Integration of fluid-based analytical model with Packet-Level Simulation for Analysis of Computer Networks

Tak Kin Yung, Jay Martin, Mineo Takai, and Rajive Bagrodia[*]
Department of Computer Science
University of California, Los Angeles

ABSTRACT

Fluid flow analytical models have been shown to be able to capture the dynamics of TCP flows and can scale well to solving for networks with a large number of flows. However, accurate closed form solutions are not yet available for wireless networks. Traditional packet-level discrete event simulations provide accurate predictions of network behavior, but their solution time can increase significantly with the number of flows being simulated. Integration of fluid flow models with packet-level simulators appears to offer significant benefits. In this paper, we describe an approach to integrate fluid flow models into QualNet, a scalable packet-level simulator. We validate the mixed model with detailed packet-level simulations for the scenarios considered in this paper. The execution time of the mixed model is significantly impacted by the frequency with which the analytical model must be solved in response to changes in the data rate at the interface of the packet-level and analytical models. We present a time averaging approach to mitigate this impact and present the results of the resulting tradeoff between prediction accuracy and model execution time.

Keywords: GloMoSim, Qualnet, network simulation, fluid flow models

1. INTRODUCTION

Simulation is a commonly used method to understand and evaluate the performance of communication networks. Discrete event simulators employ a packet-by-packet model of network activity and closely model one or more layer of the entire protocol stack. This provides accurate predictions of the network behavior, but typically at the cost of expensive and time-consuming computations. As the sizes of computer networks have been increasing significantly, networks now contain a large number of nodes, with heavy traffic going through them. Simulations of such networks for reasonable duration of times can be prohibitively expensive. Commonly used discrete-event network simulators include GloMoSim [1, 2, 12], QualNet [13], a commercial tool derived from GloMoSim, ns-2 [14] and OPNET.

One of the methods that have been proposed to reduce the long execution time of detailed packet-level simulators is by using parallel model execution. However, even though parallel execution has been shown to reduce the execution time of large models by an order of magnitude [3], it is still the case that execution time of such models, particularly for wireless networks, continues to be an impediment to their widespread use for networks with tens of thousands of nodes.

A lot of research has been done in the community to investigate the possibility of employing analytical models in simulations [5, 6, 7]. Instead of simulating transmissions of packets, these analytical models offer a closed form solution to obtain network behavior, for example, by using differential equations to model the rates of traffic changes. However, closed form solutions have heretofore not been developed for wireless networks. In this paper, we explore the use of mixed-model simulators that combine a packet-level, discrete event simulator, with a fluid-flow based analytical model. Our primary purpose in this paper is to develop the appropriate interfaces between packet-level simulators and analytical models and demonstrate that such mixed mode simulators yield results that match those obtained from detailed packet-level simulators. Our eventual objective is to use such mixed model simulators is to analyze networks that include a wired backbone network

---

[*] Correspondence: Rajive Bagrodia; Email: rajive@cs.ucla.edu. Other author information: T.K.Y.; Email: tkyung@cs.ucla.edu; J.M.: Email: jmartin@cs.ucla.edu; M.T.; Email: mineo@cs.ucla.edu. UCLA Parallel Computing Laboratory: http://pcl.cs.ucla.edu/.

This work is supported by the US Department of Defense/DARPA under the project “Maya: Next Generation Performance Prediction Tools for Global Networks” (Contract N66001-00-1-8937).

(that is amenable to closed form analytical solutions) with wireless networks at the boundaries (that are best evaluated using packet-level simulations).

Although the interfaces described in this paper are general purpose, we will be using specific examples of analytical models and packet-level simulators to illustrate our ideas and present experimental results. In particular, we will be integrating Misra et al's fluid-based analytical model [4] with QualNet [13], the next generation of the GloMoSim packet level simulator.

The remainder of the paper is organized as follows. Section 2 discusses QualNet, the discrete event simulator we will be using to produce the final integrated version, and the fluid flow analytical model. Section 3 outlines the mechanisms used to integrate fluid flow models into QualNet. Section 4 discusses some implementation details associated with our integration. Section 5 describes some simulations to validate our integrated simulator. Section 6 examines the performance aspects of our approach, and finally in Section 7 we conclude.

2. BACKGROUND

In this section we will briefly describe the existing packet-level simulator and the fluid-based analytical models that were used to develop the mixed mode simulator.

2.1. The QualNet Packet-Level Simulator

QualNet is the next generation of the scalable GloMoSim (Global Mobile Information Systems Simulator) simulator. GloMoSim was designed to simulate large-scale wireless networks with thousands of mobile nodes, each of which may have different communication capabilities via multi-hop ground, aircraft and satellite media. QualNet has extended GloMoSim's capabilities to wired networks as well as mixed wired and wireless networks. Like its predecessor, QualNet uses the parallel simulation kernel provided by the PARSEC discrete-event simulation language. As a result, QualNet is among the few simulators for wireless and wired networks that have been implemented on sequential and parallel architectures. Example parallel platforms include an 8-processor DELL™ PowerEdge 6100 running Windows NT®, a 24-processor Sun® Enterprise 5000 running Solaris™, and a 28-processor SGI™ 2000 running IRIX®. This paper focuses primarily on QualNet's capability for simulating wired networks on sequential architectures.

QualNet includes detailed models of commonly used protocols at each of the primary layers of the protocol stack depicted in Figure 1. These range from commonly used applications like file transfer (ftp) and web browsing (http) to transport, routing and MAC layer protocols. In each case, commonly used protocols in both wired and wireless networks have been modeled. For instance, we have modeled routing protocols like OSPF and RIP that are common in wired networks as well as AODV for wireless networks. Protocols for cellular networks are being developed. The current list of protocol models that are available in QualNet are listed in Table 1.

QualNet defines simple APIs between neighboring layers to enhance modular composition of protocol models developed at different layers by different designers. The APIs are kept as close as possible to the operational protocol stack, such that even operational code is easily integrated into QualNet with this layered design. As the operational code has typically been extensively tested, this provides substantial benefits as well as cost savings. The integration capability has already been demonstrated at the transport layer in QualNet by extracting the TCP Lite model from the protocol code distributed with the FreeBSD operating system. The only restriction made in the QualNet APIs is that network nodes can communicate with other nodes only through the lowest layer, and models at other layers cannot directly access data for other network nodes. A number of statistical metrics at each layer of the protocol stack are collected automatically by the simulator and can subsequently be used by the analyst to understand the application level performance metrics.

