

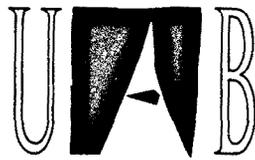


UNIVERSITAT AUTONOMA DE BARCELONA

ASPECTOS DE IMPLEMENTACIÓN Y ANÁLISIS DE  
ROBUSTEZ EN CPBM

MEMORIA PRESENTADA AL  
DEPARTAMENTO DE INFORMÁTICA  
PARA OPTAR AL GRADO DE  
DOCTOR EN CIENCIAS (INFORMÁTICA)

Javier Serrano  
Bellaterra, Noviembre 1994



Universitat Autònoma de Barcelona

UNIVERSITAT AUTONOMA DE BARCELONA

ASPECTOS DE IMPLEMENTACIÓN Y ANÁLISIS DE  
ROBUSTEZ EN CPBM

MEMORIA PRESENTADA AL  
DEPARTAMENTO DE INFORMÁTICA  
PARA OPTAR AL GRADO DE  
DOCTOR EN CIENCIAS (INFORMÁTICA)

Javier Serrano  
Bellaterra, Noviembre 1994

CÉSAR DE PRADA, Catedrático de Universidad de Automática de la  
Universidad de Valladolid,

CERTIFICA:

Que la presente memoria, *Aspectos de Implementación y Análisis de Robustez en CPBM*, ha sido realizada bajo mi dirección en el Departamento de Informática de la Universitat Autònoma de Barcelona por Javier Serrano y constituye su tesis para optar al grado de Doctor en Ciencias (Informática).



---

César de Prada  
Bellaterra, Noviembre 1994

# Agradecimientos

En primer lugar deseo hacer constar mi más sincero agradecimiento a César de Prada por su dirección y colaboración en el desarrollo de este trabajo. Sus críticas y sugerencias han sido fundamentales para la obtención de los objetivos propuestos. Los resultados habrían sido bien distintos sin sus aportaciones.

También querría agradecer al Dr. Ignasi Serra, Director del Departamento de Informática donde ha sido desarrollada esta tesis, que con sus indicaciones y continuo apoyo moral ha contribuido a que este trabajo llegase a feliz término.

A mis compañeros de despacho Romualdo Moreno y Miquel Angel Piera con los que tantas horas he compartido y tanta ayuda he recibido. Y al resto de compañeros de la Unidad: Josep LLuis de la Rosa, Juanjo Ramos, Marta Gómez, Carles Estorac, David Megías y, particularmente, a Ramón Vilanova cuyas sugerencias en algunas partes de la tesis han sido muy importantes. Todos ellos han contribuido a crear un ambiente de trabajo muy agradable.

A las chicas de la Universidad de Valladolid, Pastora Vega y María Jesús de la Fuente, que siempre que he estado allí han tratado de hacer mi estancia lo más fácil y agradable posible..

Un agradecimiento muy especial a mi familia por haber potenciado mis estudios y haber creado un ambiente idóneo para realizarlos, muchas veces a costa de renunciar a otras cosas.

Y por último, pero no menos, a Carmen, la chica que me aguanta todos los días, que con su cariño y apoyo ha conseguido que me levante cada día con ilusión.

# Indice

<b>Agradecimientos</b>	<b>iii</b>
<b>1 Introducción al CPBM</b>	<b>1</b>
1.1 ¿Qué es el Control Predictivo? . . . . .	2
1.2 Revisión Histórica . . . . .	4
1.3 Estado del Arte . . . . .	8
1.3.1 Restricciones en las variables . . . . .	9
1.3.2 Robustez . . . . .	10
1.3.3 Modelos de predicción . . . . .	10
1.4 Objetivos y Desarrollo de la Memoria . . . . .	14
<b>2 Metodología de Diseño CPBM</b>	<b>17</b>
2.1 Modelos de Predicción . . . . .	18
2.1.1 Modelos del Proceso . . . . .	18
2.1.2 Modelos de perturbaciones . . . . .	22
2.2 Trayectoria de Referencia . . . . .	24
2.3 Estructuración de la Ley de Control . . . . .	25
2.3.1 Horizonte de control . . . . .	27

2.3.2	Funciones base . . . . .	28
2.4	Cálculo de la Señal de Control . . . . .	29
2.4.1	La función de coste . . . . .	29
2.4.2	Restricciones . . . . .	31
2.4.3	La Ley de Control Predictiva . . . . .	32
2.5	Compensación Feedforward . . . . .	33
2.6	Control Predictivo Generalizado (GPC) . . . . .	34
2.6.1	Descripción de la Planta . . . . .	34
2.6.2	Predicción de la Salida . . . . .	35
2.6.3	El Algoritmo de control . . . . .	38
<b>3</b>	<b>Aspectos de Implementación de CPBM</b>	<b>40</b>
3.1	Formulación del DMC . . . . .	41
3.2	Implementación del DMC . . . . .	43
3.3	Forma de la Función de Coste . . . . .	48
3.4	Implementación de CPBM Multivariables . . . . .	54
3.4.1	DMC Multivariable . . . . .	54
3.4.2	Compensación feedforward . . . . .	57
3.4.3	Aplicación al control de un reactor químico . . . . .	60
3.4.4	Restricciones . . . . .	65
<b>4</b>	<b>Aplicación de Técnicas de Análisis Robusto a CPBM</b>	<b>68</b>
4.1	Análisis para el Diseño Robusto . . . . .	69
4.1.1	Funciones de sensibilidad y sensibilidad complementaria . . . . .	70
4.1.2	Descripción de la incertidumbre . . . . .	73

4.1.3	Objetivos nominales y robustos . . . . .	76
4.2	Expresión del GPC como un Controlador de Dos Grados de Libertad . . . . .	84
4.3	Influencia de los parámetros de sintonía . . . . .	89
4.4	Diseño Robusto de Controladores Predictivos . . . . .	102
4.4.1	Cálculo de la incertidumbre . . . . .	104
4.4.2	Sintonía para estabilidad robusta . . . . .	106
4.4.3	Caracterización frecuencial de la referencia . . . . .	108
4.4.4	Evaluación del diseño . . . . .	109
<b>5</b>	<b>Robustez en CPBM y Modelo del Ruido</b> . . . . .	<b>115</b>
5.1	Predicción y Modelo del Ruido . . . . .	116
5.2	Error de Predicción . . . . .	119
5.3	Error de predicción en el GPC: El polinomio T . . . . .	121
5.4	Métodos para Incrementar la Robustez . . . . .	129
5.4.1	Aumentar el factor de peso $\lambda$ . . . . .	133
5.4.2	Limitar los incrementos de la señal de control $\Delta u$ . . . . .	134
5.4.3	Filtrar la señal de control . . . . .	135
<b>6</b>	<b>Conclusiones</b> . . . . .	<b>139</b>
<b>A</b>	<b>Reactor Químico Exotérmico Encamisado</b> . . . . .	<b>143</b>
A.1	Modelo del Proceso . . . . .	144
A.2	Respuesta en Lazo Abierto . . . . .	147
<b>B</b>	<b>Toolbox de Análisis y Diseño de CPBM Robusto</b> . . . . .	<b>150</b>
B.1	Definición de Modelos . . . . .	151

B.2 Controladores . . . . .	154
B.3 Análisis . . . . .	167
B.4 Diseño Robusto . . . . .	179
<b>Bibliografía</b>	<b>187</b>

# Capítulo 1

## Introducción al CPBM

Los controladores predictivos se basan en la predicción del comportamiento de un proceso que deseamos controlar. Esta predicción la haremos a partir de un modelo explícito del proceso que hemos de conocer de antemano.

El concepto de control predictivo no está restringido a sistemas lineales SISO, se puede aplicar también a sistemas multivariables (MIMO) y sistemas no lineales. A nivel práctico se ha demostrado que puede aplicarse a la práctica totalidad de procesos: en procesos rápidos (mecatrónica) y en procesos lentos típicos en la industria de procesos.

A continuación se presentan algunas características del control predictivo, tanto a nivel teórico como a nivel práctico, que le confiere importantes ventajas sobre otros métodos de control:

- Permite la solución de problemas de control con comportamientos dinámicos complicados (sistemas con retardos, de fase no mínima, inestables).
- Permite solucionar de forma sencilla problemas de control multivariable con interacción.

- Introduce de forma natural compensación feedforward frente a perturbaciones medibles.
- Conceptualmente es fácil extender la estrategia a problemas con restricciones. El manejo de restricciones es necesario tanto en procesos lentos, ya que aparecen como restricciones para mantener la calidad, como en procesos rápidos debido a las limitaciones de los actuadores.
- Los parámetros de diseño tienen una relación directa con las especificaciones de diseño siendo fácil su sintonía. Así, es fácil establecer un compromiso entre el rendimiento dinámico y la robustez del sistema de control.
- Tiene lo que podríamos llamar "visión de futuro", que es útil cuando se conoce previamente como variará la señal de consigna. Esto es interesante en problemas de robótica y procesos batch.
- Es fácil de explicar y entender a nivel de operarios de planta y usuarios finales.

## 1.1 ¿Qué es el Control Predictivo?

El control predictivo es un control digital en el que la salida del controlador se calcula a partir de la predicción del comportamiento futuro usando un modelo del proceso que queremos controlar. La forma como operan estos controladores es la siguiente:

Sea  $u(k)$ ,  $y(k)$  y  $w(k)$  la salida del controlador, la salida del proceso y la señal de consigna, respectivamente, en el instante actual  $t = k$ . Además, definimos los siguientes vectores

$$\begin{aligned} \mathbf{u} &= [u(k), \dots, u(k + H_p - 1)]^T \\ \hat{\mathbf{y}} &= [\hat{y}(k + 1), \dots, \hat{y}(k + H_p)]^T \\ \mathbf{w} &= [w(k + 1), \dots, w(k + H_p)]^T \end{aligned}$$

donde  $H_p$  es el horizonte de predicción y el símbolo  $\hat{\cdot}$  significa estimación. Así, un controlador predictivo calcula una secuencia de controles futuros  $\mathbf{u}$  tal que la predicción de la salida del proceso  $\hat{\mathbf{y}}$  se acerque lo más posible a la señal de consigna  $\mathbf{w}$ . Esta señal de consigna se denomina generalmente trayectoria de referencia, ya que no tiene porqué coincidir necesariamente con la señal de consigna externa, y en particular, puede ser una secuencia de puntos arbitraria. El primer elemento de la secuencia de controles calculada ( $= u(k)$ ) se usa para controlar el proceso. En el siguiente instante de muestreo, se repite todo el procedimiento usando las últimas medidas tomadas. A esta técnica se le denomina de *horizonte móvil* y es la forma en que los controladores predictivos introducen la realimentación en el lazo de control.

La salida del proceso se predice mediante un modelo del proceso a controlar. Cualquier modelo que describa la relación entre la entrada y la salida del proceso se puede usar. Por tanto, no sólo los clásicos modelos CARMA son adecuados para la predicción, sino también los modelos respuesta escalón, modelos en el espacio de estados y modelos no lineales. Además, si el proceso está perturbado, se puede añadir una perturbación o modelo del ruido al modelo del proceso consiguiendo que el efecto de la perturbación sobre la predicción de la salida del proceso se tenga en cuenta.

Para definir lo bien que la predicción de la salida del proceso sigue a la trayectoria de referencia, se usa una función de coste. Normalmente esta función dependerá de  $\hat{\mathbf{y}}$ ,  $\mathbf{u}$  y  $\mathbf{w}$ . Una sencilla función de coste sería:

$$J = \sum_{i=1}^{H_p} (\hat{y}(k+i) - w(k+i))^2$$

Así, minimizando  $J$  respecto a  $\mathbf{u}$ , obtendremos la secuencia de control  $\mathbf{u}$  sobre el horizonte de predicción. Como resultado, se minimiza el error futuro entre la predicción de la salida del proceso y la trayectoria de referencia y si el modelo es idéntico al proceso y no hay perturbaciones ni restricciones, el proceso seguirá la trayectoria de referencia exactamente en los instantes de muestreo. En el caso que el proceso presente un retardo el enfoque no es diferente. Suponiendo que el retardo sea dos instantes de muestreo, las predicciones  $\hat{y}(k+1)$  y  $\hat{y}(k+2)$  no tiene sentido

considerarlas, ya que estos valores no dependen de las acciones de control en  $t = k$  y  $t = k + 1$ . Así, la función de coste quedaría:

$$J = \sum_{i=3}^{H_p} (\hat{y}(k+i) - w(k+i))^2$$

Claramente, calcular la secuencia de control es un problema de optimización, o más en concreto, un problema de minimización. Cuando el criterio es cuadrático, el modelo es lineal y no hay restricciones, existe una solución analítica. Si no es así, se requieren métodos numéricos para su solución. Por esta razón la gran mayoría de controladores predictivos usan criterios cuadráticos.

## 1.2 Revisión Histórica

A finales de los años 70 el desarrollo de la tecnología digital representaba un reto para los investigadores en el área del control automático. Con la disponibilidad de computadores mucho más potentes y baratos, se empezaban a reconsiderar las estrategias básicas usadas en aplicaciones de control.

A pesar del progreso tecnológico, la teoría clásica de los sistemas realimentados se seguía usando en el diseño de aplicaciones industriales. En aplicaciones aeroespaciales, la teoría moderna de control había tenido éxitos notables. Sin embargo, la implantación de tales técnicas al control industrial no había seguido la misma suerte.

Los procesos industriales tiene características especiales. Son sistemas altamente multivariables, las perturbaciones frecuentemente afectan a la estructura de la planta además de a las variables medidas y normalmente han de incorporar restricciones en muchas de sus variables.

En este entorno se trató de concebir una teoría de control donde los computadores digitales y el modelado tuvieran un papel primordial y soportaran las

características de los procesos industriales anteriormente descritas.

En 1978 Richalet et al. describe varias aplicaciones de "Model Predictive Heuristic Control", y en 1979 ingenieros de la compañía Shell (Cutler y Ramaker, 1979; Prett y Gillette, 1979) sacan a la luz el "Dynamic Matrix Control" (DMC) y muestran varias aplicaciones a un craceador catalítico. Ambos algoritmos usan un modelo dinámico explícito de la planta (modelo de convolución) para predecir el efecto de las futuras acciones de las variables manipuladas en la salida (de aquí el nombre de Control Predictivo Basado en Modelos - CPBM). Los movimientos futuros de las variables manipuladas se determinan mediante optimización con el objetivo de minimizar el error predecido, y sujeto a ciertas restricciones. La optimización se repite en cada instante de muestreo con información actualizada de la planta.

Por tanto, el problema de control en el contexto CPBM se formula como un problema de optimización dinámica. Aunque esto no sea una idea muy novedosa, constituye el primer ejemplo de optimización dinámica a gran escala aplicado en tiempo real en una industria de procesos.

El concepto CPBM tiene una larga historia. Zadeh y Whalen (1962) describieron por primera vez la estrecha relación entre el problema de control óptimo a tiempo mínimo y la Programación Lineal. Propoi (1963) propuso la estrategia del horizonte móvil ("Open Loop Optimal Feedback") que es una idea central de todos los algoritmos CPBM. Chang y Seborg (1983) descubrieron la relación entre el trabajo de Propoi y CPBM.

Desde el redescubrimiento de CPBM en 1978 y 1979, su popularidad en la industria de procesos continuos ha ido creciendo constantemente. En la década de los 80 se han descrito numerosas aplicaciones que han tenido éxito en este tipo de industrias (Mehra, 1982; García, 1984; Matsko, 1985; Martin, 1986; Cutler y Hawkins, 1987). Varias compañías ofrecen software de control que incluyen algoritmos CPBM: Bailey, DMC, Profimatics y Setpoint.

De forma bastante independiente, en entornos académicos principalmente

dedicados a temas de investigación sobre control adaptativo, surgió a mediados de los 80 una segunda rama de técnicas CPBM. Estos desarrollos estaban limitados esencialmente a sistemas SISO aunque conceptualmente es sencillo extenderlos al caso multivariable. Tampoco consideraban en detalle el tema de las restricciones. Los nombres de estos algoritmos son bastante esotéricos: Predictor-based Self Tuning Control (Peterka, 1984), Extended Horizon Adaptive Control (Ydstie, 1984), Model Algorithmic Control (Rouhani y Mehra, 1982), MUSMAR (Mosca, 1984), EPSAC (De Keyser y Van Cauwenberghe, 1985). Todos estos algoritmos, como los anteriores, tienen características comunes: solucionan un problema de optimización en un horizonte finito, usan la técnica del horizonte móvil, incorporan predicciones de la salida de la planta, tienen un número pequeño de parámetros de diseño.

El algoritmo de control que ha tenido más aceptación en entornos académicos es el "Generalized Predictive Control" (GPC), debido a Clarke, Mohtadi y Tuffs (1987). El GPC engloba como casos particulares la gran mayoría de métodos CPBM referenciados anteriormente. Vamos a ver a continuación una breve perspectiva histórica de este algoritmo.

El antecesor del GPC es el controlador de varianza mínima, descrito por Astrom (1970) y que era la base del famoso Self-Tuning Regulator de Astrom y Wittenmark (1973). El controlador se obtenía, para un modelo lineal dado, minimizando el siguiente criterio,

$$J(u, t) = E\{(y_{t+1} - r_{t+1})^2\}$$

donde  $y_t$  es la salida del sistema y  $r_t$  es la señal de consigna. La señal de control  $u_t$  se obtiene minimizando el criterio anterior. En el instante  $t + 1$  se soluciona un nuevo problema para  $u_{t+1}$ . El criterio anterior se basa en la suposición de que el modelo de la planta tiene retardo uno. Si la planta tiene un retardo  $d$ , el error se reemplaza por  $y_{t+d} - r_{t+d}$ .

Como es sabido, esta estrategia de control sólo funciona para sistemas de fase mínima. Para plantas de fase no mínima, el controlador al cancelar los ceros de la planta (incluidos los ceros inestables) provoca la pérdida de estabilidad interna

del sistema realimentado. Una forma de conseguir que esta técnica funcione para sistemas de fase no mínima, es modificar el criterio anterior de forma que incluya un factor de peso en la señal de control. Esto conduce a un nuevo criterio

$$J(u, t) = E\{(y_{t+d} - r_{t+d})^2 + \lambda u_t^2\}$$

que se denomina control de Varianza Mínima Generalizada. Aunque potencialmente esta técnica puede producir leyes de control internamente estables, no está totalmente garantizada para cualquier valor de  $\lambda$ .

Otra modificación frecuente hecha a la ley de control VMG es establecer el problema de optimización en términos no de  $u_t$  sino de  $\Delta u_t$

$$J(u, t) = E\{(y_{t+d} - r_{t+d})^2 + \lambda(\Delta u_t^2)\}$$

La razón de usar incrementos de la señal de control en el criterio anterior es que este criterio consigue error estático cero en el caso de señales de referencia constantes y distintas de cero aunque la planta no disponga de un integrador. Para el problema VMG estándar, el criterio siempre penaliza una señal de control  $u_t$  distinta de cero incluso en el caso de seguimiento de una referencia distinta de cero.

Como quiera que este controlador todavía falla para algunos sistemas inestables y de fase no mínima, y particularmente para sistemas que no se conoce bien el retardo, Clarke *et al.* desarrollaron una extensión de este método que dio lugar al GPC. El criterio que minimiza este controlador ya no es el error en un instante futuro, sino que minimiza el error a lo largo de un horizonte de predicción.

Uno de los aspectos más remarcables de este criterio es que posee un nivel de complejidad suficiente para producir controladores efectivos para un rango muy amplio de plantas, mientras que el criterio mismo depende únicamente de tres parámetros de diseño.

Tanto el GPC como muchos de los algoritmos CPBM surgidos en entornos académicos se han probado con éxito en aplicaciones industriales.

Así, en los últimos años el control predictivo basado en modelos ha conseguido un nivel significativo de aceptación y éxito en aplicaciones de control de procesos. Durante este mismo periodo, la comunidad académica ha perdido cierto interés en los temas de investigación en control adaptativo y ha dedicado muchos más esfuerzos en establecer una teoría de control CPBM. Es esta, pues, una metodología en la que la teoría y la práctica se dan soporte eficientemente. El progreso, que desde finales de los 70 ha sido continuo, se está ahora acelerando.

### 1.3 Estado del Arte

Aunque los CPBM se ha demostrado que funcionan bien en aplicaciones prácticas, todavía no se ha desarrollado totalmente una teoría formal que de soporte a estas técnicas y que provea de guías en el diseño de controladores (en vez de usar reglas heurísticas para la sintonía de parámetros). En particular, problemas fundamentales como la estabilidad y la robustez no están del todo solucionados ni siquiera para el caso de sistemas lineales con un controlador lineal.

Una forma de analizar las propiedades de estabilidad y robustez de controladores predictivos del tipo GPC es mediante la teoría del control óptimo LQG, ya que el GPC se puede ver como un caso particular de control LQG [Bitmead 90]. Obviamente esta teoría no es útil para estudiar el caso en que tengamos restricciones en alguna variable y como se ha demostrado en la práctica de una complejidad excesiva para su implementación en entornos industriales.

Obedeciendo a este deseo de establecer un teoría del control predictivo y a mejorar diversos aspectos de implementación son muchos los aspectos que se están investigando. La gran mayoría de estos se pueden agrupar en tres líneas fundamentales de investigación que a continuación se presentan.

### 1.3.1 Restricciones en las variables

Las restricciones surgen de forma natural en cualquier problema de control. Están impuestas por saturaciones en los actuadores, o por requerimientos preespecificados en alguna variable con objeto de hacer el proceso más seguro, más productivo, etc.

Las técnicas CPBM son la única metodología adecuada para manejar restricciones de una forma sistemática en el proceso de diseño e implementación del controlador.

Dada la gran importancia que tiene este tema (tanto a nivel teórico como práctico) son muchos los esfuerzos que se le están dedicando. Las cuestiones más importantes que se investigan en la actualidad son

- En el caso que la ley de control sea lineal y se pueda obtener una expresión compacta del controlador, podremos realizar el análisis del sistema de control mientras las restricciones no estén activas. En caso contrario no hay ninguna teoría general que permita analizar controladores con restricciones.
- Cuando el proceso a controlar es multivariable y con restricciones los algoritmos numéricos necesarios son de una complejidad extraordinaria. Existe la necesidad de desarrollar algoritmos numéricos más eficientes y adecuados al problema.
- Cuando hay muchas restricciones puede pasar que la región factible sea nula. En este caso hay que eliminar alguna restricción y tratar otra vez de solucionar el problema. Como elegir las restricciones que se desactivan teniendo en cuenta que el espacio paramétrico donde se lleva a cabo la minimización puede ser de dimensiones muy grandes es un tema de investigación muy importante sobre todo en aplicaciones de control de procesos.

### 1.3.2 Robustez

Por robustez se entiende que el rendimiento del lazo de control se mantenga cuando el comportamiento dinámico de la planta real es diferente del modelo de predicción usado. Como los modelos nunca representan perfectamente la realidad se necesitan métodos de diseño que incluyan algún mecanismo que provea de robustez frente a dinámicas sin modelar.

