Proceedings of the 2007 IEEE/RSJ International
Conference on Intelligent Robots and Systems
San Diego, CA, USA, Oct 29 - Nov 2, 2007



# Improved Prediction of Heart Motion Using an Adaptive Filter for Robot Assisted Beating Heart Surgery

Timothy J. Franke, Ozkan Bebek, M. Cenk Cavusoglu
Department of Electrical Engineering and Computer Science
Case Western Reserve University, Cleveland, Ohio, 44106-7221, USA
E-mail: tjf6@case.edu, ozkan@case.edu, cavusoglu@case.edu

*Abstract*— **Robot assisted heart surgery allows surgeons to operate on a heart while it is still beating as if it had been stopped. The robot actively cancels heart motion by closely following a point of interest (POI) on the heart surface—a process called Active Relative Motion Canceling (ARMC). Due to the high bandwidth of the POI motion, it is necessary to supply the controller with an estimate of the immediate future of the POI over a prediction horizon. In this paper, a prediction algorithm, using an adaptive filter to generate future position estimates, is implemented and studied. The effects of predictor parameters on tracking performance are studied. Finally, the predictor is evaluated using a 3 degrees of freedom test-bed and prerecorded heart motion data.**

## I. Introduction

Beating heart surgery cannot be effectively performed by hand due to the relatively quick motion of the heart [1]. A robot can be used to track a point on the heart surface, moving with the heart, canceling the relative motion and allowing a surgeon to operate as if the heart were stationary. This technology is especially important when performing off-pump coronary artery bypass graft (CABG) surgery where the surgeon is required to operate on blood vessels that are small (diameters ranging from 0.5 mm to 2 mm) that move quickly. In order to allow for precise operations on vessels of these sizes, we expect that an RMS position error along the order of 100 $\mu$m to 250 $\mu$m will yield satisfactory results.

The relatively fast and high bandwidth motion of a point on the heart surface, and the high precision to which it must be tracked, makes its robotic tracking difficult [2]. Causal error feedback control alone is not able to reduce the tracking error sufficiently such that surgery can be done on blood vessels on the heart surface. A predictive controller in the feedforward path was found to be necessary [2], [3]. Such a predictive controller needs an estimate of the future motion of the point of interest (POI) on the heart surface that is to be tracked. The estimate needs to be of a finite duration into the future, referred to as the prediction horizon. Figure 1 illustrates the prediction horizon, the estimator reference data, and displays typical motion of the POI.

In this paper, a heart motion prediction method based on adaptive filter techniques is studied. A recursive least squares based adaptive filter algorithm is proposed for parameterizing a linear system to predict the motion of the POI. The feasibility and effectiveness of the approach are demonstrated. The effects of system parameters are systematically studied and simulated on a hardware test bed with 3 degrees of freedom

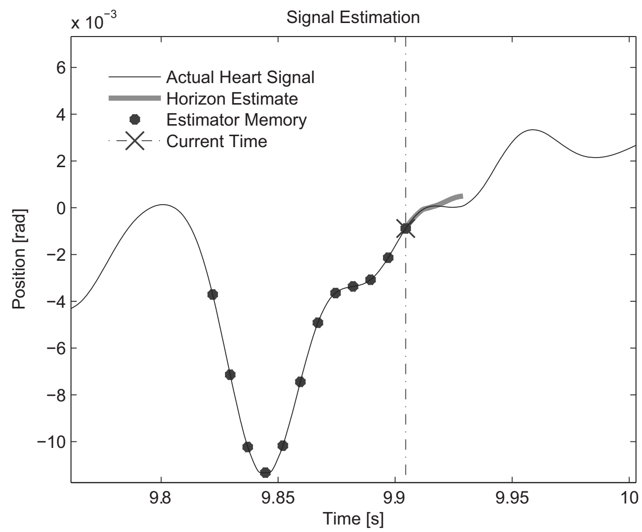


Fig. 1. Plot of heart data and the prediction made from it. The dashed line indicates the current time. The estimator memory consists of the past 11 data points and the current position which the predictor uses to estimate. Thus the predictor is $12^{th}$ order. The bold line indicate the prediction over the horizon. As can be seen, the prediction initially follows the future POI motion but diverges towards the end of the prediction horizon.

(DOF) using previously recorded heart motion data, obtained by Cavusoglu *et al.* [2], as the reference. Section II discusses how predictions were generated on similar projects. Section III and IV formally formulate the problem as well as the statistical model for the motion of the POI. The use of an adaptive filter to parameterize an estimator is demonstrated in Section V. Further, it is explained how the estimator is used to create estimations throughout the prediction horizon. Implementation details are addressed in sections VI, VII, and VIII. Finally, the evaluation results of the effectiveness of the new prediction scheme on a 3-DOF robot, using prerecorded heart motion data, is given in section IX.