• **Application**	ftp, telnet, cbr, Tcplib, http, MODSAF, synthetic traffic generators, self-similar traffic with long range dependency
• **Transport**	TCP (FreeBSD, Reno, Tahoe, New Reno), UDP
• **Routing**	Bellman-Ford, OSPFv2, RIPv2, Flooding, Fisheye, DSR, LAR, AODV, ODMRP
• **MAC**	CSMA, MACA, IEEE 802.11
• **Physical**	point-point link, wired bus, IEEE 802.11 DSSS radio
• **Propagation**	analytical (free space, Rayleigh), TIREM(*), 2-ray ground reflection, path loss trace files
• **Mobility**	random waypoint, group mobility, MODSAF, trace files

Table 1: QualNet Model Library Protocols

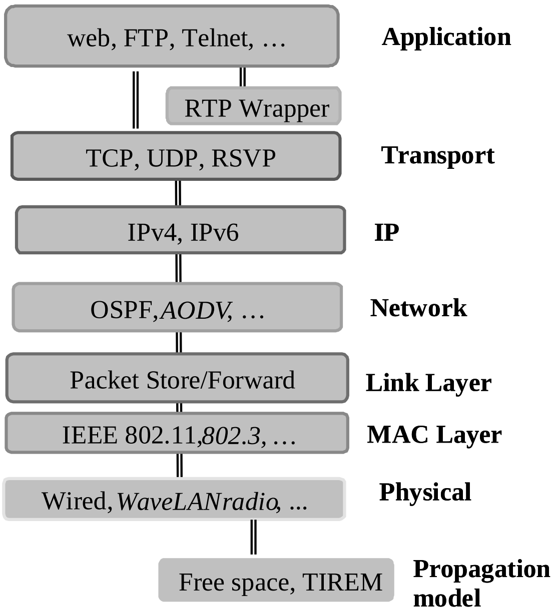

Figure 1: Protocol Stack Layers

2.2. Fluid Flow Model

In [4] Misra et al. proposed a fluid-based analytical model as a general methodology for the analysis of networks supporting TCP flows. Data traffic is modeled as a fluid. Specifically, the TCP traffic is described by a set of Stochastic Differential Equations (SDEs). Another set of differential equations is also derived for the router queuing process and AQM policy. As an application for the fluid model, they developed a scheme to calculate numerically the average behavior of network parameters, including window sizes of the TCP flows and queue lengths of the routers. The scheme has been shown to handle a large number of flows without a significant increase in computational time and complexity.

In [4] and [5] a mathematical approach has been employed to describe the analytical fluid model. For a network of active queue management (AQM) routers, a set of N+2Q differential equations can be derived, where N is number of TCP flows and Q is number of routers. The N equations (one for each flow) model the behavior of additive-increase per round trip time and multiplicative-decrease at packet loss characteristic of TCP window size. Another set of Q equations model the instantaneous queue lengths of routers as the difference between service and arrival rates of packets at each router. Each router also has a corresponding differential equation describing the average queue length as the exponential average of the instantaneous queue lengths.

Given the network parameters such as network topology, input data rates and AQM policy, the transient average behavior can be estimated analytically for a time interval. This allows us to estimate the delay of each packet traveling across a network of routers.

3. INTEGRATION OF FLUID FLOW AND PACKET LEVEL SIMULATOR

In order to incorporate the fluid flow model into QualNet, we need to design and implement an appropriate interface between a packet-level simulator the fluid flow model. This mainly involves the conversion of data from QualNet to input needed for the fluid flow model, and how QualNet makes use of the return values from fluid flow. Packets from data sources need to be converted to flows entering the analytical model, and the flows need to be converted back into packet arrival rates at the destinations.

Consider the use of mixed mode models to simulate the network shown in Figure 2. The sub-network represented by intermediate routers is to be simulated using the fluid flow model, whereas the entry/exit nodes together with the associated routers is to be modeled via packet-level simulations using QualNet.

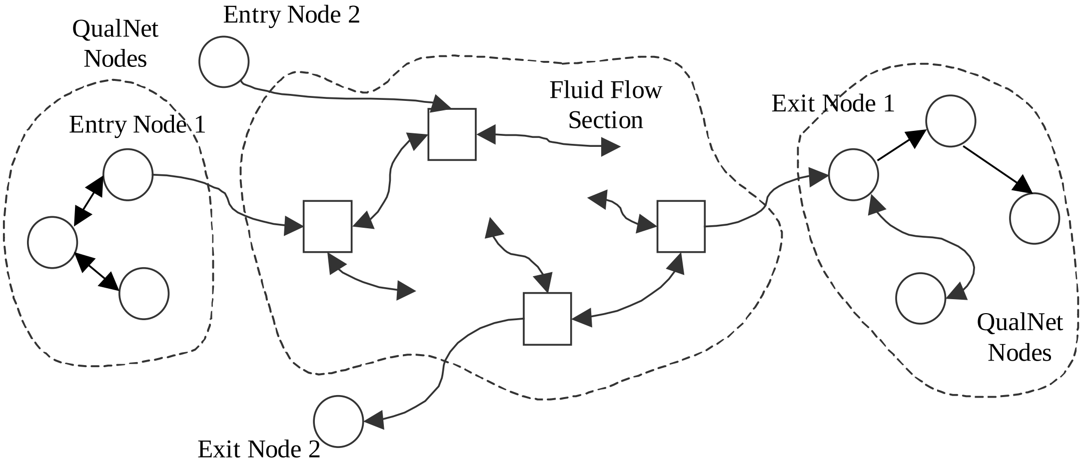

Figure 2: Simulating a network with QualNet nodes and a fluid flow section.

The mixed mode simulator must perform additional processing at the transition between packet-level simulator and the fluid flow model. The integration basically consists of three steps:

1. For each link from QualNet to fluid flow section, packets entering the fluid flow section will be converted to represent data flows, in the form expected by the fluid flow model.
2. The fluid flow analytical model will make use of data flows and other parameters to obtain network statistics by solving differential equations. In particular, queue lengths of the routers can be obtained.
3. QualNet will estimate delays for each packet based on these network statistics, then schedule packets to be delivered across the fluid flow section to appropriate QualNet nodes on the other side of the fluid flow section, as if the packets had traveled through the routers. Thus Exit Node 1 in Figure 2 does not need to be aware that incoming traffic has been simulated using the fluid flow model.

In step 1, the amount of data entering the fluid flow section will be averaged over a certain time interval. The impact of this averaging interval on the execution time and accuracy of the model is investigated in Section 6. In step 2, the fluid-flow model is solved to simulate the corresponding sub-network for the appropriate time interval. For step 3, the delay of a packet is estimated by the sum of propagation delays along the route and instantaneous queue length divided by output rate of each router.

