



Preinforme:

Control Predictivo Basado en Modelos (MPC).

EL5205-1 Laboratorio de Control Avanzado

30 de octubre, 2015

Profesores:

Doris Saez.
Manuel Duarte.
Marcos Orchard.

Auxiliar:

David Acuña.

Integrantes Grupo 1:

Matías Pavez.
Pablo Saavedra.



[Página dejada intencionalmente en blanco]

Índice

1. Introducción	2
2. Marco Teórico	3
2.1. Representación en variables de estado	3
2.1.1. Tiempo Continuo	3
2.1.2. Tiempo Discreto	3
2.1.3. Discretización de sistemas LTI continuos	4
2.2. Observabilidad y Controlabilidad	5
2.2.1. Observabilidad	5
2.2.2. Controlabilidad	5
2.3. Control Predictivo basado en Modelos (MBPC)	6
3. Procedimientos para el Laboratorio	8
3.1. Descripción del Sistema	8
3.2. Discretización de subsistemas	9
3.3. Tower: Diseño de controlador predictivo	10
3.4. Tower: Implementación en Simulink	11
3.5. Tower: Estudio del sistema	13
3.5.1. Pruebas sobre parámetros de diseño	13
3.5.2. Pruebas para reducción de Oscilaciones	13
3.6. Jib: Diseño de controlador predictivo	14
3.7. Jib: Implementación en Simulink	15
3.8. Jib: Estudio del sistema	17
3.8.1. Pruebas sobre parámetros de diseño	17
3.8.2. Pruebas para reducción de Oscilaciones	17
3.9. Controladores acoplados	17
4. Referencias	19

1. Introducción

El control predictivo basado en modelos (MBPC) tiene por objetivo resolver de forma efectiva, problemas de control y automatización de procesos que se caracterizan por presentar comportamientos dinámicos complejos, multivariable y/o inestable. El control predictivo permite abordar problemas estocásticos, procesos con retardo de tiempo, multivariabes e incluir restricciones.

En ésta actividad se trabajará con un sistema de grúa presente en el laboratorio de automática[1], como el presentado en la figura 1. El estudio y diseño de controladores para la grúa tiene un interés práctico fundamental en áreas como la construcción, donde se espera movilizar grandes cargas rápidamente, evitando gastos excesivos de energía (es decir, esfuerzo de control), de manera precisa y minimizando las oscilaciones de la carga, para así, evitar accidentes y optimizar los recursos disponibles.

Dada la naturaleza no-lineal y acoplada de los componentes del sistema, el problema será abordado en tres etapas. En primer lugar, se analizará y diseñarán MPCs para el subsistema *Tower*, el cual representa el movimiento angular de la grúa. Luego, se estudiará el sistema *Jib*, que representa el movimiento traslacional de la carga. Finalmente, se estudiará el comportamiento de ambos sistemas trabajando en conjunto.



Figura 1: Sistema “3 DOF Crane” de ©QUANSER Inc. Planta a utilizar en la experiencia de laboratorio, compuesta por un brazo rotatorio, corredera lineal y carga suspendida de la corredera.

2. Marco Teórico

2.1. Representación en variables de estado

Formalmente, se define como vector de estados $x(t)$ del sistema, el conjunto mínimo de parámetros que dada una condición inicial $x(t_0)$ y la entrada $u(t)$, permiten caracterizar la salida $y(t)$ de un sistema dinámico, para todo $t \geq t_0$.

2.1.1. Tiempo Continuo

Un sistema lineal en variables de estado de tiempo continuo se suele modelar por las ecuaciones (1) y (2), donde si $x(t)$ es de dimensión n , $u(t)$ de dimensión m , e $y(t)$ de dimensión q , entonces $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{q \times q}$ y $D \in \mathbb{R}^{q \times m}$ son matrices que representan el sistema dinámico, mediante relaciones lineales entre sus estados, entradas y salidas.

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (1)$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad (2)$$

Si además el sistema es invariante en el tiempo, se tendrá que A , B , C y D son matrices constantes.

2.1.2. Tiempo Discreto

Para el caso de tiempo discreto, la representación en variables de estado está basada en ecuaciones de diferencias, que permiten describir la dinámica del sistema. Además, se trabaja con una frecuencia de muestreo fija T_s .

De manera análoga al caso continuo y bajo las suposiciones de que la planta es un sistema LTI (Linear Time Invariant), se tendrá una representación discreta equivalente dada por las ecuaciones (3) y (4), para el estado del sistema $x(t)$ y la salida $y(t)$. Además, dado que se muestrea en intervalos fijos de tiempo, se suele trabajar con el cambio de variable $t = k \cdot T_s$, donde t es el tiempo transcurrido y k es la numeración del intervalo de tiempo actual.

$$x[k+1] = Ax[k] + Bu[k] \quad (3)$$

$$y[k] = Cx[k] + Du[k] \quad (4)$$

2.1.3. Discretización de sistemas LTI continuos

Dado que la experiencia requiere discretizar el sistema continuo, para luego implementar un controlador discreto, esta sección explica la transformación a realizar. Los controladores suelen ser implementados discretamente en la práctica, utilizando un retenedor de orden cero (ZoH) antes de pasar a la planta, el que se encarga de generar una señal continua a partir de la salida discreta del controlador. En la Figura 2 se presenta la respuesta de un ZoH ante una señal discreta.

