

Comparison of filtering methods for crane vibration reduction

ANDERSON DAVID SMITH

This project examines the utility of adding a predictor to a crane system in order to test the response with different filtering methods: notch filter, low-pass filter, and input shaping. The use of filtering and shaping methods allows payloads to be moved to their destinations faster and with less residual oscillation. Filters, though performing a similar function to the input shaper, are typically harder to find and more difficult to implement. The goal of filter and input shaper comparison is to provide the means to experimentally validate the superiority of the input shaper to the conventional filter. Previously, work was done with WinCC software in order to create a user interface in which the position of the tower crane trolley and payload are displayed on a WinCC graphical user interface (GUI) along with the available workspace and a set of controls for actuating the crane. The project further develops the user interface of the WinCC software by providing a predictor which predicts the trolley motion of the mobile tower crane in order to prove that a predictor improves user efficiency.

INTRODUCTION

Cranes, as well as all flexible systems, vibrate during their use, reducing the accuracy of their desired operations. Filters and input shapers are devices which attempt to reduce vibration in a flexible system by convolving the input signal in some way. Digital notch filtering, digital low-pass filtering, and input shaping are three well-known methods for modifying reference commands to reduce vibration in flexible systems. Reference commands are inputs which, in this case, drive the crane to a desired location and are actualized through some form of user interface. Flexible systems are systems containing a flexible body which would be modeled as a spring that causes some vibration in the system. Like filtering, input shaping has been shown to effectively reduce vibration (Smith, 1957, Bhat, 1990). In unshaped signal generation of tower cranes, the load on the crane will always have residual oscillation or swaying due to the crane's motion. The first sections of the project pertain directly to the comparison of these different filtering methods on reducing such vibration.

In mechanical systems, filtering and input shaping are primarily used for vibration reduction. Filtering reduces vibration by suppressing certain frequencies. For example, a low-pass filter allows low frequencies to pass while suppressing all other frequencies. Notch filtering allows low and high frequencies to pass while suppressing a defined range of frequencies.

Input shaping specifically reduces vibration by taking an input command into a system and modifying it by convolution into two or more impulse sequences which add up to the original signal. These impulses are timed such that the residual vibration resulting from them will be 180 degrees out of phase and thus cancel out. Figure 1 shows a hypothetical system response to a step input. The system is assumed to be undamped, meaning that it is free to oscillate without any reduction due to friction or other effects.

Suppose this input signal instead was separated into two impulses, each at 50% of the desired original impulse. Each of these impulses would have a residual vibration associated with it. Through proper spacing of these impulses in time, one can theoretically cancel the vibration by spacing these impulses such that the residual vibrations of both impulses are 180 degrees out of phase and thus cancel each other. Such a system would

Contact email: gtg207x@mail.gatech.edu
Advisors: William Singhose, PhD
School: School of Mechanical Engineering
Georgia Institute of Technology

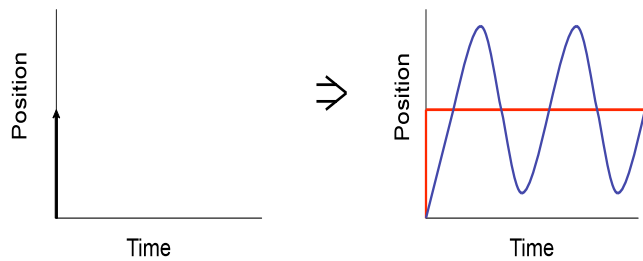


Figure 1. Hypothetical system response (right) to a step input (left). The desired step input is red, while the actual system response is in blue.

provide zero residual vibration, as illustrated in Figure 2.

Since the introduction of robust input shaping, substantial evidence has led to the assessment that input shaping is superior to both notch and low-pass filtering for suppressing vibration in mechanical systems (Singer, 1999). In this project, input shapers and conventional filters were compared using simulations based on 3 criteria: filter duration or the “filter length is a lower bound on the move time,” residual vibration in the system, and robustness to system uncertainties (Singer, 1999). It was found that, based on the aforementioned criteria, input shapers should always be equal or superior to conventional filters. In the project, experimental data was tabulated which will be used to attempt to further previous research regarding the superiority of input shapers.