A flow that extends from an entry node to some exit node is simulated as described above. Connections will be set up between QualNet nodes with routes traveling through the fluid flow section. The nodes and routers that lie completely within the sub-net simulated via fluid flow do not need to be created in QualNet, since no real packets need to be transmitted across the intermediate routers. The end-to-end delay of packets going through the fluid flow section is obtained by solving differential equations modeling the network. Packets are then handed across the fluid flow section based on the estimated delays at the time the packet enters the fluid flow section.

In the integrated model, the fluid flow model can be invoked using two different modes: the simple mode and the delay-estimate mode. The delay-estimate mode would be suitable for most situations, but in some cases the simple mode can be employed to obtain much faster execution.

In the delay-estimate mode, we model the intermediate routers in a network using the fluid flow model, then we add some TCP flows inside the fluid flow section. The TCP flows follow the usual additive-increase, multiplicative-decrease window size behavior as modeled by the fluid flow model. In this mode, the input flows of the fluid flow section will change according to the input data rates entering the routers. This results in changing data rates for each time averaging interval entering the fluid flow section. Depending on the time averaging interval we choose for data rate updates, the amount of computation required can vary.

The fluid flow model calculates a set of queue length estimates based on the data rate averaged at regular time intervals. As a packet is being sent across the fluid flow section, the total delay across the section is calculated. This delay is obtained based on the instantaneous queue lengths of all the intermediate routers along the route. To improve accuracy, we make use of delay statistics at time instants as the packet travels through each router. This simulates the real situation where the delay at a router for a packet is determined at the time the packet progresses through each router. The delays estimated by the fluid flow model depend on both the TCP connections inside fluid flow section and the QualNet end-to-end connections.

In some situations, the simple mode can be used for faster execution. If the end-to-end nodes have much lighter traffic when compared to the middle fluid flow section, so that incoming data rates have a negligible impact on the network statistics, it will be efficient to employ this mode. In this simple mode, the end-to-end QualNet connections do not affect flows inside the fluid flow section. In other words, the data rates and flows of the fluid flow section depend on the background traffic. Although this is simple and efficient to implement, it might not be accurate when end-to-end traffic rates become heavy.

Because the QualNet connections do not affect the network statistics in this mode (which will be valid only when the QualNet connections are light traffic), we can use this mode to obtain the network statistics using the fluid flow model prior to the actual simulation of packet transmissions. This results in a much faster execution time when compared to other modes.

4. IMPLEMENTATION

In this section, we discuss some implementation issues associated with the integration of a fluid flow model into a packet-level simulator. As QualNet is developed using a layered approach, we can modify the implementation details at a particular layer without affecting other layers. To integrate the fluid flow model, the MAC layer, and more specifically the Wired Link model in QualNet which simulates the point-to-point connections in a wired network, was modified.

The original fluid flow model was applied to a network with RED as the AQM policy [4], and implemented in MATLAB™ code for its differential equation solving capability. We extended and modified the code to account for data rate inputs from QualNet. In the latest implementation, we can specify two types of input flows: TCP flows, with their window sizes controlled by the analytical model internally, and flows that are controlled by end-to-end QualNet nodes. The rates of the latter flows are adjusted by QualNet applications. This allows different QualNet applications to specify different input data rates to the network. For example, a CBR application will specify a constant input data rate, while an FTP application will vary its data rate as it receives the acknowledgement packets.

We converted the fluid flow model implementation from MATLAB™ to C, using the MATLAB™ compiler. Subsequently, a wrapper function in C was added to interface with the fluid flow calculations, and to perform various utility functions including input rate calculations, delay estimation, and packet drop determination. These routines are invoked in the Wired Link model routines in QualNet, whenever the source entering the fluid flow section wants to send packet, and when it is time to update input rates for each time quantum. Figure 3 summarizes our code structure and data flow:

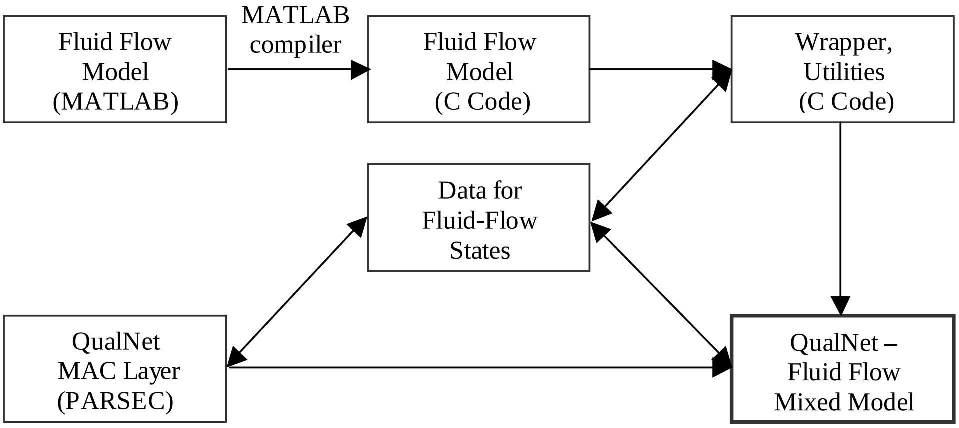

Figure 3: Mixed model code structure and data flow.

5. VALIDATION

In previous work, results from the fluid flow model have shown excellent agreement with the simulator model developed using ns-2 [4, 14]. This section describes some experiments to validate the results of the mixed-model model with an equivalent packet-level simulation using QualNet.

5.1. Experiment topology

To demonstrate the correctness of our mixed model implementation, we begin with a simple topology of a network with 6 nodes, as shown in Figure 4. The network includes two sets of connections: from nodes 0 to 1 and from nodes 4 to 5. Link capacities of 40Mbps are assigned for all links except the one between nodes 2 and 3, which is assigned a capacity of 2Mbps. Thus the link between nodes 2 and 3 will be the bottleneck. Propagation delay is assumed to be 5ms in all the experiments.

The sub-net defined by nodes 4,2,3,5 is modeled using the fluid flow model, while nodes 0 and 1 will be simulated as QualNet nodes. Conceptually there is an end-to-end link connecting nodes 0 and 1 with an estimated delay based on traffic in the fluid flow section, and packets may be dropped as they are transmitted across the fluid flow section.

