EXPERIMENTAL ANALYSIS FOR WIND INDUCED DISTURBANCE REJECTION ON BRIDGE CRANE STRUCTURES

Julia Pasternack
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
Julia.pasternack@gmail.com

Dan Thayer
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
dthayer14@gmail.com

Rohan Bansal
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
rohan@gatech.edu

Ehsan Maleki
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
ehsan.maleki@gatech.edu

Greg Tao
Mechanical Engineering
Massachusetts Institute of Technology
Atlanta, GA 30332
gregdtao@gmail.com

Kevin Kim
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
kk83kim@gmail.com

William Singhose
Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
singhose@gatech.edu

ABSTRACT

This paper describes the development of a feedback controller for rejecting wind disturbances on a crane payload. The goal of this research is to perform in-situ parameter sensitivity testing that is specific to a given crane structure. Through experimentation, this paper evaluates the robustness of a closed-loop controller that implements station keeping on a bridge crane. A set of general trends were developed to describe how altering system parameters and operating conditions affected controller performance. Such findings are not limited to single parameter characterization and can be extended to multiple variables given specific system specifications.

INTRODUCTION

Cranes operating in windy conditions run the risk of unwanted payload oscillation that present danger to their surroundings. Often times, the crane operators attempt to combat the issue manually (Omar, 2004). However, such operation can be difficult, especially when vision of the payload is limited. Attempts at combating unwanted sway through time optimal control have proven to be a challenge due to their sensitivity to system parameters (Manson, 1992) and (Auernig, 1987) and to disturbances. Thus, there exists a need for a closed-loop system that can effectively maintain both the desired position and minimal payload deflection under the effect of a disturbance on the payload. This approach is quite different than that which utilizes input shaping to reduce payload swing (Singer, 1990), (Khalid, 2005). Input shaping can successfully remove vibration induced by intended motion of the crane, but not from disturbances.

This paper discusses a closed-loop controller that implements station keep on a bridge crane. Experimental results are obtained using a portable crane (Lawrence, 2005). In particular, a set of trends for how varying system parameters and operating conditions affect the performance of the controller are reported. The experimental setup utilized in this work can also be used in future studies on bridge crane operation under user-specified wind conditions.

BACKGROUND

System Model

A bridge crane can be modeled as a simple pendulum on a cart, where the generalized displacements are the position of the trolley, x, and the angular deflection of the payload, θ, as shown in Figure 1.a. The trolley is modeled as a mass with an applied actuator force, F. The payload, which in this case is a sphere with moment of inertia, I, is suspended from a rigid and weightless string of length L. A disturbance force, D, is applied at the payload. It is assumed that the motion of the cart is unaffected by motion of the payload due to the large mechanical impedance in the drive system. Using this assumption, the model can then be partitioned into two sections, as shown in Figure 1.b. In this sectioned model, the trolley is no longer dependent on the motion of the payload, but the payload remains dependent on the motion of the trolley.

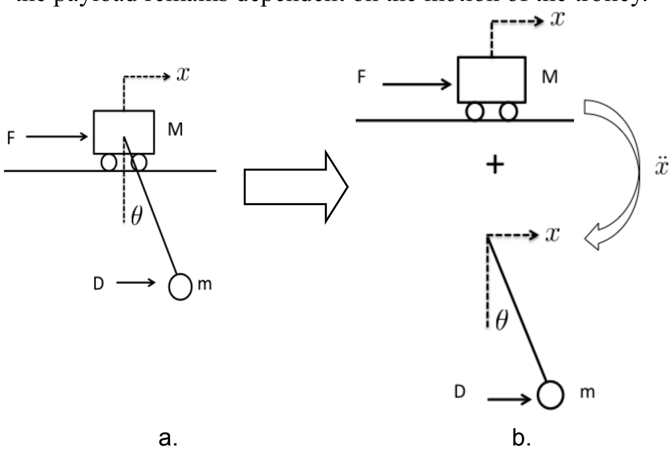

FIGURE 1. MODELS FOR BRIDGE CRANE ANALYSIS

The trolley motion is affected only by the actuator, resulting in the simple equation of motion.

