Improving the Operation Efficiency of Supermedia Enhanced Internet Based Teleoperation via an Overlay Network*

Zhiwei Cen†, Amit Goradia‡, Matt Mutka† and Ning Xi‡
†*Dept. of Computer Science and Engineering*
‡*Dept. of Electrical and Computer Engineering*
Michigan State University
East Lansing, MI 48824, USA
xin@egr.msu.edu

Wai-keung Fung and Yun-hui Liu
Dept. of Automation and Computer Aided Engineering
Chinese University of Hong Kong
Shatim, N.T., Hong Kong
wkfung@acae.cuhk.edu.hk

***Abstract*— For Internet based real-time teleoperation systems, random time delay can cause instability in the closed loop control system and hence hinder task accomplishment. Event based control systems have been proposed to overcome the instability caused by the random time delay. High latency at the transport layer can still impede effective and reliable execution of tasks with high dexterity requirements. Network QoS based dynamic resource allocation has been proposed to increase the efficiency and reliability of task execution. However, these approaches only try to mitigate or overcome the effects of random time delay and do not address the cause of latency issues in the communication channel.**

This paper addresses the efficiency and reliability requirements for supermedia enhanced teleoperated systems by reducing the end-to-end transmission latency through the use of overlay networks. The proposed system reduces the transmission latency by using multiple, disjoint paths in overlay networks. The proposed system facilitates reliable and efficient task completion for tasks with high dexterity requirements. Experimental validation of the proposed teleoperated system using the PlanetLab Network is provided for the task of teleoperating a mobile manipulator system.

***Index Terms*— Teleoperation, Supermedia, Overlay Networks, Internet.**

I. INTRODUCTION

Internet-based teleoperation has been extensively researched and studied because of its many benefits, such as increasing the reachability and safety of human operators [1]. Supermedia enhanced teleoperation involves the flow of control commands from the operator to the robot and various media feedback streams such as video, audio, haptic etc., from the robot to the operator. The feedback information enables the operator to be aware of the current state of the robot and its surroundings. This situational awareness facilitates closed loop control and effective task completion. With the control commands and the feedback information communicated through the Internet, the operator and the robot form a closed loop feedback control system. The Internet serves as an "action superhighway" instead of an information superhighway as seen by traditional networking applications. All the information flowing in a real-time teleoperation system is collectively termed as supermedia [1]. Supermedia differs from traditional multimedia in that a larger variety of media are involved, all of which are to be transmitted through a shared path in the Internet and each of them has a different Quality of Service (QoS) requirement that may even change during the execution of the task.

One of the major challenges of Internet based teleoperation is to make the unpredictability caused by the Internet to be transparent to the control loop of the system. Among all the uncertainties, time delay is one of the biggest obstacles to build a stable teleoperation control system. The performance difficulty caused by the Internet based teleoperation system is a result of using time as a reference for different system entities. The event based control model [1] has been proposed to combat the instability arising out of the random time delay in the control loop. In the event-based control approach, a non-time based reference is used. An event reference, which is a monotonically increasing parameter, is used to synchronize the operator and the robot. Using the event based control model, the system stability can be ensured. However, the effective performance of highly dexterous tasks using the teleoperation system mainly depends on the quality of transfer of the supermedia streams involved in the system. A dynamic QoS based resource allocation scheme has been presented in [2] to address the problem from the application perspective. Fung, et al. introduced a Task Dexterity Index (TDI [2]) of the robotic task associated with each supermedia stream. A Task Dexterity Index (TDI) is generated for each data stream to describe the bandwidth requirement of the robotic task. With the help of TDI, bandwidth for all data streams involved in the teleoperation system is dynamically allocated during task execution.

However, the work of [2] still cannot reduce the latency imposed by the network. In this paper we addresses the reliability and efficiency problem of Internet based teleoperation systems from the transport service perspective. We propose to use an overlay network to transport supermedia streams over multiple overlay paths in order to reduce the end-to-end latency through redundancy and disjoint paths.

*This work is partially supported by NSF Grants: IIS-9796300, IIS-9796287, EIA-9911077, INI-0314120, and HNI-0334035.

Various other efforts have also focused on providing a QoS aware transport service over the Internet. Many research efforts aim to improve QoS levels for multimedia applications over the current best effort Internet. However these transmission mechanisms do not solve issues in supermedia transmission because supermedia streams have dynamic QoS requirements associated with the task execution process. The priority of a supermedia stream will change dynamically when the teleoperation system switches from one task to another [2], [3].

