Prediction Techniques for Haptic Communication and their Vulnerability to Packet Losses

Fernanda Brandi and Eckehard Steinbach
Institute for Media Technology
Technische Universität München
Munich, Germany
{fernanda.brandi, eckehard.steinbach}@tum.de

Abstract—We introduce three different uses of linear regression for improving the prediction of samples in haptic communication and compare this technique to the commonly employed zero-order and first-order linear predictors. We couple the prediction techniques with (error resilient) perceptual data reduction approaches and evaluate their robustness when losses in the network are present. Experimental results show that the proposed prediction technique improves haptic data reduction while keeping lower signal distortion compared to the traditional prediction methods when facing adverse network conditions.

I. HAPTIC DATA REDUCTION AND COMMUNICATION

In haptic interactions it is common that the available number of degrees-of-freedom (DoF) is either 3 or 6 (e.g. 3-DoF position/force, 3-DoF rotation/torque). Considering that each DoF can be represented by a single or double floating point (which use, respectively, 32 and 64 bits) we can deduce that the amount of data contained in each haptic packet is not as large as in other media formats such as image and video. However, the acquisition and transmission rates of haptic data are significantly greater than in other media as well. This rate is typically set to 1 kHz to follow the human somatosensory system perception capabilities [1] and to maintain the stability and transparency of the system [2], and thus can potentially become a burden in the communication channel as up to 1000 packets per second need to be injected into the network.

Instead of transmitting all the acquired haptic samples, an alternative is to send only a subset of the samples while the missing data can be estimated using prediction schemes. One efficient approach for haptic packet rate reduction [3] is based on psychophysical findings that show that the human haptic sensory system is limited and relative which results in perceiving two slightly different stimuli as the same one. The mathematical presentation of such limitation is the Weber's Law of Just Noticeable Differences (JND). \[ \frac{\Delta x}{x} = k \quad \text{or} \quad \Delta x = k \cdot x \] (1)

where \( x \) represents a reference stimulus, \( \Delta x \) represents the maximum increment or decrement in stimulus \( x \) that is still considered an imperceivable change (i.e. the JND) and \( k \) is a constant called the Weber parameter. The interval \( [x - \Delta x, x + \Delta x] \) is often referred as the perceptual deadband. Essentially, if users are asked to differ the reference stimulus \( x \) from any other stimulus contained in the respective deadband thresholds defined by \( x \) and the constant \( k \), they are unable to do so and would just consider both as the same stimulus. This fact can be exploited to selectively skip packet transmissions whenever similar enough stimuli can be estimated and displayed instead.

In the perceptual deadband-based haptic data reduction scheme in [3], Hinterseer et al. decide not to transmit an acquired haptic sample, \( x_i \), whenever this sample and its respective predicted sample, \( p_i \), are perceptually indistinguishable from each other. When \( x_i \) and \( p_i \) are perceived as two different stimuli, that is, when \( x_i \notin [p_i - \Delta p_i, p_i + \Delta p_i] \), the acquired sample \( x_i \) has to be transmitted and it is referred to as an update sample, \( u_j \).

This concept was also extended to the three-dimensional (3D) space which just means that all reference samples are, in this case, 3D vectors, \( \vec{x}_i \), and the deadband regions (called deadzone in the multidimensional case) are shaped as spheres with their center in the predicted samples, \( \vec{p}_i \). In case \( \vec{x}_i \) is contained in the deadzone sphere, the prediction \( \vec{p}_i \) can be used instead (i.e. \( \vec{x}_i \) is not transmitted), otherwise, a so-called violation of the deadzone occurs and an update, \( \vec{u}_j \), is transmitted.

In order to determine the predicted samples \( \vec{p}_i \), two well-known prediction schemes are typically employed, namely, zero-order linear predictor (ZOLP) and first-order linear predictor (FOLP). A one-dimensional example of such predictors is shown in Figure 1 where the ZOLP and FOLP can be observed, respectively, in the top and bottom rows. The images...
on the left show the sender side where acquired samples (solid black/blue circles) are tested against the perceptual deadband thresholds (gray areas) defined by the prediction (dashed gray lines). It can be noted that the prediction values are used until a deadband violation occurs and an update sample (solid blue circles) must be sent. The images on the right show the signal reconstruction on the receiver side where updates (solid blue circles) are received and the respective predicted samples (dashed blue circles) are determined in the same fashion as in the sender.

