2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Forward Error Correction for Reliable Teleoperation Systems Based on Haptic Data Digitization

Jae-young Lee, *Student Member, IEEE*, and Shahram Payandeh, *Member, IEEE*

***Abstract*—In this paper, we present a forward error correction method to improve reliability of bilateral teleoperation systems or haptic interfaces over the unreliable network. Based on the digitization of haptic data, the proposed error correction method can be processed within a sampling period, which ensures a real-time process for bilateral teleoperation systems or haptic interfaces. An experimental study is performed using a haptic interface that is interacted with a virtual environment through the communication network. We conduct a psychophysical evaluation to determine the required bit resolution for the haptic data digitization method. Given the psychophysical evaluation result, we present a reliability evaluation of the proposed forward error correction method under the packet loss behavior over the communication network in addition to the noisy environment.**

I. INTRODUCTION

In a network-based *bilateral teleoperation system*, a human operator is allowed to transmit position or velocity data through a communication medium while it receives feedback force data reflected by any contact with surrounding environments of a remote teleoperator. A *haptic interface* also allows a human operator to interact with a virtual object in artificially implemented virtual environments. In these haptic applications based on the sense of touch, *haptic data* – position, velocity, and force data – are required to be transmitted in real-time over an interactive communication medium, such as the Internet, or local network.

When haptic data are transmitted through the network, errors may occur at any time because of unusual network behaviors [1]. *Packet jitter* is one of common behaviors when packets are transmitted through multiple hops with different random queue. *Packet loss* occurs when transmitted data are congested over limited network bandwidth. Furthermore, *packet corruption* and *out-of-ordered packets* are also characterized as unusual network behaviors, which eventually cause erroneous haptic data. Other than such network problems, errors may be caused by natural sources at the remote site, such as thermal or radiation environment, while a teleoperation task is performed [2].

According to the current network protocol technology, such errors can be fixed by retransmitting the erroneous data, so that reliability of a network-based system can be maintained [3]. However, this type of error correction is not generally suited to bilateral teleoperation systems since the retransmission of erroneous data causes relatively large delays and delay variations [4]. When haptic data are fully digitized, which are represented in bit sequences or digital symbols, they can be not only robust against the unusual network behaviors, but also can be processed by digital error correction methods, such as forward error correction (FEC) codes for improved reliability of teleoperation systems [5].

For digitization of haptic data, which is achieved by a quantization process in addition to a sampling process, stiff virtual objects were represented in full digital forms and their stability has been addressed in [6]. A rate-distortion optimized quantization method for haptic force data was also presented [7]. For correcting errors of transmitting haptic data, a transport layer protocol approach to address the packet loss problem has been studied [8]. An overlay network protocol, which uses a virtual network to perform error correction, was also introduced in [9].

In this paper, we present a FEC method for haptic data to improve reliability of network-based bilateral teleoperation systems or haptic interfaces. Based on the full digital forms of haptic data, the proposed FEC method is designed to correct erroneous haptic data caused by the unusual network behaviors while it is performed within a sampling period. The remaining of this paper is organized as following. In Section II, we review haptic data digitization method, which is based on the sampling and quantization processes. In Section III, we propose a FEC method based on digitized haptic position and force data. In Section IV, we present the proposed FEC method that is applied to packet loss recovery. In Section V, an experimental study is performed using a haptic interface that is interacted with a virtual environment through the local network. Finally, conclusions are drawn in Section VI.

II. HAPTIC DATA DIGITIZATION

In digital signal processing, quantization is a process of mapping a large set of continuous signals to a smaller set of discrete signals or bit sequences [10]. A quantizer is a signal processing algorithm or a device that performs the quantization. In typical analog-to-digital conversion, this quantization process needs to be performed in addition to the sampling process. Given the recommended sampling rate of haptic data (1 kHz), the quantization process for haptic position and force data is shown in Figure 1. When a number of bit resolutions n is determined, the range of haptic data can be divided into 2^n intervals. Then, any position or force sample within an interval is mapped to a quantized value that is the middle value of that interval. Each quantized value has

Jae-young Lee and Shahram Payandeh are with the Experimental Robotics Laboratory, the School of Engineering Science, Simon Fraser University, 8888 University Drive, Burnaby, British Columbia, V5A 4Y1 Canada (e-mail: jla155@ sfu.ca, payandeh@sfu.ca).

a corresponding index, which is a binary sequence pre-defined by an index book. The index book consists of binary sequences and corresponding position or force intervals given the number of bit resolutions. When haptic data are transmitted through the communication network as shown in Figure 1, transmitted haptic data from senders are sampled and quantized into binary sequences, rather than floating point haptic data. At receivers, dequantizers are used to recover original haptic data using the index book.