In related research we proposed Supermedia TRansport for teleoperation over Overlay Networks (STRON) approach to improve the reliability and efficiency of teleoperation systems [4]. STRON takes advantage of multiple disjoint overlay paths and forward error correction encodings to improve the QoS performance. TCP Friendly Rate Control (TFRC1 [5]) is used as the congestion control mechanism for each overlay connection, which ensures that the supermedia traffic remains friendly to other Internet traffic.

The remainder of this paper is organized as follows. Section II introduces overlay networks and elaborates on their advantages for supermedia transport. Section III describes the architecture of the proposed supermedia transport layer and discusses a method for selecting best paths in an overlay network. Experimental testing and analysis are provided in section IV and section V provides concluding remarks.

II. Overlay Networks

This section presents a brief overview of overlay networks and current research efforts aimed at improving multimedia applications based on their widespread use. It also introduces, in particular, the PlanetLab network which is used in the experimental implementation of the proposed approach for reliable and efficient Internet based teleoperation systems.

An overlay network is composed of a set of IP (Internet Protocol)-layer network paths. These nodes can be deployed in the same or different autonomous systems (ASes). Packets may be routed among overlay nodes according to overlay link performance measurements. Since the setup of an overlay network does not require changing the underlying network infrastructure, many overlay networks are used to deploy emerging networking applications. Common overlay networks include RON [6] and PlanetLab [7]. A Resilient Overlay Network [6] allows distributed applications to detect and recover from link failures or performance degradation, which may be much quicker than solely relying on the routing protocols. In order to meet the reliable and fast transmission requirement of teleoperation systems, we can utilize several overlay network paths supported by the overlay nodes. Research shows that when the network is under utilized, parallel connections can achieve fair utilization among common connections [8]. Statistics shows in large academic networks such as Abilene [9], most network links are usually under utilized.

In order to provide an effective transport service for the teleoperation system, the overlay paths should be as physically as disjoint as possible. Cui [10] uses a probability model to represent the disjoint degree of two overlay paths. The Control Overlay Protocol [11] divides the overlay nodes into regions. Each region has a super node. The super node probes its subordinate nodes to collect their disjoint information. A routing underlay approach for overlay networks [12] introduces a routing underlay that sits between the overlay network and the Internet. The routing underlay collects topological information from the Internet and this information is used in the overlay network to provide a more reliable and efficient service.

Various research efforts use overlay networks as a means to improve quality of service for multimedia applications [13], [14]. However those methods do not meet the latency criticl requirement of supermedia applications.

PlanetLab [7] is an overlay network used to design, evaluate and deploy geographically distributed network services. In the proposed approach for reliable and efficient teleoperation, PlanetLab nodes serve as overlay network relay agents to transmit supermedia traffic through selected overlay routes in addition to the default IP routes determined by the routing protocols. By diverting the mission-critical traffic into geographically distributed overlay paths, we expect that in the face of local networking congestion or disruption, the alternative overlay paths will still be able to deliver the application packets to the destination in a timely fashion.

III. System Architecture

A. System Overview

The basic idea of using multiple paths to route supermedia streams can be explained as follows. Each supermedia stream contains a series of messages generated by the teleoperation application in a variable or constant bit rate. The message is again chopped into packets with a certain size (such as the Maximum Transfer Unit or MTU) determined by the networking layer. Given p packets for a certain message that needs to be transmitted, the sender agent encodes the p packets into αp packets, where α $(\alpha \geq 1)$ is called the stretch factor. The encoded data packets are scheduled to be transmitted over multiple overlay paths. As soon as the receiver collects $(1+\epsilon)p$ distinctive encoded data packets, the decoding algorithm can reconstruct the original data packets. Here ϵ is called the reception overhead. The reception overhead is zero for some encoding algorithms and a small number for others. A class of erasure codes that has this property is called a digital fountain code [15]. By using a digital fountain code, the transport protocol can provide a rather reliable transport service without using acknowledgments and retransmissions. The most common digital fountain codes are Reed-Solomon codes and Tornado codes [15]. Figure 1 illustrates the mechanisms of transporting supermedia streams over multiple paths. Source data were encoded with stretch factor 2 and sent over two disjoint