There are, however, tradeoffs to a reduction in vibration. Such vibration reduction results in a longer filter and shaper duration because the command is convolved into a more complex series of pulses. This slower response causes a tendency for the user to drive the system past a desired location. To attempt to reduce this problem, a predictor was designed which predicts the end position of the crane after the user has given it a command response. By using a predictor, a user can navigate to a desired location and know that, when the system response ceases, the predicted end position will coincide with the actual end position, preventing the overshoot problem. As such, a predictor

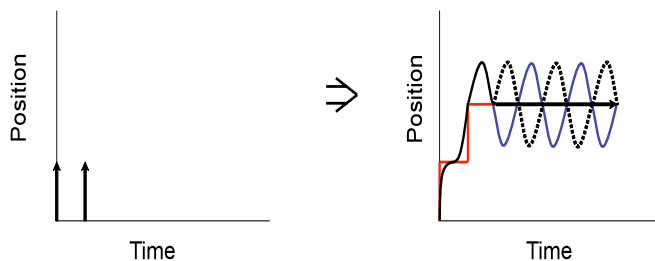


Figure 2. Example of input shaping using a hypothetical response (right) to a convolved step input (left). The blue line is the residual vibration of the first impulse, while the dotted black line is the residual vibration of the second impulse. The solid black line represents the total vibration of the system.

should allow faster response times. The purpose of the project is to provide a means by which future experiments can prove whether a predictor improves operator performance using input shaping.

METHODS

The first task of the project was to generate oscillation data for the bridge crane regarding a low-pass filter, a notch filter, and an Extra Insensitive (EI) shaper. The EI shaper is a robust form of input shaper which suppresses vibration for a wide range of frequencies. For each trial, a model tower crane was given a full velocity step input for a fixed duration. In order to ensure that these velocities remained constant, a previously generated straight line trajectory was used for the trials. The oscillation data was recorded by measuring the distance which the payload swayed from the trolley position via a camera mounted on the trolley. The camera works by generating light at a desired frequency and then detecting the reflection of that light off of a reflective surface attached to the hook of the tower crane. Then, the camera takes the data it receives from the hook at a specific angle and stores that data in a position matrix. An algorithm designed in MATLAB then determines the displacement by converting camera output data into centimeters of displacement. This is done using a conversion factor determined experimentally, by comparing a measured hook displacement to its corresponding camera output.

The three filters were then compared based on their vibration amplitude at different frequencies. Specifically, peak-to-peak vibration amplitude resulting from different shaper types for different cable lengths of the crane was measured using a tape measure. These oscillation amplitudes, resulting from the different commands, were tabulated at cable lengths of 15, 16, 18, 20, 22, 24, 26, 30, 34, 38, 42, 46, 54, and 62 inches. The vibration reduction methods that were tested include: unshaped (no filters or shaping), low-pass filter, notch filter, and the EI shaper. The next task was to generate shaper results for different pass band frequencies. The insensitivity of low-pass and notch filters was examined at different edge frequencies, ranging from 0 to 0.5, in order to determine its effect on filter duration by plotting the edge frequency versus the filter duration. The insensitivity of a filter or shaper is essentially a measure of the robustness of that filter/shaper to reduce vibration despite changes in frequency. The final task was to create a predictor for input shaper position to use with tower crane Programmable Language Controller (PLC) and Windows Control Center (WinCC) software. The PLC is used to interface the tower crane with a computer and the WinCC allows the user to interface with the PLC program via a graphical user interface. The purpose of this task was to create a predictor for use in tests to assess any increase or decrease in efficiency resulting from having the predictor.

DATA ANALYSIS FOR FILTERS AND COMMAND GENERATION

Figure 3 shows average data from three trials for each of the filters and shapers analyzed. The low-pass filter generally maintained the lowest peak-to-peak amplitude of the filters. Note that the unshaped command dips at roughly 38 inches to provide almost no residual vibration. This occurs because, at that particular length of the cable, the corresponding command acts like a filter. The filtered and shaped methods were all expected to produce less vibration than the unshaped case.