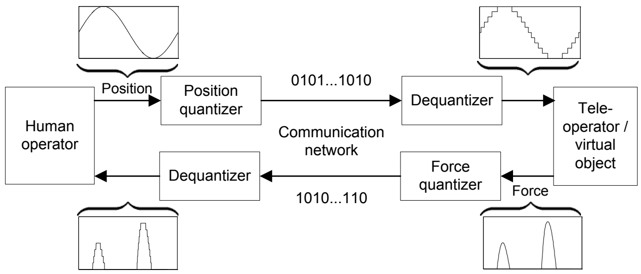


Figure 1. Position data quantization in the forward link and force data quantization in the backward link.

In this quantization process for haptic data, quantization error between the quantized and floating point haptic data is inevitably introduced after received data are recovered by the dequantizers. Hence, when haptic data are digitized, it is important to retain signal fidelity that ensures perceptual transparency of teleoperation systems while eliminating unnecessary part of haptic data. Psychophysically, it is known that a relatively small difference of haptic data may not be noticed by human perception [11]. Therefore, the number of bit resolutions, which is defined by the index book design, needs to be chosen such that the differences between quantized values and corresponding floating point haptic data are unnoticeable by human perception through a psychophysical evaluation. Note that such quantization process can be also used as a rate-distortion optimized quantization that performs a lossy compression by managing the amount of quantization errors and the number of bit resolutions [7]. In this paper, however, we focus on the quantization process simply for digitizing haptic data in order to be processed by the proposed FEC method.

III. Forward Error Correction for Haptic Data

When erroneous data are detected because of the unusual network behaviors, the current network protocol technology provides transport control protocol (TCP) that performs automatic repeat query (ARQ) method to correct the errors. In this ARQ method, a receiver sends back acknowledgement (ACK) to a sender for indication of correct packet reception. If an error or a lost packet is detected, the receiver sends back negative acknowledgement (NACK) to the sender, so that the sender retransmits the erroneous packet [3]. In bilateral teleoperation systems or haptic interfaces, however, since haptic data need to be continuously transmitted in real-time, such ARQ method causes significant delays and delay variations. Therefore, user datagram protocol (UDP) is recommended in general as a transport protocol for teleoperation systems or haptic interfaces [12].

UDP in network-based bilateral teleoperation systems provides faster transmission and smaller delay variations of haptic data, but it does not provide error correction capability as TCP does. In current digital transmission systems, such as digital multimedia and telecommunication systems, FEC codes are also widely used to detect and correct any errors during the network transmission. Unlike the ARQ error correction method, FEC codes do not use the backward link that requests retransmission of erroneous data, but it requires higher bandwidth in the forward link in order to send redundant information for error correction capability. A network-based bilateral teleoperation system or haptic interface using a FEC code is shown in Figure 2. In order to perform the FEC, haptic data need to be converted to digital forms first by using the quantization process presented in Section II. Given digitized haptic data, a FEC encoding process that adds redundancy is performed before transmitting data. Then, using the redundancy, a FEC decoder detects a limited number of errors that may occur during the network transmission, and corrects the errors without performing retransmission.

There are many types of FEC codes currently used for numerous digital applications, and they can be categorized into two groups: the block codes that use fixed size of codewords, and the convolutional codes that additionally use arbitrary length of neighboring information for error correction. In order to apply a FEC code for teleoperation systems or haptic interfaces, processing delay introduced by a FEC code needs to be as minimal as possible. In this paper, we introduce a Reed-Solomon (RS) code, which is one of the widely used block codes for many recent digital applications, such as multimedia systems, compact disk, and hard disk [5].

A. Symbol Mapping

A RS code processes digital symbol data instead of bit sequences. Hence, bit sequences, which are generated from the presented quantization method in Section II, need to be

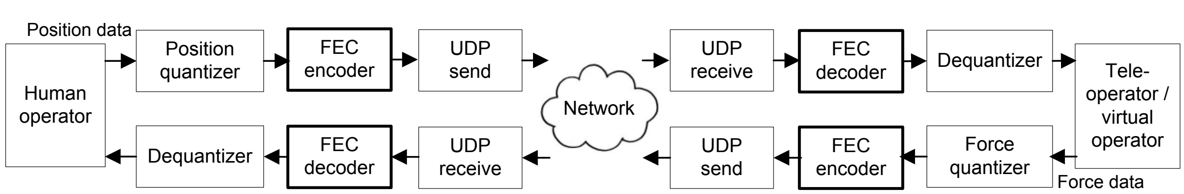
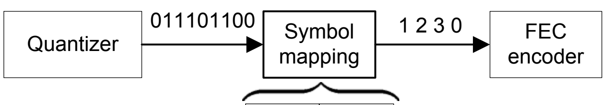


Figure 2. A network-based bilateral teleoperation system using forward error correction and digitized haptic data.

converted to symbol data. A bit-to-symbol mapping scheme before performing the FEC encode is shown in Figure 3. In this example, four symbols are used to be mapped with two bit sequences. The number of bits per symbol can be defined by considering data rate over limited network bandwidth and robustness against the unusual network behaviors. If a higher number of bits is used to map a symbol, higher data rate can be transmitted through the network. Otherwise, if a lower number of bits is used to map a symbol, which generates a smaller set of symbols, the probability of errors can be also lower compared to using the higher number of bits. After symbols are transmitted to a receiver, the received symbols, which are processed by a FEC decoder for error correction, need to be converted back to bit sequences to be dequantized.