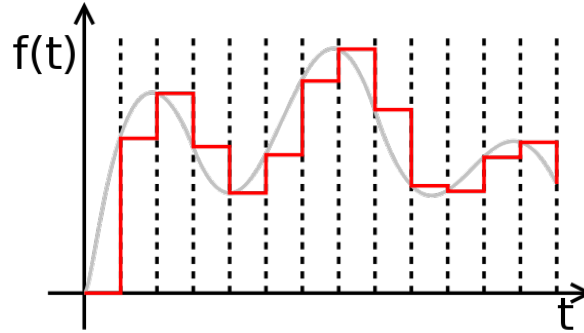


Figura 2: Función Zero-Order-Hold (ZoH). En claro, se presenta la señal original muestreada a intervalos regulares, en rojo se presenta la respuesta del ZoH.

Para diseñar controladores discretos se debe tomar en cuenta el efecto del ZoH en la respuesta de la planta, lo que se realiza considerando el subsistema (ZoH-Planta) como un todo. Un método para ello, es usar la aproximación de Tustin, lo que requiere conocer la función de transferencia del sistema, que en este caso no se posee.

Otro método consiste en discretizar directamente el sistema continuo en variables de estado, mediante la transformación presentada en la ecuación (5), donde A_d y B_d se calculan mediante las expresiones (6) y (7). La representación generada es el equivalente discreto de la representación en tiempo continuo de la ecuación (1), considerando un tiempo de muestreo T_s constante [2].

$$x[k+1] = A_d x[k] + B_d u[k] \quad (5)$$

$$A_d = e^{AT_s} \quad (6)$$

$$B_d = \int_0^{T_s} e^{A\lambda} B d\lambda \quad (7)$$

Esta discretización se puede realizar en MATLAB mediante el comando `c2d`, que al especificar el sistema en variables de estado y T_s , retorna su versión discretizada.

2.2. Observabilidad y Controlabilidad

La observabilidad y controlabilidad de un sistema son aspectos duales de un mismo problema. Garantizar la observabilidad de un sistema permitirá saber si es posible generar observadores para todos los estados internos de él, y por otro lado, la controlabilidad nos garantiza que es posible construir un controlador para llevar al sistema a un estado deseable.

2.2.1. Observabilidad

Un sistema es observable, si se pueden estimar sus estados en un tiempo finito y sólo a partir de sus entradas y salidas. En sistemas lineales la propiedad de observabilidad puede ser comprobada a partir de la matriz de observabilidad O , presentada en la expresión (8), donde se debe verificar que su rango sea igual a la cantidad de variables de estado n , o análogamente, si O tiene n filas linealmente independientes, el sistema es observable.

$$O = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (8)$$

2.2.2. Controlabilidad

Un sistema es controlable, si es posible llevarlo desde un estado particular a otro en el espacio de estados, mediante una entrada admisible y en un tiempo finito, o análogamente, que siempre exista una entrada que permita cambiar desde un estado a cualquier otro en un tiempo acotado. En los sistemas lineales la propiedad de controlabilidad puede ser comprobada a partir de la matriz de controlabilidad R , presentada en la expresión (9), de forma que R tiene un rango igual a la cantidad de variables de estado n , o dicho de otra forma, R tiene n filas linealmente independientes, el sistema es controlable.

$$R = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B] \quad (9)$$

2.3. Control Predictivo basado en Modelos (MBPC)

El Control Predictivo Basado en Modelos (MBPC o MPC, por su nombre en inglés "Model (Based) Predictive Control") es una herramienta de control que permite incorporar criterios de operación de la planta a través de una función objetivo y restricciones sobre las acciones de control. MBPC utiliza modelos matemáticos del proceso para poder predecir el comportamiento de las variables involucradas, en cuanto al horizonte de predicción este tiene que ver con la naturaleza de la planta y los objetivos de control que se requieren aplicar sobre ella. La idea es establecer una trayectoria deseada futura o referencia para las variables que se desean controlar de forma que la el control se realiza manipulando las variables controladas optimizando una función de costo, la aplicación de la política de control se realiza sobre un horizonte móvil, lo que implica que en cada instante se debe realizar una predicción futura según el horizonte seleccionado y sobre esta predicción se realiza una operación de optimización sobre las variables manipuladas para obtener la acción de control. En la figura 3 se puede ver un esquema general del funcionamiento de MBPC.

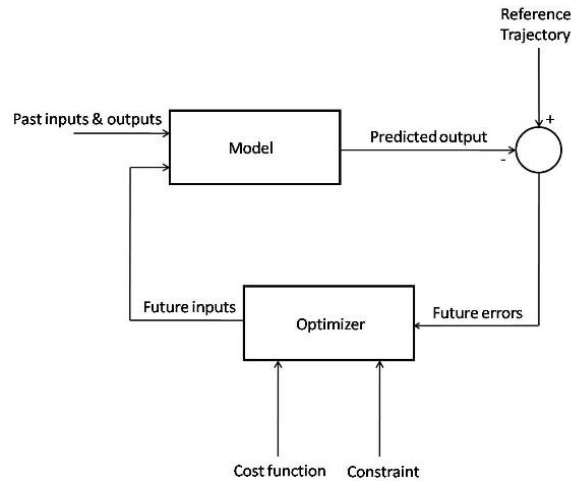


Figura 3: Esquema de como trabaja el Control Predictivo Basado en Modelos MBPC.