1) Zero-Order Linear Predictor: This approach just replicates the last transmitted/received sample. That means that whenever there is a deadzone violation, a new update, \( \bar{u}_j \), is transmitted and the following predicted samples, \( \bar{p}_i \), will just display the same value as this last update, as seen in Eq. (2).

\[
\bar{p}_i = \bar{u}_j \tag{2}
\]

2) First-Order Linear Predictor: This approach employs the last two update samples \( \bar{u}_j \) and \( \bar{u}_{j-1} \) and their respective timestamps, \( t_j \) and \( t_{j-1} \), to define the prediction’s gradient, \( \bar{g}_j \), as seen below.

\[
\bar{g}_j = \frac{\bar{u}_j - \bar{u}_{j-1}}{t_j - t_{j-1}} \tag{3}
\]

The prediction \( \bar{p}_i \) can then be determined employing the gradient, \( \bar{g}_j \), the latest update sample, \( \bar{u}_j \), its respective timestamp, \( t_j \), and the timestamp of the prediction, \( t_i \). This relation can be observed in Eq. (4).

\[
\bar{p}_i = \bar{g}_j \cdot (t_i - t_j) + \bar{u}_j \tag{4}
\]

There are two ways of utilizing this predictor: (i) only \( \bar{u}_j \) and \( t_j \) are transmitted whenever there is a deadzone violation and \( \bar{g}_j \) is calculated at the receiver (note that \( \bar{u}_{j-1} \) and \( t_{j-1} \) should be stored at the receiver as well); (ii) the gradient \( \bar{g}_j \) is calculated at the sender and also transmitted along with \( \bar{u}_j \) and \( t_j \). In lossless communication channels, the case (i) can be straightforwardly applied, however, since this is not always the scenario encountered (i.e. error-prone networks can also be employed), this prediction system becomes more sensitive to error propagation. Conversely, the case (ii) encapsulates one more piece of data (the gradient \( \bar{g}_j \)) in the packet which significantly decreases the dependency on the successful transmission of packets. In this way, even if previous packets are lost, whenever a packet is received, the correct prediction can be immediately calculated and error propagation is stopped. From now on in this work, we always consider the case (ii) whenever FOLP is mentioned.

A. Signal Distortion Metric

Since there is not yet a standard objective distortion metric for haptic communication, a common way of evaluating haptic signals is to perform extensive psychophysical tests with human users. These subjective tests are often tiresome and need to be carefully designed to avoid bias and masking effects due to long test runs and inappropriate experimental setup. Since in this work we have a few varying parameters and extensive range of test values, a subjective test would be impractical. Hence, we employ a simple objective metric introduced in [4] named Weighted Mean Squared Error (WMSE) that can be mathematically described as:

\[
WMSE = \frac{1}{N} \sum_{i=0}^{N-1} \frac{||\bar{x}_i - \hat{x}_i||^2}{||\bar{x}_i||} \tag{5}
\]

where N is the total number of samples, \( \bar{x}_i \) is the acquired signal vector and \( \hat{x}_i \) is the reconstructed signal vector (i.e. the update and predicted samples). It can be noted that the WMSE is an adaptation of the known Mean Squared Error (MSE). The additional term \( ||\bar{x}_i|| \) dividing the squared error is included in the WMSE expression to take into account the human perception relativisation mentioned earlier in this work. This term modifies the total inside the summation in a manner that for a considered error, \( ||\bar{x}_i - \hat{x}_i|| \), the perception of this distortion is inversely proportional to the intensity of the stimulus \( \bar{x}_i \). In other words, in case \( ||\bar{x}_i|| \) is small the given deviation is "amplified" and if \( ||\bar{x}_i|| \) is large the deviation is "reduced". Note that considering the nature of the proposed experiments in this work, this metric is chosen to serve as a comparison guideline between several signal reconstructions and does not intend on replacing completely subjective experiments in the future.

B. Packet Loss in Haptic Communication

Haptic communication often occurs over the internet, which means that the packet transmission is susceptible to delay, jitter and losses. Since the haptic data reduction scheme relies on sample prediction, whenever packets are lost (or severely delayed) the receiver continues on estimating missing samples based on "old" information. This causes the receiver to display an incorrect signal which naturally differs from what was determined at the sender.