$$\ddot{x} = \frac{F}{M} \tag{1}$$

The equations of motion for the pendulum system are derived using Lagrangian mechanics (Sorenson, 2007). The dynamics of the pendulum were influenced by θ, $\ddot{x}$, and F. Because the controller to be implemented would ideally keep the angular deflection of the payload small, these equations of motion can be linearized as:

$$\ddot{\theta} = \frac{\frac{D}{m} - \ddot{x} - g \cdot \theta}{L + \frac{I}{(m \cdot L)}} \tag{2}$$

Feedback Loop

The feedback controller is designed to achieve zero deflection, as well as maintain a desired trolley position. This controller consists of two feedback loops, shown in Figure 2. The outer feedback loop uses PID control to move the trolley to a desired position, x_d, ultimately serving to dampen vibration and eliminate steady-state error in x. The inner feedback loop uses PD control to push the angular deflection of the payload, θ, to zero. The choice of a PD controller stems from the fact that vibration reduction was the primary concern and eliminating steady-state error in theta is significantly less important than reaching a constant deflection. Also, an integral gain on the angle would likely introduce drift into the system in an attempt to minimize error if the disturbance force was a constant force. Furthermore, the overhead camera that senses the payload deflection angle always has some offset error that would also lead to drift under integral control.

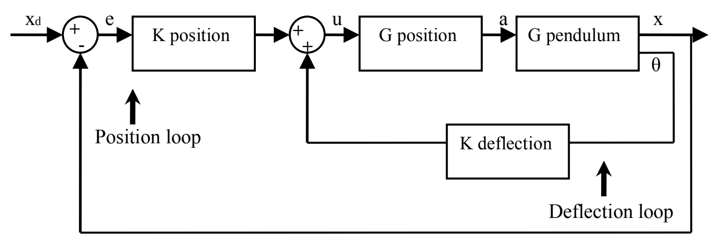

FIGURE 2. FEEDBACK LOOP SCHEMATIC

System modeling using SIMULINK/MATLAB

The system was simulated in MATLAB using Simulink. The Simulink model shown in Figure 3 implemented the two feedback controllers with a velocity command output, as was the case in the actual bridge crane used in the experiments. The velocity profile was saturated at the maximum trolley velocity and then differentiated and saturated again at the maximum