Los elementos esenciales de MBPC son los modelos del proceso, la función objetivo y la obtención de la ley de control. Un buen modelo del proceso debe asegurar un comportamiento similar a la realidad de tal forma que las predicciones que se realicen con ella sean certeras, de lo contrario a partir de este punto el control sería impredecible. El modelo representa la relación entre las entradas y las salidas medibles de un proceso, incluso el modelo puede considerar las variables no medibles como perturbaciones o ruidos para intentar compensar los errores de modelación. Los modelos más utilizados son los autoregresivos (estocásticos), función de transferencia, por respuesta al impulso y escalón, espacio de estados, etc.

En cuanto la función objetivo en general es del tipo cuadrática debido a su simpleza y a la propiedad de ser convexa de forma que se garantiza un mínimo global, de todas formas esta función se selecciona de forma tal que permita minimizar ciertas características sobre el proceso por ejemplo la energía de la señal de entrada o salida, seguimiento de referencia, interacción entre la entrada o

salida, etc. Una función de costo típica es la presentada en la ecuación (10)

$$J = [\hat{y}(t+1|t) - r(t+1)]^2 + \lambda[\Delta u(t)]^2 \quad (10)$$

Donde $r(t)$ es la referencia, $y(t)$ la salida de la planta y $u(t)$ es la salida de la planta. En el caso de la función de costo J (ver ecuación (10)) intenta minimizar el error entre la salida y la referencia y el la variación de la acción de control $\Delta u(t)$, donde este termino es ponderado por λ de forma que esta constante permite definir que tan importante es en el proceso de optimización el termino de variación de la entrada respecto al error de seguimiento de referencia. En cuanto al proceso de optimización este depende del tipo de modelo utilizado y las restricciones sobre las variables utilizadas en el proceso. Una forma típica de trabajar para obtener una acción de control para distintos tipos de modelos es transformar el modelo a una forma en donde se separe la respuesta forzada y la respuesta libre, ver tabla de la figura 4. La respuesta libre es la respuesta del sistema que depende solo de datos pasados en el caso de la respuesta forzada depende de datos de entrada y salida que todavía no se tienen y que deben ser calculadas.

Tipo de modelo	Futuro	Pasado
Modelo respuesta al escalón	$\sum_{i=1}^j g_i \cdot \Delta u(t+j-i)$	$\sum_{i=1}^N [g_{j+i} - g_i] \cdot \Delta u(t-i) + y(t)$
Modelo ARIMAX $C \neq 1$	$H_j \cdot \Delta u(t+j-i)$	$\frac{K_j}{C} \Delta u(t-1) + \frac{F_j}{C} y(t)$
Modelo en variables de estado	$\sum_{i=1}^j A^{j-i} B D \cdot \Delta u(t+i-1)$	$D A^{j-1} (A - G D) \hat{x}(t t) + D A^{j-1} G y(t)$
	Respuesta forzada $y_f(t+j t)$	Respuesta libre $y_l(t+j t)$

Figura 4: Estructura de predictores más típicos.

Para el diseño de controladores MPC, una opción es utilizar el toolbox *Model Predictive Control* de MATLAB, que provee un bloque en simulink capaz de utilizar diversas funciones de costo (en particular, funciones como la de la expresión (10)), permite diseñar controladores mediante la herramienta `mpctool`, especificar restricciones sobre entradas y salidas, entre otros. Por otro lado un aspecto importante al momento de diseñar el control MBPC es la determinación de la constante λ , el cual permite sintonizar la forma en que se optimiza la función de costo y por tanto permite variar la acción de control.

3. Procedimientos para el Laboratorio

3.1. Descripción del Sistema

El sistema con que se trabajará corresponde a la grúa “3 DOF Crane” desarrollada por la empresa ©QUANSER Inc.[3] y presentada en la Figura 1, la que está conformada por un brazo rotatorio (denominado *Tower*), una corredera lineal horizontal (denominada *Jib*) y una carga suspendida del Jib (denominada *Payload*). De los 3 subsistemas descritos, por simplicidad sólo se trabajará con Tower y Jib, dejando fuera a Payload.

Cada subsistema está compuesto por una entrada, que corresponde a la corriente eléctrica que circula por el motor involucrado para realizar un movimiento. Además, existen múltiples salidas que miden el ángulo θ de Tower o la distancia x del Jib en la que se encuentran las variables involucradas en cada subsistema. Otras salidas importantes del sistema son los ángulos correspondientes al bamboleo que sufre la carga al manipular la grúa. Éstos dos ángulos son medidos en cada instante y muestran la oscilación de la carga tanto en dirección radial (γ), como transversal (α), tomando como referencia la torre de la grúa.

En cuanto a las salidas del sistema ($\theta, \alpha, x, \gamma$), éstas son medidas por encoders ópticos de alta resolución, teniendo valores de $8.79 \times 10^{-4}[deg]$ para θ , $8.38 \times 10^{-7}[m]$ para x y de $0.0879[deg]$ para α y γ . En cuanto a las entradas del sistema, el motor de Tower acepta hasta $0.962[A]$, mientras que se tiene un límite de $0.79[A]$ para el Jib. Por último, se sabe que el largo total del Jib es de $121[cm]$, siendo su rango válido de movimiento menor a éste límite, mientras que Tower tiene un rango de movimiento aproximado de $\pm 155[deg]$. Ambos subsistemas cuentan con interruptores por hardware, encargados de limitar el movimiento de los actuadores respectivos a sus límites expuestos.