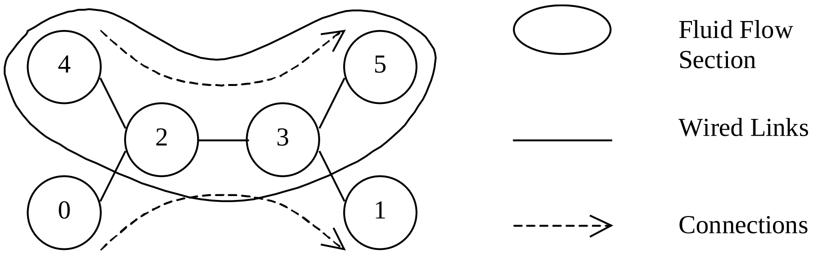

Figure 4: Network topology and connections for experiments.

5.2. Queue Length statistics

First we compare the instantaneous and average queue lengths obtained in the pure fluid flow model using MATLAB™, with the equivalent systems simulated using the packet level discrete event simulation model in QualNet, with the results predicted by the mixed mode simulator. For the first experiment, 20 TCP flows are set up from node 4 to 5, and 1 TCP flow (FTP) is established from node 0 to node 1. RED is used as the AQM policy, with a threshold of 150 packets before the router starts dropping packets, with a maximum dropping probability of 0.1 (at 200 packets). Simulation time was set to 200 seconds, which was experimentally found to be sufficient for the measured metrics to converge. As shown in Figure 5 and Figure 6, the results are comparable across the three methods, with some differences in the initial phases of network operation. This divergence is due to the fact that the fluid flow model does not model the effect of slow start.

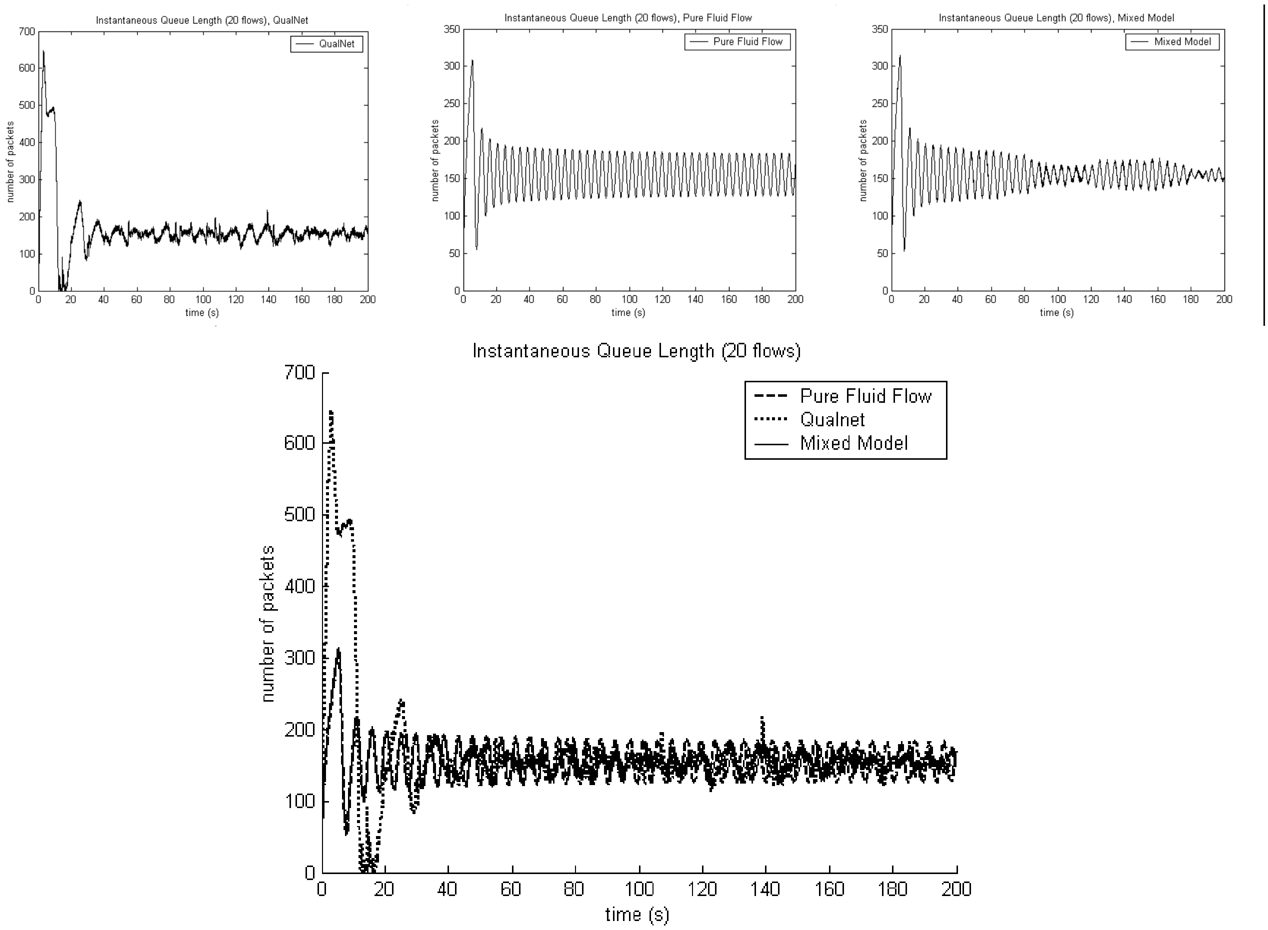

Figure 5: Instantaneous queue lengths for each model at node 2, 0→1 (1 flow), 4→5 (20 flows).

