Robust Control of an Inverted Pendulum on a Cart

Alexis Ball

ME/ECE 854 Robust Control
Final Project
Due: April 29, 2007

Abstract

This paper investigates the design and analysis of three controllers used to stabilize an inverted pendulum on a cart. This is accomplished by decomposing each control algorithm into two separate phases: swing-up control and stabilization control. A classical, $H_2$, and $H_{\infty}$ design are each considered to stabilize a nominal linearized model of the inverted pendulum on a cart in the upward vertical position. The effectiveness of each stabilizing design is evaluated through the use of the $\mu$ synthesis bound, as it applies to nominal performance, robust stability, and robust performance. Simulation results are provided to show the validity of each method.

I. INTRODUCTION

The inverted pendulum is a canonical example of a nonlinear system. The dynamics inherent in the model are often used to represent problems dealing with balance, such as bipedal walking as explored in [1], [2], and [3], unicycles [4], and the Segway Personal Transporters [5], due to the inherent instability of the system. Moreover, the inverted pendulum on a cart is often viewed as an under-actuated mechanical system, where the control inputs available are less than the degrees of freedom of the system. Thus, the natural complexity of this deceptively simple system makes it a prime candidate as a baseline study when investigating new control algorithms.

However, as with all systems, the models are never perfect. Thus, with the addition of model uncertainty paired with sensor noise and other unknown exogenous disturbances, the system previously thought to be stabilized can easily go unstable. The authors of [6] attempt to address the variability in system representation through the design of a controller using neutral networks. Although the approach is successful, no consideration is given to the possibility of outside disturbances. In [7], parametric uncertainties are dealt with via a combination of classic control theory and grey theory. In particular, the authors use a grey prediction model combined with a proportional plus derivative controller to balanced the inverted pendulum. Yet, the information received from the sensors is assumed to be flawless. Note that there are other possible methods and approaches that may be used to control an inverted pendulum, with
only a select few mentioned in this work. Moreover, the mathematical model and physical system itself
can also vary widely in the literature.

The goal of this body of work is to study the inverted pendulum on a cart within the context of model
uncertainty, sensor noise, and external disturbances, and provide a comparison between pole-placement,
\(H_2\), and \(H_\infty\) design methods as they relate to the overall system stability and performance. Section 2
fleshes out the details pertaining to the physical and mathematical description of the pendulum model used
in this study. This includes the error introduced into the model through the linearization of the nonlinear
system. The idea behind the use of two separate control algorithms and system representations to swing-up
the pendulum from rest to an upright position is addressed in Section 3; in particular, there is a swing-up
controller to be used initially and a stabilization controller to help maintain the pendulum at the vertical
upright position. Given the swing-up period is not the focus of this paper (requires no robust control
methods), the method will not be explored beyond Section 3. A entire section, Section 4, is dedicated
to providing the technical details relevant to the three separate methods used for stabilization, including
supplying the system form that lends itself to the robust control approach, performance requirements, and
the design methodology. Some preliminary closed-loop system results are provided, but not fully quantified
until Section 5. The success in terms of nominal performance, robust stability, and robust performance are
addressed using \(\mu\) synthesis in Section 5. An overall comparison between the three methods is also given.

II. PROBLEM FORMULATION AND SYSTEM DESCRIPTION

The system of interest is an inverted pendulum mounted on a motor driven cart. The rod mounted on
the cart is “free-moving” in the sense that there are no actuators dedicated to moving the pendulum alone.
Furthermore, both the cart and the pendulum are constrained to movement within the vertical plane. Any
movement that is experienced by the pendulum must be due to movement induced by the cart; this may
include outside disturbances such as wind or human interference (such as pushing the pendulum or cart) in
addition to the motion generated by the motors. However, all disturbances entering the system are assumed
to be bounded and have finite energy. This fact will later be utilized when designing the robust portion of
the control algorithm, such that all disturbances fit into a set of \(L_2\) disturbances (to be defined).