To decrease the signal deviation due to packet losses and to mitigate perceivable artifacts [5], [6], it is proposed in [5], [7] that the sender should build a binary tree that estimates all possible cases of successful and unsuccessful packet transmissions and their respective predictions and occurrence probabilities. In this way, the sender can decide on-the-fly if additional packets need to be triggered to improve the chances of synchronizing the receiver with the sender once again (i.e. receiver predicting the signal correctly). Brandi et al. test several trigger methods in [5] and show that two of them are the most effective ones to alleviate packet loss induced artifacts (while still maintaining strong data reduction). One of these triggers is based on the expected deviation between the predicted samples of all latest branches of the binary tree and the current acquired sample. The other trigger takes into account the sum of probabilities of all the latest branches that would significantly deviate from the acquired sample. In both cases, the expected deviation and the sum of probabilities are compared to experimental thresholds and if these values are not compliant, new update samples are set to be transmitted.
Although the performance of both approaches [5], [7] is comparable, there is a major advantage in [7] since it is considerably less complex (e.g. the binary tree grows linearly instead of exponentially as in [5]). Moreover, the authors also provide analytical expressions for predictions and occurrence probabilities for each branch which makes the scheme in [7] more straightforward and is the one employed in this work.

II. HAPTIC SIGNAL FILTERING

Considering that both the force and position signals are typically acquired by sensors embedded in haptic devices and robots, it is natural to assume that these signals can be noisy. Moreover, in many scenarios the velocity signal is obtained by differentiating the position signal which amplifies the position signal noise. Since the sample prediction approaches presented in Section I, especially the FOLP, are particularly sensitive to noise in the input signal, it is proposed in [8] to low-pass filter the signal before prediction using a Kalman filter. Although the use of this filter achieves good results, there are some parameters that need to be estimated and adjusted along the filtering which makes its implementation not so trivial. As an alternative to the Kalman filter, we propose in this work the use of a Moving Average (MA) [9] which is a time domain filter that averages a set of samples within a rolling time window of size $M$. The MA can be mathematically described as:

$$\tilde{x}_i = \frac{1}{M} \sum_{k=0}^{M-1} \hat{x}_{i-k}$$

(6)

In this work we employ the MA to denoise the input haptic signals. Many window sizes are tested to select the most adequate $M$ for our experiments. Figure 2 shows the test results for both force and velocity signals and the chosen values are, respectively, $M = 3$ and $M = 10$. It is taken into account that minimum distortion should be inserted in the signal while still maintaining low packet rates (i.e. closer to the bottom left corner in the two plots of Figure 2). An example of the resulting filtering can be seen in Figure 3. As a matter of simplicity, from now on all input signals utilized in this work are denoised, that is, whenever we refer to the signal $\tilde{x}$, the reader should consider it as being the filtered signal $\hat{x}$.

III. SAMPLE PREDICTION METHODS

As shown in [3] and explained in Section I, to achieve strong data rate reduction in haptic communication while providing good immersiveness to the users, prediction methods can be employed coupled with the perceptual deadband-based data reduction approach. When describing the FOLP in Section I-2, we show that the two latest update samples $\hat{u}_j$ and $\hat{u}_{j-1}$ define the gradient $\tilde{g}_j$ which indicates the direction and steepness of the future predicted samples. Considering there are times when $\hat{u}_{j-1}$ is temporally distant from $\hat{u}_j$ (i.e. $\hat{u}_j \gg \hat{u}_{j-1}$) or that the update samples are particularly noisy, it is natural to assume that the gradient may not accurately reflect the most current slope of the signal and, in consequence, contributes to inefficient prediction calculation. There are two main issues that should be considered when predictions are not accurate: (i) it leads to estimates that tend to violate the deadzone sooner (since they do not follow the curve trend) consequently increasing the transmission rate; and (ii) when haptic packets are lost these predictions tend to overshoot which potentially causes more severe artifacts [5], [6].

As pointed out in Section I-2, since $\tilde{g}_j$ can be also sent along with $\hat{u}_j$ and $\hat{t}_j$, the gradient can be determined in a different fashion than what was shown in Eq. (3). Inspired by this observation, we propose to employ a linear regression technique [10] in order to determine the trend of the signal and employ the calculated slope as the gradient $\tilde{g}_j$.

Regressions are statistical approaches that try to model a set of points in space into a polynomial curve. This approach
is particularly interesting since it can be utilized to interpret signal trends and also extrapolate data. There are several types of regression and in this work we use a simple linear regression of a line model which approximates a set of points into a straight line. Essentially, the dependent variable $X$ (intensity) is modeled based on the independent variable $T$ (time) and a set of parameters $\beta$ such as in Eq. (7).

$$X \approx f(T, \beta)$$ \hfill (7)

For the linear regression of a straight line, the model becomes the following approximation of $x_i$.

$$x_i \approx \hat{x}_i = \hat{\beta}_0 + \hat{\beta}_1 t_i$$ \hfill (8)

The parameters $\hat{\beta}_0$ (i.e. the offset of the line) and $\hat{\beta}_1$ (i.e. the slope of the line) are calculated as follows.