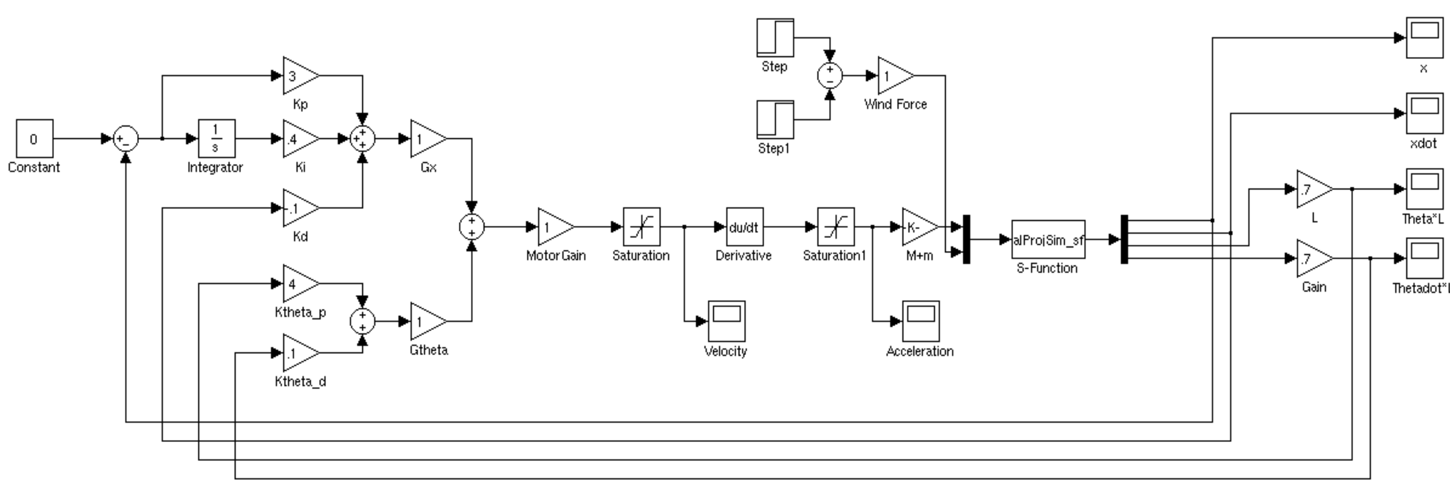

FIGURE 3. SIMULINK MODEL OF PENDULUM SYSTEM

trolley acceleration. This acceleration signal was then multiplied by the trolley mass in order to determine the force, F, required to drive the system through the desired velocity profile. Using (Co, 2004), an S-function block containing the equations of motion of the system was generated as the plant model. The Simulink model was used in order to determine the approximate PID gains to be utilized in the experimental controller. The system parameters in this Simulink model can be scaled up or down in order to test the chosen control scheme on different cranes.

EXPERIMENTAL PROCEDURE

Experimental Setup

The experimental setup involved a small, transportable bridge crane as shown in Figure 4. The portable bridge crane dimensions measure 1m in width by 1m in length by 1.6m in height. A Styrofoam ball served as the payload, as its spherical shape provided a uniform profile to the oncoming wind and its relatively light weight allowed the wind to cause a large disturbance. The setup for this experimentation is shown in Figure 4.

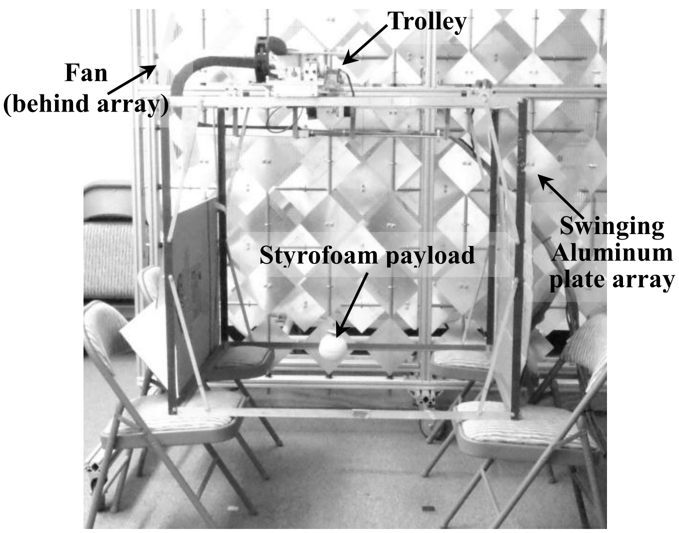

FIGURE 4: EXPERIMENTAL SETUP

The Styrofoam ball measured 10 cm in diameter and weighed 15 g before weighted washers were added to increase the mass of the payload. In order to increase the weight of the payload for some of the trials, a portion of the inside of the Styrofoam ball was carved out and filled with washers. The wind source was a 2.17 m airfoil diameter fan that had a 0.5 hp motor and a 180 RPM max speed setting (Big Ass Fans, 2010). The fan was capable of generating wind velocities of ~1 to 2 m/s and was used to simulate wind gusts through the opening and closing of an array of 19x19 cm aluminum plates that could be rotated at high speed by a series of stepper motors. Each of the aluminum plates could be controlled individually but for the purpose of this experiment, all plates were opened at one time and then closed at a later point in order to generate a uniform gust of wind from the fan over as large an area as possible.