## II. Related Works in Literature

This paper is concerned with estimating the prediction horizon for receding horizon model predictive control–a control scheme that relies on the estimate of the prediction horizon as a reference signal. There has already been several proposed ways to estimate motion of a POI on the heart surface.

Ortmaier *et al.* [4] used Takens Theorem to develop a

1-4244-0912-8/07/$25.00 ©2007 IEEE.

robust prediction algorithm, anticipating periods of lost data when a tool obscured the visual tracking system. Estimates were generated from a linear combination of embedding vectors of previous heart data. The weights were chosen such that better estimating vectors are weighted more heavily. The algorithm had a global prediction technique that correlated ECG signals to heart motion. It was able to estimate the system behavior when visual contact of the landmark was lost for some period of time.

Ginhoux *et al.* [5] separated breathing motion from heart motion in the prediction algorithm. The breathing motion was treated as perfectly periodic, since the patient would be on a breathing machine. The heart motion was predicted by estimating the fundamental frequency, as well as the amplitude and phase of the first 5 harmonics. This prediction was used to estimate disturbance so that the controller could correct for it.

Rotella [3] used the previous cycle of heart motion data as an estimate of future behavior. This lead to problems since the POI motion was not perfectly periodic. Bebek *et al.* [6] improved upon this prediction scheme by synchronizing heart periods using ECG data and separated heart and breathing motion, predicting only heart motion. Bebek noted that the prediction method still could be improved.

This paper introduces a new estimation algorithm into the controller described in the earlier work of Rotella [3] and Bebek *et al.* [6]. A new prediction technique using adaptive filters is proposed and used in place of the prediction algorithm of Bebek. Since the new linear predictor is parameterized by a least squares algorithm, the predictor is inherently robust to noise. The predictor only uses observations close to and including the present making it less susceptible to differences between heart periods than the algorithm of Bebek *et al.* [6]. Where as Ginhoux *et al* [5] formulated prediction for periodic POI motion, no assumptions are made towards periodicity of the system *a priori*, rather the predictor is unconstrained so that it can best mimic the motion of the POI.

## III. Problem Formulation

The receding horizon controller require an estimate of the immediate future of the POI on the heart surface. The prediction problem is to best estimate, in some sense, the value for the next several observations over the prediction horizon, given a finite number of known samples leading up to the present. Figure 1 provides a graphical schematic of the problem. Once a method is established to estimate the next observations, a sequence of future observations can be estimated.

Fundamentally, the predictor is a mapping from the samples in its memory to the best estimate of the next observation. The estimate for the next value is such to minimize the error between the prediction and the actual position of the POI as it will appear in the future. An $M^{th}$ order predictor has memory of the past $M-1$ observations as well as having access to the current observation. It uses these $M$ observations to generate the next expected observation. To generate the observation after that one, what was just estimated is treated as if it were actually observed and is added to the set of observations. The next observation predicted from this set is the next value in the prediction horizon. Proceeding inductively by this method, any number of future estimates can be made, stopping when all the predictions for the horizon have been made.

In order to generate predictions in this way, the one step prediction function must be known. Abstractly, the motion of the POI is represented as a continuous time dynamic system. An analogous discrete system must be created for estimation. However, the state space of the heart is not known—not even its dimension. To simplify prediction, the heart model must use a finite, and rather low order, state vector. The discrete transition function maps from the approximate model's state space onto the same space. The transition function for the discrete, approximate model is, in general, nonlinear. As is, this would be very difficult to parameterize; so the transition function will be assumed to be linear. This assumption is justified by the nature of the signal that is being predicted.

The motion of the POI on the heart surface is quasi-periodic, subject to small disturbance. Fourier analysis of the heart signal data reveals how this periodic nature is prevalent (see Figure 2). The heart position signal is primarily the superposition of two effects: motion due to the heart beating and motion due to breathing. Each of these signals closely resemble periodic signals with their own fundamental period. In the prerecorded data used by Cavusoglu *et al.* [2], which was collected from an adult porcine using a Sonomicrometry system, the lung motion is lower frequency with a fundamental period of approximately 0.4 Hz with only the primary harmonic appearing significant. The heart motion itself has a fundamental frequency of 2 Hz, corresponding to 120 bpm, with the first five harmonics being considered significant. The sharpness of these peaks indicate that the harmonics decay very little in time, meaning that the overall motion of the POI is similar to a superposition of periodic signals.

