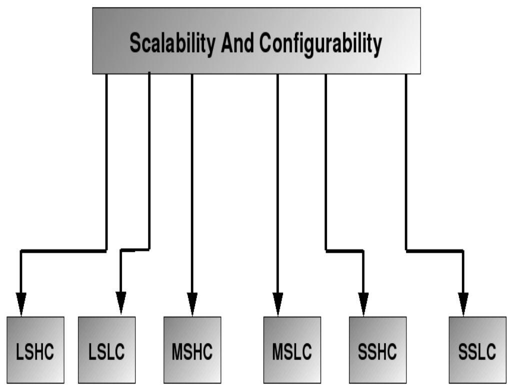

Figure 3: Classification based on Scalability and Configurability

one which can model less than or equal to 10^3 are *Small Scale* simulators. Configurability is the ability of the simulator to incorporate new features. The degree of configurability is the measure of adaptability of the simulator. It is measured on a 10 point scale. The simulators having degree of configurability greater than 8 is termed as *High Configurable* and the simulators with degree of configurability less than 8 are *Low Configurable*. Four categories of simulators follows from this classification. It is depicted in figure 3.

4.1.1 LSHC-Large Scale High Configurable Overlay Simulators

LSHC simulators have scalability greater than or equal to 10^5 nodes. The degree of configurability of these simulators scale greater than 8 on a 10 point measurement. DSSim and PlanetSim belongs to this category.

4.1.2 LSLC-Large Scale Less Configurable Overlay Simulators

LSLC simulators can model greater than or equal to 10^5 nodes with poor degree of configurability. PeerSim and TOSim are the LSLC simulators.

4.1.3 MSHC-Medium Scale High Configurable Overlay Simulators

MSHC simulators have scalability between 10^4 and 10^3 nodes. The degree of configurability of these simulators range greater than 8. QualNet is a MSHC simulator.

4.1.4 MSLC-Medium Scale Less Configurable Overlay Simulators

MSLC simulators can model nodes between 10^4 and 10^3 nodes. MSLC simulators have degree of configurability less than 8. P2PSim is a MSLC simulator.

4.1.5 SSHC-Small Scale High Configurable Overlay Simulators

SSHC simualtes nodes less than or equal to 10^3 nodes with degree of scalability more than 8. Most of the simulators that fall under this category have a detailed architecture. They rarely hide out the under lying details at the expense of scalbility. 3LS is a SSHC simulator.

4.1.6 SSLC-Small Scale Less Configurable Overlay Simulators

SSLC scales less than or equal to 10^3 nodes. These simulators provide the underlying network with lower memory footprints. SSLC has lower degree of configurabiltiy.

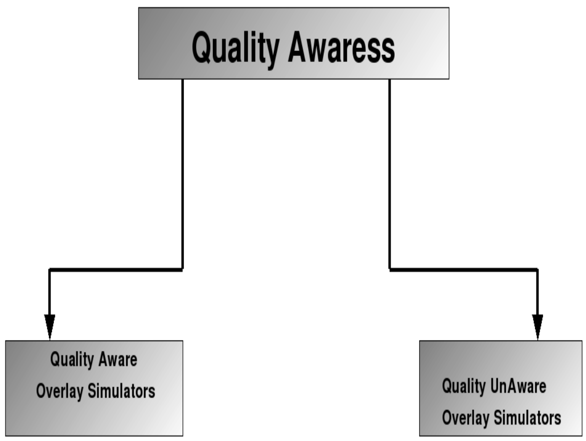

Figure 4: Classification based on Quality Awareness

4.2 Classification based on Quality Awareness

Overlay simulators are designed to offer the modeling of overlays. Some simulators are designed to give trust/Quality Awareness on the overlays they model. Trust is a subjective expectation that a component has about another future behavior based on the history of their encounters. Trust Management in the system allows it to publish messages anonymously, so that it can censor any message in the system which is malicious. Classification based on Quality Awareness is depicted in the figure 4.

