# Supermedia Transport for Teleoperations over Overlay Networks*

Zhiwei Cen[1], Matt W. Mutka[1], Danyu Zhu[1], and Ning Xi[2]

[1] Dept. of Computer Science and Engineering
Michigan State University, East Lansing, MI 48824
{cenzhiwe,mutka,zhudanyu}@cse.msu.edu
[2] Dept. of Electrical and Computer Engineering
Michigan State University, East Lansing, MI 48824
xin@egr.msu.edu

**Abstract.** In real-time Internet based teleoperation systems, the operator controls the robot and receives feedback through the Internet. Supermedia refers to robotic control commands, video, audio, haptic feedback, and other sensory information in the control system. Traditional transport services may not be able to meet the timely transmission requirements and dynamic priority changes of supermedia streams. Supermedia TRansport for teleoperations over Overlay Networks (STRON) uses multiple disjoint paths and forward error correction encodings to reduce end-to-end latency for supermedia streams. Network routes and encoding redundancy may be adjusted dynamically to meet the supermedia QoS requirements. NS2 simulations and evaluations using available bandwidth traces from globally distributed computing nodes show that STRON can significantly reduce latency compared with available transport services.

**Keywords.** Teleoperation, Overlay networks, Forward error correction.

## 1 Introduction and Related Work

Teleoperation systems allow people to control automatic systems operating at remote sites. Traditional teleoperation systems use dedicated communication channels, which have higher costs and less flexibility. In an Internet based teleoperation system the operator manipulates a control device to issue task commands to the robot through the Internet. The feedback information, including video, audio and haptic information, enables the operator to be informed of the current state of the robot. We call all the information flowing in a teleoperation system supermedia [1]. Supermedia differs from traditional multimedia in that a larger variety of media are involved, and some media types are extremely sensitive to network delay. High latency may cause the operator to lose control of the remote robot.

* This paper was supported in part by NSF Grant No. 0334035.

In this paper we present the Supermedia TRansport for teleoperations over Overlay Networks (STRON). STRON takes advantage of multiple disjoint overlay paths (such as RON [2]) and forward error correction encodings to improve the QoS performance. The networking routes and encoding redundancy may be adjusted dynamically to meet the QoS requirements of the supermedia streams. TCP Friendly Rate Control (TFRC) is used as the congestion control mechanism for each overlay connection, which ensures that the supermedia traffic is friendly to other network traffic. A more detailed version of the work can be found at [3].

## 2 System Architecture

The basic idea of STRON may 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. Given $p$ packets for a certain message that needs to be transmitted, STRON encodes the $p$ packets into $\alpha p$ packets, where $\alpha$ ($\alpha \geq 1$) is the stretch factor. The encoded data packets are scheduled to be transmitted over multiple overlay paths, one of which is the IP path. As soon as the receiver collects $(1+\epsilon)p$ distinctive encoded data packets, the decoding algorithm can reconstruct the original data packets. Here $\epsilon$ is 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 [4]. By using a digital fountain code, the transport protocol can provide a rather reliable transport service without using acknowledgments and retransmissions.

The system architecture is shown in Figure 1. At the Sender-side Overlay

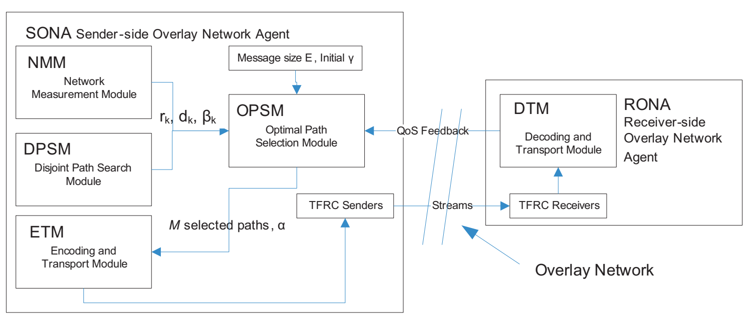


**Fig. 1.** The Architecture of STRON

Network Agent (SONA), the Disjoint Path Search Module (DPSM) runs a disjoint overlay network path searching algorithm through the connected overlay nodes. The Network Measurement Module (NMM) is usually deployed in overlay nodes as a common service. Some research efforts (such as *Pathload* [5]) have been dedicated to making accurate measurements of the network. The measurement results are fed into the Optimal Path Selection Module (OPSM).

The objectives of OPSM is to find the optimal disjoint path set to be used as active transmission paths and decide the amount of data sent over each active path. The design of the OPSM is discussed as follows. Suppose $N$ disjoint (or semi-disjoint) overlay paths were found in DPSM. For each path $k$ ($k \in [1, N]$), the following parameters are given: the single trip delay $d_k$ in terms of seconds, the average throughput $r_k$ in terms of bytes per second, and the packet loss rate $\beta_k$. The size of each message, which is the amount of effective data we need to transmit from the sender to the receiver, is $E$ in terms of bytes. We need $M$ 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 $M$ paths, we also need to calculate $V_i$, which is the amount of data injected into this path by the sender. The system redundancy coefficient $\gamma$ indicates the 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 $E_i = \frac{V_i(1-\beta_i)}{\gamma}$.