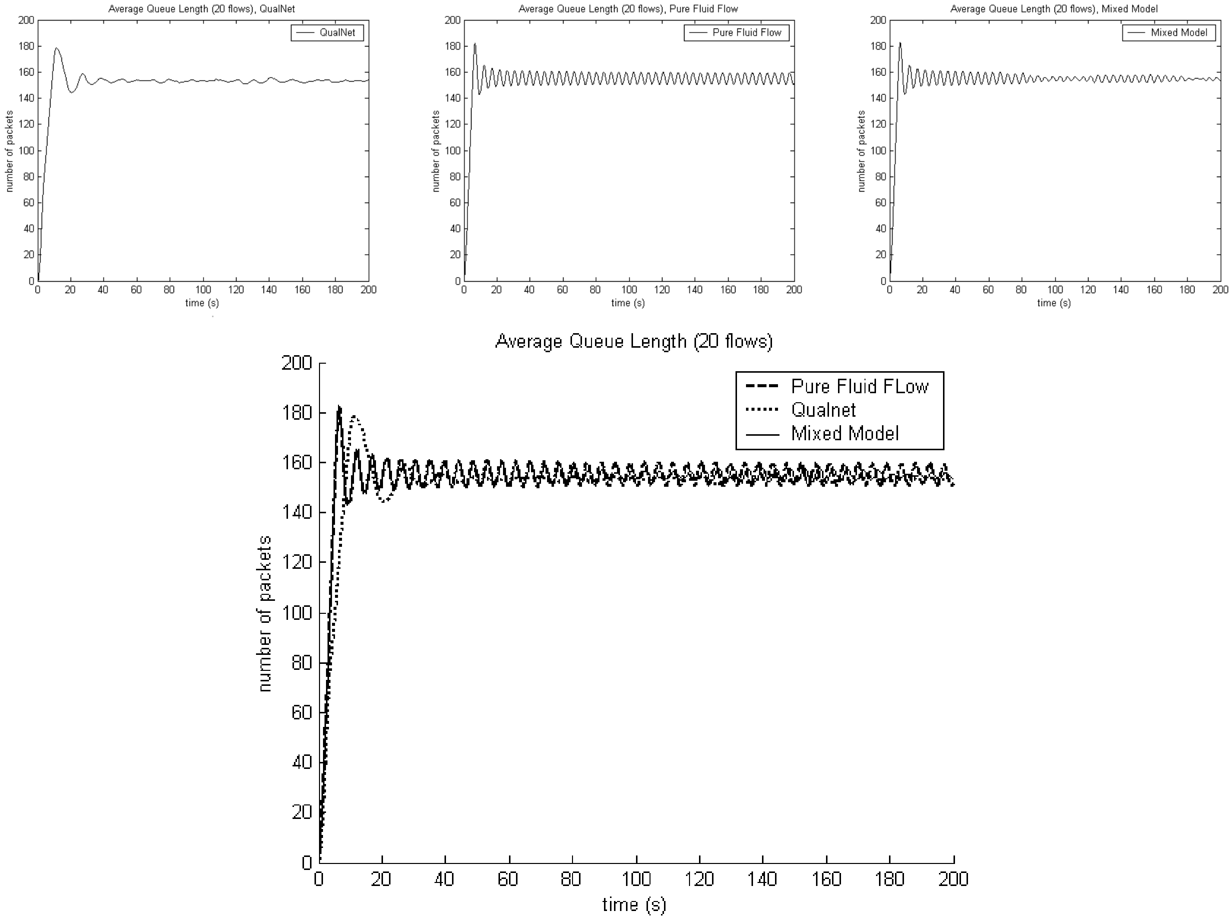

Figure 6: Average queue lengths for each model at node 2, 0→1 (1 flow), 4→5 (20 flows).

5.3. Changing Flows

The second experiment used the same topology as in the previous experiment, but we increase the number of flows and introduce some dynamic changes: initially, the network includes 15 flows each from nodes 4 to 5 and nodes 0 to 1. At time 100s, the number of flows is reduced, such that from 100s to 200s, there are only 5 flows each from node 4 to 5, as well as from node 0 to 1.

Figure 7 presents comparative results for the pure packet level and the mixed mode simulators. Again as seen from the figure, the results match quite well, reflecting the change in the flow dynamics, in the middle of the simulation. Once again, the results do not validate well in the initial phase of network operation, due to the inaccurate modeling of slow start in the pure fluid flow and mixed model cases.

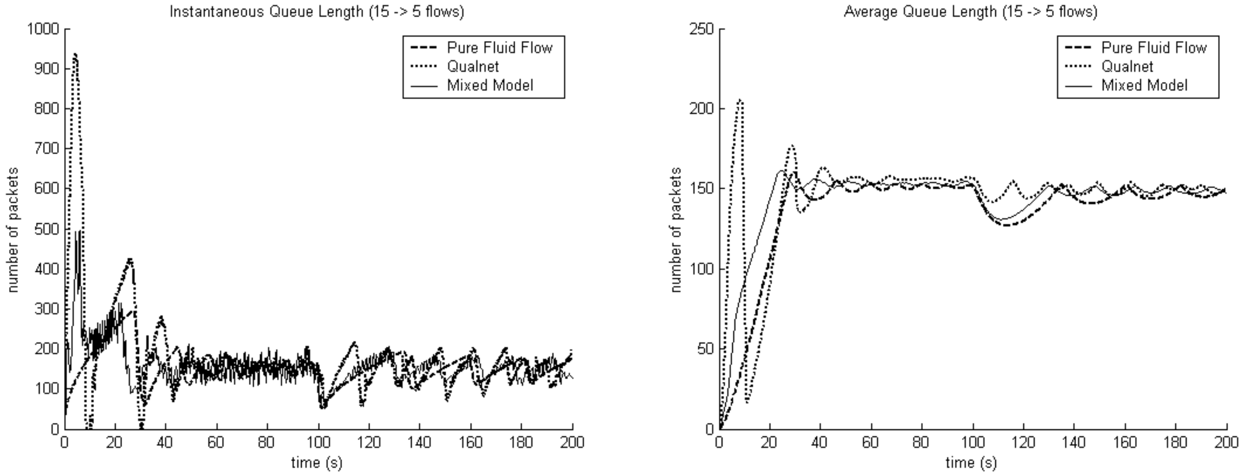

Figure 7: Instantaneous and average queue lengths at node 2, 0-100s (15 flows), 100-200s (5 flows).

5.4. End-to-end Delays

We also measured the end-to-end delay from the source to destination from node 0 to 1 in the experiment performed in section 5.2. As mentioned before this delay is estimated by the network statistics obtained in the fluid flow model. Again the results in Figure 8 show good matches with those obtained using the packet level simulator QualNet.

At this time we want to note that the fluid flow model is developed particularly for large number of flows in the network. The analytical calculations obtained are expected behavior, which approximates more closely to real situations as the number of flows in the network becomes large. In this sense, our mixed model is best suited to simulations of networks with sufficient flows inside the fluid flow section.