Bit	Symbol
00	0
01	1
10	2
11	3

Figure 3. Bit-to-symbol mapping for FEC encoding process. In this example, a symbol is converted by using 2 bits.

B. *Reed-Solomon Codes*

A RS code is a linear systematic block code based on the finite field theory. In coding theory, a systematic code is an error correcting code, in which the encoded output codeword contains the input symbols. The output codeword to be processed by a RS code consists of two parts as shown in Figure 4: the data part that is the first p number of symbols to be protected against any errors, and the parity part that is the following $2t$ number of redundant information. In general, such RS code is represented as a RS(q,p) code, where q is a codeword length and p is a data part length in the codeword. For a symbol that has a length of b bits, the length of q is given by $q = 2^b - 1$. The error correction capability of a RS code depends on the parity part length, which is given by $2t = q - p$, and this implies that a decoder of RS(q,p) code can correct up to t number of erroneous symbols in the codeword with length q.

When the proposed RS code is used as a FEC method for network-based bilateral teleoperation systems or haptic interfaces, a codeword can be obtained from multiple-DoF (m-DoF) haptic data within a sampling period. For example, if 1-DoF position sample is represented by 9-bit sequences and a symbol is mapped with 3 bits, 3-DoF position data in a sampling period require 9 symbols for the data part. In this case, we can choose a RS(15, 9) code for every sampling period, and this codeword can be used to construct a single packet transmitting from a UDP send protocol in real-time. Then, at a receiver, a RS decoder uses the parity part with length 6, and it can correct up to 3 symbols if any errors occur during the network transmission.

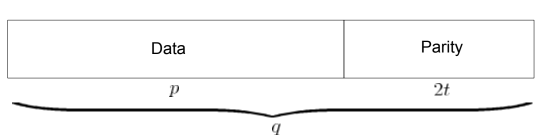


Figure 4. A Reed-Solomon codeword consists of the data part with length p and the parity part with length $2t = q - p$

IV. FORWARD ERROR CORRECTION FOR PACKET LOSS RECOVERY

When the Internet is used to transmit haptic data, some of the unusual network behaviors discussed in this paper may not be appropriate to be processed by the proposed FEC method. For example, when a packet corruption occurs and it is detected in the network layer of the Internet protocol suite, it can be simply dropped without forwarding to the application layer. Therefore, based on UDP as a transport protocol, the packet loss problem, which is due to the network congestion over limited bandwidth or the packet corruption that is intentionally dropped in the network layer, is generally considered as one of the main challenges in the real-world network -based applications.

In order to compensate for the packet loss problem of network-based bilateral teleoperation systems or haptic interfaces, one of methodologies using the proposed FEC method is illustrated in Figure 5. In this methodology, a packet consists of one sample of haptic data from the sender, and it is expressed by three symbol data. In the FEC encode stage of Figure 5, a parity packet is added for every three packets, and these packets are continuously transmitted through the network using a UDP send protocol. Since UDP does not provide any retransmission or error correction scheme, any packet can be lost during the network transmission as shown in the network stage of Figure 5. When the receiver receives the packets by a UDP receive protocol, the FEC decode process is performed in order to detect the three symbols of the lost packet, and to recover them using the parity packet. In this FEC decode process, the length of the codeword is determined by three sequential data packets and one parity packet. Therefore, the receiver is able to detect which packet is missing and also to correct the lost packet depending on the length of the parity packet.

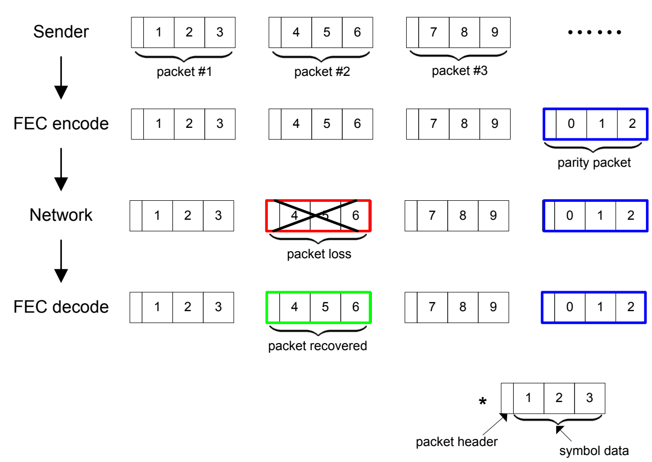


Figure 5. Haptic data loss recovery using FEC encode and decode schemes.