Tomando en cuenta un modelamiento en espacio de estado para ambos subsistemas, como el presentado en la sección 2.1.1, se definen los estados x_t y x_j según las expresiones (11) y (12) respectivamente. Las matrices correspondientes a cada sistema continuo se presentan en las expresiones (13-16) para Tower y en (17-20) para el Jib. Por último, se sabe que el tiempo de muestreo es de $T_s = 0.01[s]$.

$$x_t(t) = (\theta \quad \alpha \quad \dot{\theta} \quad \dot{\alpha})^T \quad (11)$$

$$x_j(t) = (x \quad \gamma \quad \dot{x} \quad \dot{\gamma})^T \quad (12)$$

$$A_t = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1.3108 & 0 & 0 \\ 0 & -12.5835 & 0 & 0 \end{pmatrix} \quad (13)$$

$$B_t = \begin{pmatrix} 0 \\ 0 \\ 1006.5290 \\ -939.9211 \end{pmatrix} \quad (14)$$

$$C_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (15)$$

$$D_t = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (16)$$

$$A_j = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -1.7019 & 0 & 0 \\ 0 & -13.3301 & 0 & 0 \end{pmatrix} \quad (17)$$

$$B_j = \begin{pmatrix} 0 \\ 0 \\ 18.2478 \\ 21.1299 \end{pmatrix} \quad (18)$$

$$C_j = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (19)$$

$$D_j = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (20)$$

Con lo que se tienen los subsistemas para Tower y Jib, presentados en las expresiones (21) y (22), respectivamente.

$$\begin{aligned} \dot{x}_t(t) &= A_t x_t + B_t u_t \\ y_t(t) &= C_t x_t + D_t u_t \end{aligned} \quad (21)$$

$$\begin{aligned} \dot{x}_j(t) &= A_j x_j + B_j u_j \\ y_j(t) &= C_j x_j + D_j u_j \end{aligned} \quad (22)$$

3.2. Discretización de subsistemas

Para discretizar los sistemas Tower y Jib de las expresiones (21) y (22) se utiliza la función `c2d()` de MATLAB, con la opción de “Zero Order Hold”, como se explica en la sección 2.1.3 y

tomando en cuenta el tiempo de muestreo T_s del sistema. Luego, se obtienen los sistemas discretos presentados en las expresiones (23) y (24) para los subsistemas respectivos.

$$x_t[k+1] = \begin{pmatrix} 1 & 6.5533 \times 10^{-5} & 0.01 & 2.1845 \times 10^{-7} \\ 0 & 0.9993 & 0 & 0.0099 \\ 0 & 0.0131 & 1 & 6.5533 \times 10^5 \\ 0 & -0.1258 & 0 & 0.9993 \end{pmatrix} x_t[k] + \begin{pmatrix} 0.0503 \\ -0.04700 \\ 10.06500 \\ -9.3972 \end{pmatrix} u_t[k] \quad (23)$$

$$y_t[k] = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} x_t[k]$$

$$x_j[k+1] = \begin{pmatrix} 1 & -8.5086 \times 10^{-5} & 0.01 & -2.8363 \times 10^7 \\ 0 & 0.9993 & 0 & 0.0100 \\ 0 & -0.0170 & 1 & -8.5086 \times 10^{-5} \\ 0 & -0.1333 & 0 & 0.9993 \end{pmatrix} x_j[k] + \begin{pmatrix} 0.0009 \\ 0.0011 \\ 0.1825 \\ 0.2113 \end{pmatrix} u_j[k] \quad (24)$$

$$y_j[k] = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} x_j[k]$$

3.3. Tower: Diseño de controlador predictivo

Según el método explicado en la sección 2.2, se calculan las matrices de controlabilidad y observabilidad para el sistema Tower discreto, las que se presentan en las expresiones (25) y (26), respectivamente. Ambas matrices son de rango 4, por lo que Tower es un sistema Observable y Controlable.

$$C_O = \begin{pmatrix} 0.0503 & 0.1510 & 0.2516 & 0.3522 \\ -0.0470 & -0.1409 & -0.2347 & -0.3281 \\ 10.065 & 10.0640 & 10.0610 & 10.0580 \\ -9.3972 & -9.3854 & -9.3618 & -9.3264 \end{pmatrix} \quad (25)$$

$$O_O = \begin{pmatrix} 1 & 0 & 0.00 & 0 \\ 0 & 1 & 0.00 & 0 \\ 1 & 6.5533 \times 10^{-5} & 0.01 & 2.1845 \times 10^{-7} \\ 0 & 0.9993 & 0.00 & 0.0100 \\ 1 & 0.0003 & 0.02 & 1.7473 \times 10^{-6} \\ 0 & 0.9975 & 0.00 & 0.0200 \\ 1 & 0.0006 & 0.03 & 5.8953 \times 10^{-6} \\ 0 & 0.9943 & 0.00 & 0.0299 \end{pmatrix} \quad (26)$$

La función objetivo del controlador predictivo es la presentada en la ecuación (27).

$$\begin{aligned}
 J_t = & \sum_{j=N_1}^{N_2} \lambda_1 (x_1(t+j) - r_1)^2 + \sum_{j=N_1}^{N_2} \lambda_2 (x_3(t+j) - r_3)^2 + \sum_{j=1}^{N_u} \lambda_u \Delta u(t+j-1) \\
 \text{s.a.} \quad & -3 < x_2 < 3 \text{ [deg]}
 \end{aligned} \tag{27}$$

Entonces, como se explica en la sección 2.3, se diseña un controlador MPC para la planta discreta Tower, mediante la herramienta `mpctool`, donde se consideran las siguientes configuraciones:

- Se utiliza un horizonte de predicción a 60 pasos y uno de control a 40 pasos.
- Se agrega la restricción sobre la variable $x_2 = \alpha$

3.4. Tower: Implementación en Simulink

En la Figura 5 se presenta el modelo de simulink que contiene el subsistema Tower discreto y el bloque de MPC correspondiente. En 6 y 7 se presentan los resultados de las simulaciones del controlador diseñado mediante la herramienta `mpctool` y siguiendo la configuración propuesta es las especificaciones de la experiencia. El controlador presenta un buen desempeño a simple vista, pues logra seguir la referencia deseada en un tiempo corto, pero presenta oscilaciones periódicas en la carga, las que en el laboratorio real deberían decaer en el tiempo por motivos de disipación de la energía del péndulo.