Como se ha comentado anteriormente, los CPBM sin restricciones son equivalentes a controladores LTI clásicos y por tanto se pueden aplicar todos los métodos de análisis de robustez ya desarrollados. De hecho los CPBM no son ni más ni menos robustos que los controladores clásicos. La robustez de los CPBM depende únicamente de como se diseñen. Así, uno puede incluir en el modelo de predicción parámetros que no afectan a la descripción entre la entrada y la salida, pero se obtiene mayor robustez.

De lo que no hay duda, es que un CPBM es más fácil de ajustar para conseguir robustez que un controlador clásico. Como diseñar y sintonizar un CPBM para conseguir robustez es otro importante tema de investigación.

Dentro del tema de la robustez y directamente ligado a las técnicas CPBM está el problema como evaluar la incertidumbre que tiene el modelo usado en el controlador. Los métodos de identificación global debidos a Richalet en vez de dar únicamente un conjunto de parámetros del modelo, determinan una región del espacio paramétrico donde puede estar el modelo real, lo cual nos determina un modelo de la incertidumbre.

### 1.3.3 Modelos de predicción

El elemento más importante de la metodología CPBM es el modelo. La elección del modelo tiene varias consecuencias: este ha de ser suficientemente com-

pleto para capturar las dinámicas esenciales de la planta, ha de permitir calcular las respuesta libre y forzada, y ha de permitir también realizar un análisis del controlador resultante.

Los primeros controladores predictivos aparecidos en la década pasada usaban modelos lineales. Estos controladores permiten tratar sistemas con no linealidades suaves simplemente desintonizando el controlador. En presencia de no linealidades fuertes, el uso de modelos no lineales apropiados es una opción natural.

Las representaciones de la planta habitualmente usadas para modelizar estas no linealidades son: ecuaciones diferenciales no lineales, modelos paramétricos y redes neuronales. Por otra parte, cuando el conocimiento que tenemos del proceso es incompleto una forma de describirlo es mediante modelos cualitativos. A continuación se describen estos modelos.

### Ecuaciones diferenciales no lineales

Si disponemos de un conjunto de ecuaciones diferenciales no lineales el control predictivo se puede plantear como un problema general no lineal

$$\min_u G[x(t_f)] + \int_{t_0}^{t_f} F[x(t), u(t)] dt$$

sujeto a

$$\begin{aligned} \dot{x} &= f[x(t), u(t)], \quad x(t_0) = x_0 \\ h(x, u) &= 0 \\ g(x, u) &\leq 0 \end{aligned}$$

La solución de este problema cuando no hay restricciones de igualdad y  $g(x, u)$  es una función sólo de  $u$  es un problema clásico de control óptimo. La solución se puede encontrar mediante el cálculo de variaciones o mediante la programación dinámica. Estos métodos son computacionalmente muy costosos y rara vez usados en línea.

Es más prometedor discretizar el vector de controles respecto al tiempo y convertir el problema anterior en uno de programación no lineal. Es también posible usar un modelo de simulación en vez de las ecuaciones diferenciales para calcular las predicciones y los gradientes necesarios para el algoritmo numérico de optimización. Esto se ha llevado a cabo con éxito en un proceso biotecnológico de depuración de aguas residuales por Moreno (1994).

El control óptimo linealizado resuelve un problema como el anteriormente planteado y utiliza la misma estrategia que los controladores predictivos típicos como el DMC o GPC. Cuando el estado del proceso cambia se actualiza un modelo lineal y se usa para calcular la respuesta escalón del proceso. El modelo no lineal se usa para realizar el cálculo de la respuesta libre. O sea, las ecuaciones diferenciales no lineales se integran en lineal, en paralelo con el proceso, mientras el controlador predictivo usa el modelo lineal para minimizar la función de coste.

### Modelos paramétricos

La deducción de un modelo en ecuaciones diferenciales no lineales se hace a partir de principios fundamentales. Esto en general es costoso y en ciertos sistemas incluso se desconocen los principios que lo rigen o son extraordinariamente complejos.

Una alternativa es suponer que la no linealidad tienen una determinada estructura polinomial e identificar los parámetros de ésta. Esta forma de enfocar problema es muy adecuada para el control predictivo e incluso para el control adaptativo, ya que los parámetros se pueden identificar en lineal.

Algunas descripciones típicas de sistemas no lineales son las siguientes:

- Modelo de Hammerstein generalizado. La ecuación en diferencias de este modelo es

$$A(q^{-1})y(t) = a_0 + B(q^{-1})u(t-d) + C(q^{-1})u^2(t-d)$$

- Modelo de Volterra paramétrico. La ecuación en diferencias de este modelo es

$$A(q^{-1})y(t) = a_0 + B(q^{-1})u(t-d) + D(q_1^{-1}q_2^{-1})u(t-d)u(t-d)$$

- Modelo bilineal. La ecuación en diferencias de este modelo es

$$A(q^{-1})y(t) = a_0 + B(q^{-1})u(t-d) + F(q_1^{-1}q_2^{-1})y(t)u(t-d)$$

En estas expresiones  $d$  es el retardo y los polinomios se definen de la siguiente forma

$$\begin{aligned} A(q^{-1}) &= 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_{n_a}q^{-n_a} \\ B(q^{-1}) &= 1 + b_1q^{-1} + b_2q^{-2} + \dots + b_{n_b}q^{-n_b} \\ C(q^{-1}) &= 1 + c_1q^{-1} + c_2q^{-2} + \dots + c_{n_c}q^{-n_c} \\ D(q_1^{-1}, q_2^{-1})u(t-d)u(t-d) &= \sum_{i=0}^{n_d} \sum_{j=i}^{n_d} d_{i,j}u(t-d-i)u(t-d-j) \\ F(q_1^{-1}, q_2^{-1})y(t)u(t-d) &= \sum_{i=0}^{n_f} \sum_{j=i}^{n_f} f_{i,j}y(t-i)u(t-d-j) \end{aligned}$$

Estos modelos pertenecen una clase de modelos orientados a bloques. Los dos primeros estan compuestos de un bloque dinámico lineal y una función estática no lineal. En el modelo bilineal tanto la parte lineal como la no lineal son dinámicas.

## Redes neuronales

En los últimos años, las redes neuronales se han convertido en una atractiva forma de construir modelos de procesos no lineales complejos. Esto es debido a que las redes neuronales tienen la capacidad inherente de aproximar una función no lineal arbitraria. Así, estas nos proporcionan una forma posible de identificar un modelo de un proceso no lineal.

La ventaja de esta técnica es que es posible desarrollar rápidamente modelos de procesos complejos. Sin embargo, como el procedimiento de cálculo de la señal de control es numérico, el controlador predictivo resultante sólo se puede aplicar en los casos donde el tiempo no sea un parámetro crítico.

## Modelos cualitativos

Los modelos cualitativos son una herramienta para describir modelos de sistemas dinámicos inciertos basados en principios fundamentales y combinados con conocimiento heurístico. Las características de los modelos cualitativos relevantes desde el punto de vista del CPBM son

- Describen sistemas continuos determinista ordinarios mediante un conjunto de ecuaciones diferenciales y algebraicas cualitativas, que en principio son ecuaciones diferenciales y algebraicas cuyas variables y parámetros toman valores simbólicos y en intervalos.
- Se han de usar métodos aritméticos especiales o lógicos para calcular la solución. En general las soluciones son muy conservativas.

El uso de modelos dinámicos cualitativos dentro de la metodología CPBM es muy sencillo conceptualmente: se ha de reemplazar el modelo convencional usado para la predicción por uno cualitativo. En la práctica las propiedades especiales de la predicción cuando se usan modelos cualitativos causa problemas inusuales en el establecimiento del problema de control.

## 1.4 Objetivos y Desarrollo de la Memoria

Los controladores predictivos basados en modelos ya han demostrado su utilidad práctica en aplicaciones industriales, y prueba de ello son los diversos sistemas de control comercializados basados en alguno de estos controladores. La teoría, sin embargo, no está tan desarrollada a pesar de los grandes avances que se han hecho en estos últimos años. La tesis pretende contribuir en algunos aspectos teóricos de CPBM.

Los objetivos de la tesis se pueden dividir por un lado en el estudio de

algunos aspectos de implementación de CPBM, y por otro en un estudio de robustez de éstos.

En cuanto a los aspectos de implementación se estudian tres: El primero es un aspecto muy concreto del DMC donde en algunas ocasiones ciertas expresiones de la respuesta libre provoca respuestas inesperadas e indeseables. El segundo aspecto está relacionado con la forma que toma la función de coste que se ha de minimizar en cualquier CPBM y la influencia que tienen los parámetros de sintonía en ésta. El último aspecto es referente a la implementación de sistemas multivariables y la posibilidad de implementación de éstos mediante controladores SISO con compensación feedforward para eliminar la interacción entre lazo estudiando las ventajas que esta forma de implementación presenta.

La presencia de incertidumbres en el modelo del proceso y la presencia de perturbaciones actuando sobre éste, pueden influir significativamente en la eficiencia de los controladores. De aquí la importancia del análisis de robustez de los controladores predictivos. Este estudio se realiza desde dos enfoques distintos. El primero será mediante la teoría del control robusto, estudiando como influyen los parámetros de diseño sobre la robustez y el rendimiento del sistema, así como la presentación de un método de diseño que confiere al controlador características robustas.

El segundo enfoque en el análisis de robustez es a través del modelo de ruido que use el controlador. El modelo del ruido además de modelizar la dinámica entre las perturbaciones y la salida, tiene una influencia fundamental en la robustez del sistema de control.

Para alcanzar los objetivos planteados será preciso desarrollar una librería de funciones que implementen los diversos controladores predictivos y permitan su análisis. Las implementaciones de los CPBM se realizarán sobre ACSL y MATLAB, mientras que las herramientas de análisis se desarrollarán sobre MATLAB.

El desarrollo de la presente memoria se ha llevado a cabo en los siguientes capítulos:

En el segundo capítulo se presentan los elementos comunes a todos los CPBM, que son el modelo del proceso, la trayectoria de referencia, la estructuración de la ley de control y el cálculo de la señal de control. Además se deducen dos controladores predictivos, uno es el conocido GPC y el otro un controlador no lineal.

En el tercer capítulo se estudian los aspectos de implementación planteados en los objetivos. Se empieza formulando el DMC y se pasa a comparar dos implementaciones una teniendo en cuenta un modelo de respuesta escalón infinito y otro truncado. En el siguiente punto se estudia la influencia de los parámetros de diseño en la forma de la función de coste. Por último se comparan varias implementaciones de sistemas multivariable haciendo hincapié en las ventajas que tiene la implementación mediante controladores independientes con compensación feedforward.

En el cuarto capítulo se empieza presentando las técnicas estándar de análisis y diseño robusto. A continuación se deduce la función de transferencia de un controlador predictivo LTI, el GPC. Con estas herramientas se pasa a estudiar la influencia de los parámetros de diseño sobre la robustez mediante las funciones de sensibilidad y sensibilidad complementaria. El capítulo finaliza presentado un método de diseño que le confiere al controlador predictivo características robustas.

Finalmente en el quinto capítulo se vuelve a abordar el tema del análisis y diseño robusto de controladores predictivos a través del modelo del ruido. Se empieza estudiando la relación entre el modelo del ruido y el error de predicción que se comete. A continuación para el caso del GPC se estudia la importancia del polinomio  $T$  en el error de predicción. Para acabar se proponen diversos métodos que incrementan la robustez de los controladores predictivos.

## Capítulo 2

# Metodología de Diseño CPBM

Todos los controladores predictivos tienen en común una serie de características. Vamos a presentar a continuación cuatro elementos comunes a todos ellos:

1. Un modelo del proceso a controlar. Este modelo se usa para predecir la salida del proceso sobre el horizonte de predicción. Ejemplos de estos modelos son los CARIMA o FIR.
2. La trayectoria de referencia para la salida del proceso. En muchos controladores se usa una trayectoria de primer orden.
3. Una estructuración de la ley de control, ya que en general, el problema tiene muchos grados de libertad y en la práctica conviene reducirlos.
4. Cálculo de la señal de control. Con el fin de obtener la secuencia de control óptima sobre el horizonte de predicción se establece una función de coste. Mediante un procedimiento de optimización se obtiene la solución. Si el criterio es cuadrático, el modelo lineal y no hay restricciones existe una solución analítica. En caso de que haya restricciones, la función de costo se minimizará con métodos de optimización no lineal.

Cada uno de estos elementos dispone de varias opciones, dependiendo de la elección que se haga obtendremos un método u otro de control predictivo. A continuación se describen las diversas posibilidades que existen para cada elemento de un controlador predictivo.

## 2.1 Modelos de Predicción

Conceptualmente el modelo de un sistema real se puede dividir en dos submodelos:

- Un modelo del proceso que establezca la relación entre la salida de la planta y las entradas que podamos medir. Si hay más de una entrada medible al proceso, una de ellas será la variable manipulada y las otras las consideraremos como perturbaciones medibles. El control predictivo automáticamente compensa perturbaciones medibles mediante acción feedforward.
- Un modelo de la perturbación que trata de describir la parte de la dinámica del proceso que no se explica con el anterior modelo del proceso. En aplicaciones reales, estas perturbaciones se deben a entradas al procesos no medibles, perturbaciones del entorno, ruidos en las medidas, así como a errores en el modelado.

Esta división en dos submodelos no se suele realizar en los controladores predictivos, pero a la hora de su estudio tiene ventajas didácticas.

### 2.1.1 Modelos del Proceso

Se puede usar cualquier modelo que permita calcular predicciones, ya sean continuos o discretos; lineales o no lineales; función de transferencia, espacio de estados o convolución; con parámetros fijos o adaptativo. Por tanto, el control predictivo

es un campo abierto a la investigación. Actualmente los modelos más importantes son los siguientes.

### Variables de estado

El proceso se describe mediante las ecuaciones de estado

$$\begin{aligned}\mathbf{x}(k) &= \mathbf{A}\mathbf{x}(k-1) + \mathbf{B}\mathbf{u}(k-1) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k)\end{aligned}$$

donde  $\mathbf{x}(k)$  es el estado del proceso,  $\mathbf{A}$  es la matriz del sistema,  $\mathbf{B}$  es la matriz de entrada y  $\mathbf{C}$  es la matriz de salida.

Estos modelos son adecuados para analizar la estructura interna del proceso. El método es, además, multivariable, siendo un caso particular los sistemas SISO. Sin embargo, los cálculos son más complejos que con otros modelos, y si el estado no es directamente accesible, se ha de usar observador de estado.

Este modelo, como se verá, es el más general de los modelos lineales. A partir de él se pueden deducir los otros.

### Función de transferencia

El sistema se describe mediante una ecuación en diferencias

$$y(k) + a_1y(k-1) + \dots + a_{n_a}y(k-n_a) = b_1u(k-1) + \dots + b_{n_b}u(k-n_b)$$

o

$$A(z^{-1})y(k) = B(z^{-1})u(k)$$

A partir de una descripción en variables de estado y con condiciones iniciales nulas, se puede calcular su matriz de funciones de transferencia equivalente

$$y(z) = G(z)u(z)$$

donde

$$G(z) = C(zI - A)^{-1}B$$

Una ventaja de usar modelos de función de transferencia compartida por los modelos de variables de estado es que se pueden usar con procesos inestables o muy poco amortiguados. Por otra parte su número de parámetros es mínimo.

Una desventaja es que se ha de presuponer el orden del sistema ( $n_a$  y  $n_b$ ) cuando se quiere identificar el modelo.

### Modelos de convolución

Existen dos posible modelos de convolución, muy ligados entre sí. El primero es la respuesta impulso que tiene la siguiente forma:

$$y(k) = \sum_{j=1}^{+\infty} h_j u(k-j) \approx \sum_{j=1}^n h_j u(k-j) = H(z^{-1})u(k)$$

donde  $h_j$  son los valores de la respuesta impulso en los instantes de muestreo y  $H(z^{-1}) = h_1 z^{-1} + h_2 z^{-2} + \dots + h_n z^{-n}$ .

La respuesta impulso se puede deducir tanto a partir de la función de transferencia, dividiendo simplemente numerador por denominador, como a partir de la representación en variables de estado. Si la matriz  $A$  es estable la inversa en (2.1.1) se puede expandir en series de Neuman

$$G(z) = \sum_{j=0}^{\infty} CA^j B z^{-j-1} = \sum_{j=1}^{\infty} h_j z^{-j}$$

La ventaja de los modelos FIR (Finite Impulse Response) es que no requieren apenas información a priori. Además, pueden describir de forma sencilla procesos con dinámicas complicadas. Otra importante ventaja es, que al no haber términos recursivos, la predicción es simple y poco sensible a los errores.

La desventaja es que no se puede aplicar directamente a sistemas inestables, ya que para que sea útil el modelo los valores de  $h_j$  tienen que tender a cero cuando  $j \rightarrow \infty$ . Además, estos modelos tienen muchos parámetros ( $n$  es generalmente grande, de 30 a 50).

El segundo modelo de convolución es la respuesta escalón (step).

Suponiendo que la respuesta escalón se haya estacionado después de  $n$  intervalos de muestreo, el modelo es el siguiente,

$$y(t) = y_0 + \sum_{j=1}^n g_j \Delta u(t-j) = y_0 + G(q^{-1})(1 - q^{-1})u(t)$$

donde  $g_j$  son los valores de la respuesta escalón unitario en los instantes de muestreo y  $\Delta u(t) = u(t) - u(t-1)$ . Además,  $y_0$  es el estado inicial cuando se aplica el escalón que se supone constante. En general se toma cero.

Los coeficientes de la respuesta escalón y la respuesta impulso están relacionados por las siguientes ecuaciones:

$$\begin{aligned} g_j &= \sum_{i=1}^j h_i \quad (j = 1 \dots n; g_0 = 0) \\ h_j &= g_j - g_{j-1} \quad (j = 1 \dots n; h_0 = 0) \end{aligned}$$

Los modelos de respuesta escalón tienen las mismas ventajas y desventajas que los de respuesta impulso.

Estos modelos, dadas sus características, parecen muy adecuados para la industria de procesos, donde los procesos suelen ser estables en lazo abierto y no hay limitaciones en el tiempo de cálculo de la ley de control, ya que son muy lentos.

## Modelos no lineales

Los modelos lineales vistos han dado lugar a conocidos controladores predictivos. El uso de modelos no lineales es aún un tema de investigación. En la literatura

se encuentran desarrollos de controladores predictivos basados en modelos de EDO no lineales (continuas y discretas), modelos paramétricos no lineales, modelos de redes neuronales y modelos cualitativos.

Como se comentó en el capítulo anterior cada uno de estos modelos tiene sus ventajas y sus inconvenientes.

### 2.1.2 Modelos de perturbaciones

Los controladores predictivos son adecuados para eliminar perturbaciones en un sistema de control. Sin embargo, sólo funcionarán bien si se dispone de un buen modelo de la perturbación. Por ejemplo, si el modelo de la perturbación representa un ruido coloreado de media cero con un espectro de frecuencias dado, el controlador predictivo basado en este modelo dará un buen rendimiento eliminando esta perturbación, pero no será capaz de asegurar un control sin offset cuando se produzca una perturbación tipo escalón. La elección de un modelo de perturbación apropiado es extremadamente importante.

#### Perturbaciones constantes

En aplicaciones prácticas es muy importante conseguir un control sin offset en presencia de perturbaciones con media no nula. Un sencillo modelo para este tipo de perturbaciones es

$$n(t) = d$$

o

$$n(t) = \frac{\xi(t)}{1 - q^{-1}}$$

donde  $d$  es una constante desconocida y  $\xi(t)$  se puede interpretar como una serie de impulsos con amplitudes desconocidas producidos en instantes aleatorios de tiempo (ruido blanco de media cero). El segundo modelo describe por tanto una perturbación que va dando saltos aleatoriamente.

**Drifts**

Los modelos de estas perturbaciones son una extensión de los anteriores

$$n(t) = d_1 + d_2 t$$

o

$$n(t) = \frac{\xi(t)}{(1 - q^{-1})^2}$$

**Perturbaciones ARMA y ARIMA**

Un modelo muy general para estas perturbaciones es

$$n(t) = \frac{C(q^{-1})}{D(q^{-1})} \xi(t)$$

Este modelo se llama ARMA (Auto-Regresive Moving-Average). Cuando el polinomio  $D(q^{-1})$  contiene explícitamente el factor  $(1 - q^{-1})$ , que es necesario para obtener un control sin offset, el modelo se denomina ARIMA (Auto-Regresive Integrated Moving-Average).

Los modelos ARIMA incluyen todos los modelos de perturbaciones anteriores como casos especiales:

$$\begin{array}{ll} \text{Perturbaciones constantes:} & C(q^{-1}) = 1, \quad D(q^{-1}) = 1 - q^{-1} \\ \text{Perturbaciones drift:} & C(q^{-1}) = 1, \quad D(q^{-1}) = (1 - q^{-1})^2 \end{array}$$

**EJEMPLO: Predicción de la salida en IDCOM**

El controlador IDCOM (Richalet, 1978) usa un modelo del sistema de respuesta impulso y un modelo de perturbación constante.

$$y(t) = H(q^{-1})u(t) + n(t)$$

La predicción de la salida vendrá dada por

$$\hat{y}(t+k) = H(q^{-1})u(t+k) + n(t+k)$$

La mejor predicción de la perturbación cuando es un constante es

$$n(t+k) = n(t) = y(t) - H(q^{-1})u(t)$$

Por tanto, la predicción de la salida óptima a  $k$  pasos será

$$\begin{aligned}\hat{y}(t+k) &= H(q^{-1})u(t+k) + y(t) - H(q^{-1})u(t) \\ &= y(t)H(q^{-1})(u(t+k) - u(t))\end{aligned}$$

## 2.2 Trayectoria de Referencia

En los controladores predictivos se supone implícitamente que se conoce como queremos que varíe la salida del proceso:  $\{r(t+k/t), k = 1 \dots H_p\}$ . Hay aplicaciones como robótica, servo controles o procesos batch en las que normalmente se conoce como ha de variar la salida del proceso. En estos casos el rendimiento del sistema será excelente, ya que el controlador actuará en avance (antes que cambie la señal de consigna en el instante actual) para compensar retardos y desfases en el proceso (es lo que habíamos llamado "visión de futuro"). Si no se especifica la señal de consigna futura, siempre se puede tomar  $r(t+k/t) = r(t)$ .

Las trayectorias de referencia  $w(t+k/t)$  se inicializan con el valor actual de la salida del proceso:

$$w(t/t) = y(t)$$

y se calcula a lo largo del horizonte de predicción  $H_p$ , de manera que conduzca la salida del proceso hacia la salida deseada de acuerdo a una dinámica preespecificada. Generalmente se especifica una dinámica de primer orden:

$$w(t+k/t) = \alpha w(t+k-1/t) + (1-\alpha)r(t+k/t), \quad k = 1 \dots H_p$$

donde  $\alpha$  se suele denominar factor de filtrado. Es un parámetro de sintonía que toma valores  $0 \leq \alpha \leq 1$  y nos permite especificar la dinámica del lazo cerrado ( $\alpha \rightarrow 0$  : control rápido;  $\alpha \rightarrow 1$  : control lento). Obviamente el error que va a minimizar el control predictivo es la distancia entre la trayectoria de referencia  $w(t+k)$  y la predicción de la salida  $\hat{y}(t+k)$ .