Using simple Newtonian physics and the free body diagrams shown in Fig. 1, we can begin to define
the dynamics of the inverted pendulum on a cart. It is assumed that the mass of the cart is \(M = 0.5\)kg and
the mass of the pendulum \(m = 0.5\)kg. The cart is subjected to a frictional force, where the coefficient of
friction is represented as \(b = 0.1\)N/m/sec. Naturally, this force is assumed to be in the opposite direction
of the applied force on the cart. The externally applied force \(F\) is assumed to be composed of the control
input and the external disturbances acting on the cart. The length of the pendulum to the center of the
mass and the inertia are represented by $L$ (or $l$) = 0.3 m and $I = 0.006$ kg·m$^2$, respectively. The variables of primary interest are the pendulum angle from the vertical $\theta$ and the cart position $y$. Note that the cart position is occasionally referred to as $x$. There is no appreciable difference between the two. Focusing on the free body diagram of the cart in Fig 1, we begin by summing the forces of the cart in the horizontal direction. This yields the following

$$M \ddot{y} + b \dot{y} + N = F$$ (1)

where $N$ is

$$N = m \ddot{y} + ml \ddot{\theta} \cos(\theta) - ml \dot{\theta}^2 \sin(\theta)$$ (2)

the sum of the forces in the horizontal direction of the pendulum. By substituting the representation for $N$ in (2) into (1), the following equation is obtained

$$M \ddot{y} + b \dot{y} + ml \ddot{\theta} \cos(\theta) - ml \dot{\theta}^2 \sin(\theta) = u$$ (3)

where $M = M + m$ and $u$ is the system input. Next, focusing on the sum of the forces perpendicular to the pendulum, the following equation of motion is obtained

$$P \sin(\theta) + N \cos(\theta) - mg \sin(\theta) = ml \ddot{\theta} + m \ddot{y} \cos(\theta)$$ (4)

where $P$ is pointing towards the ground, relative to the cart free body diagram in Fig. 1, originating at the point where the cart and the pendulum connect. That same point on the pendulum free body diagram is pointing up. Summing the moments around the centroid of the pendulum reveals

$$-P l \sin(\theta) - N l \cos(\theta) = I \ddot{\theta}$$ (5)
Equations (4) and (5) are used to eliminate the variables $P$ and $N$ and obtain the following representation

$$(I + ml^2) \ddot{\theta} + mgl \sin(\theta) = -ml\dot{y} \cos(\theta)$$

(6)

Thus, the dynamics of the inverted pendulum on a cart are represented by the differential equations shown in (3) and (6). Although this representation is acceptable when designing the swing-up control discussed in Section 3, a linear model is necessary for the stabilizing controller. Recall, that robust control theory is applicable to linear systems and not necessarily the nonlinear representation. Using small angle assumptions and linearizing the system about the equilibrium point $\theta = \pi$, the linear model shown in (7) and (8) is obtained.

$$M\ddot{y} + b\dot{y} - ml\dot{\phi} = u$$

(7)

$$(I + ml^2) \ddot{\phi} - mgl\phi (1 + \delta) = ml\dot{y}$$

(8)

Note that $\theta = \phi + \pi$, where $\phi$ is a small angle from the vertical upward direction. By replacing $\theta$ with $\phi$ for the system dynamics, the control objective can be formulated as a regulation problem. The small angle approximations utilized are $\sin(\theta) = -\sin(\phi) \approx -\phi(1 + \delta)$, $\cos(\theta) \approx -1$, and $\dot{\theta}^2 \approx 0$. The variable $\delta$ represents the uncertainty introduced via the linear approximation of an inherently nonlinear system.

In order to design the stabilizing controllers, the system uncertainty must be removed from (7) and (8) to achieve a nominal system representation without the noise. Note that if the noise was not removed, $\delta$ would appear as a parameter in the controller. This is not a wise decision, given that the exact value of $\delta$ is unknown; yet, it is known that $\delta$ will always satisfy the condition $|\delta| \leq 1$. However, it is convenient to place the system in the basic configuration of a linear fractional transformation, given this form is convenient when designing controllers robustly, and analyzing the resulting performance via $\mu$ synthesis. Thus, we will proceed to take the Laplace transform of the linear system given in (7) and (8), with the intent of separating out the model uncertainty by utilizing an upper linear fractional transformation. The Laplace transforms of (7) and (8), assuming the initial conditions are zero, are shown in (9) and (10), respectively.