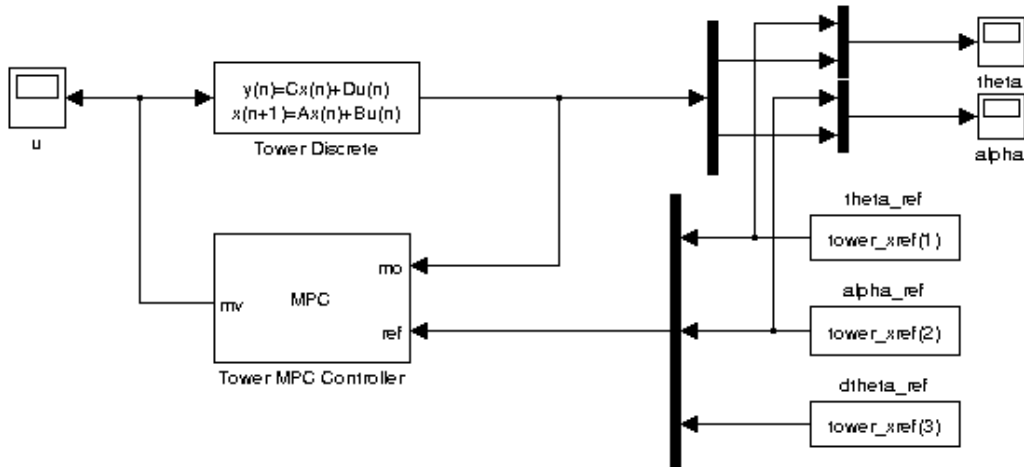


Figura 5: Implementación en simulink de sistema Tower discreto junto a controlador MPC perteneciente al toolbox *Model Predictive Control* de MATLAB.

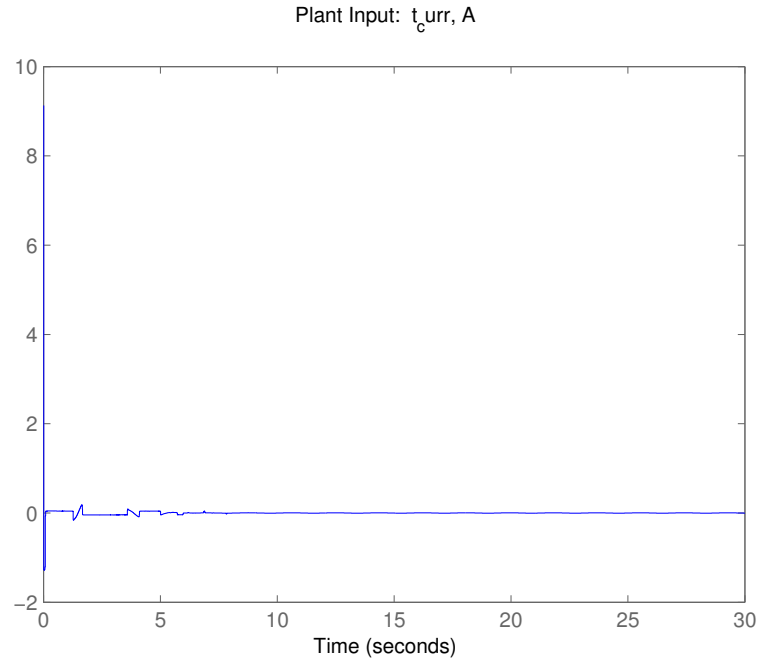


Figura 6: Entrada a sistema Tower, mediante simulación del controlador MPC diseñado con la herramienta `mpctool`. El estado inicial es nulo y se considera una referencia de $\theta = 60[deg]$.