As previously mentioned, a major measure of the robustness of a filter or shaper is its insensitivity. When modeling error increases, the actual frequency differs more from the theoretical modeled frequency and results in vibration; the insensitivity is a measure of how much the actual frequency can differ from the modeled frequency while remaining within a certain tolerable level of vibration. Insensitivity data was tabulated based on camera data associated with Figure 3, generated using previously mentioned methods. Figures 4, 5, and 6 show both theoretical and experimental magnitude curves for the low-pass filter, notch filter, and EI shaper, respectively, at various normalized frequencies. Data generally deviates from the model due to actuator limits on the crane. Actuators, in the case of the crane, are electric motors, and they are limited by physical constraints such as a maximum speed and torque. These system actuator limits prohibit the crane from accurately matching the desired signal input sent to the system. In all cases, the vertical axis is the magnitude of the insensitivity while the horizontal axis is the normalized frequency, found by dividing each frequency by their maximum values in each case. The blue line represents the expected insensitivity for each normalized frequency while the red squares represent the experimental data at various frequencies. The red bars represent an estimated error associated with the experimental values. The V_{tol} term is the tolerable amount of vibration for the system, arbitrarily decided to be 5%. The EI shaper works based on a given tolerable level of vibration and so this value may be chosen at will.

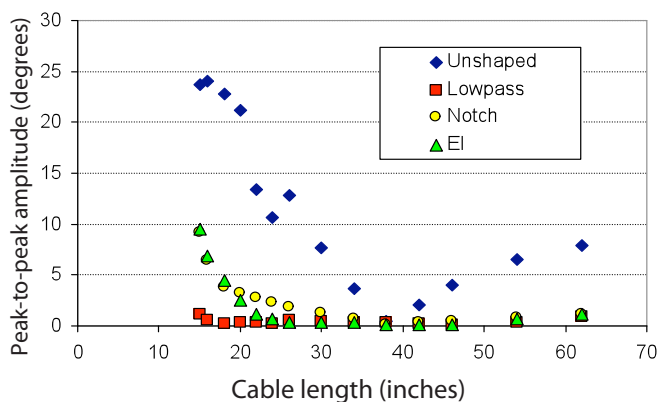


Figure 3. Peak-to-peak amplitudes for several filters, averaged across three trials each.

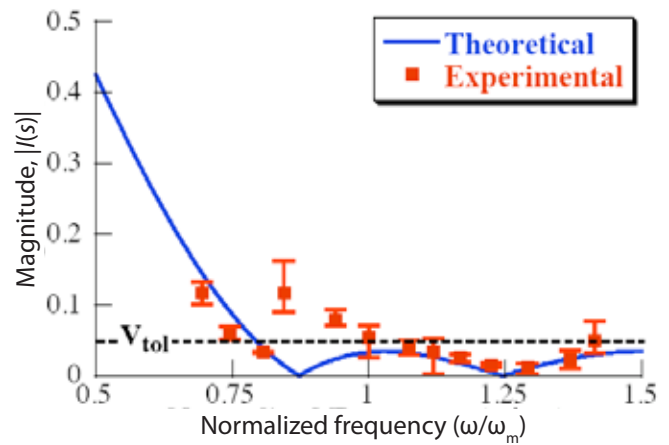


Figure 4. Magnitude of the insensitivity $|I(s)|$ versus normalized frequency for a low-pass filter, determined both theoretically and empirically.

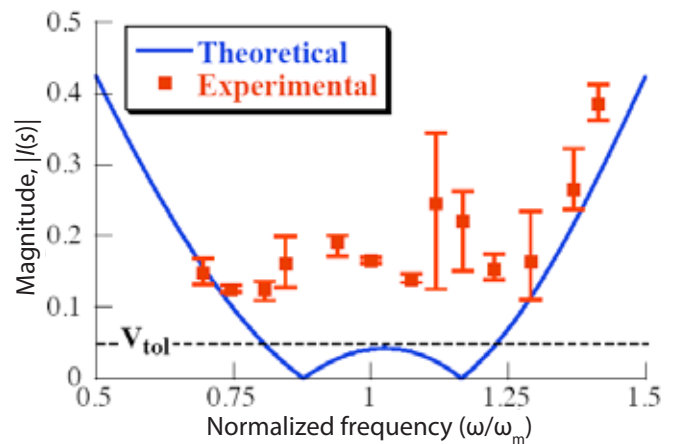


Figure 5. Magnitude of the insensitivity $|I(s)|$ versus normalized frequency for a notch filter, determined both theoretically and empirically.