$$MY(s)s^2 + bY(s)s - ml\Phi(s)s^2 = U(s)$$

(9)

$$(I + ml^2) \Phi(s)s^2 - mgl\Phi(s)(1 + \delta) = mlY(s)s^2$$

(10)

Solving for $Y(s)$ with (10) yields

$$Y(s) = \left[ \frac{I + ml^2}{ml} - \frac{g(1 + \delta)}{s^2} \right] \Phi(s)$$

(11)
Substituting (11) into (9) to solve for \( U(s) \) gives

\[
U(s) = \left[ \frac{M(I + ml^2)s^2}{ml} - Mg(1 + \delta) + \frac{b(I + ml^2)s}{ml} - \frac{bg(1 + \delta)}{s} - ml s^2 \right] \Phi(s) \quad (12)
\]

Multiplying (12) by \( s \) reveals that the order of the system is 3. Thus, dividing (12) by \( s^3 \) yields

\[
\Phi(s) = \frac{1}{\alpha s^2} U(s) - \frac{\beta_1}{s} \Phi(s) + \frac{\beta_2}{s^2} \Phi(s) + \frac{\beta_3}{s^3} \Phi(s) + \frac{\beta_4}{s^3} \Phi(s) \quad (13)
\]

where \( \alpha = \frac{M(I + ml^2) - mL^2}{ml}, \beta_1 = \frac{b(I + ml^2)}{ml\alpha}, \beta_2 = \frac{Mg}{\alpha}, \text{ and } \beta_3 = \frac{bg}{\alpha}. \) From (13), the block diagram shown in Fig. 2 is formed. According to Fig. 2, the new system inputs are \( u, v_1, \) and \( v_2. \) The system outputs are \( \phi, z_1, \) and \( z_2. \) Hence, (13) can be written as

\[
\Phi(s) = \frac{s}{\alpha(s^3 + \beta_1 s^2 - \beta_2 s - \beta_3)} U(s) + \frac{s}{s^3 + \beta_1 s^2 - \beta_2 s} V_1(s) + \frac{1}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} V_2 \quad (14)
\]

Equation (14) leads to the upper linear fractional transformation representation (15) shown graphically in Fig. 3.

\[
\begin{bmatrix}
z_1 \\
z_2 \\
\phi
\end{bmatrix}
= M
\begin{bmatrix}
\nu_1 \\
\nu_2 \\
u
\end{bmatrix}
\quad (15)
\]

where

\[
M_{11} = \begin{bmatrix}
\beta_2 s \\
\beta_3 s \\
\beta_3 s
\end{bmatrix}
\begin{bmatrix}
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} \\
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} \\
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}
\end{bmatrix},
M_{12} = \begin{bmatrix}
\beta_2 s \\
\beta_3 s \\
\beta_3 s
\end{bmatrix}
\begin{bmatrix}
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} \\
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} \\
\frac{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3}
\end{bmatrix}.
\]
\[ M_{21} = \begin{bmatrix} \frac{s}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} & \frac{1}{s^3 + \beta_1 s^2 - \beta_2 s - \beta_3} \end{bmatrix}, \text{ and } M_{22} = \frac{s}{\alpha (s^3 + \beta_1 s^2 - \beta_2 s - \beta_3)}. \]

Fig. 3. Linear Fractional Transformation Representation

In subsequent sections, it will be shown that the representation given in (15) is a key component in setting up the closed-loop system within the \( \mu \) synthesis framework.

III. CONTROL SCHEME OVERVIEW

The purpose of this section is to provide the reader with a general understanding of the control algorithm employed when bringing the pendulum from the downward to the vertical position, while remaining in the vertical for the remainder of time. Fig. 4 displays the various states of an inverted pendulum on a cart.