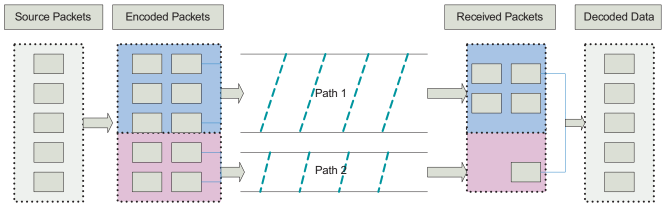

Fig. 1. An example of transporting messages over multiple paths

paths. Six packets were sent over path 1, which has a higher bandwidth and lower loss rate. Four packets were sent over path 2, whose bandwidth is low and the loss rate is high. As soon as the receiver collects 5 data packets, which are enough for the decoder to work, it signals the sender to stop sending packets and decodes the information. This form of implementation has a great advantage over traditional transport protocols because it eliminates the requirement of most retransmissions when packet loss occurs.

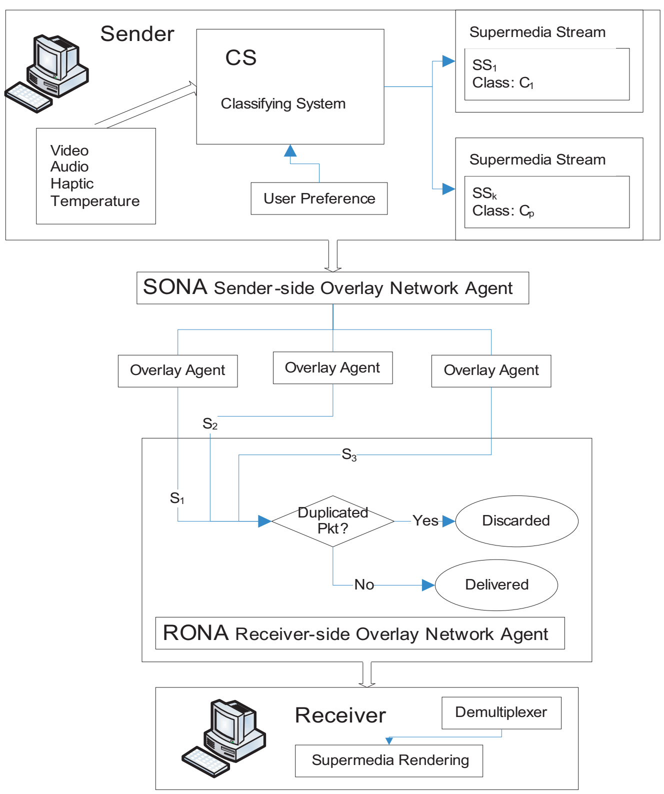

Fig. 2. The architecture of the multiple path sender and receiver modules.

B. System Architecture

The system architecture is shown in Figure 2. At the sender side, the supermedia streams are classified according to their roles in the teleoperation system by the Classifying System (CS). The function of the CS module is similar to the task driven resource allocation module presented in [2]. Each class of supermedia stream has its own QoS requirement. The various data streams are classified into the respective classes using a pre-specified user preference and the current task requirements which can be represented using the TDI's as shown in [2]. The streams are then delivered to the Sender-side Overlay Network Agent (SONA), which is responsible for transporting each class of streams according to its QoS specifications. The SONA transmits the supermedia streams via multiple routes to overlay agents at various locations over the world. The overlay agents relay the received packets to the Receiver-side Overlay Network Agent (RONA). The RONA receives and decodes the supermedia packets routed through multiple paths in the overlay network. It decodes the data from the fastest path and discards the redundant data it receives later. Afterward the sender was signaled to stop sending more packets. It then passes the decoded streams to the demultiplexer which delivers the various supermedia streams to their respective rendering agents to present to the human operator. The robot and operator both have a SONA and RONA implemented which act as transport service agents. More details on the SONA and RONA are described in [4].

C. Path Selection Problem

In this section we formulate the problem of path selection for the overlay network based on QoS characteristics. N disjoint paths in the network with measured QoS parameters are given. For each path k we have 1) single trip time delay d_k in terms of bytes per second, the average throughput r_k in terms of bytes per second, and the packet loss rate β_k. The size of the message, E, for each supermedia stream is given.