El especificar una dinámica de la trayectoria de referencia tiene efectos sobre la dinámica del lazo cerrado y sobre la robustez. Esta dinámica se puede especificar de forma general mediante una función de transferencia de un filtro  $F(q^{-1})$ .

Así, la trayectoria de referencia se calculará, inicializando esta al valor de la salida de la planta y filtrando la señal de consigna mediante  $F$ . Hay dos aspectos importantes a tener en cuenta. El primero es, que si hay un seguimiento perfecto, o sea  $y(t) = w(t)$ , sería innecesario inicializar en cada instante de muestreo  $w(t)$  y por tanto, el filtro  $F$  actúa como un precompensador, usado en muchos sistemas de control. En segundo lugar, como la consigna ( $r(t)$ ) se filtra fuera del lazo de control, esto no va a tener ninguna influencia sobre la estabilidad del sistema.

## 2.3 Estructuración de la Ley de Control

Todo sistema dinámico estable y localmente linealizable, sometido a una entrada determinada, tiene una respuesta compuesta de dos términos: la respuesta libre y la respuesta forzada. Así, la predicción de la salida la podemos considerar como

$$y(t+k) = y_l(t+k) + y_c(t+k), \quad k = 1 \dots H_p$$

$y_l(t+k)$  es la respuesta libre suponiendo acciones futuras de control nulas,  $u_l(t+k) = u(t-1)$ ,  $k \geq 0$

$y_c(t+k)$  es la respuesta forzada o controlada. Es la parte de la salida que corresponde a las acciones de control futuras (que hay que determinar),  $u_c(t-k) = 0$ ,  $k < 0$

$$y \Delta u_c(t+k) = \Delta u(t+k), \quad k \geq 0$$

La respuesta libre  $y_l(t+k)$  se puede calcular de forma sencilla a partir de la ecuación de predicción apropiada, ya que sólo depende de acciones de control pasadas conocidas en el instante  $t$ .

La respuesta forzada  $y_c(t+k)$  es debida a los futuros incrementos de control  $\{\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+k-1)\}$ . Cada incremento representa un escalón aplicado a la entrada del proceso en los instantes  $t, t+1, \dots, t+k-1$ . Por tanto la contribución de estos incrementos a la salida del proceso, suponiendo que el proceso es lineal, es

$$y_c(t+k) = g_1 \Delta u(t+k-1) + g_2 \Delta u(t+k-2) + \dots + g_k \Delta u(t)$$

donde los parámetros  $g_i$  son los coeficientes de la respuesta escalón.

Los errores de control futuros se definen en forma matricial

$$\mathbf{E} = [\mathbf{e}(t+1)\mathbf{e}(t+2)\dots\mathbf{e}(t+H_p)]^T$$

donde  $e(t+k) = w(t+k) - y(t+k)$ . Si ahora definimos

$$\begin{aligned} \mathbf{E}_0 &= [w(t+1) - y_l(t+1) \dots w(t+H_p) - y_l(t+H_p)]^T \\ \mathbf{G} &= \begin{bmatrix} g_1 & 0 & \dots & 0 \\ g_2 & g_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{H_p} & g_{H_p-1} & \dots & g_1 \end{bmatrix} \\ \mathbf{U} &= [\Delta u(t)\Delta u(t+1), \dots, \Delta u(t+H_p-1)]^T \end{aligned} \tag{2.1}$$

el vector de errores futuros lo podemos reescribir como

$$\mathbf{E} = \mathbf{E}_0 - \mathbf{G}\mathbf{U}$$

Independientemente del criterio que usemos para minimizar los errores en  $\mathbf{E}$ , siempre será una función con  $H_p$  variables independientes (el número de incrementos de control futuros). Teniendo en cuenta que el horizonte de control suele

estar en el rango de 10 a 30 muestras, solucionar este problema de optimización será computacionalmente complejo.

Esta es una razón para introducir una cierta estructuración de la ley de control, o sea eliminar grados de libertad especificando alguna relación entre los futuros incrementos de control.

Otra razón muy importante para estructurar la ley de control es que si dejamos la variable manipulada totalmente libre, esto puede producir una señal de control de alta frecuencia. En general, esto es indeseable en sistemas reales que suelen tener una característica pasa-baja y por tanto no aportaría ningún beneficio en cuanto al rendimiento del sistema de control y por contra podría llegar a producir inestabilidad si se tubieran dinámicas sin modelar de alta frecuencia (que suele ser habitual).

Existen varias técnicas de estructurar la ley de control. Se presentan a continuación dos de estas técnicas.

### 2.3.1 Horizonte de control

En el caso que tengamos una señal de consigna constante es de esperar que la señal de control también tienda a ser constante. Por tanto, es razonable especificar un horizonte de control  $H_c$ , a partir del cual la señal de control permanezca constante:  $\Delta u(t+k) = 0$  para  $k \geq H_c$ . El resultado será un vector de control de dimensión  $H_c$  en vez de  $H_p$ .

Desde el punto de vista computacional, lo que conseguimos con esta estrategia es eliminar las últimas  $H_p - H_c$  columnas de la matriz de respuestas escalón  $\mathbf{G}$ . Esto reduce la complejidad del problema ya que  $H_c$  es en general mucho menor que  $H_p$ .

Un caso límite de esta estrategia es cuando  $H_c = 1$ . en este caso sólo se ha

de calcular  $\Delta u(t)$  lo que lleva a una solución escalar extremadamente simple.

### 2.3.2 Funciones base

Suponer que la señal de control será constante en el futuro puede que no sea una buena elección para cualquier tipo de sistema. En el caso de servo sistemas con señales de consigna variables parece más adecuado usar algún otro tipo de estructuración que tenga en cuenta esta característica y mejore el rendimiento del sistema de control. La técnica que se presenta se basa en calcular la señal de control como una combinación lineal de funciones base preespecificadas:

$$u(t+k) = \sum_{i=1}^m \mu_i B_i(k), \quad k = 0, \dots, H_p - 1$$

La selección de las funciones base  $B_i$  depende de la naturaleza del proceso (tipo 0, tipo 1, etc.) y del tipo de señal de consigna. Las funciones base que se suelen utilizar son de tipo polinomial, a veces retardadas en el tiempo

$$\begin{aligned} B_1(k) &= \gamma(k) \\ B_2(k) &= \gamma(k)k \\ B_3(k) &= \gamma(k)k^2 \\ B_4(k) &= \gamma(k-n) \end{aligned}$$

donde  $\gamma(k)$  es la función escalón unitario. Así, el algoritmo de control ha de determinar los  $m$  coeficientes  $\mu$ . Si las funciones base utilizadas son escalones la ley de control resultante es equivalente a una que use la técnica del horizonte de control.

## 2.4 Cálculo de la Señal de Control

### 2.4.1 La función de coste

En los controladores predictivos la ley de control surge de la minimización de una función de coste que pesa como mínimo la diferencia entre la trayectoria de referencia futura y la predicción de salida del proceso.

Una sencilla función coste es la siguiente:

$$J = \sum_{i=1}^{H_p} (\hat{y}(k+i) - w(k+i))^2 + \rho u(k+i-1)^2$$

donde  $\rho$  es un factor de peso  $\geq 0$ . Este criterio tiene dos objetivos contrapuestos: la minimización del error de seguimiento y la minimización de la salida del controlador. El factor de peso  $\rho$  se introduce para establecer un compromiso entre estos dos objetivos.

Por otra parte, pesando la salida del controlador directamente producirá errores en estado estacionario si el proceso no contiene ningún integrador. Este efecto indeseable (para perturbaciones constantes) se puede evitar pesando en su lugar los incrementos de la señal de control:

$$J = \sum_{i=1}^{H_p} (\hat{y}(k+i) - w(k+i))^2 + \rho \Delta u(k+i-1)^2$$

Si la técnica que se ha usado para estructurar la ley de control es la del horizonte de control entonces en la función de peso anterior se tiene restricción adicional: en un cierto intervalo, determinado por el horizonte de control  $H_c$  (donde la señal de control puede variar libremente) y el horizonte de predicción, los incrementos del controlador se suponen ceros:

$$\Delta u(k+i-1) = 0 \quad 1 \leq H_c < i \leq H_p$$

Por tanto, se calculan  $u(k) \dots u(k+H_c-1)$  y se suponen constantes el resto de controles sobre el horizonte de predicción. La razón de introducir esta restricción adicional

es que el horizonte de control  $H_c$  es fácil de seleccionar y en muchas situaciones hace que el factor de peso  $\rho$  de la salida del controlador sea superfluo.

Otra extensión útil que se puede a la función de coste anterior es introducir un intervalo  $H_s$  que llamaremos horizonte de coste mínimo en los que los errores de seguimiento (predichos) no se pesan:

$$J = \sum_{i=H_s}^{H_p} (\hat{y}(k+i) - w(k+i))^2 + \sum_{i=0}^{H_c-1} \rho \Delta u(k+i-1)^2$$

Así, los errores de las primeras  $H_s - 1$  muestras no se tienen en cuenta en la optimización. La razón de hacer esto es que si el proceso presenta algún retardo no tiene sentido minimizar unos errores que no van a estar afectados por la señal de control que estamos calculando. Al intervalo  $H_s - H_p$  se denomina horizonte de coincidencia.

La forma más general de la función de coste, que incluye pesos en el error de seguimiento y las acciones de control es

$$J = \sum_{i=H_s}^{H_p} \alpha_i (\hat{y}(k+i) - w(k+i))^2 + \sum_{i=0}^{H_c-1} \beta_i \Delta u(k+i-1)^2$$

El vector de coeficientes de peso del error de seguimiento  $\alpha = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_{H_p}]^T$  introduce varias alternativas:

- eligiendo  $\alpha = [0 \ 0 \ \dots \ 0 \ 1 \ 1 \ \dots \ 1]^T$  definimos un horizonte de coincidencia.
- eligiendo el vector de peso de manera que sólo algunos elementos sean distintos de cero  $\alpha = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ \dots \ 0 \ 1]^T$  nos conduce al concepto de *puntos de coincidencia*.
- eligiendo  $\alpha_k = \lambda^k$  pesaremos de forma exponencial los errores de seguimiento en el horizonte de predicción.

Los coeficientes de peso de la señal de control  $\beta_k$  se pueden usar para mejorar la robustez y solucionar algunos problemas numéricos. Un inconveniente que

presentan es que es difícil establecer a priori su valor óptimo. Muchas veces se escogen todos los  $\beta_k$  iguales y así sólo hay que sintonizar un parámetro. De todas formas, usando una técnica adecuada de estructuración de la ley de control es innecesario usar estos coeficientes.

### 2.4.2 Restricciones

En muchos casos prácticos hay restricciones en la salida del controlador debido a que los conversores D/A y actuadores tienen un rango limitado. Por ejemplo, cuando se usa un controlador para mantener el rumbo de un barco, la señal de control va al mecanismo de guiado que controla el timón. Este mecanismo de guiado tiene dos restricciones fundamentales: el ángulo del timón (que está limitado por ejemplo entre más menos 30 grados) y la velocidad de éste (limitado, por ejemplo, a 10 grados por segundo). La primera restricción se llama restricción de nivel mientras que la segunda restricción de velocidad. Las restricciones de velocidad tienen particularmente un efecto desastroso en la estabilidad del lazo cerrado, ya que este tipo de restricciones producen grandes retardos de fase cuando están activas.

Por tanto, una característica fundamental en muchos problemas prácticos de control es la presencia de restricciones en las variables de entrada o de salida, o incluso en estados del proceso. Además de restricciones debidas a limitaciones en los equipos de control, también aparecen comunmente debidas a razones de calidad del producto, seguridad de equipos o personas, regulaciones medio ambientales, etc.

La inclusión de restricciones en el diseño de controladores CPBM es la característica más esencial que los distingue de otras técnicas de diseño. De hecho, si no hay restricciones, la mayoría de controladores CPBM son equivalentes a un controlador LTI clásico. En particular el Control Predictivo Generalizado (GPC) es idéntico matemáticamente a un LQG.

Las restricciones más habituales son en la salida del proceso, la señal de

control y los incrementos de control

$$\begin{aligned}\underline{Y}_k &\leq \hat{y}(t+k) \leq \overline{Y}_k & 1 \leq k \leq H_p \\ \underline{U}_k &\leq u(t+k) \leq \overline{U}_k & 0 \leq k \leq H_c - 1 \\ \underline{\Delta u} &\leq \Delta u(t+k) \leq \overline{\Delta u} & 0 \leq k \leq H_c - 1\end{aligned}$$

### 2.4.3 La Ley de Control Predictiva

La ley de control óptima se calcula minimizando la función de coste respecto a la secuencia de salida del controlador sobre el horizonte de predicción  $H_c$  :  $u(k), \dots, u(k + H_c - 1)$ . Si el modelo del proceso es lineal, la función de coste cuadrática y no hay restricciones, la minimización se puede hacer analíticamente.

Si queremos minimizar una función de coste  $J$  respecto al vector  $\mathbf{u}$ , cualquier mínimo local satisfará la condición:

$$\mathbf{g} = \frac{\partial J}{\partial \mathbf{u}} = \mathbf{0}$$

donde  $\mathbf{g}$  es el gradiente. La solución de la ecuación anterior será particular para cada función de coste que a su vez corresponde a cada uno de los métodos predictivos.

En el caso que tengamos restricciones, la minimización de la función de coste no se puede realizar analíticamente, hay que recurrir a métodos numéricos.

Si la función de coste es cuadrática (usa la norma 2) la solución del problema se obtiene mediante programación cuadrática. Esta solución requiere una gran potencia de cálculo, pero tiene la ventaja que cuando las restricciones no están activas se pueden usar herramientas estándar para el análisis del sistema de control.

Otra posibilidad es usar la norma 1 en la función de coste, en este caso la solución se obtiene mediante programación lineal. Esta técnica es computacionalmente más simple que la programación cuadrática. Tiene el grave inconveniente que incluso cuando las restricciones no están activas no es posible un análisis de estabilidad y rendimiento ya que no se puede expresar como un controlador LTI.

Una característica interesante de CPBM es que las señales del proceso futuras (predicciones de la salida y señales de control futuras) se pueden calcular explícitamente y comprobar si violan alguna restricción. De esta forma se pueden establecer técnicas ad hoc para el manejo de restricciones que sean más simples en cuanto a necesidades de cálculo que las dos anteriores.

## 2.5 Compensación Feedforward

Una característica importante de CPBM que ya se comentó, es la capacidad de introducir de forma natural compensación feedforward. Para ello se necesita un modelo de la dinámica entre una perturbación medible y la salida del proceso. Por ejemplo consideremos el caso de un modelo en variables de estado,

$$\begin{aligned}x(t) &= Ax(t-1) + Bu(t-1) + Fv(t-1) \\y(t) &= Cx(t)\end{aligned}$$

donde  $v(t)$  representa una perturbación medible.

Para realizar la predicción se ha de hacer alguna suposición sobre los valores futuros de la perturbación. En algunos casos estos valores son conocidos, por ejemplo, en procesos batch, cambios de flujo preprogramados, variaciones de la producción planificados, etc. Muchas veces la acción feedforward se usa para compensar perturbaciones de carga, en este caso es razonable suponer que la perturbación será constante en el futuro  $v(t+k) = v(t)$ .

La compensación feedforward puede ser una alternativa al control multivariable. Podemos considerar las acciones de control de los otros lazos como perturbaciones medibles e integrarlas en el modelo. Así, teniendo en cuenta las perturbaciones medibles en el modelo de predicción, las técnicas CPBM tratan de compensar automáticamente este efecto perturbador.

## 2.6 Control Predictivo Generalizado (GPC)

El método que se describe a continuación (GPC) queda dentro del concepto de control predictivo y por tanto tiene las características comunes a todos estos métodos. El GPC surgió como una propuesta de algoritmo de "propósito general" de control estabilizante para una gran mayoría de procesos reales. En particular, es capaz de conseguir un control estable con procesos con parámetros variables, retardos variables y con órdenes de modelos que cambian instantáneamente siempre que los datos de entrada salida sean suficientemente ricos para conseguir una identificación de la planta razonable. Por tanto, es un método adecuado para control adaptativo.

### 2.6.1 Descripción de la Planta

El primer elemento necesario en un método de control predictivo es un modelo adecuado. El modelo usado en el GPC es un CARIMA

$$A(q^{-1})y(t) = B(q^{-1})u(t) + D(q^{-1})v(t) + \frac{T(q^{-1})}{\Delta}\xi(t) \quad (2.2)$$

donde  $q^{-1}$  es el operador retardo y  $\Delta$  es el operador diferencia  $(1 - q^{-1})$ . El polinomio  $B(q^{-1})$  tendrá los primeros elementos ceros correspondientes al retardo de la planta real. La señal  $v(t)$  corresponde a una perturbación medible que no estaba incluida en la formulación inicial del GPC debida a D. W. Clarke. El incluir esta señal en el modelo nos permitirá de forma automática introducir acción feedforward en la ley de control, ya que el controlador tratará de compensar estas posibles perturbaciones.

Hay varias interpretaciones del término de ruido en este modelo:

1. Considerando  $T = 1$  y  $\xi(t)$  un ruido blanco, se puede considerar la perturbación como escalones aleatorios actuando sobre la entrada de la planta. Esta es una buena representación para una típica perturbación en carga.

2. Si  $\xi(t)$  es una secuencia de variables aleatorias independientes y  $T$  es el polinomio  $C(q^{-1})$  (conocido o estimado), la perturbación modelada será un ruido coloreado actuando sobre la salida. Una predicción basada en este modelo tendrá propiedades óptimas (como varianza mínima).
3.  $T(q^{-1})$  se puede escoger como un polinomio de diseño fijo. De esta manera las predicciones no serán óptimas, pero con una buena elección podemos conseguir otras importantes propiedades tales como robustez frente a dinámicas sin modelar.

### 2.6.2 Predicción de la Salida

La predicción de la salida  $\{y(t+j/t), k = 1 \dots H_p\}$  depende de la entrada de control futura que se asuma  $\{u(t+k/t), k = 0 \dots H_p - 1\}$ . Conceptualmente la señal de control se puede considerar como la suma de dos señales:

$$u(t) = u_f(t) + u_c(t)$$

siendo  $u_f(t)$  la señal de control suponiendo que no hay acciones de control futuras  $u_f(t+j) = u(t-1)$ ,  $j \geq 0$  y  $u_c(t)$  la futura señal de control que está por determinar  $u_c(t-j) = 0$ ,  $j < 0$  y  $\Delta u_c(t+j) = \Delta u(t+j)$ ,  $j \geq 0$ .

El modelo de predicción adoptado lo podremos reescribir de la siguiente manera:

$$A(q^{-1})y(t) = B(q^{-1})u_c(t) + B(q^{-1})u_f(t) + D(q^{-1})v(t) + \frac{T(q^{-1})}{\Delta}\xi(t)$$

y la salida del sistema podemos considerarla como la suma de dos efectos, cada uno debido a una de las partes en que hemos dividido la señal de control:

$$y(t+j) = y_i(t+j) + y_c(t+j)$$

donde

$$y_i(t+j) = \frac{B(q^{-1})}{A(q^{-1})}u_f(t+j) + \frac{D(q^{-1})}{A(q^{-1})}v(t+j) + \frac{T(q^{-1})}{\Delta A(q^{-1})}\xi(t+j)$$

es la respuesta libre suponiendo que no hay acciones futuras de control y

$$y_c(t+j) = \frac{B(q^{-1})}{A(q^{-1})} u_c(t+j)$$

es la parte controlada de la salida del sistema debidas a las futuras acciones de control.

La parte controlada de la respuesta viene dada por:

$$y_c(t+j) = (h_0 + h_1q^{-1} + h_2q^{-2} + \dots)u_c(t+j)$$

donde los parámetros  $h_i$  son los valores de la respuesta impulso

$$H(q^{-1}) = h_0 + h_1q^{-1} + h_2q^{-2} + \dots = \frac{B(q^{-1})}{A(q^{-1})}$$

Si tenemos en cuenta la relación que existe entre la respuesta impulso y la respuesta escalón

$$G(q^{-1}) = \frac{H(q^{-1})}{1 - q^{-1}}$$

podemos calcular la respuesta escalón si disponemos de la función de transferencia

$$\begin{aligned} y_c(t+j) &= \frac{B(q^{-1})}{A(q^{-1})} u_c(t+j) \\ &= g_0 \Delta u_c(t+j) + g_1 \Delta u_c(t+j-1) + \dots + g_j \Delta u_c(t) \\ &= (g_0 + g_1 q^{-1} + \dots + g_j q^{-j}) \Delta u_c(t+j) \\ &= G_j(q^{-1}) \Delta u_c(t+j), \quad j \geq 0 \end{aligned}$$

La respuesta libre  $y_l(t+k)$  se puede calcular fácilmente de la forma que se explica a continuación. La respuesta libre dependerá de los valores pasados de la señal de control, pero estos son conocidos en el tiempo  $t$ . Respecto a la perturbación medible se habrá de realizar alguna suposición sobre sus valores futuros.

$$\begin{aligned} A(q^{-1}) \frac{\Delta}{T(q^{-1})} y_l(t+j) &= B(q^{-1}) \frac{\Delta}{T(q^{-1})} u_f(t+j) \\ &\quad + D(q^{-1}) \frac{\Delta}{T(q^{-1})} v(t+j) + \xi(t+j) \end{aligned}$$

Vemos que la respuesta libre depende del polinomio observador  $T(q^{-1})$ . Para simplificar la predicción definimos primero las siguientes variables:

$$\hat{y}_f = \frac{\Delta}{T} y_l, \quad \hat{u}_f = \frac{\Delta}{T} u_f, \quad \hat{v} = \frac{\Delta}{T} v_f$$

Así, filtrando la entrada y la perturbación conocida a través de  $\Delta/T$  y suponiendo que disponemos de una predicción de  $v(t + j|t)$ , podemos calcular la predicción de la salida filtrada  $\tilde{y}_f(t + j|t)$ ,

$$A\tilde{y}_f(t + k|t) = B\tilde{u}_f(t + j|t) + D\tilde{v}(t + j|t)$$

La predicción de la respuesta libre se hace basándose en este modelo y haciendo dos suposiciones:

- $\xi(t + j|t) = 0, j > 0$  siendo  $\{\xi\}$  una secuencia de ruido no correlacionada.
- $\Delta u(t + j) = 0, j \geq 0$ , manteniendo  $u$  constante e igual a  $u(t-1)$  para la respuesta libre.

Haciendo ahora la desconvolución de la salida filtrada a través de  $T/\Delta$  obtendremos la predicción que buscábamos  $\hat{y}_f(t + j|t) = p_j$ .