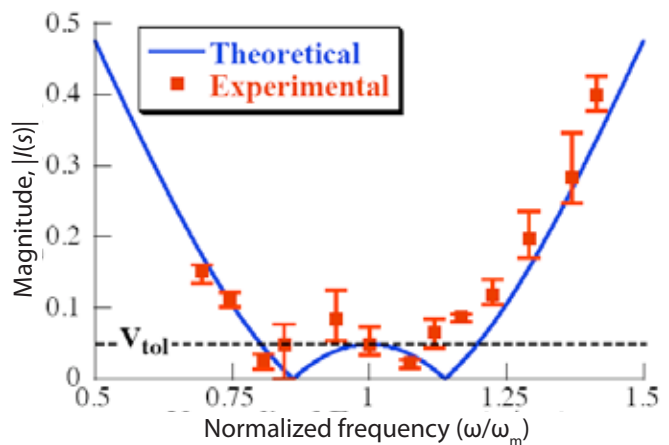


Figure 6. Magnitude of the insensitivity $|I(s)|$ versus normalized frequency for an EI shaper, determined both theoretically and empirically.

The data deviates farther from the model for a notch filter. This discrepancy arises due to the fact that the notch filter is a significantly more complex filter than the other cases discussed. Notch filter complexity makes the command more difficult for the crane to track, which leads to vibration. Therefore, the results from the notch filter are more prone to deviation from the model.

DURATION AND INSENSITIVITY DATA ANALYSIS FOR FILTERS

The next task was to generate shaper results for different pass band frequencies. Pass band frequencies are the range of frequencies that a band pass filter allows to pass while suppressing all other frequencies. Figure 7 shows an example of a band pass filter. At frequencies below ω_{p1} and above ω_{p2} , the signal is suppressed by the filter. Between those two pass band frequencies, however, the signal is not suppressed.

For this experiment frequencies ranged from 0.05 Hz to 0.5 Hz in intervals of 0.05 Hz. In order to generate these values, MATLAB's shaper toolbox was employed. These durations were generated based on theoretical models rather than direct experimentation. The goal of the band pass generation was to determine the validity of previously generated data in hopes of improving on previous results. One common trend among the filters is that the duration of the filter process increases with increasing pass bands. Moreover, increasing the frequency of the filters has little effect on the filter duration, as shown in Figure 8. Edge frequencies, ω_{p1} , were generated using 10 Hz, 20 Hz, 50 Hz, and 100 Hz sampling rates.

The sample time versus the filter duration at two different pass band frequencies ω_{p1} and ω_{p2} is shown in Figure 9. Note that as the sample time increases, the filter duration remains nearly constant for both pass band frequencies.

The next focus of the project was to evaluate the insensitivity of the low-pass and notch filters and determine how these changes in insensitivity affect the duration of the filter. The insensitivity of a filter is a built in measure of the robustness of that filter. Unfortunately, to produce a more robust filter, the filter will often require a longer duration (Vaughan, 2007). Filter insensitivities do vary from filter to filter so comparison between various filters is necessary. These effects for the low-pass filter are demonstrated in Figure 10. Note that when the insensitivity of the filter is increased, the duration for that filter also increases. Furthermore, as the pass band frequency is increased, the duration increases asymptotically.