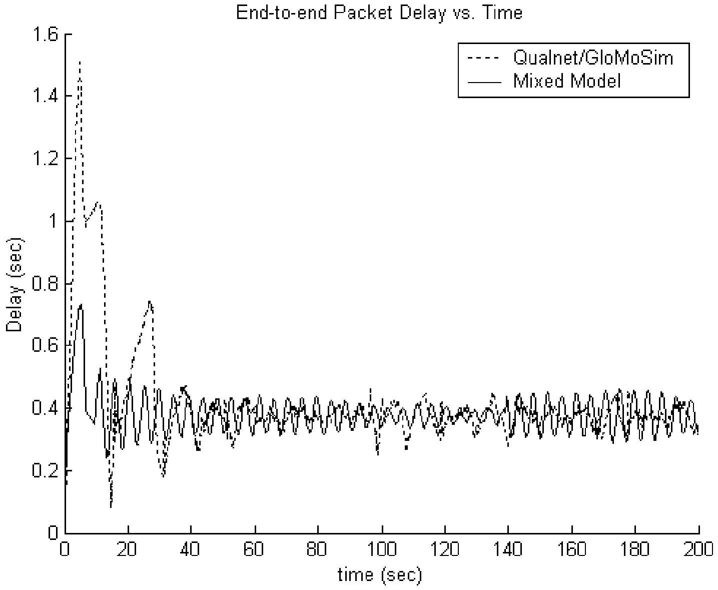

Figure 8: End-to-end packet delay from node 0 to 1, with connections 0→1 (1 flow), 4→5 (20 flows).

6. PERFORMANCE

Having studied the validity of our mixed model, we now turn to evaluate its performance.

6.1. Effect of time averaging intervals

As discussed before, the data rates for fluid flow calculations must be updated periodically in the mixed model. The data rates are averaged over the selected time interval; clearly, the number of fluid flow calculations increase as the data rate is updated more frequently. As the fluid flow model is analyzed using compiled MATLAB™ code, there is some overhead in converting data into the formats used by their compiled code. The differential equation solution of the fluid flow analysis can also be an expensive operation, depending on the nature of the network and input flows. In general, the averaging interval should be chosen to minimize the execution time while obtaining an accurate prediction of network performance.

As an example, for the first experiment shown in the validation section (Section 5.2), several time averaging intervals were used. Figure 9 shows the ratio of execution time of the mixed model to the execution time of the packet level simulation for identical scenarios, as a function of the selected averaging period. As seen from the figure, the execution time depends heavily on the selected time interval. For time intervals greater than 5s, the execution time stays relatively constant. This is because as we decrease the number of fluid flow calculations, each calculation now takes more time (for longer simulation time interval). For smaller time intervals, the calculations and overhead cost associated with each calculation increases quickly.

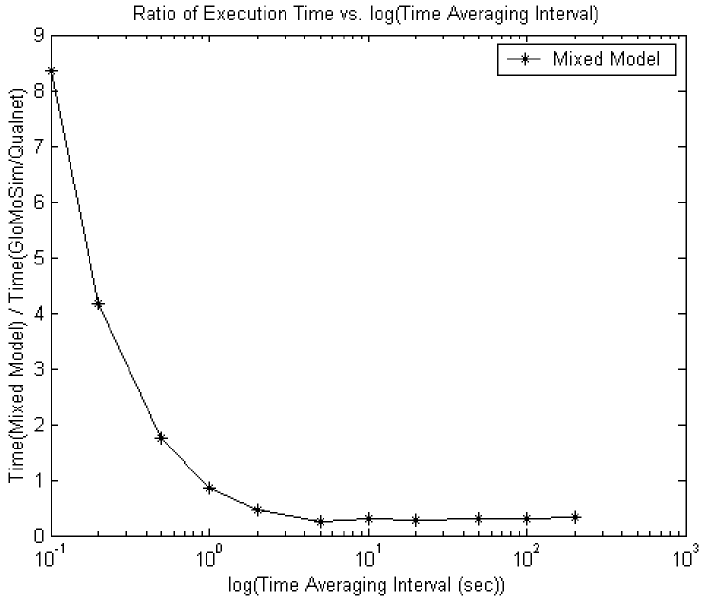

Figure 9: Ratio of Execution Time for various time averaging intervals.

The next experiment further investigates the impact of different time averaging intervals. The experiment uses 10 TCP flows that are modeled entirely by the fluid flow model, and 1 UDP (CBR) connection from node 0 to 1 that is modeled using mixed mode simulation. The UDP connection has bursts of data at 5s intervals, and each burst (2Mbps) lasts for about 0.1s. Without the bursts, the queue lengths will eventually stabilize. As Figure 10 and Figure 11 show, choice of smaller averaging time intervals can capture the bursts of data rate changes (the “spikes”), while selection of longer averaging intervals causes the data rates to be averaged over the whole interval (darker, flat line with dots after 60s). Of course, smaller time averaging intervals lead to longer execution times. Depending on the network topology and flow parameters, selection of a large value for the time averaging intervals may lead to inaccuracies in predicting instantaneous behavior.

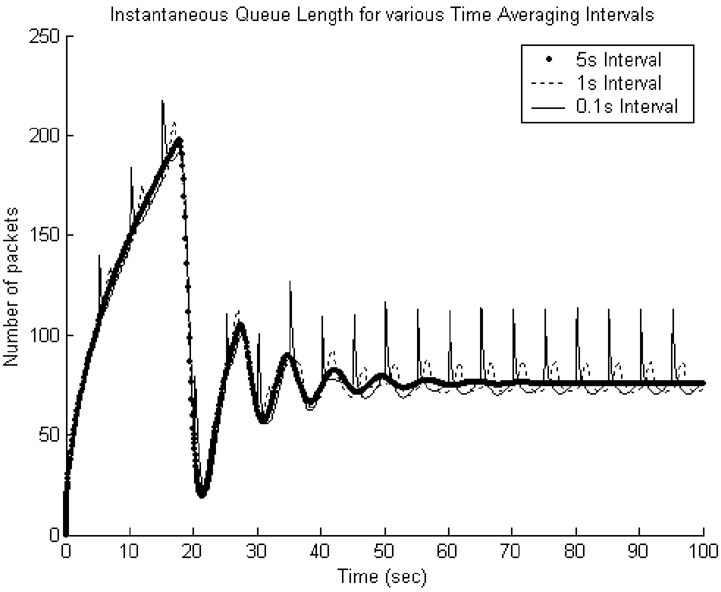

Figure 10: Queue lengths for various time averaging intervals for a network of TCP flows with bursts of CBR traffic.