Por tanto, para el cálculo de las predicciones desde  $j = 1$  hasta el horizonte de predicción se hace:

1. Mantener los valores filtrados de los datos de entrada\salida disponibles.
2. Repetir sobre el horizonte de predicción de nuestro modelo (A.B.D) usando los datos filtrados existentes como condiciones iniciales y suponiendo incrementos de la señal de control nulos.
3. Hacer la desconvolución de la secuencia resultante.

La predicción de la salida total será la suma de la parte controlada y la parte libre,

$$\hat{y}(t + j) = +G_j(q^{-1})\Delta u_c(t + j) + p_j, \quad j \geq 0$$

Combinando todas las salidas  $\hat{y}(t+h)$ ;  $j = 1 \dots H_p$  en un vector  $\mathbf{y}$  conseguimos lo que llamaremos *ecuación fundamental*

$$\hat{\mathbf{y}} = \mathbf{G}\Delta\mathbf{u} + \mathbf{p}$$

donde

$$\begin{aligned} \mathbf{y} &= [y(t+1), y(t+2), \dots, y(t+H_p)]^T \\ \Delta\mathbf{u} &= [\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+H_p-1)]^T \\ \mathbf{p} &= [p_1, p_2, \dots, p_{H_p}]^T \end{aligned}$$

La matriz  $\mathbf{G}$  es la matriz de respuestas escalón que definimos en 2.1.

### 2.6.3 El Algoritmo de control

Consideremos el vector  $\mathbf{e}$  compuesto de los predecibles futuros errores del sistema  $w(t+j) - \hat{y}(t+j)$ . El GPC calcula la secuencia de control  $\{u(t+j)\}$  en el instante  $t$ , minimizando la siguiente función de coste

$$J(H_s, H_p, H_c, \lambda) = \sum_{j=H_s}^{H_p} e^2(t+j) + \lambda \sum_{j=1}^{H_c} \Delta u^2(t+j-1) \quad (2.3)$$

donde

- $H_s$  es el horizonte de predicción mínimo.
- $H_p$  es el horizonte de predicción máximo.
- $H_c$  es el horizonte de control.
- $\lambda$  es un factor de peso del control.

sujeto a la restricción de que los incrementos de control  $\Delta u(t+j)$  para  $j > H_c$  son cero. Además, para conseguir un comportamiento del sistema aceptable se han de escoger valores para los horizontes de manera que  $H_s > 1$  y  $H_c < H_p$ .

El GPC (como otros MBPC) usa una estrategia de horizonte móvil, donde en cada instante de muestreo  $t$ , se aplica solamente el primer valor de la secuencia de control calculada  $\Delta u(t)$ , repitiéndose todo el algoritmo en el siguiente instante.

Para minimizar la función de coste el primer paso es expresar ésta en forma matricial. La predicción de la salida quedará según el apartado anterior y teniendo en cuenta los horizontes que hemos fijado

$$\hat{y} = \mathbf{G}\Delta\mathbf{u} + \mathbf{p}$$

donde  $\Delta\mathbf{u} = [\Delta u(t), \dots, \Delta u(t + H_c - 1)]^T$ , y

$$\mathbf{G} = \begin{pmatrix} g_{H_s} & g_{H_s-1} & \cdots & 0 \\ g_{H_s+1} & g_{H_s} & g_{H_s-1} & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{H_p} & g_{H_p-1} & \cdots & g_{H_p-H_c+1} \end{pmatrix} \quad (2.4)$$

Aquí  $\mathbf{G}$  es una matriz  $(H_p - H_s + 1) \times H_c$  con los elementos  $g_{ij}$  tal que  $j - i > H_s$  iguales a cero. Sustituyendo la ecuación anterior en el índice a minimizar resulta

$$J = (\mathbf{w} - \mathbf{G}\Delta\mathbf{u} - \mathbf{p})^T(\mathbf{w} - \mathbf{G}\Delta\mathbf{u} - \mathbf{p}) + \lambda\Delta\mathbf{u}^T\Delta\mathbf{u}$$

Si ahora llamamos  $\mathbf{E}_0 = \mathbf{w} - \mathbf{p}$  y desarrollamos la expresión

$$J = \mathbf{E}_0^T\mathbf{E}_0 - \Delta\mathbf{u}^T\mathbf{G}^T\mathbf{E}_0 - \mathbf{E}_0^T\mathbf{G}\Delta\mathbf{u} + \Delta\mathbf{u}^T[\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I}]\Delta\mathbf{u}$$

La solución de  $J_{min}$  si no hay restricciones es

$$\frac{\partial J}{\partial \Delta\mathbf{u}} = \frac{1}{2}[\mathbf{E}_0\mathbf{G}\Delta\mathbf{u}_{opt}] + \frac{\lambda}{2}\Delta\mathbf{u}_{opt} = 0$$

agrupando términos

$$\mathbf{E}_0 + [\lambda\mathbf{I} - \mathbf{G}]\Delta\mathbf{u}_{opt} = 0$$

finalmente despejando  $\Delta\mathbf{u}_{opt}$  obtenemos

$$\Delta\mathbf{u}_{opt} = (\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I})^{-1}\mathbf{G}^T\mathbf{E}_0 \quad (2.5)$$

Esta expresión nos permite calcular el valor óptimo de los incrementos de control futuros.

## Capítulo 3

# Aspectos de Implementación de CPBM

En el presente capítulo se analizan varios aspectos de implementación de controladores predictivos. El primero es relativo al DMC (Dynamic Matrix Controller). Cuando se deducen las ecuaciones de la respuesta libre del sistema, las expresiones resultantes son diferentes si se considera un modelo de respuesta escalón con un número infinito de parámetros o una con un número finito. Como la implementación necesariamente se ha de hacer con un número limitado de términos, en algunos casos nos encontramos con respuestas indeseables.

El segundo aspecto que consideraremos será la influencia de los parámetros de diseño en la forma de la función de coste que hay que minimizar para obtener los incrementos de control. Este es un aspecto general para cualquier controlador predictivo.

Por último se estudia la implementación de controladores predictivos multivariables. Una característica de CPBM es que es fácil incorporar acción feedforward. Aprovechando esta característica se ha implementado, por una parte, un control multivariable mediante controladores SISO con acción feedforward que com-

pensa la interacción de los otros lazos. Y por otra, se presenta una forma de hacer frente a restricciones en alguna salida del proceso no controlada.

### 3.1 Formulación del DMC

El DMC se basa en un modelo de respuesta escalón del proceso. Para un sistema SISO el modelo es

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t-i) + n(t)$$

donde  $y(t)$  es la salida del proceso,  $u(t)$  la variable manipulada,  $g_i$  los coeficientes correspondientes a la respuesta escalón del proceso y  $n(t)$  es una perturbación actuando en el instante  $t$ .

La predicción de la salida vendrá dada por

$$y(t+k) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + n(t+k) \quad (3.1)$$

si ahora suponemos que la mejor estimación de los valores futuros de la perturbación  $n(t+k)$  es el valor actual  $n(t)$

$$n(t+k) = n(t) = y(t) - \sum_{i=1}^{\infty} g_i \Delta u(t-i) \quad (3.2)$$

entonces la predicción óptima a  $k$  pasos será

$$\hat{y}(t+k) = y(t) + \sum_{i=1}^{\infty} g_i (\Delta u(t+k-i) - \Delta u(t-i)) \quad (3.3)$$

Para realizar el cálculo de la señal de control conviene dividir la predicción en dos partes, una correspondiente a la respuesta libre y otra a la respuesta forzada, y así conseguimos separar los términos que dependen de acciones de control futuras de las ya conocidas. La predicción de la respuesta libre es

$$\hat{y}_f(t+k) = y(t) + \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) - \sum_{i=1}^{\infty} \Delta g_i u(t-i) \quad (3.4)$$

$$= y(t) + \sum_{j=1-k}^{\infty} g_{j+k} \Delta u(t-j) - \sum_{i=1}^{\infty} \Delta g_i u(t-i)$$

teniendo en cuenta que los incrementos futuros de la señal de control son cero  $\Delta u(t-i) = 0$ ,  $i$

$$\begin{aligned} \hat{y}(t+k)_f &= y(t) + \sum_{i=1}^{\infty} g_{i+k} \Delta u(t-i) - \sum_{i=1}^{\infty} \Delta g_i u(t-i) \\ &= y(t) + \sum_{i=1}^{\infty} (g_{i+k} - g_i) \Delta u(t-i) \end{aligned} \quad (3.5)$$

Debido a que el controlador hay que implementarlo en un ordenador, no se pueden usar infinitos términos de la respuesta escalón. Para procesos asintóticamente estables, los coeficientes  $g_i$  tienden a un valor constante. Por tanto, el sumatorio se puede truncar

$$\hat{y}_f(t+k) = y(t) + \sum_{i=1}^N (g_{i+k} - g_i) \Delta u(t-i) \quad (3.6)$$

donde  $N$  ha de ser un número suficientemente grande para que

$$g_{i+k} - g_i \approx 0, \quad i > N, \quad k = N1 \dots N2 \quad (3.7)$$

donde  $N1 - N2$  es el horizonte de coincidencia. Nótese que, si el proceso no es asintóticamente estable, entonces  $N$  no existe y  $\hat{y}_f(t+k)$  no se puede calcular y por lo tanto el control DMC no se podrá llevar a cabo.

La segunda parte de la predicción de la salida corresponde a la respuesta forzada. Esta es debida a los futuros incrementos de control. Por tanto, tendrá la forma,

$$\begin{aligned} \hat{y}_c(t+k) &= g_k(\Delta u(t) + g_{k-1}(\Delta u(t-1) + \dots + g_1(\Delta u(t+k-1) \\ &= G_k(q^{-1}) \Delta u(t+k) \end{aligned} \quad (3.8)$$

donde  $G_k(q^{-1})$  es un polinomio de orden  $j$  en el operador retardo  $q^{-1}$ ,

$$G_k(q^{-1}) = g_1 q^{-1} + \dots + g_k q^{-k} \quad (3.9)$$

Si ahora llamamos  $\hat{y}_f(t+k) = p_j$  la predicción total de la salida será

$$\hat{y}(t+k) = G_k(q^{-k}) \Delta u(t+k) + p_j \quad (3.10)$$

que resulta ser una expresión idéntica a la que nos aparecía en el GPC.

La señal de control  $u(t)$  se obtiene de la minimización respecto de  $\Delta u(t+k)$  de la función de coste

$$J = \sum_{k=N_1}^{N_2} (\hat{y}(t+k) - w(t+k))^2 + \sum_{k=0}^{N_u-1} (\lambda \Delta u(t+k))^2 \quad (3.11)$$

donde  $w(t)$  es la trayectoria de referencia y  $\lambda$  un factor de peso. Además el DMC usa la técnica del horizonte de control  $\Delta u(t+k) = 0$  para  $k \geq N_u$  siendo  $N_u$  el horizonte de control.

Dado que tanto la función de coste como la ecuación de predicción de la salida son iguales que en el GPC la deducción de la señal de control óptima dará el mismo resultado que el GPC.

## 3.2 Implementación del DMC

Algunas veces las ecuaciones del DMC se deducen a partir de un modelo de respuesta impulso truncado (Morari 1989),

$$y(t) \approx \sum_{i=1}^N h_i u(t-i) + n(t) \quad (3.12)$$

aplicando la relación que existe entre los coeficientes de la respuesta impulso y la respuesta escalón  $h_i = g_i - g_{i-1}$  y teniendo en cuenta que el primer elemento de la respuesta escalón es  $g_0 = 0$  resulta el siguiente modelo

$$\begin{aligned} y(t) &= \sum_{i=1}^N (g_i - g_{i-1}) u(t-i) + n(t) \\ &= \sum_{i=1}^N g_i u(t-i) - \sum_{i=0}^{N-1} g_{i-1} u(t-i-1) + n(t) \\ &= \sum_{i=1}^{N-1} g_i \Delta u(t-i) - g_N u(t-N) + n(t) \end{aligned} \quad (3.13)$$

Esta expresión es válida para procesos asintóticamente estables suponiendo que  $N$  sea suficientemente grande. Sin embargo, veamos que ocurre si la usamos para calcular las predicciones de la respuesta libre. La estimación óptima de la perturbación  $n(t)$  será

$$n(t+k) = n(t) = y(t) - \sum_{i=1}^{N-1} g_i \Delta u(t-i) - g_N u(t-N) \quad (3.14)$$

y la predicción de la respuesta libre, por tanto

$$\begin{aligned} \hat{y}_f(t+k) &= \sum_{i=1}^{N-1} g_i \Delta u(t-i) - g_N u(t-N) \\ &+ y(t) - \sum_{i=1}^{N-1} g_i \Delta u(t-i) - g_N u(t-N) \end{aligned}$$

Manipulando esta expresión como lo en la hicimos sección anterior

$$\begin{aligned} \hat{y}_f(t+k) &= y(t) + \sum_{i=1}^{N-k-1} (g_{i+k} - g_i) \Delta u(t-i) - \sum_{i=N-k}^{N-1} g_i \Delta u(t-i) \\ &+ g_N (1 - q^{-k}) u(t+k-N) \end{aligned} \quad (3.15)$$

Esta expresión que es similar a (3.5) es inexacta incluso si  $N$  es suficientemente grande para incluir coeficientes de la respuesta impulso  $h_i$  que sean prácticamente cero.

Obsérvese que en (3.5) los cambios pasados en la señal de control son amortiguados por el factor  $g_{i+k} - g_i$  que tiende a cero cuando  $i$  se aproxima a  $N$ . Sin embargo, en (3.15), estos cambios están multiplicados por  $g_i$  que en general no serán nulos y que provocan unas predicciones incorrectas. Esto hará que el controlador genere una mala señal de control incluso después que el sistema haya alcanzado el estado estacionario y en ausencia de perturbaciones.

Vamos a observar este problema en un ejemplo. Para ello usaremos un modelo *benchmark* propuesto en la literatura ([Abu 90]) para comparar sistemas de control. Este sistema es de fase no mínima, de respuesta inversa y presenta un retardo,

$$G(s) = \frac{K e^{-\tau s} \omega_0 (sT_n - 1)}{(sT_n + 1)(s^2 + 2\xi\omega_0 s + \omega_0^2)} \quad (3.16)$$

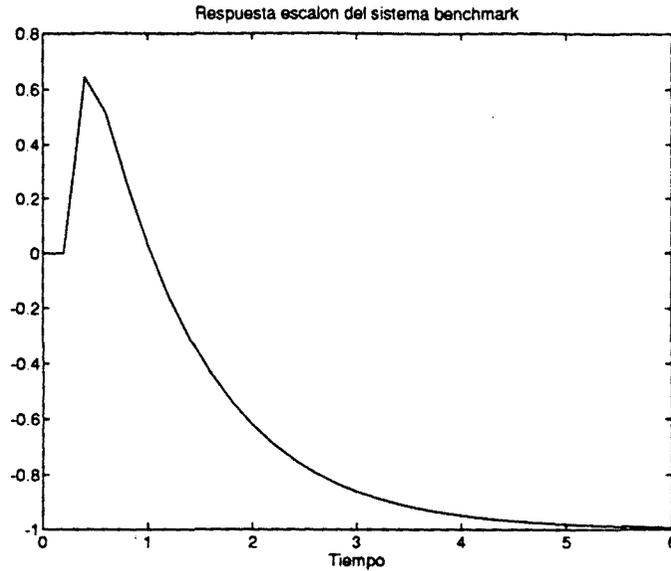


Figura 3.1: Respuesta escalón del sistema benchmark

con  $K = 1$ ,  $\tau = 0.2$ ,  $\omega_0 = 15$  y  $\xi = 1$ . En la Fig. 3.1 se muestra la respuesta escalón en lazo abierto de este sistema, que efectivamente tiene una dinámica compleja.

La discretización de este modelo para un periodo de muestreo de 0.2 seg. y con un retenedor de orden cero (ZOH) es

$$G(q^{-1}) = \frac{q^{-2}(0.6445 - 0.7176q^{-1} - 0.0906q^{-2})}{1 - 0.9183q^{-1} + 0.084q^{-2} - 0.002q^{-3}} \quad (3.17)$$

Se han realizado dos experimentos para comparar las respuestas de dos D-MC, uno haciendo las predicciones según (3.5), y otro según (3.15). En ambos casos se ha tomado el mismo número de puntos de la respuesta escalón  $N = 50$  y el resto de parámetros de sintonía han sido  $N1 = 6$  de manera que no se tuviera en cuenta la parte de la respuesta inversa,  $N2 = 30$ ,  $Nu = 1$  y sin ningún filtro en la señal de consigna ni factor de peso.

En la Fig. 3.2, usado la ecuación de predicción (3.5) podemos ver que la salida alcanza la señal de consigna y permanece estable. En la Fig. 3.3, usado la

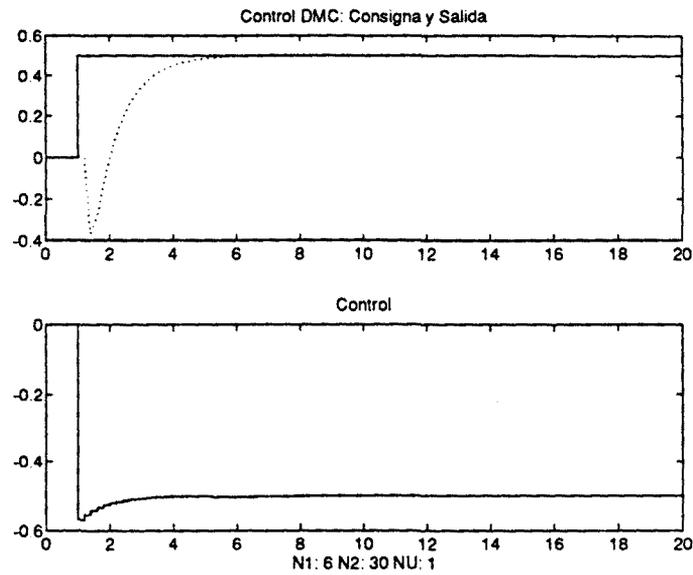


Figura 3.2: Respuesta del DMC

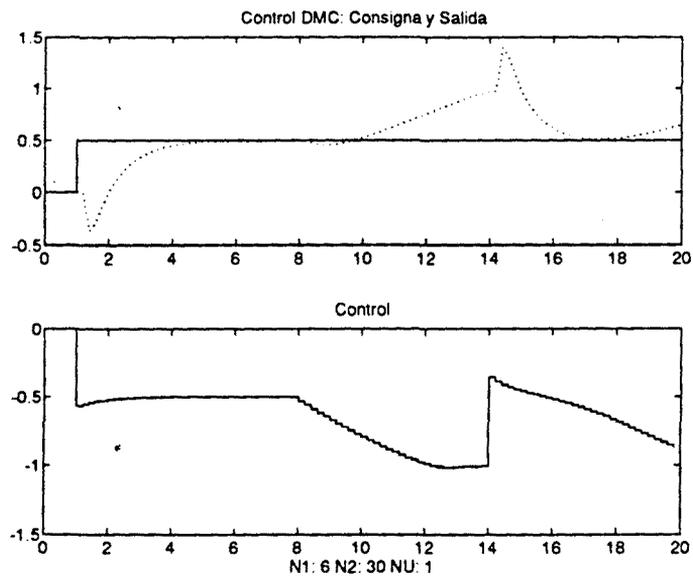


Figura 3.3: Respuesta del DMC con un modelo de respuesta escalón truncado

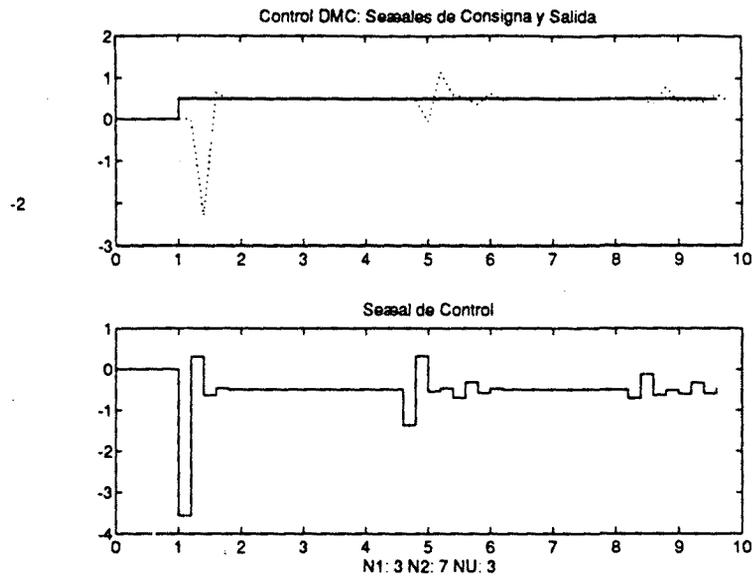


Figura 3.4: Respuesta del DMC con  $N=25$

ecuación de predicción (3.15), vemos que inicialmente la respuesta es la misma, pero al cabo de un tiempo cuando los términos pasados en (3.15) alcanzan el transitorio (o sea los términos que van multiplicados únicamente por  $g_i$ ), en el sistema aparece una perturbación producida por el mismo controlador.

Otro aspecto importante en el DMC es, como se comentó anteriormente, que si no se toma un número de muestras de la respuesta escalón lo suficientemente grande para que haya  $N2$  muestras prácticamente iguales, se cometen errores en el cálculo de la respuesta libre. Estos errores se traducen en un comportamiento anómalo incluso cuando ya se ha alcanzado el estado estacionario.

Para observar este efecto se ha realizado un experimento tomando 25 muestras de la respuesta escalón, que son suficientes para describir todo el transitorio y sintonizando el controlador para conseguir un control *deadbeat*, ya que de esta manera se aprecia mejor el efecto. En la Fig. 3.4 se muestra la respuesta de este controlador y se observa que al cabo de unos instantes se produce una perturbación. La razón es

que la diferencia  $g_{i+k} - g_i$  en la ecuación (3.5) no es lo suficientemente pequeña para poderse aproximar por 0. Por tanto, al no hacerse bien la predicción de la respuesta libre el controlador trata de corregir un error que en realidad no existe, provocando la perturbación señalada.

### 3.3 Forma de la Función de Coste

Como ya se ha comentado, un controlador predictivo obtiene la señal de control minimizando respecto a  $\Delta u(t+k)$  una función de coste. La función de coste más usada en CPBM es

$$J = \sum_{k=1}^{N_2} (\hat{y}(t+k) - w(t+k))^2 + \sum_{k=0}^{N_u-1} (\lambda \Delta u(t+k))^2 \quad (3.18)$$

Parece claro que, para un modelo dado, la forma de esta función dependerá de los valores de  $N_1$ ,  $N_2$ ,  $N_u$  y del factor de peso  $\lambda$ . También dependerá de la trayectoria de referencia y de las predicciones, pero estas cambian en cada instante de muestreo. Si queremos hacer comparaciones de la forma de la función de coste para diversas sintonías de los parámetros lo que haremos es considerar siempre el mismo instante de muestreo.

El dibujo del contorno de la función de coste  $J$  en el espacio de futuros incrementos de control será, obviamente, imposible de visualiazar en una sola gráfica si  $N_u > 2$ . Por tanto, el estudio de la forma de la función de coste se centrará en el efecto que tiene sobre esta el horizonte de coincidencia y el factor de peso.