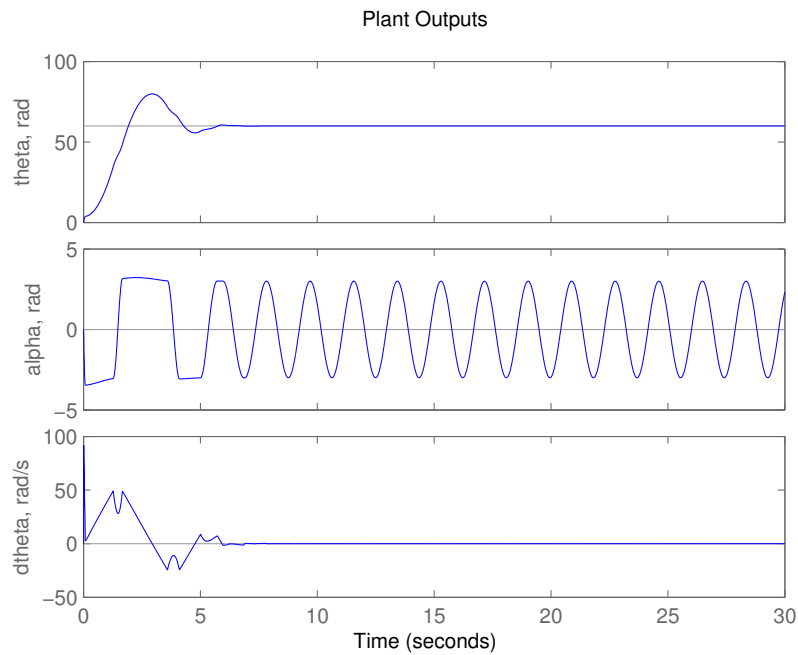


Figura 7: Salidas del sistema Tower, mediante simulación del controlador MPC diseñado con la herramienta `mpctool`. El estado inicial es nulo y se considera una referencia de $\theta = 60[deg]$.

3.5. Tower: Estudio del sistema

Para revisar el funcionamiento del control MBPC se realizarán pruebas sobre la planta real (Tower) de forma de contrastar por otro lado la respuesta obtenida por simulación. Las pruebas a realizar serán considerando cambios en la referencia, los horizontes de predicción y el factor λ de la ecuación de optimización, ver ecuación 28. Por otro lado los datos que se observarán sobre las respuestas de la planta serán el tiempo de establecimiento t_s , frecuencia natural w_n , overshoot o sobrepaso M_o , etc.

Los parámetros iniciales a utilizar son: $N_1 = 1$, $N_2 = 60$, $N_u = 40$, $\lambda_1 = 30$, $\lambda_2 = 10$ y $\lambda_u = 0.1$, ver ecuación (28). Por otro lado se realizarán las pruebas considerando las referencias $\theta = 60^\circ$ y $\theta = 90^\circ$.

$$J_t = \sum_{j=N_1}^{N_2} \lambda_1 (x_1(t+j) - r_1)^2 + \sum_{j=N_1}^{N_2} \lambda_2 (x_3(t+j) - r_3)^2 + \sum_{j=1}^{N_u} \lambda_u \Delta u(t+j-1) \quad (28)$$

s.a. $-3 < x_2 < 3$ [deg]

3.5.1. Pruebas sobre parámetros de diseño

La primera prueba será sobre los horizontes de predicción de forma que se tomarán datos de entrada y salida de la planta utilizando los mismos horizontes originales reducidos a la mitad y otros al doble de forma de tener una idea general del efecto de la magnitud del horizonte en la acción de control y el comportamiento de la planta.

La segunda prueba a realizar será sobre el factor λ , del cual se revisarán los casos: $\lambda_u = 0$ y $\lambda_1 = \lambda_2$.

3.5.2. Pruebas para reducción de Oscilaciones

Para estas pruebas se considerará cambios sobre los factores λ de la función objetivo de forma de considerar una mayor importancia asociada a la oscilación de la carga. En estas pruebas se considerará una referencia de $\theta = 0^\circ$, además de introducir una perturbación a la carga de la grúa. El perfil de perturbación consiste en usar una entrada de $0.5[A]$ durante 0.5 segundos, luego $-0.5[A]$ durante 0.5 segundos, para finalmente dejar oscilando durante 1 segundo antes de ejecutar cualquier acción de control. Este perfil se puede observar gráficamente en la figura 8.

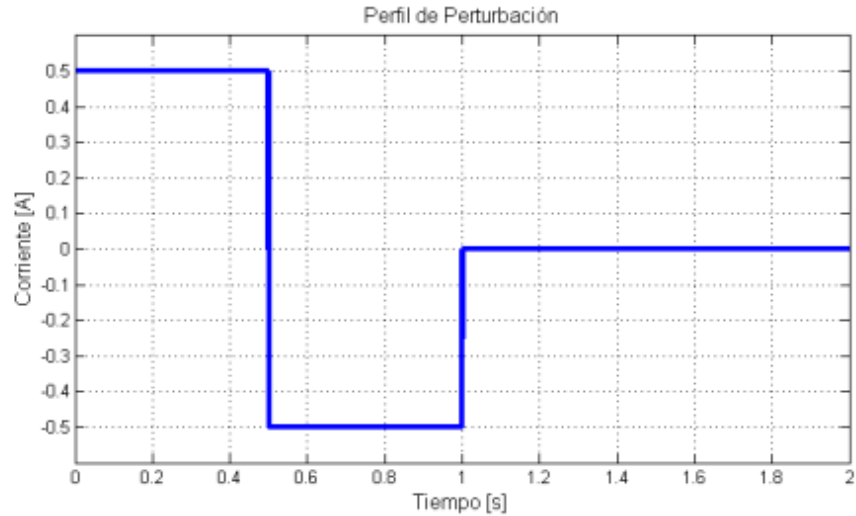


Figura 8: Perfil de perturbación para prueba de reducción de oscilaciones.

El objetivo de esta prueba es minimizar el tiempo en que la oscilación (α) queda acotado por una banda de 3σ y no vuelve a salir. Además las pruebas permitirán observar cuales de los factores λ tiene una mayor incidencia al intentar minimizar las oscilaciones sobre la carga, además de cuantizar las velocidades de convergencia. Una segunda prueba sera manipular el horizonte de predicción y observar el efecto sobre el controlador para posteriormente concluir al respecto.