4.2.1 Quality Aware Overlay Simulators

Simulators offer various degree of Trust/Quality Awareness on the overlay they simulate. TOSim offers good degree of trust on the overlay they model. TOSim can model individual malicious peers, malicious collectives, malicious collectives with camouflage, malicoius spies, sybil attack, virus-disseminators.

4.2.2 Quality UnAware Overlay Simulators

Most of the Simulators surveyed do not incorporate trust on them. This category of simulators cannot model malicious peers or its variants. Peersim,PlanetSim,QualNet,DSSim are Quality UnAware Overlay Simulators.

4.3 Classification based on Adaptability

The adaptability of the overlay simulators refers to its futuristic utility. A futuristic simulator should be *highly scalable and configurable.* It should support testing of protocols that are going to be developed in the future. A futuristic should also have the provision for incorporating future technologies, protocols and algorithms. This includes extensibility of the simulator. Figure5 gives the classification based on Adaptability.

4.3.1 Futuristic Overlay Simulators

Futuristic Overlay Simulators are Large Scale, Highly configurable and Highly Tunable simulators. There exists no Futuristic Overlay Simulators upto this date.

4.3.2 NonFuturistic Overlay Simulators

NonFuturistic Overlay Simulators are Large Scale,Highly configurable and Less Tunable simulators. All LSHC belongs to this category. DSSim belong to this category.

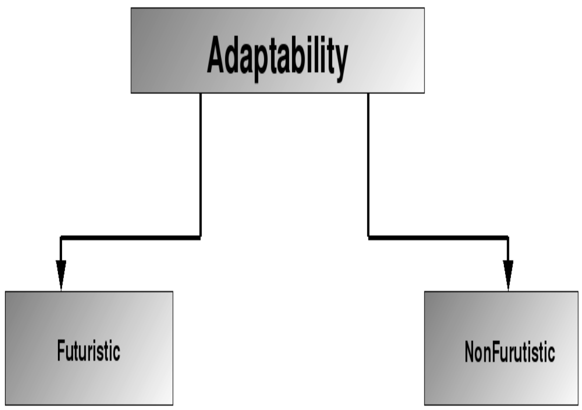

Figure 5: Classification based on Adaptability

5 Conclusion And Future Work

An exhaustive survey of well known overlay simulators have been performed. The characteristic features of simulators have been studied and compared. This comparison and categorization will help the research and student community to easily select the overlay simulator suitable to their application domain. Eventhough there are many overlay simulators, none of the prominent simulators incorporate an explicit support for fault-tolerance and client mobility. The research community can look into the prospects of designing a new simulator that supports the currently ignored aspects. As shown in Section 4,we find that currently there are no futuristic overlay simulators. The research community can proceed to design futuristic overlay simulator which are flexible enough to support testing of new protocols that are being developed. The survey in this paper can be extended to a large class of overlay simulators. The classification can also be extended further to include more number of classes and properties.

References

[1] Alberto Montresor and Ozalp Babaoglu. The bison project. *IEEE Computational Intelligence Bulletin*, 1(1), December 2002.

[2] Diego Doval and Donal O'Mahony. Overlay networks,a scalable alternative for p2p. *IEEE Internet Computing*, pages 2–5, July-August 2003.

[3] TJ Giuli and Mary Baker. Narses: A scalable flow-based network simulator. *Technical Report cs.PF/0211024, http://arXiv.org/abs/cs.PF/0211024*, November 2002.

[4] Hubert Zimmermann. Osi reference model the iso model of architecture for open systems interconnection. *IEEE Transactions on Communications*, 28(4):425 432, April 1980.

[5] http://www.aisee.com/.

[6] http://ants.etse.urv.es/planet.

[7] http://www.cs.unibo.it/bison.

[8] http://www.cis.udel.edu/ jaikaeo/qualnet/manual.html.