Usando las facilidades que da Matlab, se han hecho diversos experimentos en los que se muestra la forma de  $J$  cuando el controlador está actuando sobre el sistema benchmarh (3.16). Para decidir que instante vamos a elegir para comparar las funciones de coste, primero observaremos una secuencia temporal de la evolución de la forma de ésta. Esto es, para un controlador fijo vamos a ver que forma adquiere la función de coste. Las diferencias provendrán exclusivamente del error entre la

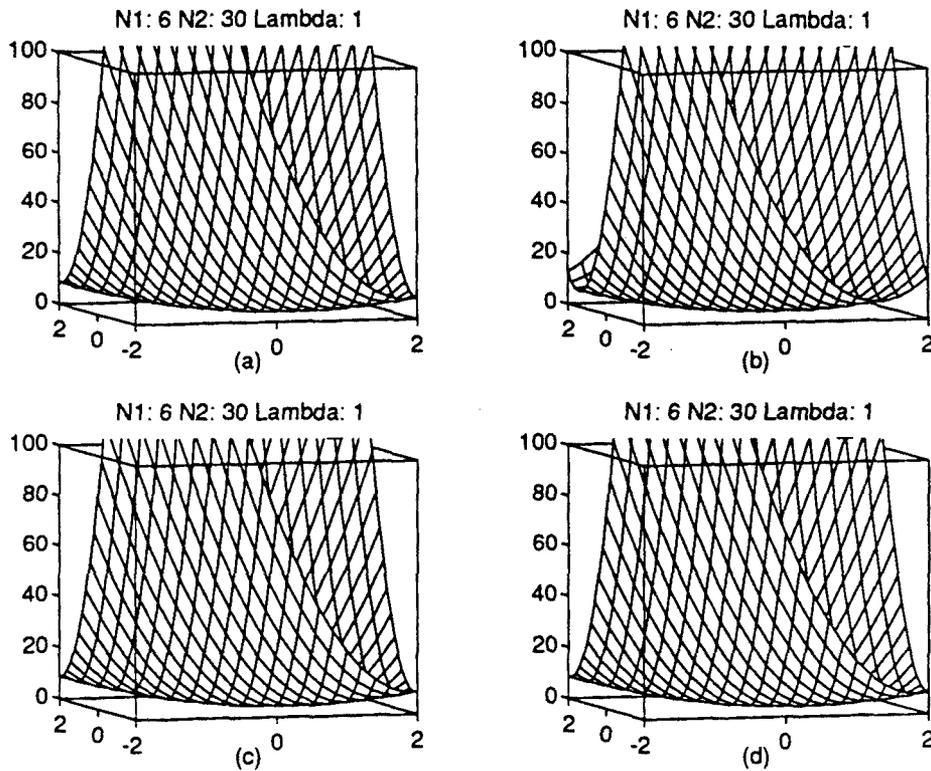


Figura 3.5: Evolución temporal de la forma de la función de coste trayectoria de referencia y la predicción de la salida.

En la Fig. 3.5 podemos ver esta evolución temporal. En (a) tenemos la función de coste justo en el instante anterior de un cambio en la referencia, en (b) en el instante que cambia la referencia, en (c) en mitad de la respuesta del sistema y en (d) cuando ha alcanzado ya la referencia. Las gráficas (a), (c), y (d) son prácticamente iguales, ya que, el error entre trayectoria de referencia y predicción de la salida es nulo o muy pequeño (en el caso de (c)) como es lógico cuando estamos en un estado estacionario. La forma de la función de coste justo en el instante en que se produce un salto en la referencia (b) es distinta a las otras. Este es un buen instante para comparar funciones de coste, ya que se tiene en cuenta el error entre referencia y

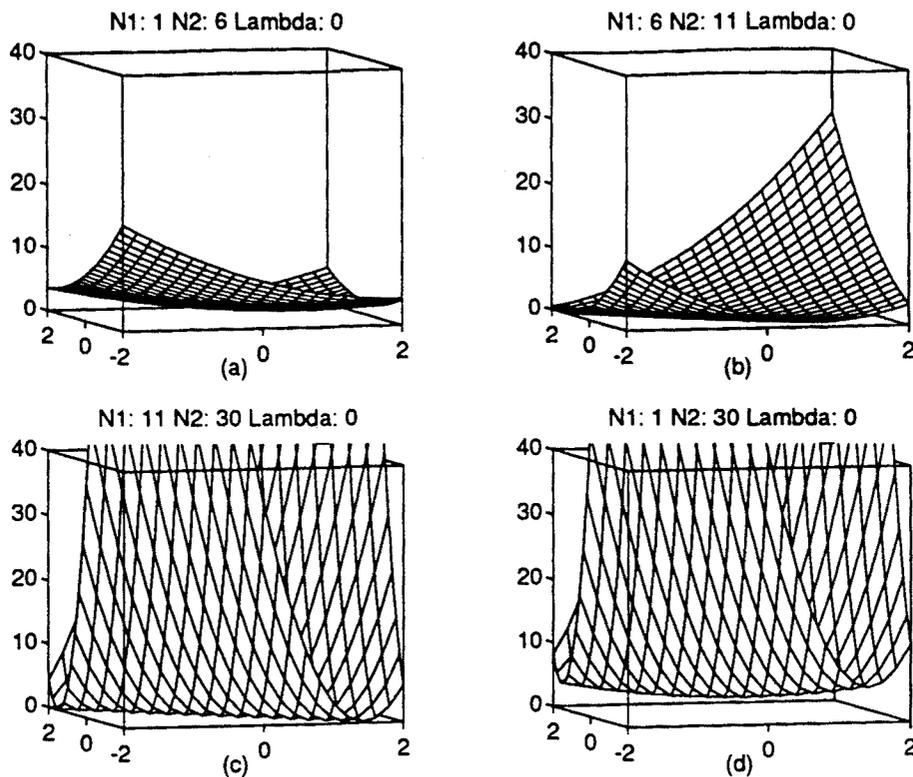


Figura 3.6: Funciones de coste con horizontes de coincidencia distintos

salida. Además, las predicciones que se hagan en este instante serán siempre iguales ya que partimos de un estado estacionario, independientemente de los parámetros de diseño.

### Horizonte de predicción

El horizonte de coincidencia es uno de los factores que influye en la forma de la función de coste. Para determinar su influencia vamos a estudiar la forma de ésta para distintos horizontes de coincidencia.

En la Fig. 3.6 se muestran las formas de la función de coste para cuatro

horizontes de coincidencia. En (a) tenemos un horizonte de coincidencia (1..6) que corresponde a la parte de respuesta inversa del sistema benchmark. Tomando este horizonte el sistema se hace inestable. La forma que adquiere la función de coste es muy plana pero con un mínimo definido. En (b) tenemos la función de coste para un horizonte de coincidencia (6..11). Aquí empieza a aparecer un valle pero con mínimo claro alrededor del punto (-2,2) que corresponde a acciones de control enérgicas. En (c) tenemos un horizonte de coincidencia (11..30). En la función de coste aparece un profundo valle donde el mínimo no está nada definido y el valor de la función de coste en el valle es prácticamente cero. Por último, en (d) tenemos la función de coste para un horizonte (1..30). Este horizonte de coincidencia incluye los tres anteriores y la forma de la función de coste de alguna manera también. Así, podemos observar que la función de coste tiene un valle profundo pero que en la parte más baja de la función no vale cero y, por otra parte, se tiene un mínimo bastante definido. Estas son en definitiva las características más claras de las tres gráficas anteriores.

### Factor de peso

El segundo parámetro que influye en la forma de la función de coste es el factor de peso. Este parámetro tiene un efecto muy claro. En la Fig. 3.7 se pueden observar las funciones de coste para dos controladores iguales pero con diferente factor de peso. Si el factor de peso es cero aparece un valle en la función de coste, en cambio cuando el factor de peso es 4 la función de coste es mucho más redondeada y el mínimo está perfectamente definido.

Esto sugiere que cuando tenemos grandes valles en la función de coste como es el caso del factor de peso cero, si tenemos que buscar el mínimo usando algoritmos numéricos (que sería necesario si tuviéramos restricciones en las variables del problema) podríamos tener problemas para encontrar el mínimo, lo que provocaría un control más deficiente. Por el contrario, un valor de  $\lambda$  distinto de cero, produce mínimos más definidos, con contornos más redondeados, que hacen la optimización más sencilla.

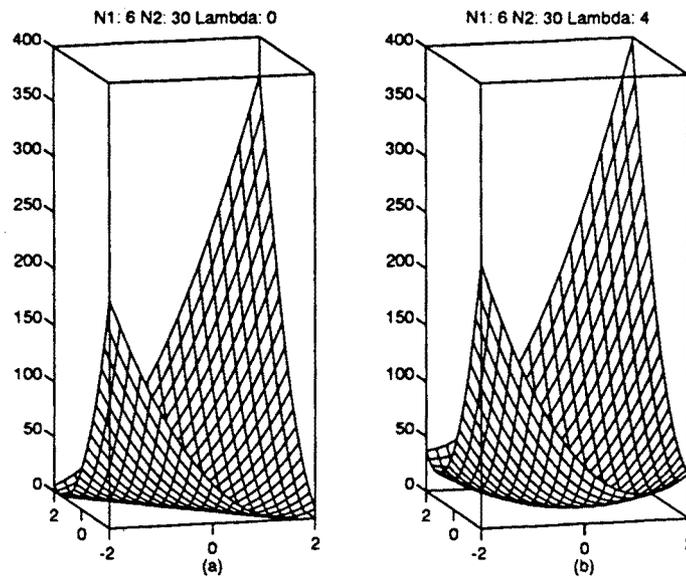


Figura 3.7: Funciones de coste con factores de peso distintos

Para ilustrar este punto, se han comparado dos controladores DMC con los mismos parámetros de sintonía y sobre el mismo modelo *benchmark*. La única diferencia es que para minimizar la función de coste en un caso se ha hecho numéricamente (Fig. 3.8) y, en el otro, analíticamente mediante la expresión (2.5). La minimización numérica se ha llevado a cabo con una rutina de programación cuadrática que incluye restricciones en los incrementos de la señal de control, en la señal de control y en la salida. Las restricciones se han fijado lejos de los límites en que se mueven las señales, de manera que, en principio, se deberían obtener las mismas respuestas.

Sin embargo, debido a las razones anteriormente mencionadas, podemos observar en las Fig. 3.8 y 3.9 que las respuestas son diferentes. En el caso del algoritmo numérico se producen oscilaciones adicionales debidas al problema numérico de encontrar el mínimo de la función de coste. En el caso de la minimización analítica a partir del instante 2.5 tanto la salida como el control permanecen constantes.

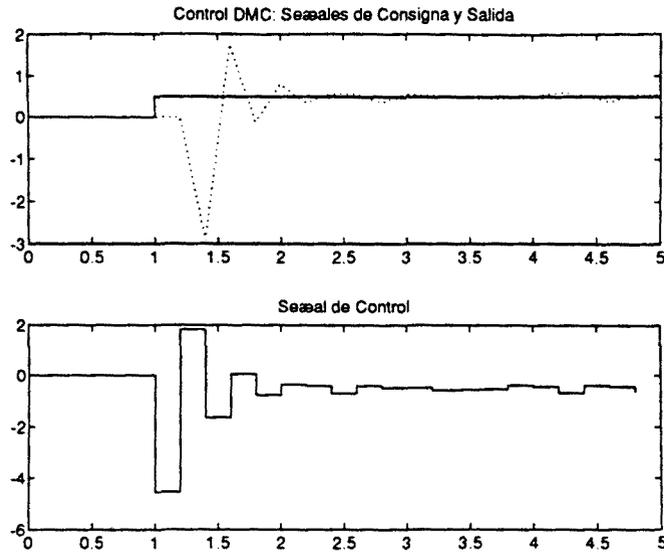


Figura 3.8: Respuesta de un DMC haciendo la optimización de forma numérica

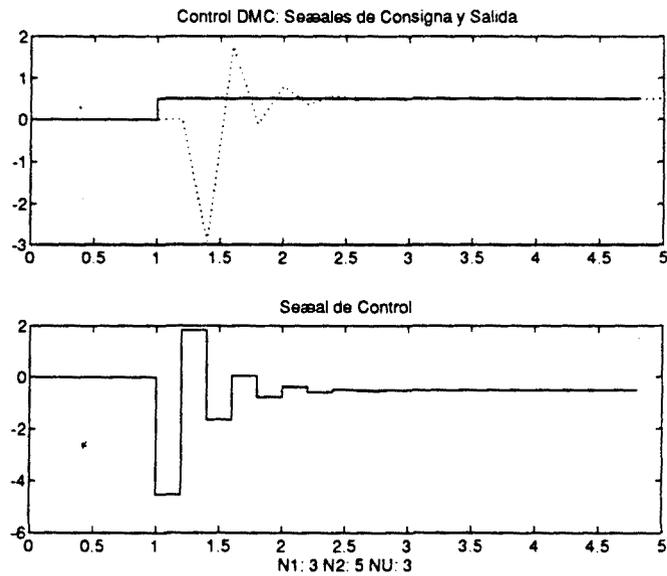


Figura 3.9: Respuesta de un DMC haciendo la optimización de forma analítica

### 3.4 Implementación de CPBM Multivariantes

Una de las características de las técnicas CPBM es que se pueden extender al caso multivariable. Todos los principios que se han presentado en los capítulos anteriores son válidos y su ampliación a sistemas multivariantes será más o menos sencilla dependiendo del modelo del proceso que se utilice. Para modelos en variables de estado la extensión es inmediata, dado que estos modelos ya son multivariantes. Para modelos de convolución la extensión es sencilla y se presenta a continuación. Para modelos de función de transferencia la extensión es más intrincada, pero posible.

Dado que la extensión dependerá del modelo del proceso, vamos a desarrollar un caso en concreto, el DMC, pero las conclusiones que obtengamos serán generales para cualquier controlador predictivo multivariable.

#### 3.4.1 DMC Multivariable

El modelo del proceso será un conjunto de respuestas escalón entre las entradas y las salidas,

$$\begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_p(t) \end{bmatrix} = \begin{bmatrix} G^{11}(q^{-1}) & G^{12}(q^{-1}) & \dots & G^{1m}(q^{-1}) \\ G^{21}(q^{-1}) & G^{22}(q^{-1}) & \dots & G^{2m}(q^{-1}) \\ \vdots & \vdots & \ddots & \vdots \\ G^{p1}(q^{-1}) & G^{p2}(q^{-1}) & \dots & G^{pm}(q^{-1}) \end{bmatrix} \begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_m(t) \end{bmatrix} + \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_m(t) \end{bmatrix} \quad (3.19)$$

donde los elementos de la matriz  $G^{ij}(q^{-1})$  son polinomios cuyos coeficientes son la respuesta escalón entre la entrada  $j$  y la salida  $i$ :

$$G^{ij}(q^{-1}) = g_1^{ij}q^{-1} + g_2^{ij}q^{-2} + \dots + g_n^{ij}q^{-n}, \quad i = 1 \dots p; j = 1 \dots m \quad (3.20)$$

La expresión (3.3) de la predicción óptima a  $k$  pasos puede generalizarse

para la salida  $i$  de la siguiente forma:

$$\hat{y}_i(t+k) = y_i(t) + G^{i1}(q^{-1})(\Delta u_1(t+k) - \Delta u_1(t)) + \dots + G^{im}(q^{-1})(\Delta u_m(t+k) - \Delta u_m(t)) \quad (3.21)$$

Siguiendo los mismos pasos que en el DMC SISO, la predicción de la salida la dividiremos en la parte libre y la forzada. La respuesta libre es la parte de la respuesta que no depende de los incrementos de control futuros. Para la salida  $i$  tendremos,

$$\begin{aligned} \hat{y}_i^f(t+k) = y_i(t+k) &+ \sum_{j=1}^N (g_{j+k}^{i1} - g_j^{i1}) \Delta u_1(t-j) + \dots \\ &+ \sum_{j=1}^N (g_{j+k}^{im} - g_j^{im}) \Delta u_m(t-j) \end{aligned} \quad (3.22)$$

La respuesta forzada se calcula tomando condiciones iniciales nulas. Para la salida  $i$  tendremos,

$$\hat{y}_i^c(t+k) = G_k^{i1} \Delta u_1(t+k) + \dots + G_k^{im} \Delta u_m(t+k) \quad (3.23)$$

donde  $G_k^{ij}(q^{-1})$  es el polinomio

$$G_k^{ij}(q^{-1}) = g_1^{ij} q^{-1} + g_2^{ij} q^{-2} + \dots + g_k^{ij} q^{-k} \quad (3.24)$$

Combinando todas las salidas en un vector conseguimos la ecuación fundamental del controlador

$$\hat{\mathbf{y}} = \mathbf{G} \Delta \mathbf{u} + \mathbf{p} \quad (3.25)$$

donde

$$\begin{aligned} \hat{\mathbf{y}} &= [\hat{y}_1(t+1) \dots \hat{y}_1(t+N2) \dots \hat{y}_p(t+1) \dots \hat{y}_p(t+N2)]^T \\ \Delta \mathbf{u} &= [\Delta u_1(t) \dots \Delta u_1(t+N2-1) \dots \Delta u_m(t) \dots \Delta u_m(t+N2-1)]^T \\ \mathbf{p} &= [\hat{y}_1^f(t+1) \dots \hat{y}_1^f(t+N2) \dots \hat{y}_p^f(t+1) \dots \hat{y}_p^f(t+N2)]^T \\ \mathbf{G} &= \begin{bmatrix} G_{11} & G_{12} & \dots & G_{1m} \\ G_{21} & G_{22} & \dots & G_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ G_{p1} & G_{p2} & \dots & G_{pm} \end{bmatrix} \end{aligned}$$

Cada uno de los elementos  $G_{ij}$  de la matriz  $\mathbf{G}$  es una matriz de respuestas escalón entre la entrada  $j$  y la salida  $i$  de la forma

$$G_{ij} = \begin{bmatrix} g_{N_1}^{ij} & g_{N_1-1}^{ij} & 0 & \cdots & 0 \\ g_{N_1+1}^{ij} & g_{N_1}^{ij} & g_{N_1-1}^{ij} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N_2}^{ij} & g_{N_2-1}^{ij} & g_{N_2-2}^{ij} & \cdots & g_{N_1-N_u+1}^{ij} \end{bmatrix} \quad (3.26)$$

Los parámetros  $N_2$  y  $N_u$  que ya se definieron son el horizonte de predicción y el horizonte de control respectivamente. Para un sistema MIMO con  $m$  entradas y  $p$  salidas la función de coste que se minimiza para obtener los futuros incrementos de control se debe extender a

$$J = \sum_{k=n_1}^{N_2} E^T(t+k)Q_k E(t+k) + \sum_{k=0}^{N_u} \Delta u^T(t+k)R_k \Delta u(t+k) \quad (3.27)$$

donde

$$\begin{aligned} E(t+k) &= w(t+k) - \hat{y}(t+k) && \text{vector de errores en el instante } k \\ &&& \text{(p componentes)} \\ \Delta u(t+k) &&& \text{vector de incrementos de la señal de} \\ &&& \text{control en el instante } k \text{ (m componentes)} \\ Q_k, R_k &&& \text{matrices de peso diagonales} \end{aligned}$$

Las matrices  $Q_k$  ( $p \times p$ ) y  $R_k$  ( $m \times m$ ) pueden contener diferentes factores de peso para cada entrada y salida. Eligiendo adecuadamente estas matrices se pueden obtener diferentes optimizaciones y horizontes de control para cada entrada y salida.

Definiendo el vector  $\mathbf{E}_0 = \mathbf{w} - \mathbf{p}$  podemos reescribir la función de coste en forma vectorial

$$J = [\mathbf{E}_0 - \mathbf{G}\Delta\mathbf{u}]^T Q [\mathbf{E}_0 - \mathbf{G}\Delta\mathbf{u}] + \Delta\mathbf{u}^T R \Delta\mathbf{u} \quad (3.28)$$

Las matrices compuestas  $Q$  y  $R$  corresponden a

$$Q = \begin{bmatrix} Q_{N_1} & 0 & \cdots & 0 \\ 0 & Q_{N_1+1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & Q_{N_2} \end{bmatrix} \quad R = \begin{bmatrix} R_1 & 0 & \cdots & 0 \\ 0 & R_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & R_{N_u} \end{bmatrix} \quad (3.29)$$

El mínimo de esta función respecto a  $\Delta \mathbf{u}$  es la solución ya conocida

$$\Delta \mathbf{u}_{opt} = (\mathbf{R} + \mathbf{G}^T \mathbf{Q} \mathbf{G})^{-1} \mathbf{G}^T \mathbf{Q} (\mathbf{w} - \mathbf{p}) \quad (3.30)$$

Esta expresión es totalmente equivalente a la solución de la versión SISO y lo único que hay que tener en cuenta es que los vectores y matrices tienen más componentes. Conviene notar que la dimensión de la matriz  $\mathbf{G}$  crece al aumentar el número de entradas y salidas de forma geométrica, al igual que la matriz que hay que invertir en la solución que tendrá las mismas dimensiones que  $\mathbf{R}$ .

El cálculo de la solución puede llegar a ser numéricamente complejo. Si además consideramos restricciones en las variables la complejidad de la solución se dispara. Veamos a continuación una implementación mediante compensación feed-forward que reduce considerablemente los cálculos.

### 3.4.2 Compensación feedforward

Una posibilidad obvia de control de un sistema multivariable cuando no hay interacción entre sus lazos es mediante controladores SISO. Una alternativa entre un control SISO y un diseño totalmente multivariable es considerar las señales de control de los otros lazos como perturbaciones medibles.

Para realizar este tipo de diseño primero hay que tener en cuenta que el número de entradas ha de ser como mínimo igual que el de salidas. Si el número de entradas es mayor no importa, ya que todas se tratan igual, como perturbaciones medibles. Por otra parte, hay que realizar un estudio de la interacción para hacer el emparejamiento de los lazos.

Para cada lazo tendremos un controlador. La respuesta forzada será la misma que en caso del DMC SISO, mientras que la respuesta libre se modificará de la siguiente manera.

La respuesta libre tendrá una parte que será exactamente igual que en el

caso multivariable debida a señales de control pasadas y conocidas, más una parte debida a la influencia de las futuras señales de control del resto de lazos. Por tanto, habrá que hacer una estimación de estas futuras señales de control. Esto no es un problema, ya que los controladores predictivos calculan en cada periodo de muestreo un vector de señales de control futuras en el horizonte de predicción. A pesar que sólo el primer elemento de este vector de señales de control se aplica realmente parece una suposición razonable tomar el resto de elementos como la estimación de la futura señal de control. La predicción de la respuesta libre para el lazo  $i$  quedará,

$$\begin{aligned}
 \hat{y}_i^f(t+k) = y_i(t+k) &+ \sum_{j=1}^N (g_{j+k}^{i1} - g_j^{i1}) \Delta u_1(t-j) + \dots \\
 &+ \sum_{j=1}^N (g_{j+k}^{im} - g_j^{im}) \Delta u_m(t-j) \\
 &+ \sum_{h=1}^m \sum_{j=0}^{k-1} g_{k-j}^{ih} \Delta u_h(t+j) \quad h \neq i \quad (3.31)
 \end{aligned}$$

Esta estrategia nos proporciona un rendimiento mucho mejor que un control SISO con un coste computacional mucho menor comparado con el caso multivariable.