Similar trends occur in the notch filter as shown in Figure 11. As the insensitivity of the filter increases, the duration also increases. Likewise, as the bandpass frequency is increased, the duration increases. Further, as pass band frequencies are in-

creased, durations approach an asymptote and their calculation no longer becomes feasible. Therefore, the low-pass filter was only run at pass band frequencies ranging from 0.05 to 0.45 Hz while the notch filter ran at frequencies ranging from 0.05 to 0.40 Hz. The tremendous increase in filter duration required to get robust filters shows the inherent superiority of using in-

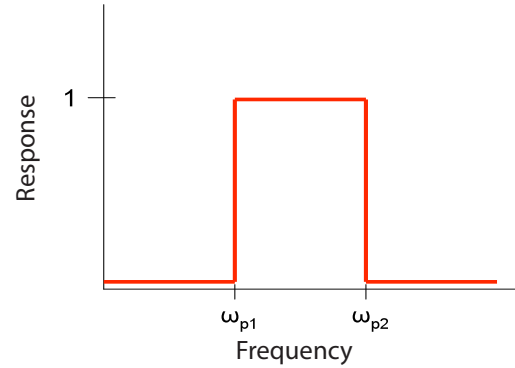


Figure 7. Theoretical band pass filter. Such a filter allows the passage of a signal between frequencies ω_{p1} and ω_{p2} , but not elsewhere.

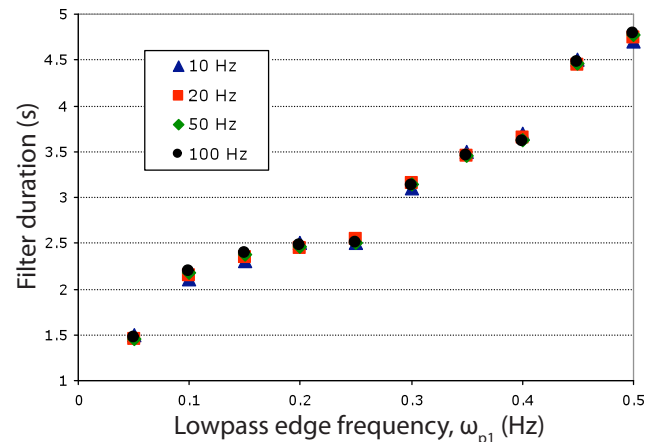


Figure 8. Lowpass edge frequency vs. filter duration.

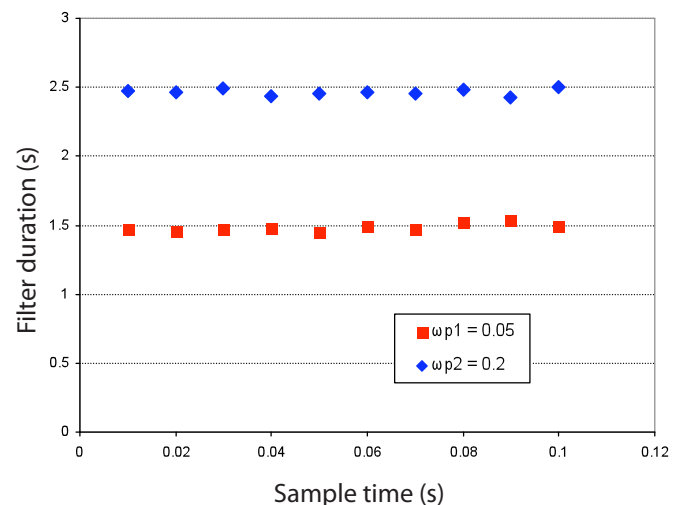


Figure 9. Sample time versus filter duration for frequencies of $\omega_{p1} = 0.05$ and $\omega_{p2} = 0.2$ Hz.