The justification for using a linear predictor is this: a linear system can easily be constructed that has a frequency response that mimics the heart signal's Fourier representation. The transient response would then resemble the observed heart data. Since each peak in the system's frequency response would be due to a pair of complex eigenvalues, the above five significant harmonics would only need a tenth order system to reproduce the behavior. This system would have the special property that, given the current state of the actual heart signal as initial values for the system, the transient response would follow the actual heart motion—giving a prediction. Finally, if the state was formulated as a stacked vector of past observation then the determination of the initial state would be trivial. A linear system of the above specifications would meet the requirements for the heart model transition function. However, the model would still need to be parameterized in a way to statistically minimize the error of the prediction.

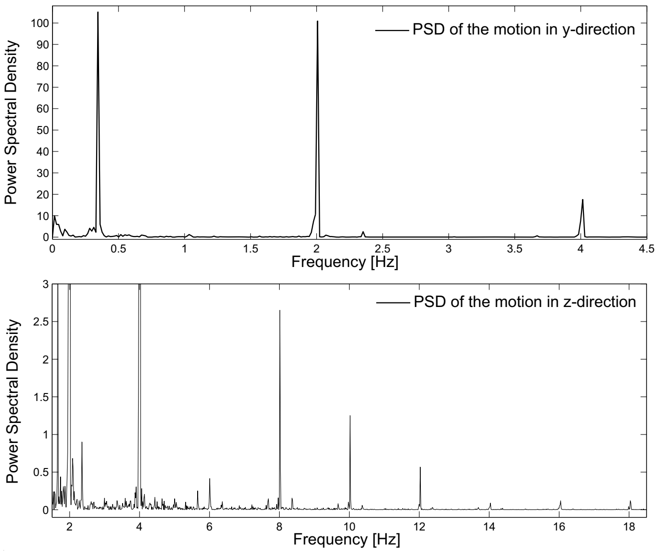


Fig. 2. Power Spectral Density of the heart motion in the y and z directions. Tall, narrow peaks with the absence of intermittent frequencies indicate largely periodic motion of the heart.

## IV. Statistical Model of Heart Motion

The heart position data consists of 3-dimensional vectors representing position. These vector samples are assumed to be generated from a vector autoregressive model (VAR). A VAR process has multiple output signals which are correlated with each other. The model is given by (see [7]):

$$\vec{\gamma}_k = \sum_{i=1}^{M} A_i \vec{\gamma}_{k-i} + \vec{\gamma}_k \tag{1}$$

In this case, it is an $M^{th}$ order VAR model. Each observation is given by a weighted sum of past observations, and is perturbed by noise given by $\vec{\gamma}_k$. Noise vector $\vec{\gamma}_k$ is assumed to be zero mean white noise. Since the linear combination of past observations account for correlation between observations, for any two noise vectors $\vec{\gamma}_k$ is uncorrelated with $\vec{\gamma}_i$ for $i \neq k$. Since the noise vector is assumed to be white, it is not useful when generating predictions of future values. Therefore, when parameterizing the equation for the purpose of prediction, only the weighting matrices need to be estimated.

The prediction is treated as if the heart statistics are stationary. In practice, the heart statistics may likely change during surgery. Should these changes be gradual, an adaptive predictor will be able to adjust to these changes sufficiently quickly. However, if the statistics change abruptly the predictor cannot adapt in time and actions must be taken to minimize the effect of poor predictions. Future work will have to address how to detect these events as well as what actions should be taken.

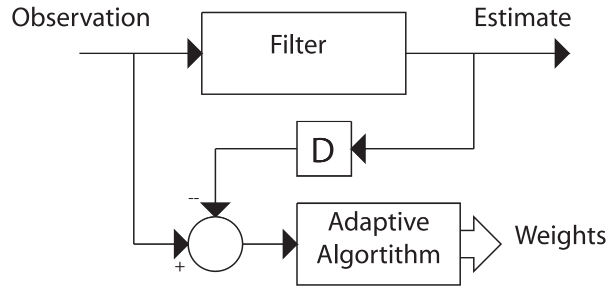


Fig. 3. The adaptive filter is arranged to minimize the error between the estimate for the current observation, calculated in the last iteration, and the actual observed value. In this way, the weights of the filter are statistically optimized to estimate one step ahead.