El ahorro de cálculo lo podemos evaluar de la siguiente forma. Para el caso sin restricciones y funcionando en modo no adaptativo los incrementos de la señal de control los calculamos con (3.30) donde

$$\mathbf{K} = (\mathbf{R} + \mathbf{G}^T \mathbf{Q} \mathbf{G})^{-1} \mathbf{G}^T \mathbf{Q} \quad (3.32)$$

sólo se ha de calcular una vez al principio y no influye en el número de operaciones que hay que hacer cada instante de muestreo. Lo que calculamos en cada instante es el producto  $\Delta \mathbf{u} = \mathbf{K} \mathbf{E}_0$ .

En el caso multivariable la matriz  $\mathbf{K}$  tiene dimensiones  $(m \cdot Nu) \times (p \cdot N2)$  y el vector  $\mathbf{E}_0 = \mathbf{w} - \mathbf{p}$  tiene dimensión  $(p \cdot N2)$ . Teniendo en cuenta que sólo necesitamos calcular el primer incremento de la señal de control el número de productos necesario es  $m \times (p \cdot N2)^2$ .

En el caso SISO con compensación feedforward tendremos  $m$  matrices  $\mathbf{K}$  con dimensiones  $Nu \times N2$  y  $m$  vectores  $\mathbf{E}_0$  con dimensión  $N2$ . En este caso necesitamos calcular los  $Nu$  incrementos de la señal de control con lo que el número de productos necesario es  $m \times Nu \times N2^2$ .

Por otra parte, para calcular la respuesta libre hemos de realizar los cálculos correspondientes a la compensación feedforward que suponen  $p \times m \times N2^2/2$  productos adicionales.

Para poder comparar más fácilmente ambas implementaciones vamos a considerar un sistema con igual número de entradas y salidas  $n$ . En este caso vemos que el número de productos en el caso multivariable depende de  $n^3$  mientras que en caso de la implementación feedforward depende de  $n^2$ . La diferencia de coste será mayor cuanto mayor sea el número de variables.

En el caso que tengamos restricciones los cálculos son distintos, ya que el algoritmo numérico de minimización evalúa directamente la función de coste. El factor determinante de la función de coste en la complejidad de los cálculos es la matriz de respuestas escalón  $\mathbf{G}$ . En un control multivariable esta matriz crece de forma geométrica respecto al número de entradas y salidas,  $(m \cdot Nu) \times (p \cdot N2)$ . En el otro caso resolveremos  $m$  problemas donde la matriz  $\mathbf{G}$  tendrá dimensión  $Nu \times N2$ .

Lo mismo sucede cuando el controlador predictivo trabaja de forma adaptativa o con el controlador predictivo no lineal presentado en el capítulo anterior. En ambos casos en cada instante de muestreo hay que calcular el vector  $\mathbf{K}$ . El elemento crítico de este cálculo es la inversión de una matriz que en el caso multivariable crece de forma geométrica mientras que el otro lo hace aritméticamente. Por otra parte, el número de cálculos necesarios para invertir una matriz también crece geoméricamente respecto al número de elementos. Todo esto hace que el coste computacional del controlador multivariable se dispare respecto al caso del control con compensación feedforward.

La implementación de controladores SISO con compensación feedforward

tiene interés, sobre todo, en aplicaciones donde el tiempo sea un factor crítico y por otro lado el proceso tenga interacción entre sus lazos y sea necesario desacoplarlos.

### 3.4.3 Aplicación al control de un reactor químico

Las dos implementaciones de controladores predictivos multivariables presentadas en los apartados anteriores se van a comparar sobre un reactor químico exotérmico encamisado. La descripción y el desarrollo del modelo matemático se pueden ver en el Apéndice A.

Los objetivos de control en un reactor químico de estas características son dos. Uno es mantener una temperatura constante en el interior del reactor para que se produzca la reacción de una sustancia A que se transforma en una B. El segundo objetivo es mantener la concentración de la sustancia B a un determinado valor.

Para tratar de cumplir estos objetivos de control disponemos de dos variables manipuladas. Una es el flujo de líquido que entra al reactor conteniendo la sustancia A, y la otra es el flujo de refrigerante que entra a la camisa de refrigeración. El emparejamiento de lazos elegido para realizar el control es: el flujo de líquido que contiene la sustancia A ( $F_l$ ) para controlar la concentración de B ( $C_b$ ) y el flujo de refrigerante ( $F_r$ ) para la temperatura en el interior del reactor ( $T_l$ ). Este emparejamiento es innecesario en la implementación multivariable ya que el controlador tiene en cuenta la interacción entre lazos.

El esquema de control se puede ver en la Fig. 3.10. Se han usado dos controladores en cascada que actúan sobre las válvulas que regulan los flujos de líquido y de refrigerante. Las dinámicas de estos lazos son mucho más rápidas que las del reactor, así que no se han tenido en cuenta en el modelo.

El primer paso para implementar un controlador predictivo es conseguir un modelo adecuado del proceso. Dado que los controladores son DMC, los modelos que necesitamos son respuestas escalón entre las entradas y las salidas. Llamaremos

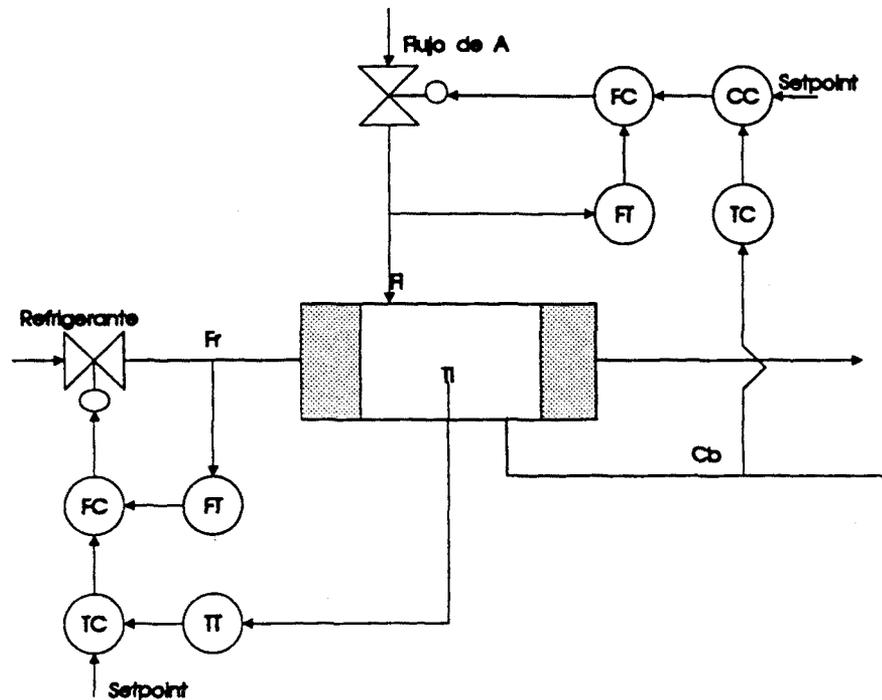


Figura 3.10: Esquema de control del reactor químico

- $G^{11}$ : Respuesta escalón entre  $F_l$  y  $C_b$
- $G^{12}$ : Respuesta escalón entre  $F_r$  y  $C_b$
- $G^{21}$ : Respuesta escalón entre  $F_l$  y  $T_l$
- $G^{22}$ : Respuesta escalón entre  $F_r$  y  $T_l$

Estas respuestas escalón las hemos identificado mediante el método de minimización paramétrica NLMS (Normaliced Least Mean Square). El número de muestras tomadas para cada una de las respuestas es 60. Con este número conseguimos que aproximadamente los 25 últimos valores de cada una de las respuestas sean prácticamente idénticos. De esta forma N2 podrá tomar valores hasta 25 sin que se deteriore el control porque no se haya estacionado la respuesta.

Se han realizado tres experimentos para comparar las diferentes implementaciones del control DMC presentadas anteriormente. Los experimentos consisten en variar en el instante  $t = 2h$  la consigna de la concentración de B en el tanque de 4 a

6  $Kgmol/m^3$ , y en  $t = 52h$  se modifica la consigna de temperatura que pasa de 60 a  $50^{\circ}C$ . En estos experimentos se han utilizado los mismos parámetros de sintonía para todos los DMC, que son,  $N_1 = 1$ ,  $N_2 = 10$ ,  $N_u = 4$  y un factor de peso  $\lambda = 0.025$ . No se ha usado ningún filtro en la referencia.

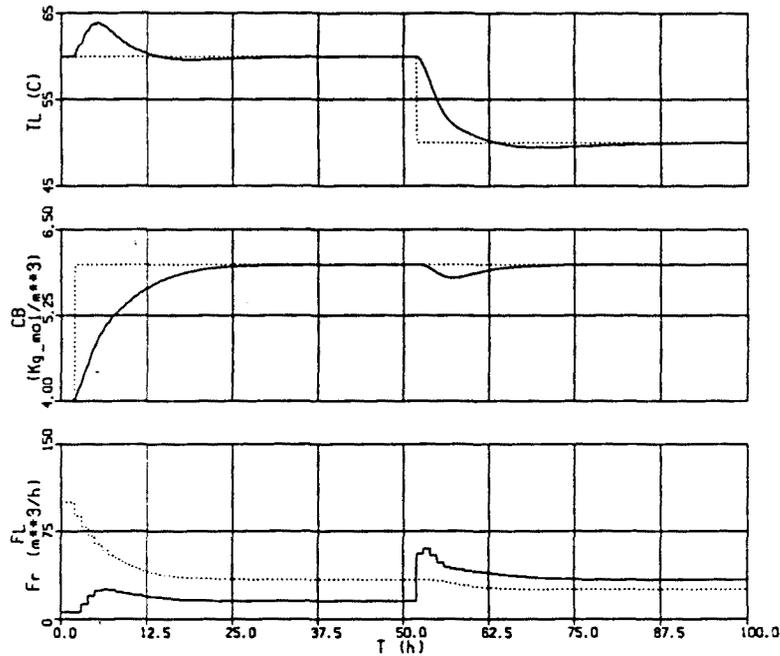


Figura 3.11: Control mediante dos DMC independientes

### Control mediante dos DMC independientes

En esta primera implementación se han usado dos DMC trabajando independientemente uno del otro. En la Fig. 3.11 podemos ver la respuesta del sistema. Observamos que la respuesta en el lazo de concentración cuando variamos esta consigna es bastante lenta y provoca una perturbación en el lazo de temperatura de unos  $4^{\circ}C$ . Cuando cambia la consigna de temperatura la respuesta de este lazo es también algo lenta y con un pequeño sobrepico, y produce una perturbación en la

concentración de varias décimas.

### Control mediante un DMC multivariable

La segunda implementación consiste en un DMC completamente multivariable. En la Fig. 3.12 podemos ver la respuesta del sistema. Observamos que la respuesta del lazo de concentración cuando variamos esta consigna es algo más rápida que en la implementación anterior pero lo que realmente mejora es la perturbación del lazo de temperatura que es apenas perceptible.

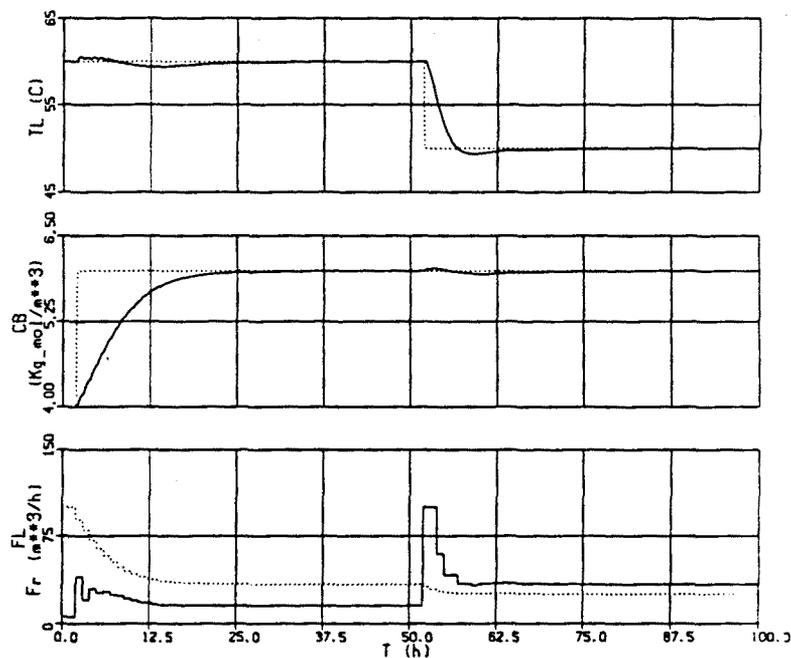


Figura 3.12: Control mediante un DMC multivariable

Cuando cambiamos la consigna de temperatura la respuesta de este lazo es también algo mejor que en la implementación anterior y otra vez el lazo de concentración no se ve apenas afectado. Queda claro que esta implementación tiene en cuenta la interacción entre lazos y cuando cambia una consigna no se ve afectado el otro lazo.

### Control mediante dos DMC con compensación feedforward

Por último se ha implementado un control con dos DMC SISO con compensación feedforward. En la Fig. 3.13 podemos ver la respuesta del sistema. Observamos que la respuesta del lazo de concentración cuando variamos esta consigna es similar a la del DMC multivariable aunque provoca una perturbación en la temperatura mucho más grande de unos  $3^{\circ}\text{C}$  que con todo tiene una magnitud menor que en el caso de dos DMC independientes.

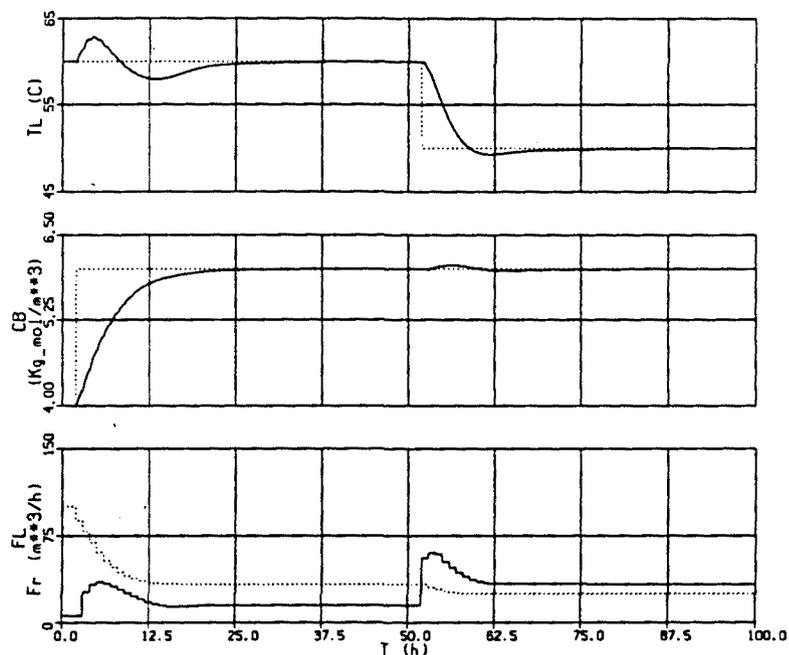


Figura 3.13: Control mediante dos DMC con compensación feedforward

Al variar la consigna de temperatura, la respuesta de este lazo es también similar a la del DMC multivariable y la perturbación que provoca en el lazo de concentración es apenas perceptible. Como se ve el rendimiento de esta implementación es mejor que la implementación con dos DMC independientes y algo peor que el DMC multivariable. Esto es algo lógico ya que esta implementación usa más información

(y por tanto más cálculos) que los DMC independientes, pero menos que el DMC multivariable. Esta implementación tiene otra ventaja que veremos a continuación.

### 3.4.4 Restricciones

El implementar un control multivariable mediante controladores independientes que tengan en cuenta la interacción por medio de la compensación feedforward da cierta flexibilidad. Esta forma de implementación permite abordar el tema de las restricciones desde una óptica que no pasa por solucionar un problema de programación cuadrática.

En la industria muchas veces no encontramos con procesos que tienen más variables controladas que variables manipuladas. En estos casos sólo un número de variables controladas igual al de variables manipuladas se podrá conseguir que sigan una referencia. El resto de variables de salida, ya que no se pueden controlar, puede interesar que no se salgan de ciertos límites.

Una forma posible de tratar estas restricciones en una variable no controlada (la variables controladas se supone que siguen a la consigna) es mediante una estructura override. Esta normalmente tratará que la variable controlada a la referencia. Pero si una variable no controlada alcanza una restricción, el sistema conmutará de controlador y éste tratará de mantener la variable (ahora controlada) al nivel de la restricción. Esto seguirá así hasta que cambien las condiciones de operación y se vuelva a conmutar a la situación normal cuando ya no haya restricciones activas.

Se ha implementado esta estructura en el lazo de control de la concentración del reactor químico. Este lazo, además de tener en cuenta la interacción del lazo de temperatura, trata de mantener la concentración de producto B al valor de la referencia y vigila que la concentración de producto A no se salga de unos límites. Si esto ocurre la estructura override conmuta y pasa a controlar la concentración de producto A de manera que se mantenga al valor de la restricción, o sea, la referencia

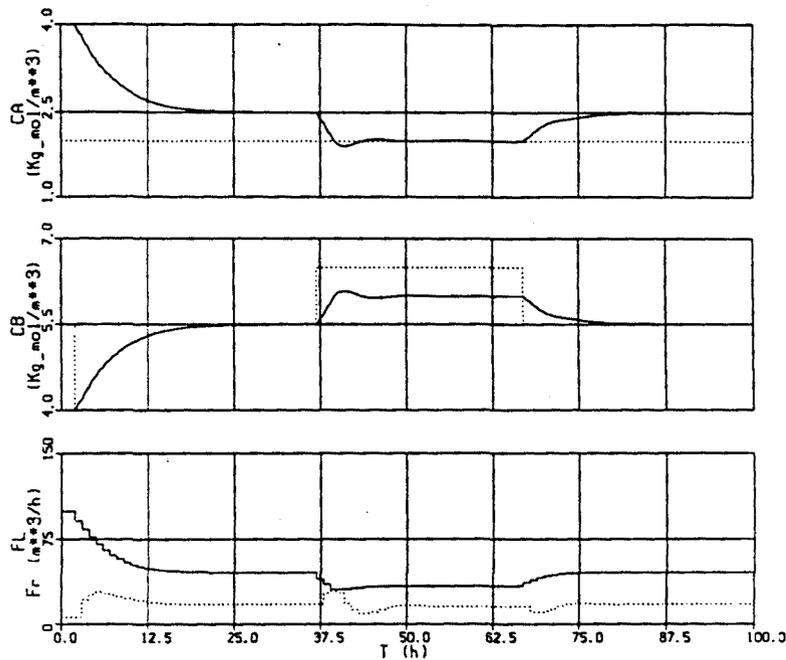


Figura 3.14: Control DMC con un restricción en  $C_a$

del controlador pasa es el valor de la restricción. Cuando cambien las condiciones y la concentración de A pueda estar dentro de los límites permitidos, se volverá a conmutar de controlador y se pasará a la situación primera.

En la Fig. 3.14 podemos ver como responde el lazo de concentración cuando tenemos una restricción en la concentración de producto A en el interior del tanque que no queremos que sea inferior a  $2 \text{ Kg mol/m}^3$ . El experimento consiste en variar la consigna de concentración de B y observamos que mientras no está activa la restricción la concentración de B sigue a la consigna pero cuando ésta se activa el sistema cambia de controlador y empieza a regular la concentración de A de manera que no baje de la restricción. Una vez que la restricción se desactiva se vuelve a conmutar de controlador y la concentración de B sigue la consigna.

Así, hemos resuelto un problema con restricciones en una variable no con-

trolada sin tener que recurrir a costosas rutinas de programación cuadrática. El único incremento de cálculo que supone la estructura override es debido al mecanismo de decisión que escoge el controlador que hay que usar en cada momento.

## Capítulo 4

# Aplicación de Técnicas de Análisis Robusto a CPBM

Los controladores predictivos como su nombre indica están basados en un modelo, que llamaremos a partir de ahora *nominal*. Estos modelos son necesariamente inexactos. En la práctica es muy importante que los controladores funcionen bien, incluso cuando el comportamiento dinámico del proceso real sea diferente del descrito por el modelo nominal.

Así, todo controlador predictivo tiene que contemplar ciertos aspectos de robustez: buena operación de la planta en presencia de falta de información sobre el funcionamiento de algunas partes de la misma. La mejor manera de conseguir características de robustez en un sistema de control es intentar representar de alguna manera esta falta de información para poder incorporarla en el proceso de análisis y diseño. Esto permite tener en cuenta, de manera directa, la robustez del sistema de control que se está diseñando.

En la primera sección se presentan las técnicas estándar para diseño robusto. Estas técnicas usan una representación basada en la respuesta en frecuencia a partir de una función de transferencia. Así pues, para analizar un controlador pre-

dictivo mediante estas técnicas será necesario poderlo expresar como una función de transferencia. Prácticamente todos los controladores predictivos basados en modelos lineales y sin restricciones se pueden expresar de esta forma, por lo que no representa un inconveniente. Dado que los controladores predictivos son discretos, así como los modelos que usan, la planta a controlar se representará también de forma discreta. Los resultados que se obtengan son prácticamente idénticos a los que se obtienen para plantas y controladores continuos.

En la segunda sección se deduce la función de transferencia para un controlador predictivo, el GPC. En la siguiente sección se hace un estudio de la influencia de los parámetros de sintonía de un controlador predictivo sobre la robustez, bajo el paraguas de las técnicas que se han presentado. El capítulo finaliza presentando un método de diseño que le confiere al controlador predictivo características robustas.

## 4.1 Análisis para el Diseño Robusto

La mayoría de los problemas que surgen al tratar con un esquema de control realimentado provienen del mismo hecho, el desconocimiento o falta de información: INCERTIDUMBRE. En el caso que se dispusiera de toda la información del proceso que se intenta controlar, entonces no haría falta la realimentación, sino que se podría realizar el control en lazo abierto. Está claro que la presencia de este desconocimiento o incertidumbre afectará al control que podamos hacer sobre el sistema y, por tanto, habrá que tenerlo en cuenta.

La incertidumbre del modelo puede tener diversas fuentes. La más importante es debida a las no linealidades de los procesos reales. Si el modelo del proceso se obtiene via linealización, entonces éste será adecuado sólo alrededor del punto de trabajo donde se haya realizado la linealización. Aunque el proceso se pueda representar con bastante exactitud mediante el modelo lineal, al cambiar el punto de operación puede conducir a variaciones de los parámetros del modelo lineal.