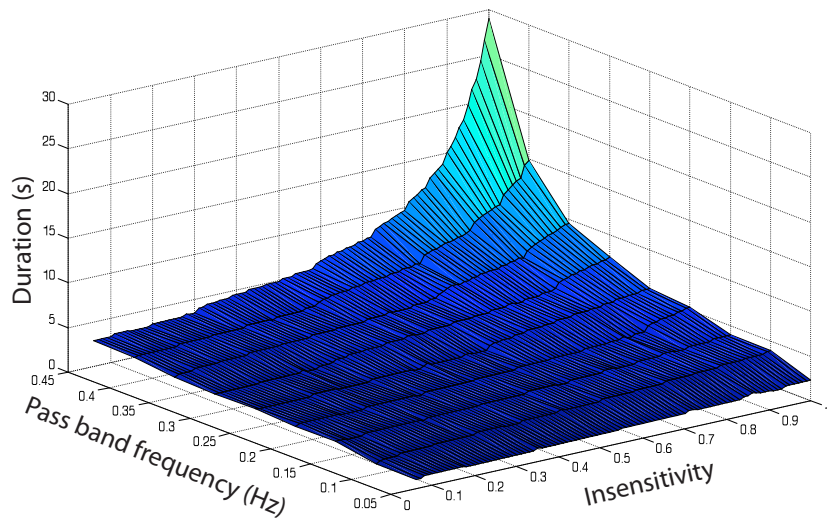


Figure 10. Effect of insensitivity change on both duration and frequency for a lowpass filter.

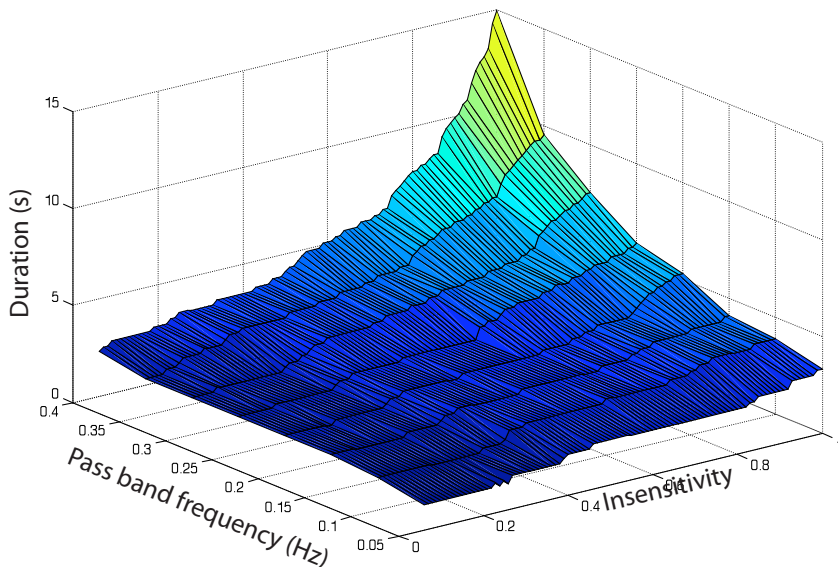


Figure 11. Effect of insensitivity change on both duration and frequency for a notch filter.

put shaping which does not require such increases in duration (Singer, 1999).

FUNCTIONALITY OF A WINCC PREDICTOR FOR A TOWER CRANE PLC

The final goal of the project was to create a predictor for input shaper position for use with the tower crane PLC and WinCC software. In order to create the predictor, the associated parts of the GUI including buttons and subroutines had to be created. This predictor will then be used in future experiments to determine if it too improves the control effort. Previous work had been done with the WinCC software, provided by Siemens, in order to create a user interface in which the position of the tower crane trolley and payload are displayed on the WinCC along with the available workspace and a set of controls for ac-

tuating the crane. The basic workspace design for the crane is shown in Figure 12.

This workspace was modified by adding a predictor which predicts the final destination of the input shaper as a total displacement of the unshaped case. Recall that by adding a shaper to an unshaped command, the shaper essentially modifies the input signal at percentages of its maximum allowable motion. The total distance output by the shaper command will match that of the unshaped command but using a longer duration, thereby creating the potential for the user to overshoot the destination. Therefore, by using the unshaped command as a predictor, a driver will know by the predictor where the trolley will eventually finish when the shaped command is passed through the system thus predicting its final destination. Figure 13 shows the modified user interface.

Likewise, Figures 14(a) and Figure 14(b) show the predictor being actuated on a path. In Figure 14(a), the predictor has begun moving while in Figure 14(b), the predictor has finished its motion. As the predictor moves along the path at the unshaped command velocity, the actual trolley position trails behind at its respective percentage of the maximum velocity. Eventually, the predictor and trolley come to rest at the desired end location, shown in Figure 15.