A Siemens PLC generates a series of velocity setpoints for the motors. The crane is driven by Siemens AC synchronous servomotors that move the trolley and bridge axes via two timing belts. A direct-drive DC motor is used for hoisting. A Siemens digital camera is attached to the trolley to measure the payload swing (Lawrence, 2005). The crane can be operated through either a manual controller or through the use of a GUI, as shown in Figure 5.

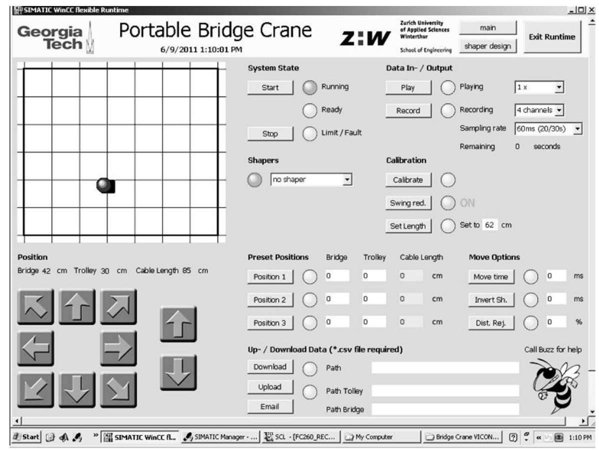

FIGURE 5: BRIDGE CRANE GUI

The upper left portion of the GUI shows a real-time animation of the crane from an overhead view using the camera and encoder measurements of the crane motion. The square is the trolley position and the circle is the hook position.

The objective of the control system was to reject a short burst of wind by returning the trolley back to its original position while simultaneously damping out payload oscillations. System robustness tests were carried out by varying three different system parameters: string length, payload mass, and wind disturbance pulse time.

The gains for both PID and PD controllers were experimentally determined for the middle cases of each of the three parameters, and were held constant throughout all trials. This set of controller gains is listed in Table 1.

TABLE 1. SYSTEM GAINS FOR ADVANCED EXPERIMENTATION

Payload K_P	1.0
Payload K_D	0.1
Trolley K_P	0. 1
Trolley K_I	0.001
Trolley K_D	0.01

The P, I, and D subscripts denote the proportional, integral, and derivative gain values respectively. There were a total of five variations for each of the three parameters. These values are listed in Table 2.

TABLE 2. PARAMETER VALUES FOR SYSTEM VARIATIONS

Disturbance Pulse Time (s)	Payload Mass (g)	String Length (cm)
2	18	75
3	28	80
4	**38**	**85**
5	48	90
6	58	95

For this study, three repeated trials were performed for each combination of parameters. In total, 45 trials were carried out on 15 different parameter configurations, keeping two out of the three parameters constant at values specified in bold font in Table 2.

The resulting data for both sets of trials were analyzed in order to determine the maximum overshoot and settling time of the payload deflection time responses.

RESULTS AND DISCUSSION

Many different cases for the Simulink model were run using the experimental system parameters described in Table 2 above. The wind force used in the Simulink model was determined using:

$$\vec{F}_d = -\frac{\rho v^2 c_d A}{2}\hat{v} \tag{3}$$

Where F_d is the drag force, ρ is the density, v is the relative speed of the payload, A is the surface area in contact, and c_d is the drag coefficient, which is 0.48 for a rough sphere. For the particular payload configuration used in this experiment, this wind force on the payload was approximated as 0.015 N. The results of the Simulink simulation for varying hoist lengths are shown in Figure 6.

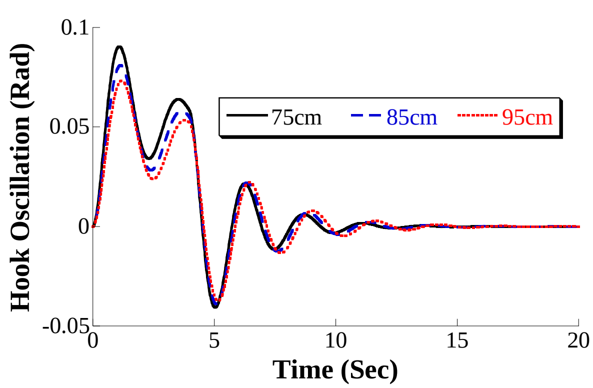

FIGURE 6.SIMULATED TRANSIENT RESPONSE FOR VARYING HOIST LENGTHS

In comparison to these simulation results, the time responses of the selected experimental results for varying hoist length are shown in Figure 7. Three trials are shown for each hoist length.