En el dos caso anterior, la fuente y la estructura de la incertidumbre se conoce con cierta exactitud. Existe otro tipo de incertidumbre incluso cuando el proceso es esencialmente lineal. Esta es debida a que los parámetros físicos del proceso nunca se conocen con exactitud, y a que las dinámicas rápidas no se suelen tener en cuenta en el modelo. Así, a altas frecuencia, incluso el orden del modelo es desconocido.

A continuación se presentan las técnicas que nos permitirán realizar el análisis y diseño de los controladores predictivos teniendo en cuenta la incertidumbre de los modelos que usan estos controladores.

#### 4.1.1 Funciones de sensibilidad y sensibilidad complementaria

Como se comento anteriormente, el análisis robusto trabaja en el dominio de la frecuencia y los procesos se describen mediante la función de transferencia. Así pues, partiendo de un proceso a controlar que tengamos representado por un modelo discreto  $G_0(z)$ , se plantea el esquema de control que se muestra en la figura 4.1.

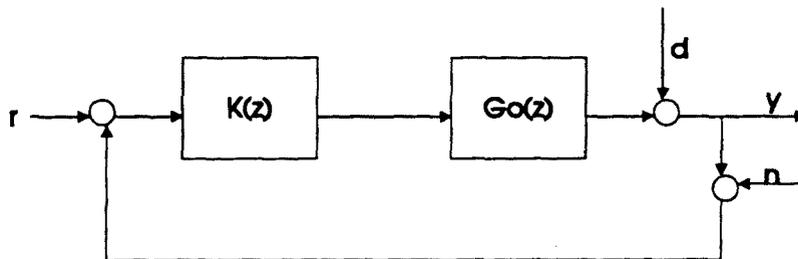


Figura 4.1: Esquema de control realimentado

Como se puede ver, en este sistema de control se ha considerado también la presencia de las señales de perturbación  $d$  y de ruido en la medida  $n$ .

El problema de control consiste en mantener la variable controlada  $y$  a un cierto valor deseado. El controlador realizará la tarea correctora a partir de la señal de

error construida como la diferencia existente entre la salida real del proceso y el valor deseado para la misma. Observar que en presencia de las señales de perturbación mencionadas, la señal de error que entrará al controlador vendrá dada por

$$e = r - (y + d + n) \quad (4.1)$$

en lugar de la señal de error esperada

$$e = r - y \quad (4.2)$$

Así, podemos ver las señales  $d$  y  $n$  como fuentes de incertidumbre sobre la salida del proceso, las cuales se habrán de tener en cuenta a la hora de diseñar el controlador. Por otra parte, éste trabajará con una señal diferente de la señal a partir de la cual ha sido diseñado, lo que sin duda es un problema.

Otro punto a tener en cuenta en el proceso de diseño, es el hecho que este diseño se realizará a partir del modelo nominal  $G_0(z)$ , y que la dinámica de este modelo diferirá de la del proceso real. Esto añade más incertidumbre en la señal de error esperada por el controlador, en el cual la componente correspondiente a la salida no será del tipo  $y = G_0(z)u$ .

Por tanto, vemos que tenemos tres fuentes de incertidumbre que causan una cierta falta de información en el esquema de control realimentado. Para poder tener en cuenta esta falta de información en el diseño del controlador, es conveniente ver como son las ecuaciones del lazo cerrado en presencia de estas señales y que papel juega el controlador. Así, tenemos que la salida viene dada por

$$y = \frac{G_0(z)K(z)}{1 + G_0(z)K(z)}(r - n) + \frac{1}{1 + G_0(z)K(z)}d \quad (4.3)$$

donde, si definimos las funciones de *Sensibilidad*  $S_0(z)$  y de *Sensibilidad Complementaria*  $T_0(z)$  como

$$S_0(z) = \frac{1}{1 + G_0(z)K(z)} \quad (4.4)$$

$$T_0(z) = \frac{G_0(z)K(z)}{1 + G_0(z)K(z)} \quad (4.5)$$

obtenemos

$$y = T_0(z)r - T_0(z)n + S_0(z)d \quad (4.6)$$

## Especificaciones

Si el objetivo de control es obtener  $y \approx r$ , a partir de la ecuación (4.6), vemos que deberíamos tener

$T_0(z) \approx 0$  : Para eliminar el ruido de medida  $n$

$S_0(z) \approx 0$  : Para eliminar la perturbación  $d$

$T_0(z) \approx 1$  : Para conseguir  $y \approx T r \approx r$

Vemos por tanto, que se nos presentan dos objetivos aparentemente contradictorios, dado que hemos de conseguir simultáneamente  $T_0(z) \approx 0$  y  $T_0(z) \approx 1$ .

No obstante, la ecuación (4.6) nos describe una relación en el dominio frecuencial y, por tanto, las dos situaciones mencionadas no son del todo contradictorias. Se puede pensar en separar los dos requerimientos frecuencialmente y tener  $T_0(z) \approx 0$  en un cierto rango de frecuencias y  $T_0(z) \approx 1$  en otro. Estos rangos de frecuencia nos vendrán determinados por el contenido frecuencial de las señales  $d$ ,  $n$  y  $r$ . Así, para eliminar el efecto del ruido en la medida, será necesario conocer el contenido frecuencial de  $n$  y hacer  $T_0(z) \approx 0$  en estas frecuencias. De la misma forma, según sea el contenido frecuencial de las referencias a seguir por el sistema, convendrá tener  $T_0(z) \approx 1$  en este rango.

Observar que esto limita el conjunto de señales de referencia posibles, a aquellas que tengan un cierto contenido frecuencial. En otras palabras, esto limitará el ancho de banda del sistema. En los controladores predictivos se dispone del filtro de la señal de consigna para conseguir que la trayectoria de referencia tenga el contenido frecuencial adecuado.

Otra limitación proviene del hecho de tener que conseguir  $S_0(z) \approx 0$  para que las perturbaciones no afecten a la salida. En este caso, el problema es que no será posible conseguir  $S_0(z) \approx 0$  en cualquier rango de frecuencias. El rango posible vendrá determinado por la división frecuencial que hayamos hecho de la función  $T_0(z)$ .

Esto es debido a que ambas funciones no son independientes, sino que están ligadas por la relación

$$S_0(z) + T_0(z) = 1 \quad (4.7)$$

(De aquí el nombre que se le da a estas funciones). A partir de esta relación, podemos ver que no es posible tener a una misma frecuencia de manera simultánea  $|S(e^{j\omega})| < 1/2$  y  $|T(e^{j\omega})| < 1/2$ , ya que en este caso tendríamos

$$1 = |S(e^{j\omega}) + T(e^{j\omega})| \leq |S(e^{j\omega})| + |T(e^{j\omega})| < 1, \quad 0 \leq \omega \leq \pi \quad (4.8)$$

Así, si en un cierto rango de frecuencias tenemos  $T_0(z) \approx 0$  hemos de tener en este mismo rango  $S_0(z) \approx 1$ . La conclusión es que no podemos evitar que afecten a la salida las componentes de perturbación y ruido en la medida que aparezcan en una misma frecuencia.

Desde el punto de vista frecuencial, el diseño del controlador se ha de hacer a partir del papel que este juega en la definición de las funciones de Sensibilidad y Sensibilidad Complementaria para poder conseguir que estas tengan la forma deseada que resulta de la caracterización frecuencial de las señales que intervienen en el lazo de realimentación. Esta caracterización la podemos representar por medio de la construcción de unas determinadas *Funciones de Peso*, que obedecen al comportamiento frecuencial deseado. Así, si representamos el comportamiento de la función de Sensibilidad con el peso  $W_S^{-1}(z)$  y el de la función de Sensibilidad Complementaria con  $W_T^{-1}(z)$ , el diseño consistirá en encontrar un controlador de manera que consigamos

$$\begin{aligned} S_0(z) &\approx W_S^{-1}(z) \\ T_0(z) &\approx W_T^{-1}(z) \end{aligned} \quad (4.9)$$

### 4.1.2 Descripción de la incertidumbre

El hecho que el modelo no refleje fielmente la dinámica del proceso, hace que el controlador diseñado en base a este modelo, pueda no responder a las prestaciones esperadas cuando se aplique sobre el proceso real.

Para tratar este problema es necesario ver primero como describir la incertidumbre asociada a la dinámica del proceso y deducir después las ecuaciones que resultan al considerar esta incertidumbre. En este apartado nos centraremos en el primer punto.

Partimos de un modelo  $G_0(z)$  del proceso (nominal) que representa más o menos bien el proceso real. Este proceso real, debido a variaciones en las condiciones de operación, degradación de componentes, etc, no será un proceso fijo con una dinámica fija, sino que presentará unas características variables en el tiempo. Dado que el controlador que diseñamos ha de operar bajo todas estas variaciones, el proceso real se describirá a partir de las variaciones que puede experimentar en relación al modelo nominal. Esto nos proporciona el conjunto de plantas para las cuales el controlador diseñado tendrá que conseguir las especificaciones.

Así, se tendrá en cuenta la presencia de incertidumbre a la hora de diseñar el controlador, considerando que la dinámica de la planta no está descrita por un único modelo LTI sino por una familia  $\mathcal{F}$  de modelos LTI.

Dado que la descripción de los procesos es mediante la función de transferencia, esta familia  $\mathcal{F}$  la describiremos en el dominio de la frecuencia. El hecho de tener una familia de modelos quiere decir que para una determinada frecuencia no tendremos unos valores únicos de magnitud y fase, sino un conjunto de valores posibles. Así, una función de transferencia se representará en el plano polar asociando no un punto a cada frecuencia, sino una región determinada. En esta región tendremos los posibles valores de magnitud y fase asociados a una determinada frecuencia.

Estas regiones pueden tener formas muy complejas, lo que dificulta su descripción matemática y, por tanto, su posterior tratamiento para poder extraer alguna conclusión útil de cara al control a diseñar. Es por esto que se adopta una simplificación de estas regiones considerando una descripción de las mismas tipo disco. De hecho, cualquier región por compleja que sea se puede aproximar por un disco.

Para describir estos discos se especifica para cada frecuencia el centro y

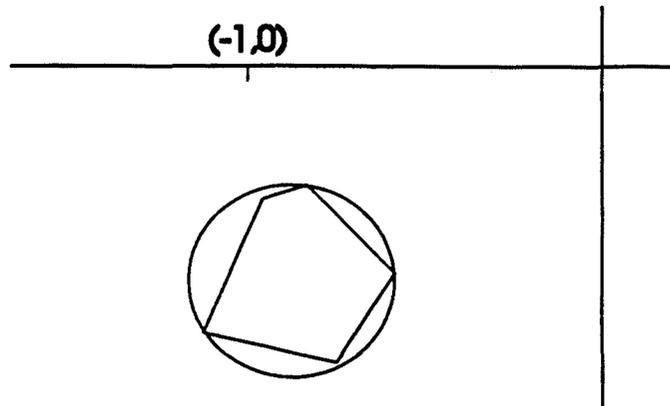


Figura 4.2: Región de incertidumbre aproximada por un disco

el radio del disco. Debido a que estamos describiendo variaciones alrededor de un modelo nominal, se toma este como centro, mientras que el radio viene dado por una función real  $r_a(\omega)$  que expresa la incertidumbre asociada a cada frecuencia.

Así, la familia  $\mathcal{F}$  de modelos estará descrita por

$$\mathcal{F} = \{G(e^{j\omega}) : |G(e^{j\omega}) - G_0(e^{j\omega})| \leq r_a(\omega)\} \quad 0 \leq \omega \leq \pi \quad (4.10)$$

de manera que cualquier miembro de la familia adoptará la forma

$$G(e^{j\omega}) = G_0(e^{j\omega}) + \Delta_a(e^{j\omega}) \quad 0 \leq \omega \leq \pi \quad (4.11)$$

donde

$$|\Delta_a(e^{j\omega})| \leq r_a(\omega) \quad 0 \leq \omega \leq \pi \quad (4.12)$$

Esta incertidumbre  $\Delta_a(e^{j\omega})$  se llama *Incertidumbre Aditiva*, ya que se perturba la planta aditivamente. Una descripción alternativa de esta familia de plantas  $\mathcal{F}$  se obtiene a partir de definir

$$\Delta_m(e^{j\omega}) = \frac{\Delta_a(e^{j\omega})}{G_0(e^{j\omega})} \quad 0 \leq \omega \leq \pi \quad (4.13)$$

$$r_m(\omega) = \frac{r_a(\omega)}{|G_0(e^{j\omega})|} \quad 0 \leq \omega \leq \pi \quad (4.14)$$

de manera que podemos escribir  $\mathcal{F}$  como

$$\mathcal{F} = \left\{ G(e^{j\omega}) : \frac{|G(e^{j\omega}) - G_0(e^{j\omega})|}{|G_0(j\omega)|} \leq r_m(\omega) \right\} \quad 0 \leq \omega \leq \pi \quad (4.15)$$

pudiéndose escribir cualquier miembro de la familia como

$$G(e^{j\omega}) = (1 + \Delta_m(e^{j\omega}))G_0(e^{j\omega}) \quad 0 \leq \omega \leq \pi \quad (4.16)$$

donde

$$|\Delta_m(e^{j\omega})| \leq r_m(\omega) \quad 0 \leq \omega \leq \pi \quad (4.17)$$

Esta incertidumbre  $\Delta_m(e^{j\omega})$  se llama *Incertidumbre Multiplicativa*.

El análisis posterior se hará básicamente sobre incertidumbres multiplicativas, pero conviene remarcar que además de estos dos tipos de descripción de incertidumbre existen otros. El propósito de estas descripciones es poder escoger la que mejor refleje nuestro conocimiento de la planta y de la incertidumbre asociada. En la Tabla 4.1 se resumen las descripciones de la incertidumbre más comunes.

Multiplicativa	$G = (1 + \Delta_m)G_0$
Aditiva	$G = G_0 + \Delta_a$
Multiplicativa Inversa	$G = G_0(1 + \Delta_{mi}G_0)^{-1}$
Aditiva Inversa	$G = G_0(1 + \Delta_{ai})^{-1}$

Tabla 4.1: Descripciones de la incertidumbre

### 4.1.3 Objetivos nominales y robustos

Uno de los objetivos que se persiguen en el diseño de un sistema de control es que el controlador responda a las expectativas cuando se ponga a funcionar sobre la planta real. Este objetivo suele ser muy difícil de cumplir. Por esto resulta ventajoso dividirlo en dos partes. Esta división se basa en el hecho que la representación que

tengamos de la planta a partir de un cierto modelo nominal  $G_0(z)$  es tan solo una aproximación y, por tanto, tendremos asociada una incertidumbre a través de la cual, tal y como hemos visto, se contemplan las diferentes dinámicas en que es posible que se encuentre el controlador cuando se implemente para operar con la planta real.

La existencia de un modelo nominal y de funciones de incertidumbre nos permite hablar de

**Objetivos nominales** : donde se tiene en cuenta el funcionamiento del controlador únicamente sobre el modelo nominal  $G_0(s)$  del sistema.

**Objetivos robustos** : donde se tiene en cuenta el funcionamiento del controlador para los miembros de la familia  $\mathcal{F}$  de plantas especificada a partir de las funciones de incertidumbre.

Como podemos ver, en estos dos grupos los objetivos que se persiguen son los mismos, pero aplicados a diferentes situaciones. En cuanto a los objetivos, por una banda tenemos el requisito mínimo de la estabilidad, en el sentido que el controlador ha de conseguir que, como mínimo, el sistema en lazo cerrado sea estable. Así, tendremos que en el primer caso se exigirá que el controlador estabilice el modelo nominal dando lugar al requerimiento de *Estabilidad Nominal*. En el segundo caso, se exigirá que el sistema en lazo cerrado sea estable para cualquier planta perteneciente a la familia de posibles plantas. Es el requerimiento de *Estabilidad Robusta*.

Por otra parte, el objetivo del controlador, además de estabilizar, es el de mantener el error entre la salida de la planta  $y$  y la referencia  $r$  lo más pequeño posible, teniendo en cuenta que el proceso se verá afectado por las señales externas  $d$  y  $n$ . En el caso que se halla de mantener este error lo más pequeño posible tan sólo para el modelo nominal, tendremos el requerimiento de *Rendimiento Nominal*. En el segundo caso, donde se exige lo mismo para cualquier planta perteneciente a la familia especificada, tendremos el requerimiento de *Rendimiento Robusto*.

Estas distinciones nos permiten afrontar el problema de diseño de una forma

más estructurada. Así, lo que nos interesa es obtener unas condiciones de análisis para los requerimientos robustos. El proceso que se sigue es el siguiente: obtener primero las condiciones de análisis para los requerimientos nominales, y ver después como se modifican estas condiciones, como consecuencia de incorporar las funciones de incertidumbre para obtener las condiciones de análisis para los requerimientos robustos.

### Objetivos nominales

Veamos en primer lugar como se establecen las condiciones para conseguir los objetivos nominales.

#### Estabilidad Nominal

El requerimiento de estabilizar el sistema nominal puede expresarse en términos de los resultados clásicos de la teoría de la estabilidad, que establece criterios para asegurar que ciertos polinomios tengan sus raíces en una cierta región.

#### Rendimiento Nominal

Para mantener pequeño el error entre la salida del sistema y la salida de referencia, obsérvese que estas dos señales están relacionadas de la siguiente forma

$$\frac{e}{r} = S_0(z) = \frac{1}{1 + K(z)G_0(z)} \quad (4.18)$$

y por tanto será necesario que la ganancia de la función de Sensibilidad sea la más pequeña posible. Dado que la mayoría de sistemas son estrictamente propios, tendremos que

$$\lim_{\omega \rightarrow \pi} G_0(e^{j\omega}) = 0 \quad (4.19)$$

lo que implica

$$\lim_{\omega \rightarrow \pi} |S_0(e^{j\omega})| = \lim_{\omega \rightarrow \pi} \left| \frac{1}{1 + K(e^{j\omega})G_0(j\omega)} \right| = 1 \quad (4.20)$$

y será imposible de conseguir  $|S_0(e^{j\omega})| = 0$  para todas las frecuencias. Para esto, será

necesario especificar el rango de frecuencia para el cual nos interesará tener  $|S_0(e^{j\omega})| = 0$ ; es decir, que clase de señales de referencia interesa que siga el sistema.

Por otra parte, la función de Sensibilidad es también la función de transferencia entre la perturbación externa  $d$  y la salida. Así, exigiendo que  $y \approx r$  en un cierto rango de frecuencias, estamos exigiendo a la vez el rechazo de las perturbaciones que tengan su contenido frecuencial en este rango. Los dos requerimientos son de hecho equivalentes.

Tal y como se ha visto anteriormente, este comportamiento deseado de la función de Sensibilidad se puede especificar a través de una función de peso  $W_S^{-1}(z)$ , teniendo planteada la condición de rendimiento nominal como

$$|S_0(e^{j\omega})| \leq |W_S^{-1}(e^{j\omega})| \quad 0 \leq \omega \leq \pi \quad (4.21)$$

que podemos escribir también como

$$|W_S(e^{j\omega})S_0(e^{j\omega})| \leq 1 \quad 0 \leq \omega \leq \pi \quad (4.22)$$

condición equivalente a

$$\|W_S(z)S_0(z)\|_\infty \doteq \sup_{|z|=1} |W_S(z)S_0(z)| \leq 1 \quad (4.23)$$

con lo que obtenemos una condición sobre la norma infinito de la función  $W_S(z)S_0(z)$ .

De forma equivalente se podría establecer una condición de rendimiento nominal sobre la función de sensibilidad complementaria  $T_0$ . En este caso el comportamiento deseado de la función de sensibilidad complementaria lo especificaríamos a través de otra función de peso  $W_T^{-1}(z)$ , quedando la condición de rendimiento nominal como

$$\|W_T(z)T_0(z)\|_\infty \leq 1 \quad (4.24)$$

### Objetivos robustos

Una vez deducidas las ecuaciones de análisis para el caso nominal, hemos de ver como podemos incluir condiciones de robustez teniendo en cuenta la presencia de

incertidumbre.

### Estabilidad Robusta

Vamos a deducir alguna condición que nos permita asegurar que un cierto controlador satisface el requerimiento de estabilidad robusta, es decir, que el sistema en lazo cerrado que resulte con cualquier planta de la familia  $\mathcal{F}$  ha de ser estable. Aplicaremos el criterio de Nyquist, haciendo la suposición que todos los miembros de la familia  $\mathcal{F}$  tienen el mismo número de polos inestables.

Sea  $N(-1, H(z))$  el número de rodeos al punto  $(-1, 0)$  de la imagen del contorno de Nyquist a través de la función  $H(z)$ . Un controlador  $K(z)$  satisfará la condición de estabilidad robusta si y sólo si

$$N(-1, K(z)G(z)) = -n \quad \forall G(z) \in \mathcal{F} \quad (4.25)$$

donde  $n$  es el número de polos inestables. Un caso particular de la condición anterior, lo tenemos cuando consideramos la planta nominal  $G_0(z)$ , ya que para esta planta el controlador  $K$  también ha de estabilizar el lazo cerrado. Por tanto,

$$N(-1, K(z)G_0(z)) = -n \quad (4.26)$$

Suponiendo que esta última condición se satisface, la condición de estabilidad robusta se satisfará, si y sólo si, el número de rodeos no varía para ningún miembro de la familia  $\mathcal{F}$ . Esto es lo mismo que decir que la banda de Nyquist, formada al recorrer  $G(z)$  todo  $\mathcal{F}$ , no incluya el punto  $(-1, 0)$ .

A partir de la Fig. 4.3, se puede ver que esta condición es equivalente a decir que la distancia desde  $K(e^{j\omega})G_0(e^{j\omega})$  al punto  $(-1, 0)$  no sea más grande que el radio  $|K(e^{j\omega})G_0(j\omega)|r_m(\omega)$  del disco. Es decir

$$|1 + K(e^{j\omega})G_0(e^{j\omega})| > |K(e^{j\omega})G_0(e^{j\omega})|r_m(\omega), \quad 0 \leq \omega \leq \pi \quad (4.27)$$

que se puede expresar como

$$\left| \frac{K(e^{j\omega})G_0(e^{j\omega})}{1 + K(e^{j\omega})G_0(e^{j\omega})} \right| r_m(\omega) = |T_0(e^{j\omega})| r_m(\omega) < 1, \quad 0 \leq \omega \leq \pi \quad (4.28)$$

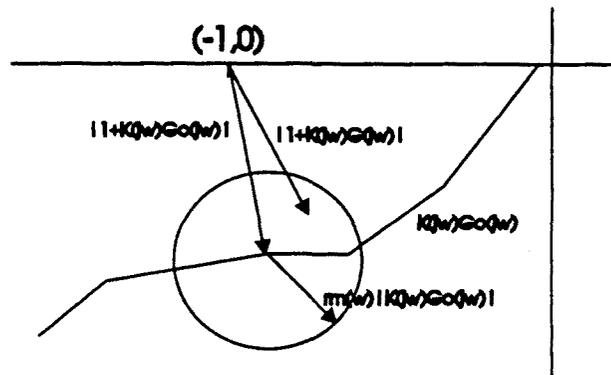


Figura 4.3: Esquema geométrico para la estabilidad robusta

o bien

$$\|T_0(e^{j\omega})r_m(\omega)\|_\infty \doteq \max_{\omega} |T_0(e^{j\omega})r_m(\omega)| < 1, \quad 0 \leq \omega \leq \pi \quad (4.29)$$

Obtenemos así, que el requerimiento de Estabilidad Robusta impone una condición sobre la norma infinito de la función de Sensibilidad Complementaria sobre la que actúa la función de incertidumbre  $r_m(\omega)$  como función de peso.