3.6. Jib: Diseño de controlador predictivo

Según el método explicado en la sección 2.2, se calculan las matrices de controlabilidad y observabilidad para el sistema Jib discreto, las que se presentan en las expresiones (29) y (30), respectivamente. Ambas matrices son de rango 4, por lo que Jib es un sistema Observable y Controlable.

$$C_o = \begin{pmatrix} 0.0009 & 0.0028 & 0.0046 & 0.0064 \\ 0.0011 & 0.0032 & 0.0053 & 0.0078 \\ 0.1825 & 0.1824 & 0.1824 & 0.1823 \\ 0.2113 & 0.2110 & 0.2104 & 0.2096 \end{pmatrix} \quad (29)$$

$$O_o = \begin{pmatrix} 1 & 0 & 0.00 & 0 \\ 0 & 0 & 1.00 & 0 \\ 1 & -8.5086 \times 10^{-5} & 0.01 & -2.8363 \times 10^{-7} \\ 0 & -0.0170 & 1.00 & -8.5086 \times 10^{-5} \\ 1 & -0.0003 & 0.02 & -2.2686 \times 10^{-6} \\ 0 & -0.0340 & 1.00 & -0.0003 \\ 1 & -0.0008 & 0.03 & -7.6540 \times 10^{-6} \\ 0 & -0.0510 & 1.00 & -0.0008 \end{pmatrix} \quad (30)$$

3.7. Jib: Implementación en Simulink

En la Figura 9 se presenta el modelo de simulink que contiene el subsistema Jib discreto y el bloque de MPC correspondiente. En 10 y 11 se presentan los resultados de las simulaciones del controlador diseñado mediante la herramienta `mpc_tool` y siguiendo la configuración propuesta es las especificaciones de la experiencia. El controlador presenta un pésimo desempeño, pues se no sigue la referencia y presenta un gran error (a simple vista!), sin embargo, mantiene la carga balanceada dentro de los límites establecidos. Se intentó diseñar con diversas elecciones de parámetros del controlador, obteniendo resultados análogos en todos los casos.

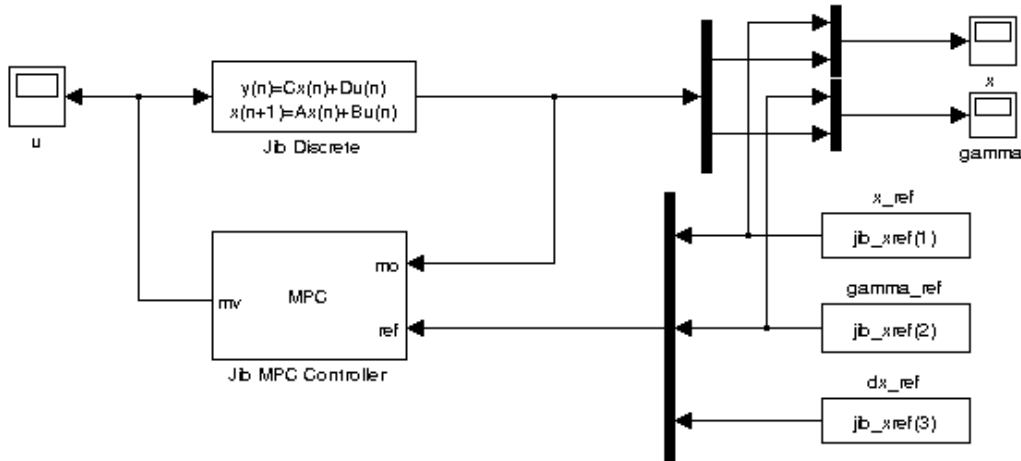


Figura 9: Implementación en simulink de sistema Jib discreto junto a controlador MPC perteneciente al toolbox *Model Predictive Control* de MATLAB.

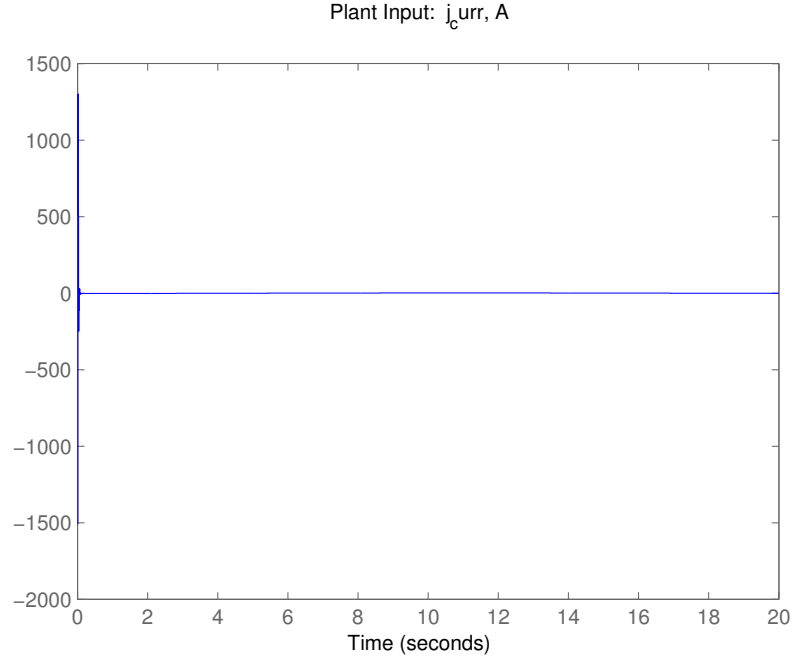


Figura 10: Entrada a sistema Jib, mediante simulación del controlador MPC diseñado con la herramienta `mpctool`. El estado inicial es nulo y se considera una referencia de $x = 0.2[m]$.