Fig. 4. States of the Inverted Pendulum on a Cart [8]

In this particular study, the interest lies in three distinct positions. In Fig. 4.a, the pendulum is considered to be stabilized at an inherently unstable equilibrium point. When the pendulum is in this position, the control is labeled as the stabilizing controller. When the system matches the position shown in Fig. 4.c, the pendulum is considered to be in the downward position. At this point, the swing-up control will be applied to move the pendulum to the upward position. With Figure 4.b, the pendulum is not quite stabilized nor completely in the downward position. This particular configuration will be referred to as the transitional period in-between the swing-up and stabilizing controllers.
A. Swing-up Control

It is assumed that the system starts at rest, as shown in Fig. 4.c, where the pendulum is in the stable equilibrium position. In order to stabilize the pendulum in the vertical upward position shown in Fig. 4.a, the pendulum must be brought near the vertical to activate the second stabilizing controller. One method discussed in [8] considers basing the movement of the cart, and therefore the pendulum, on the amount of energy present in the system. Thus, the goal of the controller is to supply the system with energy. The amount of energy applied at any given time corresponds to the disparity in the amount of energy currently in the system, and the amount present when the pendulum is near the vertical unstable equilibrium point. When the pendulum is “close enough” to the upright position, the stabilizing controller is switched on to maintain and balance the pendulum in the vertical upward position. The definition of “close enough” is heavily dependent on the control approach used when designing the stabilizing controller. In order to guarantee system stability in this pseudo-switched system, the control task can only switch from the swing-up controller to the stabilizing controller once the trajectories of the system (while influenced by the swing-up controller) are within the region of attraction of the system under the stabilizing controller. So long as this requirement is heeded, the system is ensured to remain stable throughout normal operation.

For convenience the system energy is defined to be zero when the pendulum is in the upright vertical position. Thus, the system energy is represented as

\[ E = mgl \left( \frac{1}{2} \left( \frac{\dot{\theta}}{\omega} \right)^2 + \cos(\theta) - 1 \right) \]  

where \( \omega = \sqrt{\frac{mg}{I}} \). Note that (16) is a function of the state variables \( \theta \) and \( \dot{\theta} \) instead of \( \phi \) and \( \dot{\phi} \) as defined in the latter equations of Section 2. This is due to the fact that in the swing-up control, the nonlinear model is invoked in the design process; recall that \( \phi \) and \( \dot{\phi} \) only appear in the linearized version of the inverted pendulum on a cart model. The control law that is implemented to input energy into the system in the swing-up phase is

\[ E_u = \text{sat}(2(E - E_d))\text{sign}(\dot{\theta} \cos(\theta)) \]  

where \( E_d \) is the energy of the inverted pendulum system when the pendulum is in the vertical upright position and “sat” is the saturation function intended to saturate the control at 3 volts. In simulation \( E_d \) is set to 0, but in practice this value could be slightly larger and needs to be tuned to each individual setup. Given this portion of the control is not the focus of the paper, no further analysis is provided.
B. Stabilizing Control

Once the pendulum has reached a region near the unstable equilibrium, the stabilizing controller takes over. The primary goal of the stabilizing control is to maintain the inverted pendulum in the upright orientation. This is achieved using three separate controllers that use either a pole-placement, $\mathcal{H}_2$, or $\mathcal{H}_\infty$ design approach. The particulars of each design is discussed in Section 4.

Given that the actuator, the cart motor, cannot instantaneously generated the torque necessary to move the cart, a time delay is imposed on all three controllers of the form

$$\frac{1}{\tau s + 1}$$

where $\tau$ is time constant for the transfer function in (18). Note that the value of $\tau$ is chosen relative to the time constant of the inverted pendulum on a cart system. Moreover, the control magnitude is limited to a value of 5 to ensure a fair comparison among all three control approaches.

C. Transition Control

Although the transition from the swing-up to the stabilizing controller should experience no instabilities, there is no guarantee that the system will remain stable if a switch is required from the stabilizing to the swing-up controller. If, for instance, the pendulum fails to satisfy the small angle approximations made when constructing the linear model (i.e. the pendulum angle $\phi$ diverges too far from the vertical), the pendulum could drop down and inadvertently reactivate the swing-up controller. To avoid exciting un-modeled dynamics and parametric uncertainties a smoothing algorithm was utilized to ensure a smooth transition from one control law to the next. This was done by averaging the outputs of the two controllers near the region where the control switching occurs. However, including this algorithm in the overall control scheme appears to provide no appreciable improvement to system performance. Thus, it has been removed from the control architecture.