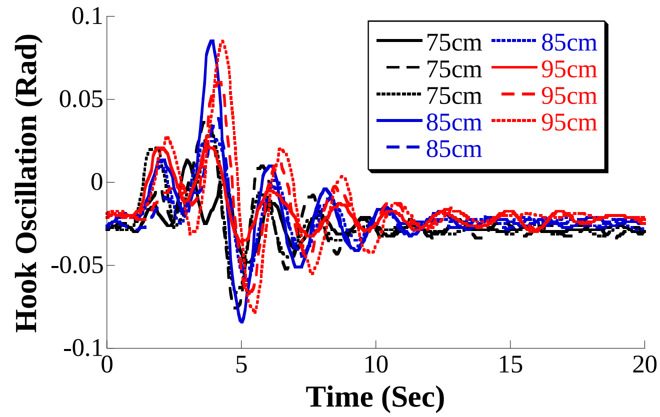

FIGURE 7.EXPERIMENTAL TRANSIENT RESPONSE FOR SELECTED HOIST LENGTHS

The results of the Simulink model share important similarities with the experimental results. However, there are noticieable differences because it is very difficult to construct an experimental setup that is capable of producing clean pulses. Furthermore, the bridge crane data acquisition process was performed separately from the wind-gust controller. Therefore, the setup was unable to provide precise synchronization between the timing of pulses and the timing of data acquisition, which resulted in slight delays between the separate trials.

Other differences stems from the fact that the exact transfer function of the bridge crane is not known. In the simulated model, a trolley velocity signal was used directly as the input, whereas for the actual crane, the velocity signal calculated by the PLC was transformed into a current signal as the input to the motors. The Simulink model assumed that the motor was capable of generating the exact velocity command it received, whereas the actual motor would likely have had some inherent lag. The optimal PID gains were also different in the simulated and experimental setups.

Regardless of these differences, both plots show the successful rejection of wind disturbance under a wide variety of parameter variations and thus demonstrate the robustness of the proposed controller. Furthermore, it should be noted that such behavior is evident when changing payload masses and wind pulse duration times, however such graphs have been omitted for the purpose of brevity.

In order to further evaluate the level of robustness to varying system parameters, the following sets of plots investigate controller response during experimentation. To evaluate performance attributes, both the maximum overshoot and settling time were evaluated. For this experimental setup, the settling time was designated to be within 1.6 degrees for the payload oscillation and 1.2 mm for trolley positioning. Each plot contains the median and mean of the values for the three trials run at each data point. In each case, the maximum overshoot of the hook/payload coincided with the highest deflection seen during the burst of wind. Figure 8 shows the overshoot of both the trolley (in terms of cm) and payload deflection (in terms of degrees).