### A. State Space VAR Model

The VAR model given in (1) can be reformulated in state space canonical form as

$$\begin{aligned} \vec{X}_k &= \Phi \vec{X}_{k-i} + \Gamma \vec{v}_k \\ \vec{\sigma}_k &= C \vec{X}_k \end{aligned} \tag{2}$$

This system can be reformulated using an arbitrary state vector, however a stacked vector of past observations simplifies the determination of the initial state, parametrization of the matrix $\Phi$, and generation of the prediction horizon. In this case, $\Phi$ is in canonical form and can be written as:

$$\Phi = \begin{pmatrix} A_1 & A_2 & \cdots & A_M \\ I & 0 & \cdots & 0 \\ 0 & I & & \vdots \\ \vdots & & \ddots & 0 \end{pmatrix} \tag{3}$$

Future observations of the system are given by solving the state space solution at time $n$. In order to find the expected trajectory, we take the expectation of (2) and find that the solution takes the form

$$E\{\gamma_{n+k}\} = C \Phi^k \vec{X}_n \tag{4}$$

Where the above formula gives the horizon estimate made at time $n$ for a value $k$ steps into the future. Note that since $\Phi^k$ is only computed for $k < K$, where $K$ is the horizon length, $\Phi^k$ always remains finite. Therefore, stability of $\Phi$ is not a concern. Since $\vec{v}$ is unknown, but its expected value is zero by construction, it does not appear in the solution to the expected trajectory.

## V. Adaptive Filter

The adaptive predictor consists of two principle parts: a linear filter and an adaption algorithm. The input-output relation of the adaptive filter is determined by the linear filter. The adaptive filter's response is the response of the linear filter to the system's input. In this case, the linear filter will be a transversal filter. The adaptive algorithm changes the filter's weights in order to make the filter's output match the desired response in a statistical sense. The adaptive algorithm changes the filter's weights, so the filter is in fact not linear time-invariant. However, when the adaptive filter is adapting to a stationary signal, it will converge to a steady state, after which point it can be treated as being linear time-invariant.

If the adaptive algorithm is able to forget the past, just as it was able to converge to a stationary signal, it can track a signal with changing statistics [8]. In the special case that the statistics change slowly relative to the algorithm's ability to adapt, then the filter can track the ideal time-varying solution. Further, if the statistics change slowly relative to the length of the prediction horizon as well as the length of the state vector, then the adaptive filter can be considered to be locally linear time-invariant. The two afore mentioned conditions are the case with modeling the heart motion during most normal situations.

The adaption algorithm uses an exponential window to weight past observations so that more recent observations carry more weight. The exponential window was chosen because it can easily be implemented recursively. Due to this windowing, the adaptive predictor is able to track the heart signal even if the statistics of the heart signal change slowly with time.

### A. Parametrization

Traditional system identification problems using adaptive filters arrange the filter such that the input to the filter is the system's input and the desired response is the system's desired response. In this way, the filter converges towards an approximation of the system's input-output relation. However, (2) is driven by white noise input vector $\vec{v}$. This input is unknown and unable to be predicted for future observations. Thus, deriving an input-output relationship for the heart motion would be impractical. Instead, the adaptive filter is arranged as a one-step predictor. The desired response is the heart position's current observation and the input to the adaptive filter is the previous heart observation. The adaptive filter adjusts its filter weights such that it generates the statistically best estimate for the next observation, given only the current and past observations.

In order to generate the predictions, the coefficient matrices, $A_i$, from (1) need to be estimated. Equivalently, the matrix $\Phi$ from (2). This matrix is in controllable canonical form, so estimating $A_i$ is sufficient to parameterize the estimated matrix, denoted $\hat{\Phi}$. As can be seen from (1), the matrices $A_i$ correspond to tap weights in a transversal filter. In a one-step predictor, when it has converged to a solution, its filter weights are precisely the matrices needed to parameterize $\hat{\Phi}$. In this way the adaptive algorithm estimates the matrix $\hat{\Phi}$.

### B. Recursive Least Squares

Recursive least squares (RLS) was chosen to be the adaptive algorithm to update the filter weights. RLS is a method that updates a least squares solution when a new piece of data is added. In practice, the RLS solution will approach the actual solution, even if the initial estimates for the solution were wrong. To formulate the RLS algorithm for vector samples, the one step prediction problem needs to be stated as a least squares problem.

$$\begin{pmatrix} \gamma_{n-1}^T & \gamma_{n-2}^T & \cdots & \gamma_{n-M}^T \\ \gamma_{n-2}^T & \gamma_{n-3}^T & \cdots & \gamma_{n-M-1}^T \\ \vdots & & & \vdots \end{pmatrix} W = \begin{pmatrix} \gamma_n^T \\ \gamma_{n-1}^T \\ \vdots \end{pmatrix} \quad (5)$$

where the objective is to find $W$ such that the square of the error between the two sides of the equation is minimized. Since $\left( \gamma_{n-1}^T \quad \gamma_{n-2}^T \quad \cdots \quad \gamma_{n-M}^T \right) W = y^T$ where $y$ is the expected one step estimate, it is clear that

$$W^T = \left( A_1 \quad A_2 \quad \cdots \quad A_M \right) \quad (6)$$

where $A_i$ are the weighting matrices from (1).