IV. Stabilizing Control Design

Recall the upper linear fractional transformation representation given in Section 2 in (15). This form only includes the model uncertainty. However, when analyzing the success of the stabilizing controllers, it is required that the external disturbances as well as the sensor noise be included in the model. The diagram shown in Fig. 5 shows the closed-loop system in the context of all uncertainties affecting the system performance. The variable $d_i$ is the external disturbances acting on the inverted pendulum system. These disturbances could include wind or a human interacting with the pendulum, with the caveat that the disturbance energy is finite and belongs to a set of $L_2$ signals, such that the magnitude is bounded.
The variable $n$ represents sensor noise that is assumed to satisfy the condition $|n| \leq 1$. The red blocks, or “$W$” transfer functions represent the weighting matrices to be defined when designing the robust controllers. These matrices are used to shape the system response. The control is denoted by $K$. Using this visualization, the following configuration is obtained

$$
\begin{bmatrix}
  z_\delta \\
  e \\
  \hat{u} \\
  z 
\end{bmatrix} = P
\begin{bmatrix}
  v_\delta \\
  d_i \\
  n \\
  u 
\end{bmatrix}
$$

(19)

where $z_\delta$ is the signal from the plant $M$ to the set of model uncertainties $\delta$, and $v_\delta$ the signal from the model uncertainties to the plant $M$. The matrix $P$ is defined as

$$
P = \begin{bmatrix}
  M_{11} & M_{12}W_d & M_{12} \\
  W_eM & W_eM_{22}W_d & 0 \\
  0 & 0 & W_u \\
  M_{21} & M_{22}W_d & W_n \\
  M_{22} & W_n & M_{22} 
\end{bmatrix}
$$

(20)

Equation (19) can be graphically depicted in the general framework shown in Fig. 6. Following the notation given in Chapter 10 of [9], the $P$ matrix can also be represented by

$$
P = \begin{bmatrix}
P_{11} & P_{12} & P_{13} \\
P_{21} & P_{22} & P_{23} \\
P_{31} & P_{32} & P_{33} 
\end{bmatrix}
$$

Let the matrix $G$ be defined as

$$
G = \begin{bmatrix}
P_{22} & P_{23} \\
P_{32} & P_{33} 
\end{bmatrix}
$$

(21)
The matrix $G$ defined in (21) is the system representation used when designing the stabilizing controllers. Other portions of the $P$ matrix will be used when analyzing the system stability and performance under the selected controllers.

### A. Nominal Control Design

The form of the nominal controller, chosen to stabilize the inverted pendulum in the vertical upward direction, is the classical pole-placement algorithm. For ease of design, (21) is transformed into the following state-space representation

$$G = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix}$$

(22)

Utilizing the system representation in (22), the closed-loop system poles are chosen such that the settling time is approximately 10 seconds. This corresponds to dominant poles placed at $-0.3 \pm j0.52$. Note, all poles are chosen to reside in the open-left half plane to ensure closed-loop system stability. The nominal control design is denoted as $K_{nom}$.

### B. Robust Control Design

The robust control design consists of two separate methods: ${\mathcal{H}}_2$ and ${\mathcal{H}}_\infty$. Referring back to Fig. 5, it is now appropriate to define the weighting transfer functions to meet certain closed-loop system requirements. The magnitude frequency response of 3 out of the 4 weighting components is shown in Fig. 7. The value for the weighting component $W_i$ is chosen as 5, given $|d_i| \leq 5$, to satisfy the robust control criteria. The weight on the signal $u$, $W_u$, is chosen such that a large importance is placed on the control within the confines of the closed-loop system bandwidth, while placing essentially no penalty on the higher frequencies. This is done, because placing a large weight outside of the system bandwidth is only likely to amplify the noise and degrade system performance. Moreover, the system becomes difficult to control.
outside the closed-loop system bandwidth. The weighting function $W_n$ is chosen such that it models the frequency contents of the noise output by the sensors. It should be clear that by examining Fig. 7 that the sensors are assumed to have noise with primarily low frequency content. Weighting function $W_e$ is chosen to correspond with the closed-loop system bandwidth to ensure acceptable noise attenuation at low frequencies, and avoid amplifying the noise across the relevant frequency band. Before designing the robust controllers, note that the minimal realization of the transfer function $G$ in (22) is obtained using the MATLAB command `minreal` and is written in the following form

$$G = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & 0 & D_{12} \\ C_2 & D_{21} & 0 \end{bmatrix} \tag{23}$$