$$\hat{\beta}_0 = \bar{x} - \hat{\beta}_1 \bar{t}$$ \hfill (9)

$$\hat{\beta}_1 = \frac{\sum_{r=0}^{R} (t_{i-r} - \bar{t})(x_{i-r} - \bar{x})}{\sum_{r=0}^{R} (t_{i-r} - \bar{t})^2}$$ \hfill (10)

In the equations above, $\bar{t}$ and $\bar{x}$ are the respective mean of $t$ and $x$ in a given interval of $R+1$ samples in the regression window such as shown below.

$$\bar{t} = \frac{1}{R+1} \sum_{r=0}^{R} t_{i-r} - k \quad \bar{x} = \frac{1}{R+1} \sum_{r=0}^{R} x_{i-r} - k$$ \hfill (11)

An experimental test is performed to define the most appropriate regression window size for the force and velocity signals. Figure 4 shows the results contrasting once again the signal distortion and the packet rate. The chosen $R$ for the force and velocity signals are, respectively, $R = 2$ and $R = 10$. We propose three different uses of the regression data and detail them in the following subsections.

A. Linear Regression Applied to Sample Gradient

The gradient $\vec{g}_j$ is no longer calculated as described in Eq. (3), instead, the slope of the straight line model $(\hat{\beta}_1)$ is utilized as the gradient, which means:

$$\vec{g}_j = \hat{\beta}_1$$ \hfill (12)

Note that we can extend the calculation of the regression to a 3D space to analogously determine the three coordinates of the parameters so they can be utilized as the gradient.

B. Linear Regression Applied to Sample Gradient and Update

Additionally to Eq. (12) which uses the parameter $\hat{\beta}_1$ as the gradient $\vec{g}_j$, we propose to also change the update sample replacing it by the regression estimate, $\vec{x}_i$, for that moment in time, $t_j$. In this case, instead of transmitting $\vec{u}_{ij} (= \vec{x}_i)$, $t_j$ and $\vec{g}_j$ (as in Eq. (3)) as originally described in Section I-2, the sender encloses $\vec{u}_{ij}$ (as in Eq. (13)), $t_j$ and $\vec{g}_j$ (as in Eq. (12)) in the haptic packet.

$$\vec{u}_{ij} = \vec{\beta}_0 + \vec{\beta}_1 t_j$$ \hfill (13)

Theoretically, this approach works also as a low-pass filter which makes its use particularly interesting for unfiltered/noisy signals. Even though the haptic signals in this work are filtered, this approach is tested to check the impact on the performance when replacing updates by the estimates $\vec{u}_{ij}$.

C. Linear Regression Adaptively Applied to Sample Gradient and Update

This third approach is a hybrid between the approaches described in the two previous subsections. The core idea here is to employ the method in Section III-B except when the new update $\vec{u}_{ij}$ differs significantly from the original update $\vec{u}_{ij}$. Whenever this happens, the method in Section III-A prevails, which means that the gradient $\vec{g}_j$ is calculated as in Eq. (12) while the update sample assumes again the value of $\vec{x}_i$. This decision process can be observed in Eq. (14).

$$\vec{u}_{ij} = \begin{cases} \vec{\beta}_0 + \vec{\beta}_1 t_j & \text{if } \|((\vec{\beta}_0 + \vec{\beta}_1 t_j) - \vec{x}_i)\| \leq \|\vec{x}_i\| \cdot k \\ \vec{g}_j & \text{otherwise} \end{cases}$$ \hfill (14)

To decide if the two possible updates are too dissimilar it is necessary to test if the difference between $\vec{u}_{ij} (= \vec{\beta}_0 + \vec{\beta}_1 t_j)$ and $\vec{u}_{ij} (= \vec{x}_i)$ violates the deadzone thresholds defined by the original update ($= \vec{x}_i$). Basically, if a violation does not occur, the method in Section III-B is utilized, otherwise, the method in Section III-A takes over.

This third approach tries to avoid a delay in displaying sudden changes in the signal (such as typically observed in force signals when contact happens or ceases to exist) as it could occur in the approach in Section III-B.

IV. EXPERIMENTAL SETUP AND PROCEDURE

We employ a SensAble PHANTOM Omni haptic interface coupled with a virtual environment (VE) based on the
The VE presents a static torus that can be freely touched by a spherical end-effector controlled by the user. We run three different 60-second haptic sessions and record all the transmitted haptic data flowing between the operator and teleoperator (i.e., force and position/velocity signals). This data is then post-processed applying the moving average filter described in Section II with $M = 3$ for the force signal and $M = 10$ for the velocity signal.