Using the statement of the least squares problem for the one step estimator in (5), the RLS algorithm can be derived. The derivation of the vector valued RLS algorithm is analogous to Haykin's derivation of the scalar case [8]. Further, an exponential weighting factor can be introduced to produce a weighted least squares problem. This factor, $\lambda$ is multiplied to each observation at each iteration, producing an exponential weighting of observations.

The RLS algorithm was formulated with past observations exponentially windowed such that the algorithm has the ability to forget the distant past. The exponential window parameter $\lambda$ is referred to as the forgetting factor. When $\lambda = 1$, the RLS algorithm does not forget old observations, instead it has infinite memory. When $\lambda < 1$, observations are reduced in importance such that the least squares solution places a greater importance on minimizing error for the more recent observations and their prediction than on older ones. From the combination of weighted memory and convergence to the optimal solution, if the statistics of the heart motion change in time, the RLS algorithm is able to adapt to the new heart behavior.

### C. Prediction

Following from (4) , the one-step prediction is:

$$\gamma_n = \hat{W} \begin{pmatrix} \gamma_{n-1} \\ \gamma_{n-2} \\ \vdots \\ \gamma_{n-M} \end{pmatrix} \quad (7)$$

which is precisely the expected value of $\gamma_n$ from (1).

The prediction horizon of length $K$ starting at time $n$ is the solution to (4) with initial condition vector being the stacked vector of the past $M$ observations.

In the actual implementation, predictions over the horizon length are generated by iterating this function several times. This avoids the computational complexity of calculating $\Phi^k$ and using it directly to compute the predictions. The calculation of $\vec{X}_n = \Phi \vec{X}_{n-1}$ is simplified by calculating $\gamma_{n+k}$ by (7), shifting the stacked observation vector $\vec{X}$ down by one observation size and making the first observation the current estimate. In this way, the computational complexity of iterating the state variable increases proportional to $M$,

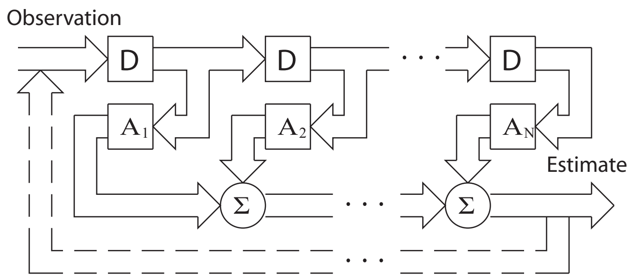


Fig. 4. The one step estimate is generated by use of a transversal filter weighting the past observations to produce an estimate for the next expected observation. When generating predictions for the horizon, the path is closed as the last estimate is treated as the current input. The prediction sequence is the collection of the estimate output each time the filter is iterated.

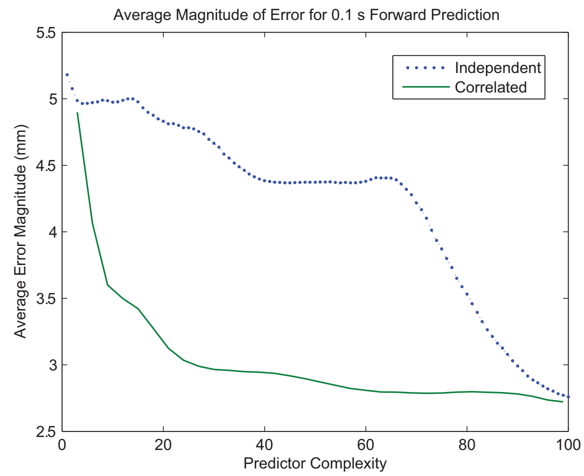


Fig. 5. Comparison plot for independent and correlated signal predictions. The predictor complexity is relative to the complexity of three independent RLS predictors of the same order. Therefore correlated predictor complexities are three times the estimator order. The trend of the plot is that for the computational effort required, accounting for correlation between signals yields better results.

opposed to $M^2$. Since the observation matrix $C$ from (2) simply retrieves the first observation from $\vec{X}$, multiplication by $C$ is not necessary because the observation can be directly indexed and removed.