Since the methodology of haptic data loss recovery described in Figure 5 requires a multiple number of packets in order to perform the FEC decode, this process inevitably introduces a constant group delay. In other words, since a codeword consists of three packets, 3 ms group delay needs to be introduced in the case of 1 kHz sampling rate. When 3-DoF haptic data are transmitted through the network, which is more general case to perform a teleoperation task, another methodology to recovery haptic data loss is illustrated in Figure 6. In this methodology, a packet consists of each DoF sample, and one of the 3-DoF samples in every sampling period is assumed to be lost during the network transmission. Since a parity packet is assigned to every sampling period as shown in this figure, the FEC decoder uses this parity packet to recover the lost packet. In this example, since the FEC decode process is performed at every sampling period, it does not introduce a group delay.

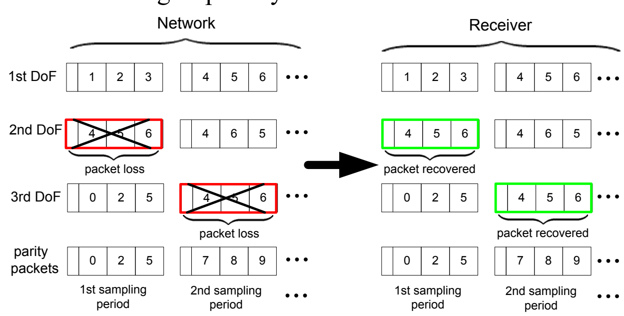


Figure 6. A FEC method for haptic data loss recovery when multiple DoF haptic data are transmitting through the network.

V. Experimental Study

A. Experimental Setup

The experimental setup to evaluate the haptic data digitization and the proposed FEC method for a network-based haptic interface is shown in Figure 7. We used a SensAble PHANToM Omni haptic device as a human operator in PC #1, and a virtual 3D teleoperator, which consists of three rotational and one sliding joints, implemented in PC #2. A local area network (LAN) cable was directly connected between these two Windows-based PCs. From PC #1, 3-DoF position data were generated by manipulating the human operator, and transmitted through the LAN cable in order to remotely operate the virtual teleoperator. In PC #2, when the end-effector of virtual teleoperator was in contact with solid spherical objects on a plane, virtual force data were computed and transmitted back to the human operator through the LAN cable. To bilaterally transmit 3-DoF position and force data, we implemented standard UDP send and receive protocols in the two PCs. The sampling rate of both position and force data was 1 kHz, and each sample was packetized to be transmitted through the UDP protocols. Using the experimental setup in this paper, we conducted the following experimental studies: 1) the psychophysical evaluation of the haptic data digitization method, and 2) the reliability evaluation of the proposed FEC method.

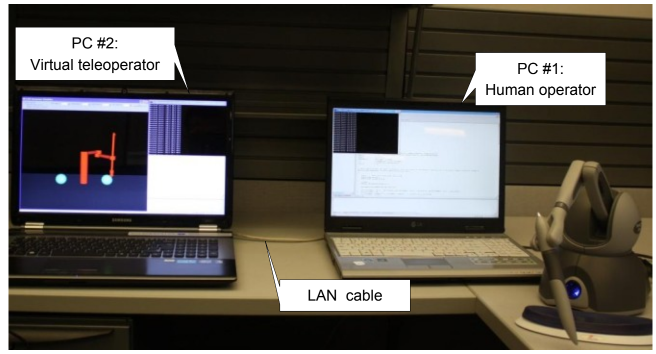


Figure 7. Experimental setup: A PHANToM Omni haptic device is used as a human operator to manipulate a virtual teleoperator. Haptic position and force data are exchanged through a LAN cable.

B. Psychophysical Evaluation of Haptic Data Digitization

For the psychophysical evaluation of haptic data digitization, we implemented uniform quantizers for 1) haptic position data, and 2) haptic force data when the number of bit resolutions was varied. Data ranges of 3-DoF haptic position were (-210, 210 mm), (-105, 145 mm), and (-100, 120 mm), whereas 3-DoF haptic force data had (-3.3, 3.3 N). Based on the experimental setup, two spherical objects were implemented on the virtual workspace of the slave side in order to conduct a *pairwise comparison* that determines any perceptual difference between the original floating point haptic data and the quantized haptic data [13]. The virtual workspace for psychophysical evaluation is shown in the left figure of Figure 8. As shown in this figure, we implemented that floating point haptic data were transmitted if the end effector of virtual teleoperator was in contact with one of the spherical objects whereas quantized haptic data were transmitted if the end effector was in contact with the other spherical object.