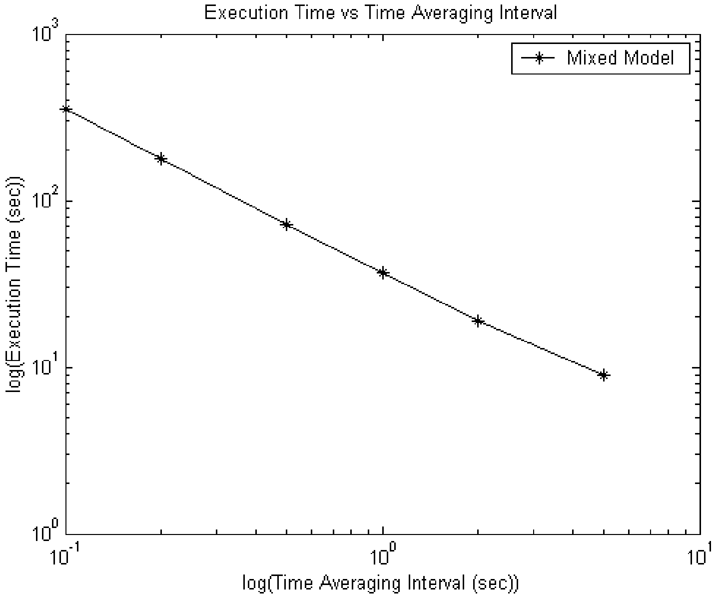

Figure 11: Execution times for various time averaging intervals for a network of TCP flows with bursts of CBR traffic.

We repeat the previous experiment using TCP connections. 25 flows are defined from node 4 to 5 that run unchanged from 0s to 200s. However, from node 0 to 1 the flows are varied as follows: the TCP connections from node 0 to 1 start in groups of 5 FTP flows, and each flow is on for 100s. The groups of flows start from 0s, 20s, 40s, 60s, 80s, and 100s. In other words, from 0 to 20s there are 5 flows from node 0 to 1, from 20s to 40s 10 flows, 40s to 60s 15 flows, and so on. The process is reversed starting at time 120s, where the number of flows is decreased by 5 flows every 20s. The bottleneck link capacity is 2Mbps.

Similar to the CBR case, the execution time is seen to increase quickly with decreasing time averaging interval (Figure 12). We also obtain the data rates from QualNet nodes entering the fluid flow section for various time averaging intervals (Figure 13). For smaller intervals, we can capture more bursty data rate values. From the figures, as the time interval decreases from 1s to 0.05s, the instants with great data rate changes can be captured.

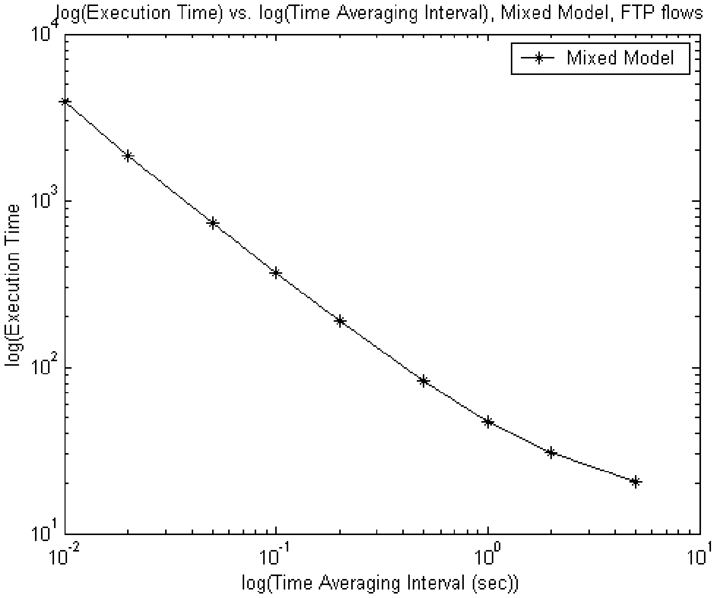

Figure 12: Ratio of Execution times for various time averaging intervals for TCP flows.

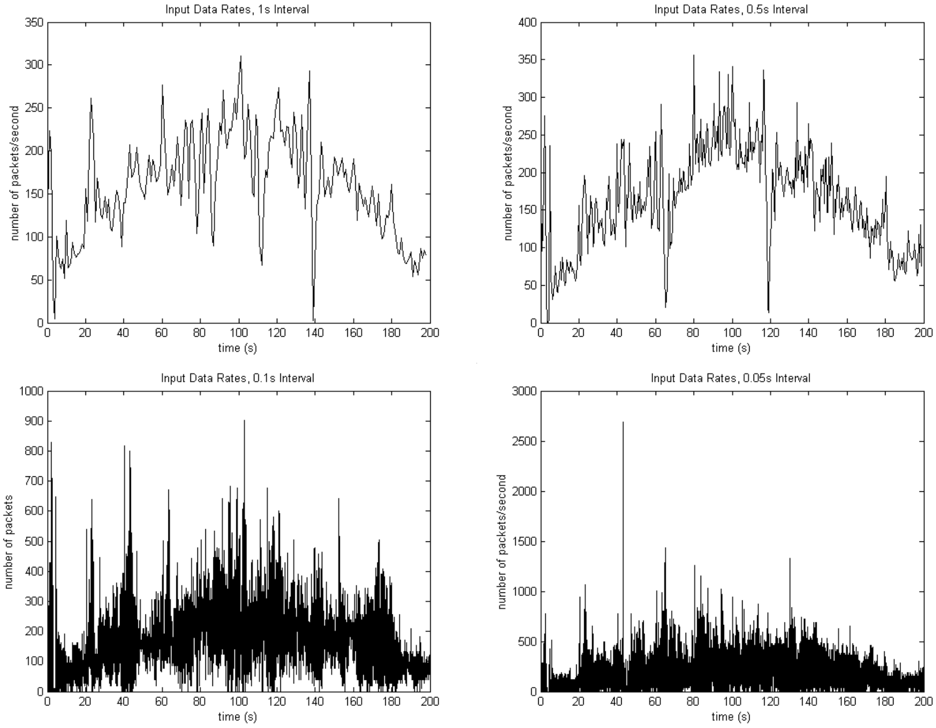

Figure 13: Data rates from QualNet nodes entering fluid flow section for various time averaging intervals.

6.2. Increasing Link Capacity

Next, we examine the impact of increasing the link capacity of the bottleneck link between nodes 2 and 3 of the network described in the previous section. We set up 100 TCP flows from nodes 4 to 5 inside fluid flow section, and 10 TCP flows from nodes 0 to 1. Then the bottleneck link capacity is increased from 2Mbps to 20Mbps. Time averaging interval is 1s.

As the link capacity is increased, more packets can be sent across the network, and since the packet level version of QualNet needs to simulate more packets than the mixed model, we expect the execution time of the discrete-event simulator will increase faster. This is shown to be the case in Figure 15 and Figure 16. We also show the average throughput of the 10 connections for both simulators, which are closely matched. This scenario demonstrates the scalability of the mixed mode simulator with respect to link capacities for large flows in the fluid flow section as compared with a pure packet-level model.