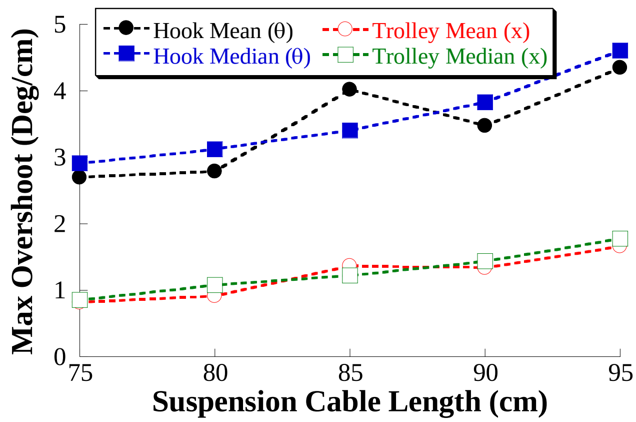

FIGURE 8. OVERSHOOT FOR VARYING HOIST LENGTHS

The maximum overshoot of the payload and, correspondingly, the trolley increases as the hoist length is increased, as shown in Figure 8. Though the increase in payload overshoot first appears counterintuitive, it occurs because as the hoist length increases, the trolley itself must travel farther (as indicated by its increasing overshoot values) and thus takes longer to catch up to the payload. In effect, this allows the payload to reach a larger deflection value for a system response to a non-ideal impulse of wind. Furthermore variation occurs simply because of the nature of the non-uniform wind distribution that is produced by the fan across the entire workspace. The controller produces a low overshoot over the approximately 20% variation in hoist length.

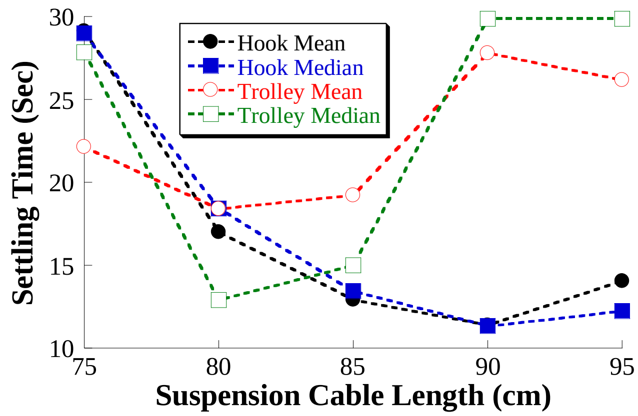

FIGURE 9. SETTLING TIME FOR VARYING HOIST LENGTHS

The plot of settling time for varying hoist lengths in Figure 9 shows that the set of PID gains chosen for the system favor settling the payload as fast as possible at the expense of the trolley settling time. The PID gains for the setup were chosen to get an optimal response at the middle set of parameters. The controller is capable of settling quickly for longer hoist lengths, but as the hoist length decreases, the natural frequency of the system increases which causes the controller's settling time to suffer.

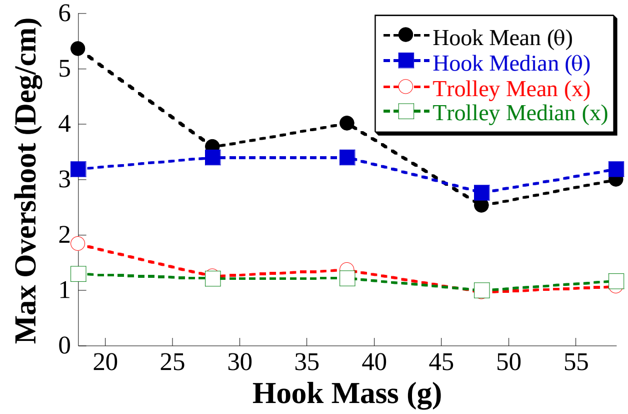

FIGURE 10. OVERSHOOT FOR VARYING PAYLOAD MASSES

The slight downward trend seen in Figure 10 is due to the decreased effect of the wind force as the mass of the payload is increased. It should be noted that the 5 degree payload overshoot caused by the wind on the lightest payload is unrealistically high and would likely not be seen except in highly treacherous wind conditions.

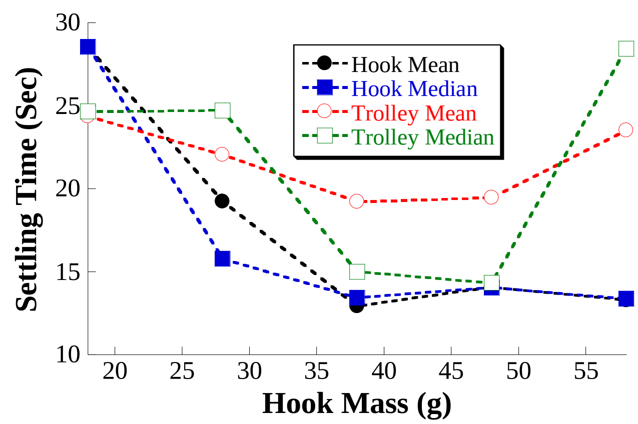

FIGURE 11: SETTLING TIME FOR VARYING PAYLOAD MASSES

The plot of the payload settling time in Figure 11 shows that the controller works well for a large payload mass but its performance degrades if the mass is low because the wind causes too much of an initial payload displacement.