12 students (10 males and 2 females) from Simon Fraser University (SFU) participated in this experimental study. While the floating point and quantized haptic data were represented in the two spherical objects and transmitted through the LAN cable, the participants were asked to conduct arbitrary tasks in the virtual workspace, which include contact with the spherical object and touch the surface of the spherical object to feel continuous 3-DoF force data as described in the right figures of Figure 8. They were not told which spherical object was represented in quantized haptic data or floating point haptic data. An evaluation was conducted while bit resolutions of the quantization are randomly changed, and the participants were not given the information about the bit resolutions either. For each evaluation, the participants were allowed to conduct the pairwise comparison for ten seconds. After ten seconds, they had to determine whether the spherical objects are different or not. Based on the answers from the participants, “yes” was considered that the quantization caused artifacts or degraded perceptual transparency compared to the floating point haptic data.

The psychophysical evaluation results of these haptic position and force data quantizations with different bit

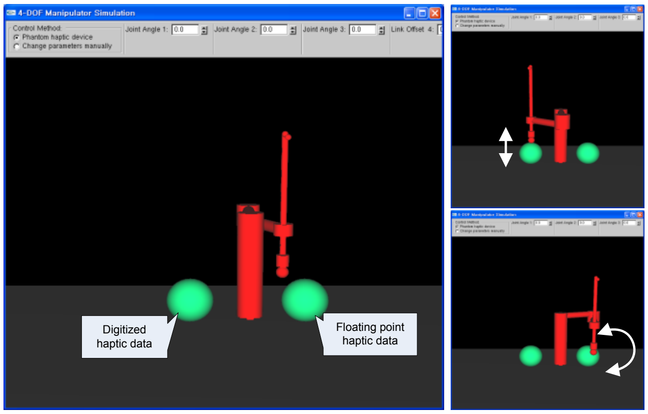


Figure 8. Psychophysical evaluation of haptic data quantization: A pairwise comparison of floating point and quantized haptic data (Left), Psychophysical tasks including contact with the spherical object (Top-right), and touch the surface of the spherical object for generating continuous 3-DoF haptic data (Bottom-right).

resolutions are shown in Table I. For both haptic position and force data quantizations, all 12 participants felt differences between the floating point and quantized haptic data when bit resolutions were less than or equal to 7. In the case of haptic position data quantization, some participants did not detect artifacts from 8 bits, and all the participants did not feel any differences when the bit resolutions are greater than or equal to 11 bits. For the haptic force data quantization, some participants did not detect artifacts from 8 bits, and all the participants did not feel any differences when the bit resolutions are greater than or equal to 10 bits. During this experimental study, it was reported that artifacts were hardly detected when the participants simply made contact with the spherical object as shown in the top-right figure of Figure 8. However, the artifacts and perception degradations were mostly detected when they touched the spherical object and moved the end-effector on the surface, which generated continuous 3-DoF haptic data as shown in the bottom-right figure of Figure 8. According to the psychophysical evaluation results, we used 11 bits and 10 bits for digitizing haptic position and force data, respectively. Therefore, these bit resolutions were used to implement and conduct the performance evaluation of the presented FEC method of haptic data in Section V-C.

TABLE I. PSYCHOPHYSICAL EVALUATION RESULTS OF THE HAPTIC POSITION AND FORCE DATA QUANTIZATIONS. THE NUMBER OF PARTICIPANTS WHO DID NOT DETECT ARTIFACTS WAS REPRESENTED IN PERCENTAGE.

Number of bits	Position data	Force data
6 bits	0 %	0 %
7 bits	0 %	0 %
8 bits	8.3 %	8.3 %
9 bits	33.3 %	58.3 %
10 bits	83.3 %	100 %
11 bits	100 %	100 %
12 bits	100 %	100 %

C. Performance Evaluation of Haptic Data Error Correction

In this experimental study, we performed the reliability evaluation of the proposed FEC method using the experimental setup in Section V-A. Given 11 bit and 10 bit resolutions for fully digitized haptic position and force data, respectively, we implemented the FEC codes for haptic position and force data and demonstrated them under 1) the additive noise, and 2) the packet loss behavior.