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

$$V_i = (t - d_i)r_i \tag{1}$$

For all Paths, we have $\sum_{i=1}^{M} E_i = E(1+\epsilon)$ where $\epsilon$ ($\epsilon \in [0, 1)$) is the reception overhead of the digital fountain code, or $E = \frac{\sum_{i=1}^{M}[(t-d_i)r_i(1-\beta_i)]}{(1+\epsilon)\gamma}$. Solving $t$ yields

$$t = \frac{E(1+\epsilon)\gamma + \sum_{i=1}^{M} r_i(1-\beta_i)d_i}{\sum_{i=1}^{M} r_i(1-\beta_i)} \tag{2}$$

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 (2). The most straightforward method is to enumerate all the $\binom{N}{M}$ sets to find the subset $s$ of $[1, N]$ that minimizes (2). When $N$ is small, which is true in most cases, this method works well. The volume that needs to be sent over a selected disjoint path $i$, which is $V_i$, may be found by using (1). The stretch factor $\alpha$ of the digital fountain encoding may be calculated as $\alpha = \frac{\sum_{i=1}^{M} V_i}{E}$. A *transport plan* consists of a series of active overlay network paths, QoS parameters of these paths, the redundancy factor $\gamma$, stretch factor $\alpha$ and the number of packets (or volume of data) to be sent over each active overlay path. The Encoding and Transport Module (ETM) encodes the supermedia streams using $\alpha$ and passes the encoded packets to the overlay transport service. The stretch factor $\alpha$ is found by the Optimal Path Selection Module.

At the receiver side, the Decoding and Transport Module (DTM) of the Receiver-side Overlay Network Agent (RONA) decodes the packet streams received by the TFRC receivers and notifies the sender when enough packets have been collected for a certain message or a message is lost if a timeout occurs. The feedback to the sender also includes information telling the sender to what degree the redundancy in the encoding is effective so that the sender can adjust the redundancy level accordingly.

## 3 Simulation

To evaluate the effectiveness and efficiency of the system, an Optimal Path Selection Module was implemented in the application layer of the NS2 simulator. In order to simulate the behavior of the system under lossy network conditions, Poisson arrival process and Markov Modulated Poisson Process (MMPP) were used to simulate the packet loss process.

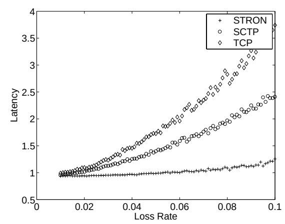


**Fig. 2.** Latency vs Loss Rate: STRON, TCP and SCTP (with 1 path)

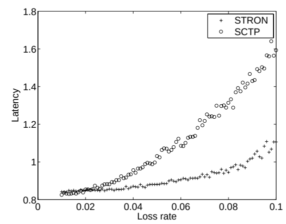


**Fig. 3.** Latency vs Loss Rate: STRON and SCTP (with 2 Paths)

| **Simulation** | **Average** | **STD** |
|---|---|---|
| STRON 1p | 3.9268 | 0.4947 |
| STRON 2p | 2.4604 | 0.5886 |
| TCP 1p | 6.0985 | 1.8439 |
| SCTP 2p | 4.0613 | 0.9387 |

**Fig. 4.** The Average and Standard Deviation of the Latencies

In the simulations, two candidate disjoint overlay network paths were established. Figure 2 shows the result of using one path. While the packet loss rate increases, STRON performs better than TCP and SCTP. In Figure 3 two paths were used for the transmission with STRON and SCTP. Although when the packet loss rate is low, SCTP performs a little better than STRON, STRON may take better advantage of the overlay paths as the loss rate increases. We also simulated the behavior of STRON using available bandwidth traces collected from globally distributed computing nodes. The average and standard deviation of the latency can be seen in Figure 4. We can see that STRON is also effective in reducing the variance of network latencies.

## References


1. Fung, W.K., Xi, N., Lo, W.t., Song, B., Sun, Y., Liu, Y.h.: 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)
2. Andersen, D., Balakrishnan, H., Kaashoek, F., Morris, R.: Resilient Overlay Networks. In: 18th ACM Symposium on Operating Systems Principles. (2001)
3. Cen, Z., Mutka, M., Zhu, D., Xi, N.: Supermedia Transport for Teleoperations over Overlay Networks. Technical Report MSU-CSE-05-01, Department of Computer Science and Engineering, Michigan State University (2005)
4. Byers, J.W., Luby, M., Mitzenmacher, M., Rege, A.: A Digital Fountain Approach to Reliable Distribution of Bulk Data. In: SIGCOMM. (1998)
5. Jain, M., Dovrolis, C.: Pathload: A Measurement Tool for End-to-end Available Bandwidth. In: Proc. of Passive and Active Measurement Workshop, Fort Collins, CO (2002)