We need to choose M disjoint paths out of the N given paths to minimize the latency between the sender and receiver. For each path i ($i \in [1, M]$) of the M path, we also need to calculate V_i, which is the amount of data injected into this path by the sender. The system redundancy coefficient γ shows how much redundancy the system has over unexpected packet loss. Assume E_i is the effective data gathered from path i that is used in the data construction process, and we have

$$E_i = \frac{V_i(1-\beta_i)}{\gamma}. \tag{1}$$

Assuming t is the time for the receiver to receive and reconstruct the original data, we have

$$t = \frac{V_i}{r_i} + d_i \tag{2}$$

and

$$V_i = (t - d_i)r_i \tag{3}$$

According to (1)

$$E_i = \frac{r_i(t-d_i)(1-\beta_i)}{\gamma} \tag{4}$$

We have

$$\sum_{i=1}^{M} E_i = E(1+\epsilon) \tag{5}$$

where ϵ ($\epsilon \in [0, 1)$) is the reception overhead of the digital fountain code, or

$$E = \frac{\sum_{i=1}^{M}[(t - d_i)r_i(1 - \beta_i)]}{(1 + \epsilon)\gamma} \tag{6}$$

Solving t yields

$$t = \frac{E(1 + \epsilon)\gamma + \sum_{i=1}^{M} r_i(1 - \beta_i)d_i}{\sum_{i=1}^{M} r_i(1 - \beta_i)} \tag{7}$$

To solve the problem, we may try set $s = \{(i_1, i_2, \cdots, i_M)|i_p \in [1, N], p \in [1, M]$, and for any $p \neq q, p \in [1, M], q \in [1, M], i_p \neq i_q\}$, which is a combination of set $[1, N]$ to minimize (7), which is

$$t = \frac{E(1 + \epsilon)\gamma + \sum_{j \in s} r_j(1 - \beta_j)d_j}{\sum_{j \in s} r_j(1 - \beta_j)} \tag{8}$$

The most straightforward method is to enumerate all the $\binom{N}{M}$ sets to find the subset s of $[1, N]$ that minimizes (7). When N is small, which is true in most cases, this method works well. When N is large, some heuristic approaches can be used.

The volume that needs to be sent over a selected disjoint path i, which is V_i, can be found by using (3).

IV. Experimental Implementation and Testing

The advantages of the proposed algorithm are demonstrated using experiments with a teleoperated mobile manipulator robot.

A. Description of the Experimental Platform and Setup

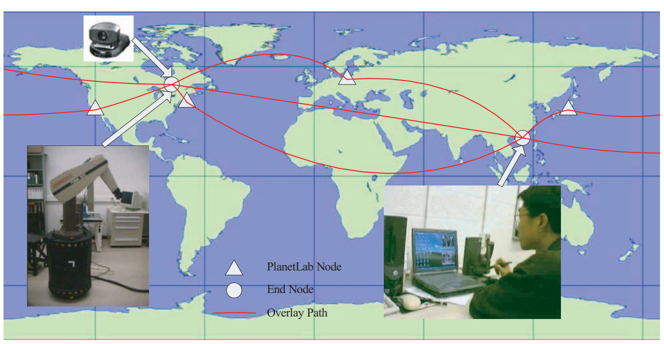

Fig. 3. The experimental test platform

Figure 3 shows the hardware structure of the Experimental test platform. It consists of a robot manipulator, a haptic device and a vision system connected through the Internet. The haptic device (phantom) is used to obtain velocity commands from the operator and render force feedback from the robot. The vision system is used for visual feedback from the robot and its surroundings.

The communication channel is composed of several nodes in the PlanetLab overlay network. The nodes were selected at myriad locations over the world from several distinct Autonomous Systems (AS). Multiple routes from the source to the destination for the various supermedia streams were implemented via the overlay network. The stretch factor α for the various supermedia streams was preset to be equal to the number of paths being used. The reception overhead factor, ϵ, was set to 0.