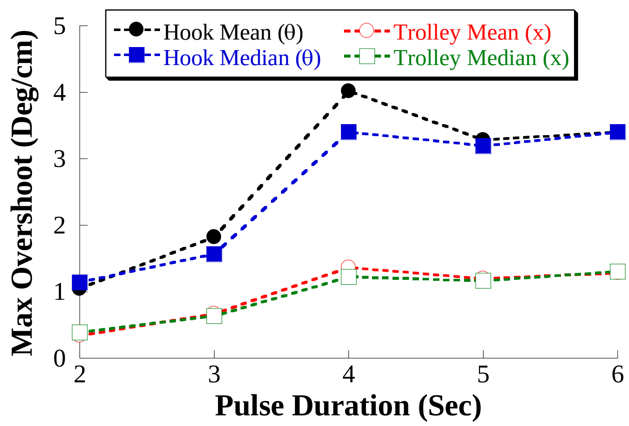

FIGURE 12: OVERSHOOT FOR VARYING WIND PULSE DURATIONS

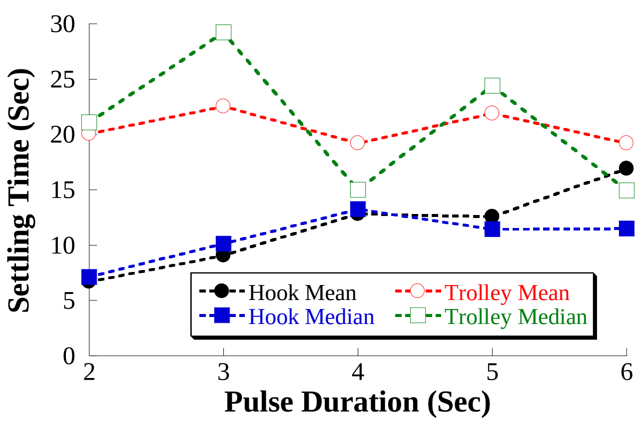

FIGURE 13: SETTLING TIME FOR VARYING WIND PULSE DURATIONS

Figures 12 and 13 show that the pulse duration of the wind gust can significantly change the overshoot of the payload but the controller performs well even for long gusts of wind.

CONCLUSION

This paper focused on the development of a control scheme for rejecting wind disturbances on the payload of a bridge crane. In order to develop the controller, a Simulink model was constructed that modeled the dynamics of the bridge crane under the dual loop control scheme chosen. An experimental setup was also constructed that allowed for the testing of crane operation under the effects of wind gusts.

The dual loop controller developed was capable of stabilizing the payload subject to a wind disturbance over a wide range of system parameters. The controller only presented poor performance results when there was an extremely light payload or a short suspension length. The wind force used on the lightest set of payloads caused an unrealistically high maximum payload deflection due to the scale of the setup. This situation would not occur on a full size crane unless it was operating under dangerously high wind levels. The difficulty seen with shorter hoist lengths is also somewhat realistic because the oscillation frequency of real cranes never approaches the extreme values tested in this project. Future work could include a study of how varying wind speed affects controller performance, as well as a study on wind disturbance rejection during the execution of a trajectory tracking task.

ACKNOWLEDGMENTS

The authors would like to thank Siemens Energy and Automation and the Manufacturing Research Center at Georgia Tech for their support of this work and for Dr. Mark Costello and Sam Sarovy for the use of their Indoor Flight Facility for the experimentation.