The $\mathcal{H}_2$ controller is designed using the MATLAB command `h2syn`, where the arguments are chosen as the the system realization in (23) and the input and output dimensions both as 1. The $\mathcal{H}_\infty$ controller is designed using the MATLAB command `hinfsyn`. The arguments entered into the function are the system realization in (23), the input and output dimensions both as 1, the lower bound on $\gamma$ as 0.001 and a upper bound of 50. The tolerance in the final $\gamma$ values is chosen as 0.01. The $\mathcal{H}_2$ and $\mathcal{H}_\infty$ controllers are denoted as $K_2$ and $K_\infty$, respectively.

C. Closed-loop System Response

The purpose of this section is to compare the closed-loop system response under the nominal, $\mathcal{H}_2$, and $\mathcal{H}_\infty$ controllers. This will serve as a qualitative cursory view of the system stability and performance, whereas the next section will investigate the results from a more quantitative approach with the use of $\mu$ synthesis. Figs. 8 and 9 show the pendulum angle and the amount of control required to stabilize the
pendulum in the vertical upward direction, respectively. The disturbance is taken as a gust of wind, modeled as an impulse response with a value of 0.1. Fig. 8 shows that for this simulation, the $\mathcal{H}_\infty$ controller appears to produce the least amount of variation in the angle from the vertical. The $\mathcal{H}_2$ controller is a close second, however, the system using the nominal controller experiences the greatest amount of angle variation. This is logical, given that the pole-placement algorithm has no guaranteed robustness against uncertainty in the system. Moreover, the non-minimum phase zero appears to have the greatest affect on the system performance for the nominal controller.

As for the controller effort, Fig. 9 shows that the $\mathcal{H}_\infty$ controller applied the largest magnitude of control, relevant to any one point in time. However, the overall control effort for the $\mathcal{H}_\infty$ controller is significantly less than that of the other two controllers. The $\mathcal{H}_2$ controller applies a slightly less amount of control than the nominal controller. Therefore, it appears as if the $\mathcal{H}_\infty$ design produced was the most successful controller in terms of minimizing angle variation and control effort.
When the magnitude of the wind disturbance is made closer to the upper bound of 5, the nominal and $\mathcal{H}_2$ controllers are unable to stabilize the inverted pendulum; this is regardless the amount of control effort. However, the $\mathcal{H}_\infty$ controller manages to not only stabilize the inverted pendulum, but do so without saturating the control at the actuator limit. For the sake of brevity, the plots are not shown here. Once again, the $\mathcal{H}_\infty$ controller appears to be the most successful.

V. NOMINAL AND ROBUST SYSTEM MEASURES

The purpose of this section is to investigate the nominal performance, robust stability, and robust performance of the system under the influence of each of the aforementioned controllers. The main mechanism to be utilized is the method of $\mu$ synthesis.

A. Nominal Performance

For a system to be nominally stable implies that the controller $K$ internally stabilizes the system, when uncertainty is absent. Nominal performance is achieved when the performance objectives of the system, without uncertainty, are reached. Given that satisfying the requirements of nominal performance implies nominal stability, only the former is investigated.

Before evaluating the nominal performance under each of the controllers, the system must be transformed into an acceptable representation. The MATLAB function `starp` is utilized to transform the representation shown in Fig. 6 into the linear fractional transformation shown in Fig. 3. Thus, the inputs to the function are the matrix $P$ defined in (20) and the controller of current interest; $K_{\text{nom}}$, $K_2$, and $K_\infty$ for the nominal, $\mathcal{H}_2$, and $\mathcal{H}_\infty$ controllers, respectively. The necessary inputs are zeroed. Next, the $M_{22}$ portion of the matrix output by the function `starp` is entered into the MATLAB function `hinfnorm` to calculate the infinity-norm of that component. If the following requirement is satisfied for all frequency values

$$\|M_{22}\|_\infty < 1 \quad (24)$$

then the system under that controller achieves nominal performance.

Using the criteria just described, all three controllers were found to satisfy the requirement of nominal performance and, in turn, nominal stability. This is to be expected, given the closed-loop system (regardless the controller) was shown to be stabilized in the absence of uncertainty.