The two prediction approaches, namely the ZOLP and the FOLP, described in Section I are tested against the three uses of linear regression introduced in Section III with $R = 2$ and $R = 10$ for the force and the velocity signals, respectively. Furthermore, we implement both the deadband-based data reduction approach [3] and the low-complexity error-resilient data reduction approach [7] with the expected deviation and sum of probabilities triggers. In all cases the Weber parameter is set to 10% for the force and 15% for the velocity signal. Lastly, we include a round trip delay of 40 milliseconds and we test different packet loss probabilities to evaluate the robustness of each scheme. The packet loss probability is set to the following values [0%, 2%, 4%, 6%, 8%, 10%, 15%, 20%, 25%, 30%, 40%, 50%]. All tests are repeated 50 times each so the mean and the standard deviation can be properly determined.

V. Experimental Results

The experimental results for the force and the velocity signals can be respectfully observed in Figure 5 and Figure 6. We compare the performance of the five prediction methods (abbreviated as zero-order, first-order, regression v1, regression v2 and regression v3 in the plots and depict them with different line colors) while using the three aforementioned data reduction schemes (abbreviated as none, exp dev and sum prob in the three distinct plots of each figure) with changing packet loss probabilities. We evaluate the results based on both the achieved data reduction and the reconstructed signal distortion.

It can be observed in Figures 5(a) and 6(a) that for both the force and velocity signals, the signal distortion grows significantly with increasing packet loss rate while the data rate remains unchanged. This is expected since the perceptual deadband approach [3] does not address the loss of packets and, consequently, does not try to compensate for the missing data. However, it is interesting to note that the packet rate reduction is further improved with the use of the regression technique for the force (8% reduction for regression v1 in comparison to FOLP, 7% for regression v2 and 2% for regression v3) and also for the velocity signal (4% reduction for regression v1 in comparison to FOLP). Moreover, the signal distortion is significantly decreased (38% in average) when regression v3 is employed in the force signal while a slight decrease (1% in average) is also noted for regression v1 in the velocity signal.

When the error-resilient data reduction approach [7] is implemented using the expected deviation trigger in the force signal as seen in Figure 5(b), the distortion drops considerably (refer also to Figure 5(a)) while still preserving a robust data reduction. It can also be observed that the three uses of the regression technique achieve lower packet rates than the FOLP (respectively, 7%, 6% and 3% reduction in average) while the distortion is also smaller (1%, 2% and 17% reduction in average). Regarding the velocity signal results in Figure 6(b) it can be noted that regression v1 performs better than the FOLP both in terms of data reduction and distortion (respectively, 3% and 2% reduction in average).

Lastly, when the approach [7] is applied using the sum of probabilities trigger (refer to Figure 5(c)) it can be observed that both the distortion and the data reduction of all predictors perform worse than the previous case where the expected deviation is employed. Nevertheless, the data reduction performance of the regression is still better than the FOLP (5%, 4% and 1% reduction in average) while regression v3 also performs significantly better in terms of distortion (35% reduction in average). Regarding the velocity signal in Figure 6(c), although the data rate tends to increase compared to the previous cases, the distortion is the lowest of all cases. Moreover, comparing the regression cases to the FOLP it is encouraging to note that the data rates are significantly smaller for the regression (16%, 8% and 8% reduction in average) while the distortion of regression v1 is slightly smaller as well (1% reduction in average).

VI. Conclusions

In this work we show that the use of a moving average filter is a viable option to denoise haptic signals in the time domain. We also propose the use of a linear regression technique to improve the prediction of haptic samples. We describe and evaluate three different uses for the linear regression data and compare their performances to the zero-order and first-order linear predictors. Our experimental results show that using the regression technique (especially regression v3 for the force and regression v1 for the velocity) can achieve improved data reduction while decreasing the overall signal distortion when coupling it to the deadband-based data reduction approach [3] and the low-complexity error-resilient data reduction scheme [7] regardless of the network conditions.

VII. Acknowledgment

This work has been supported by the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007-2013) / ERC Grant agreement no. 258941.

References

Fig. 5. Experimental results (mean and standard deviation) for the force signal for varying packet loss probability. Each plot represents the use of one of the three data reduction approaches, namely (a) deadband-based data reduction, (b) low-complexity error-resilient data reduction using the expected deviation trigger and (c) using the sum of probabilities trigger. The packet loss probability increases from the left to the right and from the bottom to the top.

Fig. 6. Experimental results (mean and standard deviation) for the velocity signal for varying packet loss probability. Each plot represents the use of one of the three data reduction approaches, namely (a) deadband-based data reduction, (b) low-complexity error-resilient data reduction using the expected deviation trigger and (c) using the sum of probabilities trigger. The packet loss probability increases from the left to the right and from the bottom to the top.