For the first experiment in this section, we demonstrated the proposed FEC codes under the additive noise environment, which was modeled as additive white Gaussian noise (AWGN). Since such additive noise is commonly used for gaining insight into the underlying performance of the proposed method before considering more realistic network behaviors, we used it only to generate erroneous haptic data and to evaluate the error correction performance of the proposed FEC codes. The additive noise was emulated when 1-DoF position and force data were transmitted through the LAN cable, and such emulation to 1-DoF haptic data was enough for the participants to feel disturbances while they performed the virtual teleoperation tasks as shown in Figure 8. In order to correct any errors caused by the additive noise, we implemented the RS(15,7) code for haptic position data in the forward link and the RS(7,5) code for haptic force data in the backward link. Before the FEC encoding, a bit-to-symbol mapping scheme was used with 2 bits per symbol rate. Since we used 10 bits for the force data digitization, the data part of codeword was simply $p = 5$. However, for the haptic position data digitization, we had to add 3 zero bits to generate a codeword with $p = 7$, which fully expresses 11 bits of position data.

The error correction performances of the proposed FEC codes for haptic position and force data are shown in Figure 9 and Figure 10, respectively. Figure 9 (a) and 10 (a) show that floating point position and force data were corrupted by the additive noise with signal-to-noise ratio (SNR) of 30 dB and 25 dB, respectively. Note that the amount of additive noise is determined by SNR denoted in dB, and a higher SNR represents a lower amount of additive noise. When these haptic data are digitized and transmitted, the receivers find digital symbols instead of floating point data. Hence, digitized haptic data can be more robust against the additive noise with relatively higher SNRs. However, in relatively lower SNRs, the receivers can easily make incorrect decisions, which result in erroneous symbols. Haptic position and force data that were digitized and recovered by such incorrect decisions are shown in Figure 9 (b) and 10 (b), respectively. Finally, the errors shown in these figures were corrected by using the proposed FEC codes, and the error correction results of haptic position and force data are shown in Figure 9 (c) and 10 (c), respectively. As shown in these figures, the proposed FEC method corrected all the errors given the level of additive noise.

In this experiment, we also evaluated the FEC codes for haptic position and force data, which were RS(15,7) and RS(7,5), respectively, under different levels of additive noise.

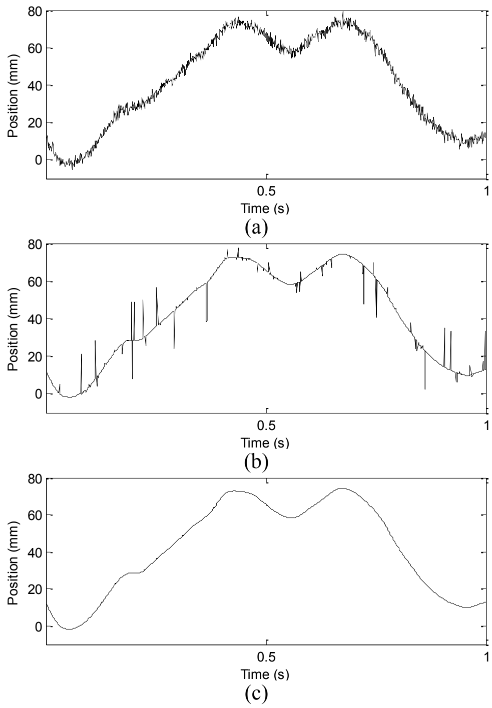


Figure 9. Forward error correction for position data: (a) Analog position data corrupted by additive noise, (b) Digitized position data received at the teleoperator without error correction, (c) Error corrected position data at the teleoperator

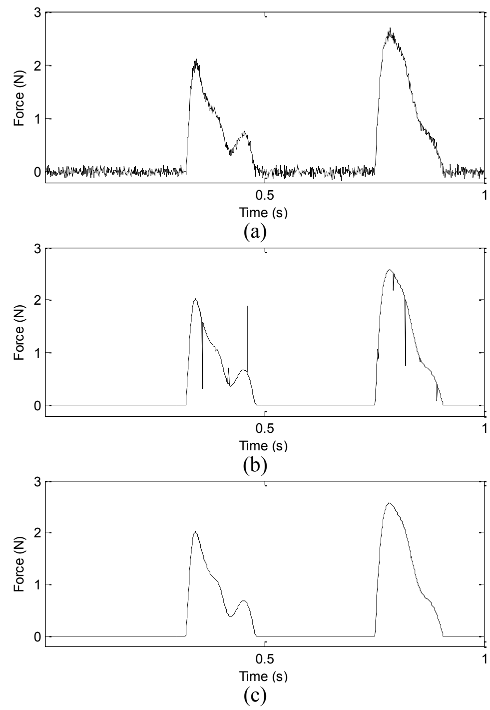


Figure 10. Forward error correction for force data: (a) Analog force data corrupted by additive noise, (b) Digitized force data received at the human operator without error correction, (c) Error corrected force data at the human operator

When the noise levels were varied, we measured the number of bit errors between the decoded haptic data at the receivers and the original haptic data at the senders for ten seconds operation. The bit error rates (BER) of the FEC codes for a ten-second interval are shown in Figure 11. As shown in this figure, errors of haptic position and force data were found up to SNRs of 24 dB and 29 dB, respectively. Note that such BER performance depended on the choice of FEC codes, bit-to-symbol mapping schemes, and types of haptic data. As shown in Figure 11, in the case of haptic force data, the number of errors was relatively small at lower SNRs because the non-contact forces could be easily decoded by making decisions to zeroes.