B. Robust Stability

For a system to be robustly stable, means that the controller of interest is able to internally stabilize the system even when a defined uncertainty class is present. In order to evaluate the robust stability of the system under the influence of each controller, the following procedure must be completed. Once again,
the MATLAB function `starp` is utilized to transform the representation shown in Fig. 6 into the linear fractional transformation shown in Fig. 3. Thus, the inputs to the function are the matrix $P$ defined in (20) and the controller of current interest; $K_{\text{nom}}$, $K_2$, and $K_\infty$ for the nominal, $H_2$, and $H_\infty$ controllers, respectively. The necessary inputs are zeroed. Next, the $M_{11}$ portion of the matrix output by the function `starp` is entered into the MATLAB function `mu` to calculate the $\mu$ bound. The arguments entered into this function include $M_{11}$ and an array describing the perturbation block structure, denoted as “blk”. In this case, “blk” is defined as

$$\text{blk} = \begin{bmatrix} 1 & 0 \\ 2 & 2 \end{bmatrix}$$

(25)

Given this system satisfies the necessary uncertainty dimension requirements listed in [9], the output of this function is defined to be the bound on $\mu$. If the following requirement is satisfied

$$\mu_\delta(M_{11}) < 1, \forall \omega$$

(26)

then the system under that controller is considered to be robustly stable.

Using the criteria just described, only the two robust controllers were found to be robustly stable. This is logical, given the nominal controller fails to take any uncertainty into account during the design phase. Note that robust stability requires that the system be stabilized for the entire class of uncertainties define. Thus, showing that the nominal controller is able to stabilize the system in the presence of some uncertainties does not guarantee robust stability (as shown by the simulation provided in Section 4). The $\mu$ bounds for the system under the nominal, $H_2$, and $H_\infty$ controllers were 3.23, 0.54, and 0.38, respectively.

C. Robust Performance

For a system to achieve robust performance, the performance objectives must be met even when a particular class of uncertainties is present in the system. If the system satisfies the requirements of robust performance, that implies that it is also robustly stabilized. Similar to the previous two cases, the system must be transformed into a robust control friendly representation. The MATLAB function `starp` is used to transform the representation shown in Fig. 6 into the linear fractional transformation shown in Fig. 3. Thus, the inputs to the function are the matrix $P$ defined in (20) and the controller of current interest; $K_{\text{nom}}$, $K_2$, and $K_\infty$ for the nominal, $H_2$, and $H_\infty$ controllers, respectively. The necessary inputs are zeroed. The matrix $M$, which is the output from the function `starp`, is then entered into the MATLAB function `mu` to calculate the $\mu$ bound. The arguments entered into this function include $M$ and an array describing the perturbation block structure, denoted as “blk”. The “blk” array is as defined in (25). If the
following requirement is satisfied

\[ \mu_\delta(M) < 1, \forall \omega \]  \hspace{1cm} (27)

then the system is said to meet the requirements of robust performance. Only the \( H_\infty \) controller was able to satisfy the requirement given in (27). The \( \mu \) bounds on both the nominal and \( H_2 \) controllers exceeded the required value of 1. However, the \( H_\infty \) controller managed to produce a \( \mu \) bound of approximately 0.87.

VI. CONCLUSION

This paper investigated the design and analysis of three controllers used to stabilize an inverted pendulum on a cart. This was accomplished by decomposing each control algorithm into two separate phases: swing-up control and stabilization control. A classical, \( H_2 \), and \( H_\infty \) design are each considered to stabilize a nominal linearized model of the inverted pendulum on a cart in the upward vertical position. All three controllers were found to meet the requirements of nominal stability and nominal performance. However, only the robust controllers were able to achieve robust stability. Although all controllers were able to stabilize the inverted pendulum in the vertical upward direction for some limited uncertainty, not all uncertainties chosen within the acceptable set of finite energy bounded disturbances resulted in a stabilized system. The \( H_\infty \) controller was shown to achieve robust performance, whereas the other controllers allowed the system to go unstable in the presence of large exogenous disturbances. Overall, a controller designed with uncertainty in mind will not always be able to guarantee robust performance. This point rings particularly true when considering that there are no guaranteed stability margins when designing a \( H_2 \) controller.

REFERENCES