[9] http://www.neurogrid.net.

[10] http://p2p.stanford.edu/www/projects.html.

[11] http://pdos.csail.mit.edu/p2psim/.

[12] http:// overlayweaver.sourceforge.net/.

[13] http://sourceforge.net/projects/narses.

[14] http://tools.ietf.org/html/draft-irtf-p2prg-survey-search 00.html. Accessed on October 2007.

[15] Ingmar Baumgart Bernhard Heep and Stephan Krause. Oversim: A flexible overlay network simulation framework. *IEEE Global Internet Symposium*, pages 79–84, May 2007.

[16] Jordi Pujol Ahullo Marc Sanchez Artigas and Pedro Garca Lopez. Planetsim- an overlay network simulation framework,user and developer tutorial.

[17] Mario T. Schlosser Tyson E. Condie and Sepandar D. Kamvar. Simulating a file-sharing p2p network. *First Workshop on Semantics in P2P and Grid Computing*, May 2003.

[18] Mark Jelasity Alberto Montresor Gian Paolo Jesi and Spyros Voulgaris. http://peersim.sourceforge.net.

[19] Mark Jelasity Alberto Montresor and Ozalp Babaoglu. A modular paradigm for building self-organizing p2p applications. *In Engineering SelfOrganising Systems: Nature-Inspired Approaches to Software Engineering,Springer-Verlag*, (2977):265–282, Apr 2004.

[20] Markus Bernat. Network modeling and evaluation tools for higher education. *University of Applied Sciences*, September 2004.

[21] Nyik San Ting. A generic peer-to-peer network simulator. *Department of Computer Science, University of Saskatchewan,Saskatoon, Saskatchewan.*

[22] Pedro Garcia CarlesPairot Ruben Mondejar Jordi Pujol Helio Tejedor RobertRallo. Planetsim: A new overlay network simulation framework. *Ubiquitous Computing,Springer Berlin / Heidelberg*, pages 123–136, March 2005.

[23] Peter Pietzuch. Hermes:a scalable event-based middleware-technical report-590. *UCAM-CL-TR-590*, June 2004.

[24] QualNet simulator version 3.1 User Manual. Scalable network technologies.inc. 2001.

[25] Rajive .L.Bagrodia. Language support for parallel discrete-event simulations. *Proceedings Of 1994 Winter Simulation Conference,Computer Science Department*, pages 1324–1331, 1994.

[26] Sam Joseph and T. Hoshiai. Decentralized meta-data strategies: Effective peer-to-peer search. *IEICE Transactions on Communications*, pages 1740–1753, 2003.

[27] Mario T. Schlosser Sepandar D. Kamvar and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. *In Proceedings of the 12th international conference on World Wide Web -ACM*, pages 640–651, 2003.

[28] S Naicken A Basu B Livingston S Rodhetbhai. A survey of peer-to-peer network simulators. *Department of Informatics,University of Sussex.*

[29] S. Naicken B. Livingston A. Basu S. Rodhetbhai I. Wakeman D. Chalmers. The state of peer-to-peer simulators and simulations. *ACM SIGCOMM Computer Communication Review*, 37(2):94–98, April 2007.

[30] Ting N. Deters. R. 3ls - a peer-to-peer network simulator. *In Third International Conference on Peer-to-Peer Computing*, 2003.

[31] Todd R. Andel and Alec Yasinac. On the credibility of manet simulations. *IEEE Computer*, 39(7), July 2006.

[32] Wolf Siberski and Uwe Thaden. A simulation framework for schema-based query routing in p2p-networks. *Springer Berlin / Heidelberg, Current Trends in Database Technology - EDBT 2004 Workshops*, 3268/2004:436–445, November 2004.

[33] Wei Wnag Yan Zhang and Shunying Lu. Simulating trust overlay in p2p networks. *Springer Berlin / Heidelberg*, pages 632–639, July 2007.