The robot system was located at Michigan State University (MSU), USA and the operator was located at Chinese University of Hong Kong (CUHK), Hong Kong, China. Overlay routing nodes for implementing the multiple path approach were selected from many different location over the world. Four disparate paths were heuristically chosen for the overlay routing links. The four paths chosen can be tabulated as 1) east coast of USA path, 2) west coast of USA path, 3) a path through Europe and 4) a path through Asia. Experiments were carried out utilizing 1, 2, 3 and 5 paths (including the default Internet path). In order to

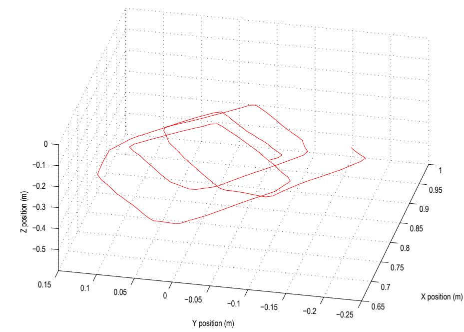

Fig. 4. The commanded position for teleoperation task.

achieve repeatable operation to compare the results of the various experiments, the teleoperation task was simulated using a stream of actual teleoperation commands recorded in a previous trial. The task involved moving the robot end-effector in concentric circles in the $X-Y$ plane as is shown in figure 4. This task has high dexterity requirements as is explained in [3]. The same set of commands was played back at the Hong Kong peer for each experimental run.

B. Experimental Data and Analysis

The teleoperation task was carried out for different number of paths used to transport the supermedia streams. The Round Trip Time (RTT) delay, motion of the robot and effective time to completion of the task are recorded and analyzed.

1) Experimental Results and Discussions: First, the teleoperation task was carried out using only the default route as a single data path between the operator and the robot. Next the proposed system using a total of up to 5 routes (4 routes via the overlay network and one default route), was used to route supermedia streams involved in the teleoperation task. Figures 5 and 6 depict a 1 minute time slice of typical sample runs for the two experimental cases implemented with and without the use of the proposed framework.

Figure 5 shows the motion of the MSU robot in the $X-Y$ plane commanded from the CUHK peer for both single and multiple routes. The initial positions of the robot are the same for both the trials. Sub figures 5(a) and 5(c) depict the motion of the robot in the X direction and 5(b) and 5(d) shows the motion of the robot in the Y direction

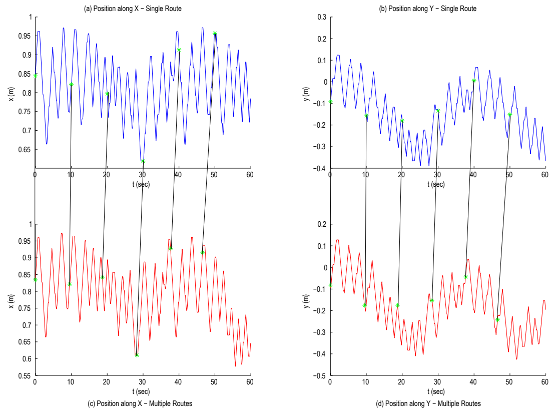

Fig. 5. Motion of the robot in the $X - Y$ plane without drops.

for the two experimental cases. Sampled event references are marked on the sub-figures and the corresponding event references for the two sample runs are connected in the appropriate sub-figures. We observe that the event reference lines start getting crooked as the task progresses indicating the speeding up of the task execution in case of the proposed approach due to a faster transport service and reduced end-to-end latency.

Figure 6 shows the measured round trip time delay trace during the task execution for both the trial runs. It can be seen that the RTT delay is significantly higher in the case of the single path. The proposed multiple paths via overlay networks approach significantly reduces the end-to-end latency in the control loop which results in efficient task execution.

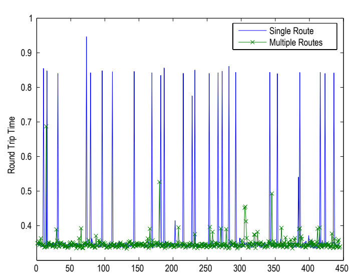

Fig. 6. Round Trip Time (RTT) delay with drops.