For the second experiment in this section, we emulated the packet loss behavior during the network transmission, and demonstrated the performance of the proposed FEC codes. Transmitted packets through the LAN cable were randomly selected to be dropped for 1-DoF haptic position and force data, and such emulation was enough for the participants to feel disturbances while they performed the virtual teleoperation tasks as shown in Figure 8. Examples of emulated packet loss behavior for 1-DoF haptic position and force data are shown in Figure 12 (a) and Figure 13 (a), respectively. Note that in these specific examples, the loss rates of position and force data were 3 % and 5 %, respectively.

In order to recover the packet losses by using the proposed FEC codes, we implemented one parity packet per every three data packets for the both forward and backward links as illustrated in Figure 5. For haptic position and force data, a bit-to-symbol mapping scheme was used with 4 bits per symbol rate. We also added 12 redundant bits for each sample of haptic position and force data, so that every packet had a fixed size of 3 symbols. After the bit-to-symbol mapping, we implemented the RS(15,9) codes for both haptic position and force data, therefore, the data part of codeword was $p = 9$ and the proposed FEC codes were able to recover errors up to 3 symbols.

The haptic position and force data after packet loss recovery using the proposed FEC codes are shown in Figure 12 (b) and Figure 13 (b), respectively. In both cases, we emulated that one of three data packets was randomly dropped during the transmission, but we also assumed that parity packets, which need to be transmitted after every three

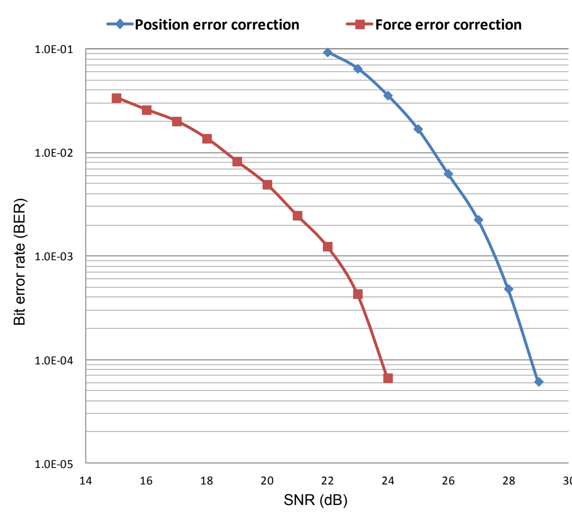


Figure 11. Bit error rates of position and force data error corrections.

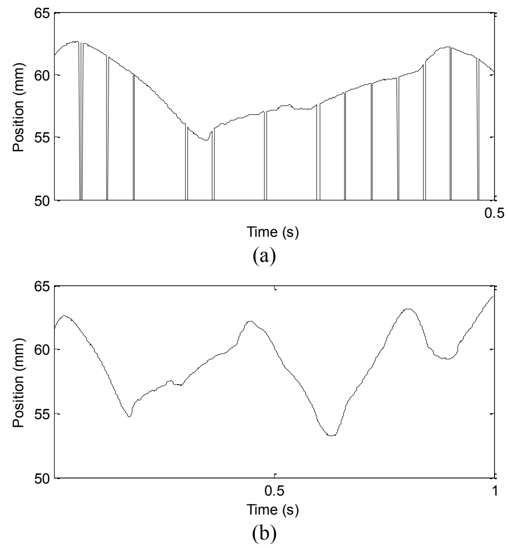


Figure 12. Forward error correction in the case of packet losses: (a) Haptic position data with packet loss, and (b) Recovered position data by using the FEC method.

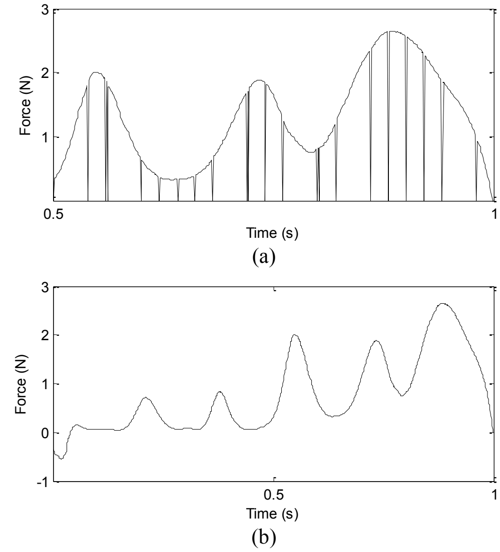


Figure 13. Forward error correction in the case of packet losses: (a) Haptic force data with packet loss, and (b) Recovered force data by using the FEC method.