REFERENCES

[1] Omar, F., Karray, F., Basir, O., Yu, L., 2004. "Autonomous Overhead Crane System Using a Fuzzy Logic Controller," *J. Vib. Control,* vol. 10, pp. 1255-1270.

[2] Manson, G., 1992. "Time-optimal control of and overhead crane model", Optimal Control Applications & Methods, Vol. 3, No. 2, pp. 115-120.

[3] Auernig, J., Troger, H., 1987. "Time optimal control of overhead cranes with hoisting of the load", Automatica, Vol. 23, No. 4, pp. 437- 447.

[4] Co, T., 2004. "Short Tutorial on Matlab Part 5. Using S-function Blocks in Simulink". See also URL http://www.chem.mtu.edu/~tbco/cm416/MatlabTutorialPart5.pdf

[5] Sorensen, K., Singhose, W., Dickerson, S., 2007. "A controller enabling precise positioning and sway reduction in bridge and gantry cranes". *Control Engineering Practice* 15, pp. 825–837.

[6] Lawrence, J., Singhose, W., 2005. "Design of minicrane for education and research". *6th Int. Conference on Research and Education in Mechatronics,* Annecy, France.

[7] Singer, N., Seering, W., 1990. "Preshaping command inputs to reduce system vibration". *Journal of Dynamic Systems, Measurement, and Control,* 112, March, pp. 76–82

[8] Khalid, A., Singhose, W., Huey, J., Lawrence, J., Frakes, D., 2005. "Human operator performance testing using an input-shaped bridge crane". Accepted for publication in *Journal of Dynamic Systems, Measurement, and Control.*

ANNEX A

SIMULINK S CODE

```
%============================ FinalProjSimDot ===========================
%  function xdot = f(t, x)
%
%  Performs integration of the dynamics associated to FinalProjSim
%
%  xdot initialized to a zero column vector so that way it forces Matlab
%  to treat it as a column vector.
%============================ FinalProjSimDot ===========================
function xdot = FinalProjSimDot(t,x,f)

%System Parameters
g = 9.8;

M = 1000;               % [kg] Mass of the trolley
% m = 0.13;               % [kg] Mass of the payload
% L = .7;                 % [m]  Length of hoist
m = f(3);           % [kg] Mass of the payload
L = f(4);
b = 0;                  % [kg/s]damping

r = .01;                % Radius of payload sphere shell
I = (2/3)*m*(r^2);      % Moment of inertia of the hollow spherical payload
%I = (1/2)*m*(r^2);     % Moment of inertia of the payload

u=f(1);
d=f(2);
xdot = zeros(4,1);         % Initialize to zero column vector.

%x1 - position
%x2 - velocity
%x3 - angle
%x4 - angular velocity
```

```
a = u/(M+m);
alpha = (d/m-a-g*x(3))/(L+I/(m*L));

xdot(1) = x(2);                         % This is x dot.
xdot(2) = a;                            % This is x ddot.
xdot(3) = x(4);                         % This is theta dot.
xdot(4) = alpha;                        % This is theta ddot.

end

function  [sys,x0,str,ts] = FinalProjSim_sfnc(t,x,u,flag,xinit)
switch flag
    case 0  % initialize
        str=[]                  ;
        ts = [0 0]              ;
        s = simsizes            ;
        s.NumContStates = 4     ;
        s.NumDiscStates = 0     ;
        s.NumOutputs = 4        ;
        s.NumInputs = 4         ;
        s.DirFeedthrough = 0    ;
        s.NumSampleTimes = 1    ;
        sys = simsizes(s)       ;
        x0 = [xinit]            ;
    case 1                           % derivatives
        sys = FinalProjSimDot(t,x,u)      ;
    case 3  % output
        sys = x;
    case {2 4 9}                     % 2:discrete
                                     % 4:calcTimeHit
                                     % 9:termination
        sys =[];
    otherwise
        error(['unhandled flag =',num2str(flag)])   ;
end
```