Table I lists the various statistical quantities of interest for the various experiments conducted using the proposed approach. The mean RTT delay decreases as a function of the number of paths used. Further the variance of the RTT delay decreases quite rapidly with the increase in the number of paths used for the supermedia transport. This decrease in the variance of the RTT with the number of paths used is very significant as it improves the reliability and performance of the communication channel used. The mean time per event is an indication of the overall efficiency of the system. A lower mean time per event indicates the system performs the task more efficiently. These results reinforce the idea that by using multiple overlay paths to

TABLE I

STATISTICAL DATA FOR EXPERIMENTS CONDUCTED

No. of Paths	No. of Events	RTT (mean) (sec)	RTT (variance)	Execution time (sec)	Mean time per event (sec/event)
1	4014	0.4488	0.7256	1857.371	0.4627
2	4289	0.3949	0.1345	1789.598	0.4173
3	4194	0.3742	0.0473	1598.912	0.3811
5	5549	0.3580	0.0052	2047.581	0.3690

provide the supermedia transport service, the data receiver will always take the data packet that arrives first from the fastest path, thus the round trip time of the packets is reduced. It is typical that network congestion or router capacity overflows occur in a restricted geographical area. The multiple paths provide a convenient and resilient way to circumvent the local networking performance degradation without introducing much overhead. Using multiple overlay paths does increase the networking traffic e.g., four overlay paths increase the networking traffic from about 1KBPS to 5KBPS when transmitting the latency critical data. However, this increase is not at all significant when compared to the streaming rates of audio and video data which are close to 500KBPS. What's more, the TFRC protocol in the STRON approach ensures the supermedia traffic is friendly to other traffic.

Table II indicates the percentage of time a particular path has the lowest latency. It shows the contribution of the individual paths to the overall supermedia transport system. It can be seen that the default route (Path #1) is the most used route. This is expected, however in the event of network congestion in the default path, various other routes can be used to maintain low latency.

TABLE II

ROUTING PERCENTAGE PER PATH

Number of Paths	Path #1 (%)	Path #2 (%)	Path #3 (%)	Path #4 (%)	Path #5 (%)
1	100	-	-	-	-
2	97.73	2.27	-	-	-
3	96.13	1.89	1.98	-	-
5	55.1	39.9	2.06	1.64	1.30

2) Experiments with Simulated Packet Loss: Due to the difficulty in capturing the characteristics of a network route in the limited time span of conducting the real-time teleoperation task, a packet loss simulation was also implemented in one of the sets of experiments reported. For this experiment, a 10 % packet loss across all paths in the system was simulated. The loss model is implemented using a stochastic packet loss model which drops a subset of packets according to an exponential processes. Some measurements [16] show that the packet loss events of the Internet may be independent and fit an exponential distribution.

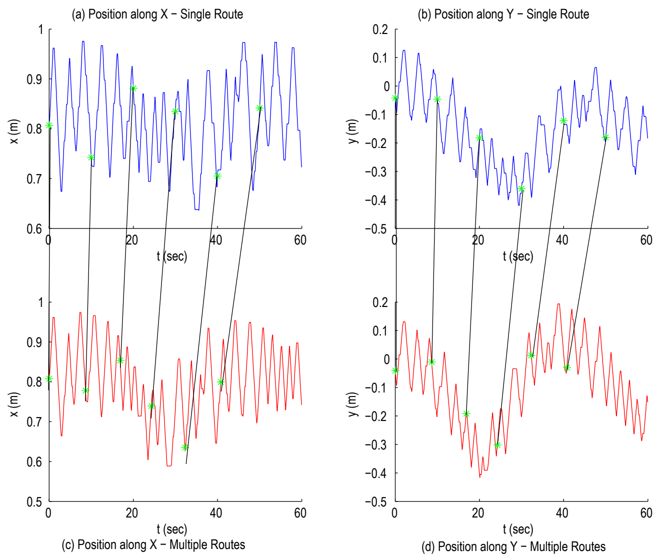

Fig. 7. Motion of the robot in the $X - Y$ plane with drops.

Figure 7 shows the X and Y position of the robot and the figure is similar to figure 5. However, in this case we notice that the combined effect of random delay and packet loss is much more severe for the single path model than the multiple path model as the event lines are more slanted. Figure 8 shows the RTT delay trace during the task execution for both the trial runs. It can be seen from the figure that packet loss affects RTT delay for single routes more severely than the proposed approach. These results are in tune with the fact that the proposed method is more robust to the congestion and failure of individual network paths.

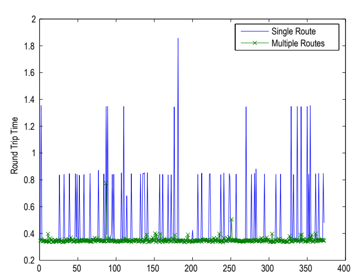

Fig. 8. The Round Trip Time (RTT) delay with drops.