CREATION OF THE WINCC AND PLC PREDICTOR

In order to communicate between the WinCC and PLC program, entities called tags are created in the WinCC. These tags are identified with various components of the user interface. Each tag references a symbolic variable in the PLC program. In turn, this symbolic variable communicates with various parts of the PLC program and may be modified and re-entered into the WinCC to update data values. Symbolic variables may also update data blocks in the PLC. Data blocks are used to store data passing into the PLC through symbols, the PLC program and the tower crane. Figure 16 shows a graphical representation of the WinCC and PLC interface with the tower crane.

The PLC program is itself divided into various types of blocks: data blocks provide storage for variable values and other forms of data; function blocks contain both data and functions. Functions contain the bulk of the script which tells the PLC pro-

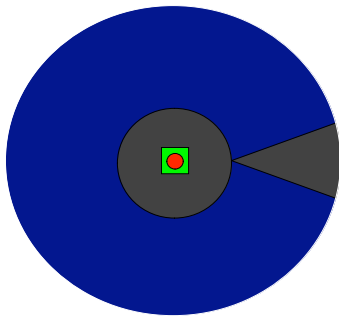


Figure 12. Basic tower crane workspace, which the user sees when operating the crane via the WinCC software. The blue area represents the available workspace, the green square represents the crane's trolley position, and the red circle represents the crane's payload position as relayed by a camera mounted on the trolley.

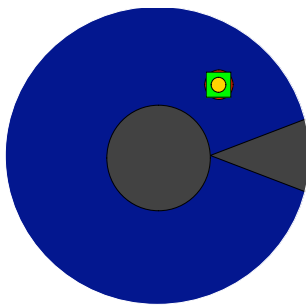


Figure 13. Modified interface, with predictor. The yellow circle now represents the payload position, while the green rectangle is again the trolley position. The red target (behind the rectangle) is now the predictor position.

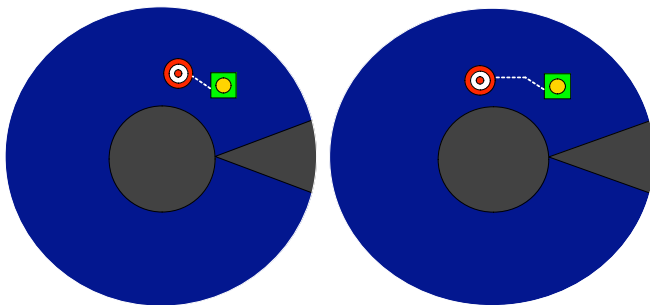


Figure 14. (a) Actuated predictor motion begins. (b) Actuated predictor motion ceases.

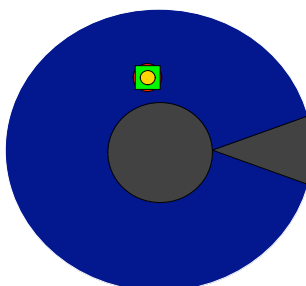


Figure 15. Trolley position ends at predictor position.

gram what to do. One such function, the organization block, is type of master function which drives the other functions. This function is necessary to control the subroutines.

A function was created to determine and update the predictor position so that the user can visually account for and adjust to the command delay that is caused by input shaping and filtering methods. The function is cycled every 100 ms and inputs the current trolley and slewing velocities. The trolley velocity is measured by the rate at which the trolley position moves radially in or out. The slewing velocity is the velocity of the trolley moving either clockwise or counterclockwise around its axis. After the function inputs the current velocities, it then outputs an updated predictor position to the data blocks. At the start of the program, a counter starts which will set the position of the predictor to the position of the trolley while the counter is less than 10 cycles (1 second). The counter cycles each time the function is called so the counter will set the predictor to the position of the trolley for the first 1 second of the program. This counter functionally resets the predictor for the user when the user performs a new motion. The function also contains a variable corresponding to the reset predictor button on the WinCC. Whenever the reset predictor button is pushed, the counter is reset to zero, causing the predictor to realign itself with the position of the trolley for 1 second. A reset button is necessary for the predictor because there is no feedback between the predictor and the trolley position. Without feedback, the predictor would become more inaccurate with continued use. Predictor inaccuracy could occur for several reasons: sometimes the motor speed will perform differently, causing a discrepancy between the predictor position and the actual position. The calibration of the predictor is performed experimentally and is subject to some calibration error. As such, there is need for the reset button.