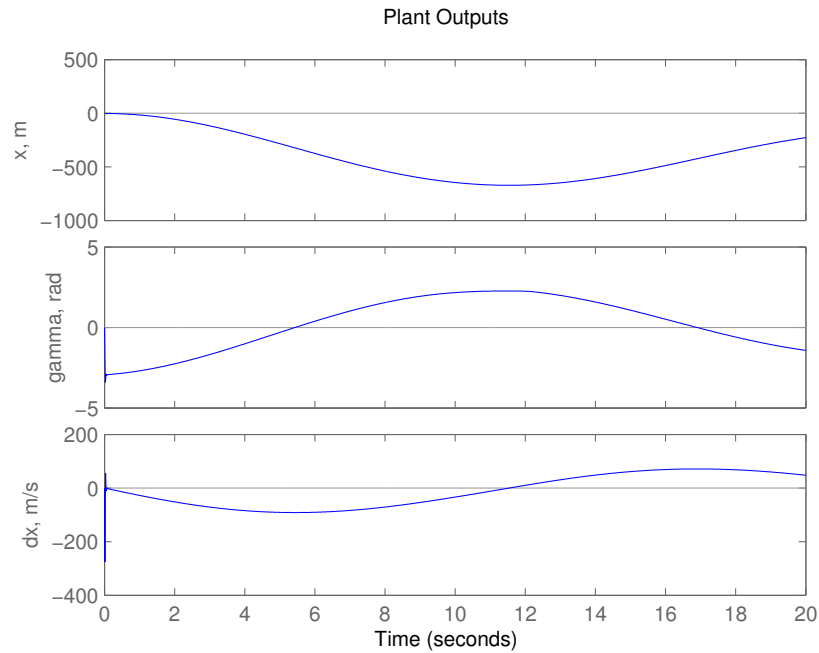


Figura 11: Salidas del sistema Jib, mediante simulación del controlador MPC diseñado con la herramienta `mpctool`. El estado inicial es nulo y se considera una referencia de $x = 0.2[m]$.

3.8. Jib: Estudio del sistema

Al igual que en el caso de Tower se realizarán pruebas similares para revisar el funcionamiento del control MBPC además de contrastar la respuesta obtenida por simulación y la respuesta real. Las pruebas a realizar serán considerando cambios en la referencia, los horizontes de predicción y el factor λ de la ecuación de optimización, ver ecuación (28). Por otro lado los datos que se observarán sobre las respuestas de la planta serán el tiempo de establecimiento t_s , frecuencia natural w_n , overshoot o sobrepaso M_o , etc.

Los parámetros iniciales a utilizar son: $N_1 = 1$, $N_2 = 60$, $N_u = 40$, $\lambda_1 = 30$, $\lambda_2 = 10$ y $\lambda_u = 0.1$, ver ecuación (28). Por otro lado se realizarán las pruebas considerando las referencias $x = 0.2[m]$ y $x = -0.2[m]$.

3.8.1. Pruebas sobre parámetros de diseño

La primera prueba será sobre los horizontes de predicción de forma que se tomarán datos de entrada y salida de la planta utilizando los mismos horizontes originales reducidos a la mitad y otros al doble de forma de tener una idea general del efecto de la magnitud del horizonte en la acción de control y el comportamiento de la planta.

La segunda prueba a realizar será sobre el factor λ , del cual se revisarán los casos: $\lambda_u = 0$ y $\lambda_1 = \lambda_2$.

3.8.2. Pruebas para reducción de Oscilaciones

De forma análoga a las pruebas que se realizarán sobre Tower, para estas pruebas se considerarán cambios sobre los factores λ de la función objetivo de forma de considerar una mayor importancia asociada a la oscilación de la carga. En estas pruebas se considerará una referencia de $x = 0[m]$.

El objetivo de esta prueba es minimizar el tiempo en que la oscilación (γ) queda acotado por una banda de 3σ y no vuelve a salir. Además las pruebas permitirán observar cuáles de los factores λ tiene una mayor incidencia al intentar minimizar las oscilaciones sobre la carga, además de cuantizar las velocidades de convergencia. Una segunda prueba será manipular el horizonte de predicción y observar el efecto sobre el controlador para posteriormente concluir al respecto.

3.9. Controladores acoplados

Para esta etapa de las pruebas en laboratorio se observará el comportamiento del sistema en conjunto, es decir Tower más Jib, de forma que es necesario acoplar ambos controladores predictivos



en la grúa del laboratorio y realizar pruebas considerando las siguientes referencia:

- $\theta = 60^\circ$, $x = 0.2[m]$
- $\theta = 0^\circ$, $x = 0[m]$, en esta prueba es necesario incluir perturbaciones como los de la figura 8.

Lo que se espera de esta prueba es ver como afectan las diferentes configuraciones de los parámetros del controlador, en el desempeño del controlador.



4. Referencias

- [1] Laboratorio de Automática, Departamento de Ingeniería Eléctrica, Universidad de Chile. <http://ingenieria.uchile.cl/investigacion/presentacion/laboratorios/90676/laboratorio-de-automatica>
- [2] Universidad Nacional de la Plata. Apuntes de control moderno. Versión online, consultada el 7/10/2015: <http://www.ing.unlp.edu.ar/controlm/electronica/archivos/apuntes/cap6.pdf>
- [3] ©Quanser Inc. Especificaciones técnicas del sistema “3 DOF Crane”. http://www.quanser.com/products/3dof_crane