Using the above described method for obtaining an estimate, the horizon is generated by collecting the next $K$ estimates of the heart surface trajectory. Each time the process starts, the current state vector is used to start the process. As each prediction is generated, the state vector is updated with the newest observation and the observation is saved into a collection. This collection of estimates is the expected trajectory of the heart surface given the past $M$ observations. When each horizon is generated, the predictor starts with the current set of actual observations. This way, if the transition matrix $\Phi$ is unstable then the predictions will not diverge in the finite horizon window.

## VI. Correlation Between Signals

The described method for generating estimates use the matrices $A_i$ as weights for the vector observations. This allows for motion along one axis to be correlated with motion on the other two. This feature comes at a significant computational cost. A less computationally intense method would treat motion of the POI on each axis as being independent. Since it would be using 3 scalars to weight each past data sample, opposed to a $3 \times 3$ matrix, it would require one third of the computational effort to process the same number of past observations. This would allow, in the same time, for more samples of past data to be processed when generating the next prediction.

In order to decide which option is best for implementation, the effectiveness of each estimate per the computational effort needs to be determined. The complexity corresponds to the order of an independent predictor of equivalent order, or a correlated predictor with order being a third of the complexity. The simulation was done by processing all of the data at once, calculating the least squares fit for the weights, and simulating predictions. The average of the Euclidean norm of the error is shown for both systems in Figure 5.

The results shown in Figure 5 reveals that treating the heart signals as being correlated yields better estimates when the computation effort is low. Typical complexity for an on-line estimator would fall in the 20 to 50 region.

Also, it should be mentioned here that these errors are not necessarily monotone with respect to the filter order. This is because the predictor minimizes the error of the one step prediction. The one step error is monotone decreasing in magnitude because if the order was increased and the new weights were held to be zero, we would have the same error that was in lower order. This way, the one step error will never increase with order. However, minimizing the one step error does not necessarily correspond to minimizing the error at some arbitrary time in the prediction horizon. Rather, Figure 5 reveals that as the order increases, and so the one step prediction error decreases, the error in the prediction horizon tends to decrease as well.

## VII. Prediction Error within the Time Window

Section VI studied the effect of predictor order on the prediction error for a fixed amount of time in the future. However, the error varies based upon how far it is in the future. Figure 6 shows the error across the prediction horizons of the correlated predictions for several complexities—as calculated in Section VI. The figure was created in the same manner as described in Section VI.

The behavior of these plots appear to be linear for times in the immediate future, and holds particularly well for the lower complexity cases. The monotone increasing error with lead time displayed in this plot reflects that the quality of the estimation decreases as you attempt to estimate further into the future. This generalization will be useful for allowing the predictive controller to properly weight estimates in the horizon when calculating the control law.

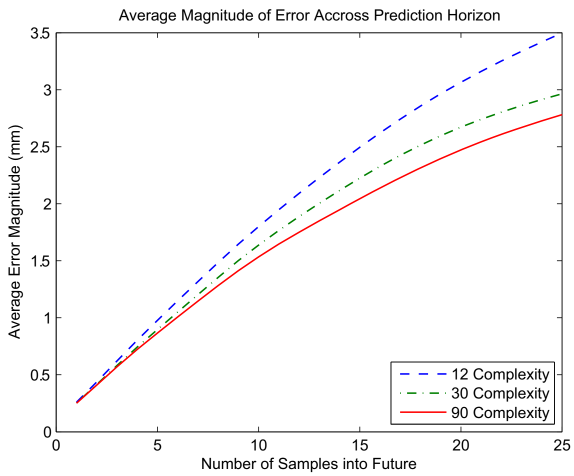


Fig. 6. Plot showing how the magnitude of the error varies throughout the prediction horizon. The data is processed at a sampling rate of 250 Hz. The prediction horizon used in the model predictive controller would correspond to seven samples in length at this frequency. Additional points are displayed to illustrate the trend.

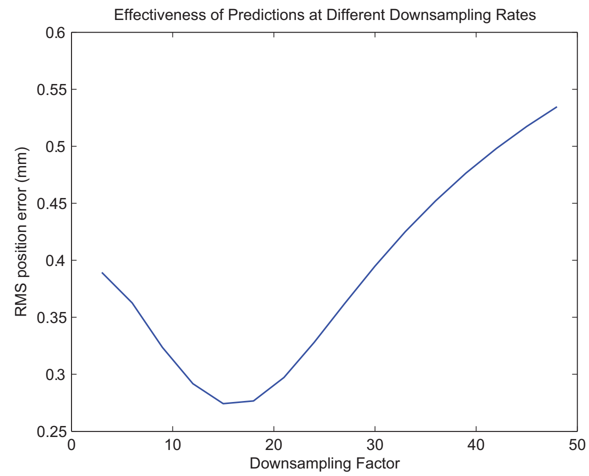


Fig. 7. The predictions can be made from a downsampled version of the 2 kHz POI signal. High frequency processing lends to problems with numerical accuracy and low frequency processing can misrepresent the dynamics by excessive interpolation. This plot illustrates this tradeoff and suggests an optimal downsampling factor of 15 for the predictor.