Several changes were also made to the WinCC program aside from the creation of the predictor. A button was added to tog-

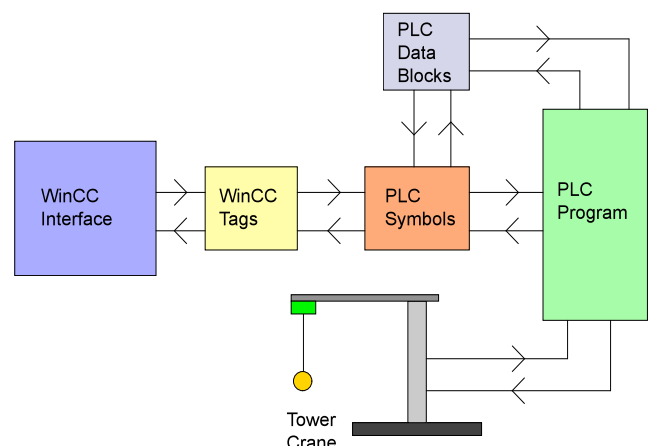


Figure 16. Tower crane interface with WinCC and PLC.

gle the predictor on and off. This option is done in order to perform comparative tests to demonstrate that operator effectiveness increases with the addition of a predictor. For example, one trial would be done with the predictor activated and another without it in order to compare the trials. Moreover, button sizes were enlarged on the GUI in order to make operating the trolley easier using the WinCC.

CONCLUSIONS

Results of the comparison between different filtering methods conformed with previous experimentation. However, tabulated data in Figure 3 does not take the filter duration into account, skewing the results toward favoring the low-pass filter. Figure 8 illustrates that, for a low-pass filter, the edge frequency causes drastic changes to the filter duration. Edge frequency values in Figure 4, using different band pass frequencies, contained higher filter durations than values previously tabulated which was probably due to the usage of MATLAB's filter toolbox which uses less robust techniques. Edge frequencies were calculated using a best initial guess representing the number of pulses. Previously generated data may have used more robust techniques for determining an optimal value, thereby producing a shorter filter duration. Additionally, the insensitivity of low-pass and notch filters was examined at different edge frequencies in order to determine the effect on filter duration. Results of this analysis yielded an increase in filter duration correlated with both an increase in edge frequencies as well as increasing insensitivity as shown in Figures 10 and 11.

Once the superiority of the input shaper was evaluated experimentally, the added effect of a predictor could be evaluated. A predictor was constructed which accurately predicts the final location of the tower crane motion. This predictor provides real-time predictor position as the tower crane is actuated by the operator, allowing the operator to more accurately determine the end location of the trolley and greatly reducing the likelihood of overshoot. The purpose of the last task was to create a predictor for the user interface such that later tests would be conducted to verify its utility. These tests will consist of a user navigating an obstacle course using an unshaped command, an input shaper, and an input shaper with a predictor. The improved usefulness of the predictor will be measured based on number of collisions, time required to navigate the course, final displacement from the target, and number of button presses. Should the predictor prove to provide faster navigation, it could greatly improve the efficiency of moving payloads to desired destinations in numerous real world applications.

REFERENCES

Bhat S. P. & Miu D. K., (1990) Precise point-to-point positioning control of flexible structures. *Journal of Dynamic Systems, Measurement, and Control*, 112(4), 667–674.

Singer N. C. & Seering W. P. (1990). Preshaping command inputs to reduce system vibration. *Journal of Dynamic Systems, Measurement, and Control*, 112(1), 76–82.

Singer N., Singhose W., & Seering W. (1999). Comparison of Filtering Methods for Reducing Residual Vibration. *European Journal of Control*, 5(2–4), 208–221.

Smith O. J. M. (1957). Posicast control of damped oscillatory systems. *Proceedings of the IRE*, 45(9), 1249–1255.

Vaughan J., Yano A., & Singhose W. (2007). Comparison of Robust Input Shapers. *Journal of Sound and Vibration*, 315(4–5), 797–815.