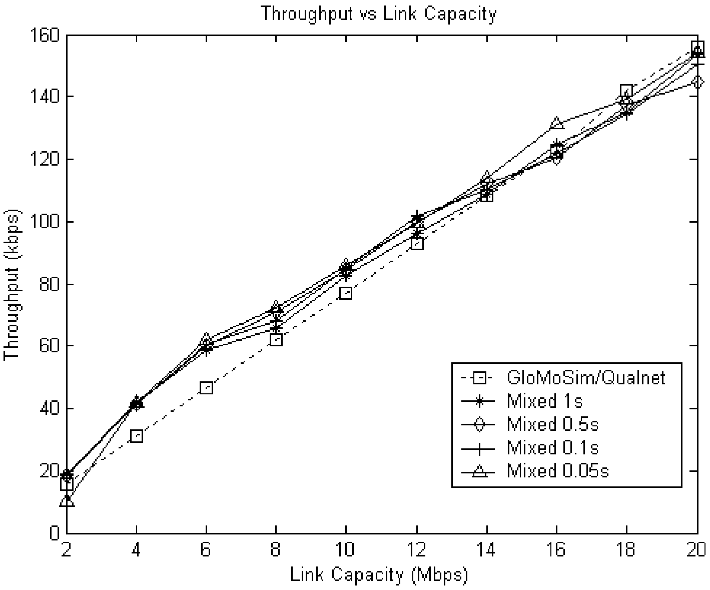

Figure 15: Throughput versus link capacity for QualNet and mixed models.

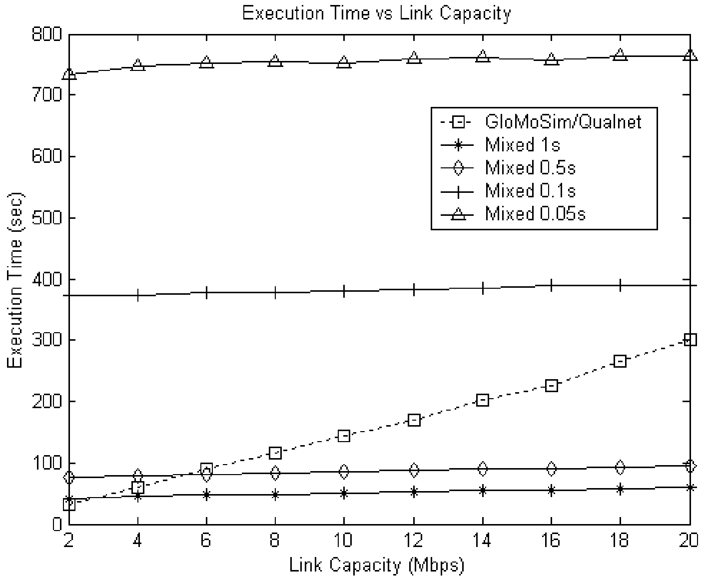

Figure 16: Execution time versus link capacity for QualNet and mixed models.

7. CONCLUSIONS

This paper presented an approach to the integration of a network analytical model into a packet level simulator. The fluid flow analytical model [4] has been shown to be able to capture the dynamics of TCP flows with RED as the network AQM policy, and can scale well to a large number of flows. The paper showed how the fluid flow model may be incorporated into QualNet, a packet-level network simulation library. The paper showed that for the preliminary set of scenarios considered in this paper, the resulting mixed mode simulator shows good validation with the results obtained from pure packet-level simulation.

In terms of performance, the use of the mixed mode simulator is clearly beneficial in cases where a section of the network with relatively heavy traffic can be identified. In these situations using pure packet level simulators requires transmitting a lot of messages and packets in the simulation, leading to a substantial increase in the execution time of the model. On the other hand, using a mixed model can take advantage of the scalability of analytical models. Nevertheless, some overhead cost is associated with the setup and solution of the fluid flow model. We employed a time averaging approach on data input rate changes to reduce the number of more expensive fluid flow calls. The results of our experiments show that in order to derive significant benefits from the mixed mode simulator, the time averaging interval for data rate updates and fluid flow model calculations must be selected based on a careful understanding of network characteristics.

8. REFERENCES

1. Lokesh Bajaj, Mineo Takai, Rajat Ahuja, Rajive Bagrodia, "Simulation of Large-Scale Heterogeneous Communication Systems", Proceedings of MILCOM'99, November 1999.
2. Xiang Zeng, Rajive Bagrodia, Mario Gerla, "GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks", Proceedings of the 12th Workshop on Parallel and Distributed Simulations -- PADS '98, May 26-29, 1998 in Banff, Alberta, Canada.
3. M. Takai, R. Bagrodia, K.Tang, M. Gerla, "Efficient Wireless Network Simulation with Detailed Propagation Model", ACM Wireless Network Journal, December 2000.
4. V. Misra, W. Gong, D. Towsley, "Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED", Proceedings of ACM/SIGCOMM, 2000.
5. V. Misra, W. Gong, D. Towsley, "Stochastic Differential Equation Modeling and Analysis of TCP-window size Behavior", Technical Report ECE-TR-CCS-99-10-01, 1999. Presented at Performance 99, October 1999 Istanbul.
6. Y. Guo, W. Gong, D. Towsley, "Time-stepped Hybrid Simulation (TSHS) for Large Scale Networks", Proceedings of IEEE Infocom 2000, Tel-Aviv, Israel, March 2000.
7. A. Yan and W. Gong, "Time-Driven Fluid Simulation For High-Speed Networks With Flow-Based Routing", Proceedings of the Applied Telecommunications Symposium '98, pp. 153-158, Boston, MA, April 1998.
8. S. Floyd and V. Jacobson, "Random Early Detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking 1(4), August 1997.
9. S. Floyd, "Recommendation on using the "gentle_" variant of RED", http://www.aciri.org/floyd/red/gentle.html, March 2000.
10. M. Vojnovic, J. Y. Le Boudec, C. Boutremans, "Global fairness of additive-increase and multiplicative-decrease with heterogenous round-trip times", Proceedings of IEEE INFOCOM 2000, March 2000.
11. V. Jacobson, "Congestion avoidance and control", Computer Communications Review, vol. 18, August 1988.
12. GloMoSim: Global Mobile Information Systems Simulation Library. http://pcl.cs.ucla.edu/projects/glomosim/.
13. QualNet User Manual. http://www.scalable-networks.com/.
14. The Network Simulator – ns-2. http://www.isi.edu/nsnam/ns/.