## VIII. Sampling Time

Since the prediction scheme for the heart data is a system that can be locally approximated as linear time invariant for short periods of time, so long as the heart signal is predicted at a sampling rate higher than the Nyquist rate, sampling rate is theoretically arbitrary. However, in practice, this did not prove to be the case. Running at a lower sampling rate means that there is a larger numerical difference between samples—important in the finite precision implementation. Also, to predict over a fixed length horizon at a lower sampling rate, fewer iterations of the predictor need to be computed. Finally, for a predictor of a fixed length, the sampling rate corresponds to how much time is between each sample—translating to how far back the system has memory. In Figure 1, the predictor order translates to the number of past observations available to the predictor, whereas the sampling rate corresponds to the spacing between those points.

The control algorithms run at a sampling rate of 2 kHz. This is well above the Nyquist rate for heart motion signal which, for 120 beats per minute and allowing for six harmonics, is about 25 Hz. The effects on RMS position error caused by changing the sampling rate are plotted in figure 7. From this plot, it appears the ideal downsampling rate in this controller is 15—corresponding to a processing frequency of 133 Hz. This optimal value is a trade off between the numerical problems associated with calculating predictions using a finite number of samples at high sampling rates and the inaccuracy caused by interpolation and aliasing.

## IX. Test on 3-DOF Robotic Test Bed

The proposed algorithm was tested on a 3-DOF robotic test bed. The 3-DOF test bed is a PHANToM Premium 1.5A haptic device which acts as our surgical robot. The trials used the prerecorded heart data mentioned in Section III. The system was ran with a sampling time of 0.5 ms using these data points in place of online measurements of the POI. The encoder positions on the PHANToM were recorded at each time sample and these positions were transformed into end effector positions. The reported RMS errors are calculated from the difference between the prerecorded target point and the actual end effector position at each time sample.

The controller from [6] was modified to include the new prediction algorithm. It was implemented in Simulink for xPC Target and ran in real time on a 2.6 GHz Pentium 4 PC . The controller compensates for gravity and the Coriolis effect. The linearized model was controlled using a model predictive controller. The MPC was formulated to track the horizon estimate weighted by a quadratic objective function.

The estimator used during these trials was a $25^{th}$ order correlated signal estimator, processing data at 133 Hz. Simulations using the estimation scheme were ran ten times with the estimation algorithm and again with the actual heart motion data as future signal reference for the prediction horizon. The later case represents a 'perfect' estimation. The RMS position errors in millimeters are reported in Table I under the simulation heading. Due to the uncertain nature of the hardware trials, the 95% confidence interval of each RMS position error is reported in Table I.

TABLE I

RMS End-Effector Position Errors for 56 s of Heart Motion Tracking

| Errors in mm | Simulation | Experiment |
|---|---|---|
| Exact Reference | 0.283 | $0.2867 \pm 0.0003$ |
| Estimator | 0.258 | $0.449 \pm 0.003$ |

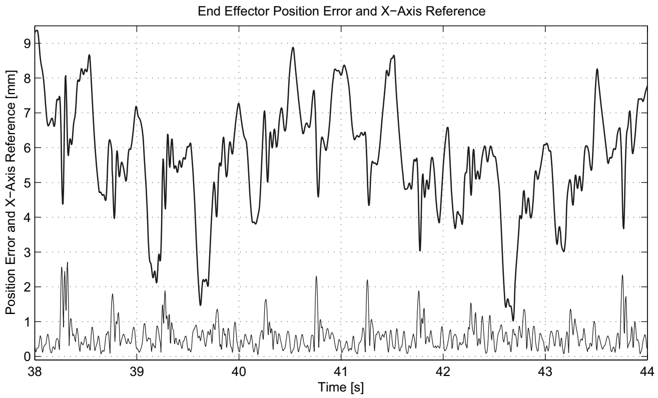


Fig. 8. Plot showing The magnitude of the end effector error (below) superimposed with the reference signal for the x-axis.