packets for the FEC decoding process, were not dropped. Hence, the packet loss rates in the both forward and backward links were 25 %. Since we used the RS(15,9) codes, which could correct errors up to $t = (15-9)/2 = 3$ symbols, one dropped packet that consists of 3 symbols could be fully recovered. In this experimental scenario, since the FEC decoding process was performed after every three data packets were received, a constant group delay, which is 3 ms, was inevitably introduced, but such small constant delay did not cause any problem on transparency and stability of virtual teleoperation tasks.

VI. Conclusion

In this paper, the FEC method for haptic position and force data was studied under the unusual network behaviors. In the case of packet loss, which is considered as one of the main challenges in the current network technology, the methodology performed in the experimental study (Figure 5) introduced a relatively small group delay, which could be negligible. However, when we choose a FEC method with relatively larger length of codeword, more errors can be corrected, but a larger group delay may cause instability of overall teleoperation systems. In order to prevent such instability problem, the passivity based control methods may be additionally adopted [14].

In this paper, we introduced the RS code as a FEC method to compensate for erroneous haptic position and force data. As the RS code is widely used for other multimedia and telecommunication systems, it is also used as a concatenated form with other types of error correcting codes, such as a convolution code. For further reliability improvement of transmitting haptic data, such concatenated code will be considered as a future remark. The error correcting code can be also combined with an estimation or prediction method, which is based on a probabilistic approach for further performance improvement [15].

References

[1] V. Paxson, "End-to-end Internet packet dynamics," *IEEE/ACM Trans. on Networking*, vol. 7, no. 3, pp. 277–292, June 1999.

[2] M. Sitti and H. Hashimoto, "Tele-nanorobotics using atomic force microscope," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Victoria, BC, Canada, Oct. 1998, pp. 1739–1746.

[3] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, Boston, MA: Pearson Education, 2008.

[4] Y. Uchimura, T. Yakoh, and K. Ohnishi, "Bilateral robot system on the real-time network structure," *IEEE Trans. on Industrial Electronics*, vol. 51, no. 5, pp. 940–946, Oct. 2004.

[5] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, Englewood Cliffs, NJ: Prentice Hall, 1983.

[6] N. Diolaiti, G. Niemeyer, F. Barbagli, and J. K. Salisbury, "Stability of haptic rendering: discretization, quatization, time delay, and coulomb effects," *IEEE Trans. on Robotics*, vol. 22, no. 2, pp. 256–268. April 2006.

[7] J.-y. Lee and S. Payandeh, "Signal processing techniques for haptic data compression in teleoperation systems," in *Proc. IEEE Haptics Symposium*, Vancouver, BC, Canada, March 2012, pp 371–376.

[8] S. Lee and J. W. Kim, "Priority-based haptic event filtering for transmission and error control in networked virtual environments," *Multimedia Systems*, vol. 15, no. 6, pp. 355–367, 2009.

[9] Z. Cen, M. W. Mutka, D. Zhu, and N. Xi, "Supermedia transport for teleoperations over overlay networks," in *Proc. Networking*, Waterloo, ON, Canada, 2005, pp. 1409–1412.

[10] Y. Wang, J. Ostermann, and Y-Q. Zhang, *Video Processing and Communications*, Upper Saddle River, NJ: Prentice Hall, 2002.

[11] W. M. B. Tiest and A. M. L. Kappers, "Cues for haptic perception of compliance," *IEEE Trans. Haptics*, vol. 2, no. 4, pp. 189 –199, Oct. 2009.

[12] E. J. Rodríguez-Seda, D. J. Lee, and M. W. Spong, "Experimental comparison study of control architectures for bilateral teleoperators," *IEEE Trans. on Robotics*, vol. 25, no. 6, pp. 1304-1318, Dec. 2009.

[13] S. Payandeh, J. Dill, and J. Zhang, "A study of level-of-detail in haptic rendering," *ACM Trans. Applied Perception*, vol. 2, no. 1, pp. 15–34 Jan. 2005.

[14] G. Niemeyer and J. J. E. Slotine, "Stable adaptive teleoperation," *IEEE Journal of Oceanic Engineering*, vol. 16, no. 1, pp. 152–162, Jan. 1991.

[15] J.-y. Lee and S. Payandeh, , "Stability of Internet-based teleoperation systems using Bayesian predictions," in *Proc. IEEE World Haptics,* Istanbul, Turkey, June 2011, pp. 499–503.