V. Conclusion

The work presented in the paper provides a fundamental QoS based framework to improve the efficiency and reliability of real-time teleoperation tasks. The approach adopted is aimed at designing a reliable and low latency transport service for various types of supermedia streams used in Internet based teleoperation systems.

Experiments using multiple paths routing of supermedia streams via an overlay network infrastructure for a real-time teleoperation task are described. The results obtained demonstrate that the proposed approach can be effectively used to reduce the mean and the variance of the end-to-end transmission latency and hence increase the efficiency of teleoperation systems. Further, using the proposed approach, the system exhibits improved reliability in case of path congestion. The proposed approach reduces the time for task completion and increases the efficiency and reliability of task accomplishment for teleoperated systems.

The results presented in this paper indicate that there is potential to exploit overlay networks to increase the efficiency and reliability of teleoperation systems and further meet the QoS objectives for supermedia systems in order to provide and Internet based ‘action superhighway‘ on the existing network infrastructure.

References

[1] Imad Elhajj, Ning Xi, Wai Keung Fung, Yun hui Liu, Tomoyuki Kaga, and Toshio Fukuda, “Supermedia in Internet-Based Telerobotic Operations,” in *Management of Multimedia Netowrks and Services 2001*, 2001.

[2] Wai Keung Fung, Ning Xi, Wang-tai Lo, BooHeon Song, Yu Sun, and Yun hui Liu, “Task Driven Dynamic QoS based Bandwidth Allocation for Real-time Teleoperation via the Internet,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003.

[3] Long Pan, Amit Goradia, and Ning Xi, “Dynamic Multi-objective Optimal Task Distribution for Teleoperated Mobile Manipulators,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.

[4] Zhiwei Cen, Matt Mutka, Danyu Zhu, and Ning Xi, “Supermedia Transport for Teleoperations over Overlay Networks,” *Technical Report MSU-CSE-05-01, Department of Computer Science and Engineering, Michigan State University*, 2005.

[5] Mark Handley, Sally Floyd, Jitendra Padhye, and Joerg Widmer, “TCP Friendly Rate Control (TFRC): Protocol Specification,” *RFC 3448, Proposed Standard*, 2003.

[6] David Andersen, Hari Balakrishnan, Frans Kaashoek, and Robert Morris, “Resilient Overlay Networks,” in *18th ACM Symposium on Operating Systems Principles*, 2001.

[7] PlanetLab, “PlanetLab: An open platform for developing, deploying, and accessing planetary-scale services,” *http://www.planet-lab.org/*.

[8] Thomas J. Hacker, Brian D. Noble, and Brian D. Athey, “The Effects of Systemic Packet Loss on Aggregate TCP Flows,” *Conference on High Performance Networking and Computing*, 2002.

[9] Abilene, “Indiana University Abilene NOC Weathermap,” *http://loadrunner.uits.iu.edu/weathermaps/abilene/*.

[10] Weidong Cui, Ion Stoica, and Randy H. Katz, “Backup Path Allocation Based on a Correlated Link Failure Probability Model in Overlay Networks,” in *10th IEEE International Conference on Network Protocols (ICNP’02)*, Paris, France, 2002.

[11] Rebecca Braynard and Amin Vahdat, “Masking Failures Using Anti Entropy and Redundant Independent Paths,” in *SOSP Work in Progress Presentation*, 2001.

[12] Akihiro Nakao, Larry Peterson, and Andy Bavier, “A Routing Underlay for Overlay Networks,” in *SIGCOMM*, Karlsruhe, Germany, 2003.

[13] L. Subramanian, I. Stoica, H. Balakrishnan, and R.H.Katz, “OverQoS: Offering Internet QoS using Overlays,” *Proc. HotNet-I Workshop, October 2002*, 2002.

[14] Thinh Nguyen and Avideh Zakhor, “Path Diversity with Forward Error Correction (PDF) System for Packet Switched Networks,” in *IEEE INFOCOM*, 2003.

[15] John W. Byers, Michael Luby, Michael Mitzenmacher, and Ashutosh Rege, “A Digital Fountain Approach to Reliable Distribution of Bulk Data,” in *SIGCOMM*, 1998.

[16] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, “Measurement and Modeling of the Temporal Dependence in Packet Loss,” *Proceedings of IEEE INFOCOM’99*, 1999.