As can be seen from Table I, in the simulation the estimator out performed the exact heart signal. This is likely due a combination of two factors. First, the simulation model is a linearized, reduced order model of the actual hardware. Second, the estimator has a robustness characteristic that makes its output less noisy than the actual heart data. The combination of these two factors yields good results in the linear case. However, when the experiment is performed on the hardware, the effects of the nonlinearities are seen when the performance of the estimator-driven controller decreases. It should be noted that though the simulation provides valuable insight to the effectiveness of the controller, it is the experimental trials that are the best indicator of performance.

While the RMS end effector position error gives a sense of the average tracking performance of the robot, it is important to look at the large position errors to determine when and why they occur since they are the errors that will ultimately determine whether or not the system is effective in tracking the POI motion. Figure 8 shows that these peaks tend to occur periodically and at approximately the same part in the POI's cycle. As can be seen from the reference signal, the POI moves very rapidly during that time period (approximately a tenth of a second or less). The low bandwidth mechanical system experiences large errors during this time. The largest error peak observed had a maximum at 5.0 mm. Measurements during these periods may be inaccurate due to errors caused by the susceptibility of the Sonomicrometry system to high frequency noise. Future data collection will in addition have an inertial sensor, and use sensor fusion techniques to attain an accurate measurement of the POI.

## X. Conclusions

From the experiments in Sections VI through IX, the following general conclusions were reached. Higher order estimators yield predictions with, on average, lower error. Since the estimation process is implemented in real time, only lower computational complexities are possible. Therefore, parameterizing a model that correlates motion of the POI between axes is preferred. Also, error in the prediction horizon increases as the predictions become further in the future. The manner in which they increase is approximately linear, especially for shorter prediction horizons. There is an optimal processing rate for producing predictions. It was found in this case to be approximately 133 Hz.

Using future heart data as the estimate of the prediction horizon yielded an experimental RMS end-effector position error 0.287 mm when tests were ran on a 3-DOF test-bed. Since this case has an exact estimate of the future, this number is considered the upper limit of performance for the current robot and controller. Bebek [6] reported an experimental RMS end-effector error of 0.653 mm. The experimental RMS error of 0.449 mm obtained using the estimator described in this paper represents a significant improvement in prediction performance.

There is future work that still remains to be done. The controller should be changed so that error in the horizon can be estimated and predictions can be accordingly weighted to improve the effectiveness of the control law. Also, the adaptive filter will track a change in heart statistics provided the underlying dynamics of the POI motion are slowly varying. However, how long it takes to respond to changes, how quickly do these dynamics typically change, and what should be done while the estimator is readjusting to a sudden change still needs to be answered.

## Acknowledgment

The authors would like to thank Dr. Mark Ratcliffe for his help during collection of the experimental data.

This work was supported in part by National Science Foundation under grants CISE IIS-0222743, EIA-0329811, and CNS-0423253, and US DoC under grant TOP-39-60-04003.

## References

[1] A. Trejos, S. Salcudean, F. Sassani, and S. Lichtenstein, "On the feasibility of a moving support for surgery on the beating heart," in *Proc. of Medical Image Computing and Computer-Assisted Interventions (MICCAI)*, 1999, pp. 1088–1097.

[2] M. C. Cavusoglu, J. Rotella, W. S. Newman, S. Choi, J. Ustin, and S. S. Sastry, "Control algorithms for active relative motion cancelling for robotic assisted off-pump coronary artery bypass graft surgery," in *Proc. of the 12th International Conference on Advanced Robotics (ICAR)*, Seattle, WA, USA, July 2005, pp. 431–436.

[3] J. Rotella, "Predictive tracking of quasi periodic signals for active relative motion cancellation in robotic assisted coronary artery bypass graft surgery," M.S. Thesis, Case Western Reserve University, Cleveland, OH, USA, August 2004.

[4] T. Ortmaier, M. Groeger, D. H. Boehm, V. Falk, and G. Hirzinger, "Motion estimation in beating heart surgery," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 10, pp. 1729–1740, October 2005.

[5] R. Ginhoux, J. A. Gangloff, M. F. DeMathelin, L. Soler, J. Leroy, M. M. A. Sanchez, and J. Marescaux, "Active filtering of physiological motion in robotized surgery using predictive control," *IEEE Trans. Robotics*, vol. 21, no. 1, pp. 67–79, February 2005.

[6] O. Bebek and M. C. Cavusoglu, "Intelligent control algorithms for robotic-assisted beating heart surgery," *IEEE Trans. Robotics*, vol. 23, no. 3, pp. 468–480, June 2007.

[7] W. Wei, *Time Series Analysis - Univariate and Multivariate Methods*. Addison-Wesley Publishing Company, New York, 1990.

[8] S. Haykin, *Adaptive Filter Theory*, 4th ed. Upper Saddle River, New Jersey: Prentice Hall, 2001.