Como se ha visto anteriormente, el ruido en la medida impone también una condición sobre la función de Sensibilidad Complementaria. Por tanto, a partir de las dos condiciones, habrá que construir un función de peso de forma que ambas restricciones queden incluidas.

Para otras descripciones de la incertidumbre se obtienen condiciones similares de estabilidad robusta. En la Tabla 4.2 se resumen las condiciones de estabilidad robusta para otras descripciones de la incertidumbre.

### Rendimiento Robusto

Para el requerimiento de Rendimiento Robusto será necesario que la condición de Rendimiento Nominal (4.23) se satisfaga para cualquier planta  $G(s) \in \mathcal{F}$ . Es

Descripción	Condición
$(1 + \Delta_m)G_0$	$\ T_0 r_m\ _\infty < 1$
$G_0 + \Delta_a$	$\ K S_0 r_a\ _\infty < 1$
$G_0(1 + \Delta_{mi}G_0)^{-1}$	$\ G_0 S_0 r_{mi}\ _\infty < 1$
$G_0(1 + \Delta_{ai})^{-1}$	$\ S_0 r_{ai}\ _\infty < 1$

Tabla 4.2: Condiciones de estabilidad robusta

decir,

$$\|W_S(z)S(z)\|_\infty \leq 1 \quad \forall G(z) \in \mathcal{F} \quad (4.30)$$

Lo siguiente es reescribir esta condición en términos de las funciones nominales. A partir de la Fig. 4.3 se puede observar la siguiente desigualdad

$$|1 + K(e^{j\omega})G(e^{j\omega})| \geq |1 + K(e^{j\omega})G_0(e^{j\omega})| - |K(e^{j\omega})G_0(e^{j\omega})| \quad \forall G(z) \in \mathcal{F} \quad (4.31)$$

o bien

$$|S(e^{j\omega})| = \left| \frac{1}{1 + K(e^{j\omega})G(e^{j\omega})} \right| \leq \frac{|S_0(e^{j\omega})|}{1 - |T_0(e^{j\omega})|r_m(\omega)} \quad \forall G(z) \in \mathcal{F} \quad (4.32)$$

Con esta desigualdad la condición de rendimiento robusto se puede escribir como

$$\frac{|W_S(e^{j\omega})S_0(e^{j\omega})|}{1 - |T_0(e^{j\omega})|r_m(\omega)} < 1, \quad 0 \leq \omega \leq \pi \quad (4.33)$$

o bien

$$|T_0(e^{j\omega})r_m(\omega)| + |W_S(e^{j\omega})S_0(e^{j\omega})| < 1, \quad 0 \leq \omega \leq \pi \quad (4.34)$$

Con lo que obtenemos otra condición sobre la norma infinito de una determinada función. En este caso

$$\| |T_0(e^{j\omega})r_m(\omega)| + |W_S(e^{j\omega})S_0(e^{j\omega})| \|_\infty < 1 \quad (4.35)$$

Podemos observar en esta ecuación que para satisfacer el requerimiento de Rendimiento Robusto se han de satisfacer las de Estabilidad Robusta y Rendimiento Nominal

con un cierto margen. Así, si conseguimos  $|T_0(e^{j\omega})r_m(\omega)| < \beta(\omega)$  y por otra parte

$$|W_S(e^{j\omega})S_0(e^{j\omega})| < 1 - \beta(\omega) \tag{4.36}$$

donde  $\beta(\omega) < 1$  entonces conseguiremos satisfacer la condición de Rendimiento Robusto.

En la Tabla 4.3 se resumen las condiciones de rendimiento robusto para cuatro formas de describir la incertidumbre y para dos formas de especificar la condición de Rendimiento Nominal. En algunas casillas aparece *complejo* y significa precisamente eso, que la expresión resultante es compleja.

Descripción	Condición de Rendimiento Nominal	
	$\ W_S S_0\ _\infty < 1$	$\ W_T T_0\ _\infty < 1$
$(1 + \Delta_m)G_0$	$\   W_S S_0  +  T_0 r_m  \ _\infty < 1$	complejo
$G_0 + \Delta_a$	$\   W_S S_0  +  K S_0 r_a  \ _\infty < 1$	complejo
$G_0(1 + \Delta_{mi}G_0)^{-1}$	complejo	$\   W_T T_0  +  G_0 S_0 r_{mi}  \ _\infty < 1$
$G_0(1 + \Delta_{ai})^{-1}$	complejo	$\   W_T T_0  +  S_0 r_{ai}  \ _\infty < 1$

Tabla 4.3: Condiciones de Rendimiento Robusta

Es interesante comentar que actualmente el diseño de controladores con rendimiento robusto es todavía un problema abierto. Comúnmente, se satisfacen únicamente las condiciones de estabilidad robusta y la de rendimiento nominal. Así, se asegura para todo el conjunto de plantas posibles el requisito mínimo de estabilidad, mientras que el rendimiento en condiciones diferentes de las nominales se verá deteriorado.

## 4.2 Expresión del GPC como un Controlador de Dos Grados de Libertad

El GPC como otros controladores predictivos se puede expresar como un controlador LTI clásico. Este es el primer paso para poder ser estudiado mediante la teoría del control robusto.

El modelo usado por este controlador es un CARIMA que en su versión estándar no incluye ningún término de perturbación medible a diferencia del que se uso en el capítulo 2 para deducir el GPC,

$$A_0(q^{-1})y(t) = B_0(q^{-1})u(t) + \frac{T(q^{-1})}{\Delta}\xi(t) \quad (4.37)$$

donde  $B(q^{-1})$  tiene como mínimo un retardo de 1. Este modelo nos conduce a las ecuaciones de predicción de la salida en términos de la respuesta libre y forzada de la siguiente forma,

$$\hat{y}(t+j) = G_j(q^{-1})\Delta u(t+j) + p_j, \quad j \geq 0 \quad (4.38)$$

La respuesta libre  $p_j$  la habíamos calculado iterando de manera recursiva sobre el modelo (4.37). Otra forma de calcularla es resolviendo las siguientes ecuaciones diofánticas

$$\begin{aligned} T(q^{-1}) &= A_0(q^{-1})\Delta E_j(q^{-1}) + q^{-j}F_j(q^{-1}) \\ E_j(q^{-1})B_0(q^{-1}) &= G_j(q^{-1})T(q^{-1}) + q^{-j}H_j(q^{-1}) \end{aligned} \quad (4.39)$$

De esta forma la predicción de la respuesta libre nos queda

$$p_j = \frac{F_j(q^{-1})}{T(q^{-1})}y(t) + \frac{H_j(q^{-1})}{T(q^{-1})}\Delta u(t-1) \quad (4.40)$$

Una vez obtenidas las ecuaciones de predicción, los incrementos futuros de la señal de control se obtenían minimizando la función de coste (2.3). La solución de este problema de optimización nos proporcionaba la ley de control

$$\Delta \mathbf{u}_{opt} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{w} - \mathbf{p}) \quad (4.41)$$

Dado que los CPBM usan la estrategia del horizonte móvil, no hace falta calcular todos los incrementos futuros de control, sino sólo el primero. Así, podemos definir los coeficientes  $k_i$ ,  $i = 1, \dots, N_2$  correspondientes a

$$(k_1 \ \dots \ k_{N_2}) = (1 \ 0 \ \dots \ 0)(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T \quad (4.42)$$

y de esta forma se puede reescribir la ley de control como

$$\Delta u(t) = \sum_{i=N_1}^{N_2} k_i (w(t+i) - p_i) \quad (4.43)$$

Introduciendo en esta ecuación la expresión de la predicción de la respuesta libre obtenemos

$$\Delta u(t) = \sum_{i=N_1}^{N_2} k_i w(t+i) - \sum_{i=N_1}^{N_2} k_i \frac{H_i(q^{-1})}{T(q^{-1})} \Delta u(t-1) - \sum_{i=N_1}^{N_2} k_i \frac{F_i(q^{-1})}{T(q^{-1})} y(t) \quad (4.44)$$

Y esto nos conduce a la expresión de un controlador de dos grados de libertad<sup>1</sup>

$$R_p \Delta u(t) = T_p w(t) - S_p y(t) \quad (4.45)$$

siendo

$$\begin{aligned} R_p &= T + \sum_{i=N_1}^{N_2} k_i H_i q^{-1} \\ T_p &= T \sum_{i=N_1}^{N_2} k_i q^i \\ S_p &= \sum_{i=N_1}^{N_2} k_i F_i \end{aligned} \quad (4.46)$$

La estructura del GPC como un controlador de dos grados de libertad se puede ver en la Fig. 4.4.

Esto demuestra que el GPC es una forma de sintetizar controladores LTI mediante un criterio de optimización. Lógicamente, cualquier controlador lineal independientemente de método de síntesis, tendrá una estructura como la anterior, aunque haya casos en que sea difícil encontrarla.

<sup>1</sup>Por claridad notacional a los tres polinomios del controlador de dos grados de libertad se les ha denominado  $R_p$ ,  $S_p$  y  $T_p$ .

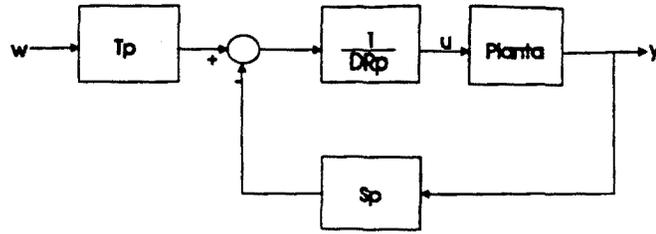


Figura 4.4: Estructura del GPC como un controlador de dos grados de libertad

Combinando el modelo de la planta (4.37) con la expresión del controlador (4.45) y descomponiendo  $T_p = TT_1$  obtenemos la ecuación del sistema en lazo cerrado,

$$(A_0\Delta R_p + B_0S_pq^{-1})y(t) = B_0TT_1w(t) + TR_p\xi(t) \quad (4.47)$$

Usando las definiciones de  $R_p$ ,  $S_p$  y  $T_1$  y después de algunos cálculos, se puede demostrar que,

$$\begin{aligned} A_0\Delta R_p + B_0S_pq^{-1} &= A_0\Delta T + \sum_{i=N_1}^{N_2} k_i(A_0\Delta H_i + BF_i)q^{-1} \\ &= T(A_0\Delta + \sum_{i=N_1}^{N_2} N_2k_i(B_0 - A_0\Delta G_i)q^{i-1}) \\ &= TP_{c0} \end{aligned} \quad (4.48)$$

y la ecuación en lazo cerrado queda finalmente

$$y(t) = \frac{B_0T_1}{P_{c0}}w(t) + \frac{R_p}{P_{c0}}\xi(t) \quad (4.49)$$

donde se observa que  $T$  se cancela en la función de transferencia en lazo cerrado como es habitual con las dinámicas de los observadores, y la estabilidad y rendimiento nominal del sistema en lazo cerrado está gobernado por las raíces del polinomio  $P_{c0}$ .

### Funciones de sensibilidad y sensibilidad complementaria

Una vez obtenida la expresión de GPC vamos a ver como definir las funciones de sensibilidad  $S_0$  y sensibilidad complementaria  $T_0$ . Primero, vamos a distinguir entre

el modelo nominal usado para el diseño del GPC ( $G_0 = B_0/A_0$ ) y el modelo real de la planta ( $G = B/A$ ) que no es conocido exactamente.

La función de sensibilidad complementaria vimos que era la función de transferencia entre la referencia  $w$  y la salida de la planta  $y$ . Por tanto, esta función ya la tenemos definida en (4.49),

$$T_0 = \frac{B_0 T_1}{P_{c0}} \quad (4.50)$$

La función de sensibilidad es la función de transferencia entre la perturbación  $d$  y la salida  $y$ . Observando la Fig. 4.4 vemos que la perturbación  $d$  tiene la siguiente expresión,

$$d = \frac{T}{\Delta A_0} \xi \quad (4.51)$$

Combinando esta expresión con el segundo término de la ecuación del sistema en lazo cerrado (4.49) obtenemos la función de sensibilidad

$$S_0 = \frac{A_0 \Delta R}{T P_{c0}} \quad (4.52)$$

En la siguiente sección haremos un estudio de la influencia de los parámetros de diseño del GPC sobre estas funciones, pero antes vamos a ver como se define la estabilidad robusta a partir, también, de éstas.

### Estabilidad robusta

La estabilidad nominal del sistema en lazo cerrado se garantiza si las raíces de  $P_{c0}$  están dentro del círculo unidad. Vamos a ver que condición se ha de cumplir para que el sistema sea estable para un conjunto de plantas, dentro del cual estará la planta real.

Así, si las plantas consideradas las describimos mediante incertidumbre multiplicativa, su relación con la planta nominal es

$$G(e^{j\omega}) = (1 + \Delta_m(e^{j\omega}))G_0(e^{-j\omega}), \quad 0 \leq \omega \leq \pi \quad (4.53)$$

La incertidumbre asociada a esta planta respecto a la planta nominal se puede calcular como

$$\begin{aligned}\Delta_m &= \frac{G(e^{j\omega}) - G_0(e^{j\omega})}{G_0(e^{j\omega})} \\ &= \left| \frac{A_0 B - A B_0}{A B_0} \right|\end{aligned}\quad (4.54)$$

La familia de plantas  $\mathcal{F}$  definirá una cota máxima para la incertidumbre multiplicativa que caracterizará esta familia,

$$|\Delta_m(e^{j\omega})| \leq r_m(\omega), \quad 0 \leq \omega \leq \pi \quad (4.55)$$

La condición de estabilidad robusta podemos expresarla ahora como

$$\left| \frac{B_0(e^{j\omega})T_1(e^{j\omega})}{P_{c0}(e^{j\omega})} \right| < \frac{1}{r_m(\omega)}, \quad 0 \leq \omega \leq \pi \quad (4.56)$$

suponiendo que todas las plantas de la familia  $\mathcal{F}$  tienen el mismo número de polos en el semiplano derecho y que el sistema nominal en lazo cerrado es estable.

Para otras descripciones de la incertidumbre se pueden deducir otras expresiones para la condición de estabilidad robusta, pero todas resultan más complejas.

## Rendimiento robusto

La condición de rendimiento robusto para incertidumbres multiplicativas está expresada en (4.35). El término  $W_S^{-1}$  permitía describir el comportamiento deseado del sistema.

Una interpretación más útil de este término en control predictivo es considerarlo como una descripción del espectro de energía del conjunto de señales de referencia que esperamos o que vamos a permitir entrar en el sistema. Esto tiene mucho interés ya que el filtro de la referencia no permitirá eliminar las frecuencias que no se ajusten a este espectro y obtener una trayectoria de referencia con un contenido frecuencial que esté incluido en  $W_S$ .

### 4.3 Influencia de los parámetros de sintonía

El conocimiento de como influyen los parámetros de sintonía de los controladores predictivos en la robustez y el rendimiento del sistema de control ha sido uno de los temas de estudio desde que salió a la luz el primer controlador predictivo. En esta sección el estudio se va a enfocar usando la teoría del control robusto presentada anteriormente.

Debido a que los controladores predictivos calculan la señal de control a partir de un modelo interno, la ley de control resultante no depende únicamente de los parámetros de sintonía. Es por esto que el estudio se llevará a cabo sobre tres sistemas distintos: un sistema de primer orden, un sistema de segundo orden y el sistema benchmark.

En la industria de procesos una gran mayoría de procesos se aproximan por un sistema de primer orden y, por tanto, es muy adecuada su elección. Consideremos un sistema de primer orden general

$$G_0(s) = \frac{K}{\tau s + 1} \quad (4.57)$$

Para los experimentos se ha elegido un modelo nominal de ganancia  $K = 1$  y constante de tiempo  $\tau = 0.6$  y el sistema se ha muestreado con periodo  $0.2s$ .

Mediante sistemas de segundo orden también son muchos los procesos que se pueden aproximar, en particular son adecuados para procesos cuya respuesta tenga sobrepico. Consideremos un sistema de segundo orden general

$$G_0(s) = \frac{\omega_o^2}{s^2 + 2\xi\omega_o s + \omega_o^2} \quad (4.58)$$

Para los experimentos se ha elegido un sistema nominal con frecuencia natural  $\omega_o = 2.5$ , factor de amortiguamiento  $\xi = 0.6$  y también se ha muestreado con periodo  $0.2s$ .

El tercer sistema que usaremos en el estudio ya se ha visto con anterioridad y tiene la característica de ser un sistema complicado que presenta retardo, fase

no mínima y respuesta inversa. Veremos que al usar un modelo de este sistema en el controlador, la ley de control que resulta tendrá también un comportamiento complicado.

Para los experimentos que se van a realizar utilizaremos un GPC por las razones que antes ya hemos comentado y consisten en ir variando los valores de los parámetros  $N_1$ ,  $N_2$ ,  $N_u$ ,  $\lambda$  y  $T$ , y observar como se modifican las curvas de sensibilidad y sensibilidad complementaria para los tres sistemas que se han descrito. El objetivo es sacar patrones de comportamiento de estas curvas cuando varías alguno de los parámetros de manera que seamos capaces de conseguir la forma deseada de las curvas de sensibilidad y sensibilidad complementaria.

La variación de los parámetros se ha hecho a partir de unos valores base. Así, para el inicio del horizonte de coincidencia se ha fijado  $N_1 = 1$  excepto para el sistema benchmark que como presenta un retardo se ha fijado a 2. Para el horizonte de predicción se ha fijado  $N_2 = 15$ , para el horizonte de control  $N_u = 1$ , para el factor de peso  $\lambda = 0$  y para el polinomio  $T = 1$ .

### Influencia de $N_1$

El primer parámetro que se estudia es el inicio del horizonte de coincidencia y los valores que va ha tomar son 1, 2, 5 y 10. En las Fig. 4.5, 4.6 y 4.7 se muestran las curvas de sensibilidad y sensibilidad complementaria para el sistema de primer orden, de segundo orden y el benchmark respectivamente.

Lo primero que observamos es que este parámetro no influye en absoluto en la sensibilidad y, por tanto, en el rechazo de perturbaciones.

En cuanto a la sensibilidad complementaria, o sea la respuesta en lazo cerrado, observamos que para los sistemas de primer y segundo orden no tiene ninguna influencia, sin embargo para el sistema benchmark si. El sistema benchmark como se había comentado presenta un retardo de 2 periodos de muestreo y tiene respuesta

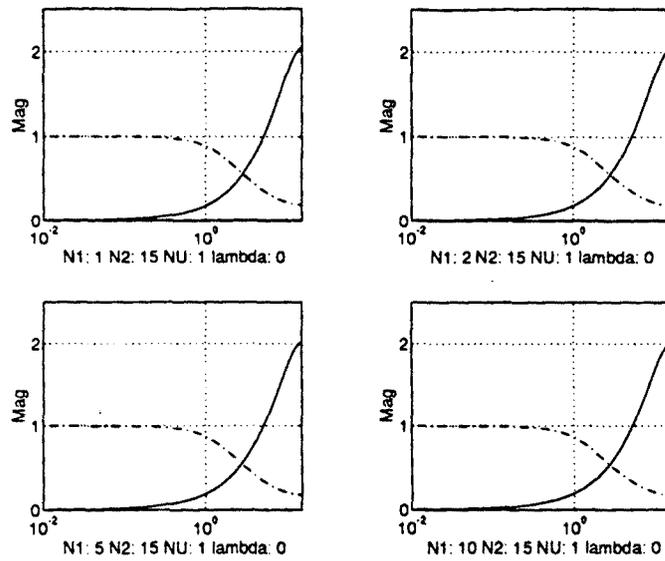


Figura 4.5: Funciones S y T para un sistema de primer orden variando  $N_1$

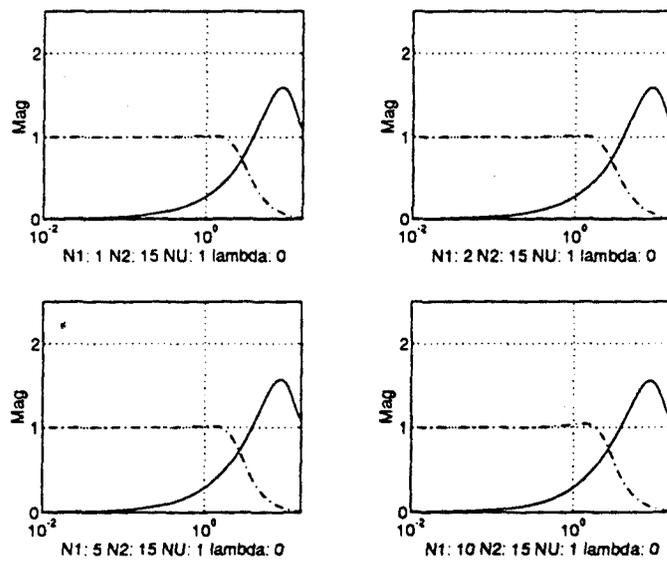


Figura 4.6: Funciones S y T para un sistema de segundo orden variando  $N_1$

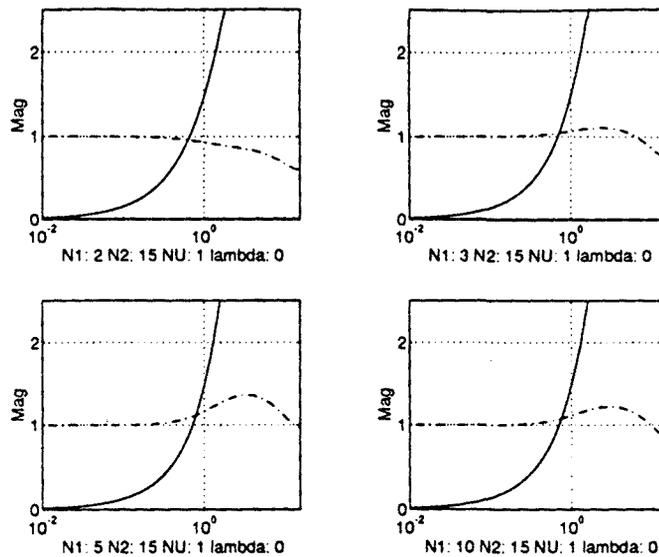


Figura 4.7: Funciones S y T para el sistema benchmark variando  $N_1$

inversa. Así, si  $N_1$  es igual o menor que el retardo (2), el sistema tiene un ancho de banda pequeño, pero si es mayor que el retardo, el ancho de banda aumenta e incluso presenta un pico de resonancia que se atenúa cuando aumenta  $N_1$ .

Por tanto, para sistemas sencillos el parámetro  $N_1$  no influye, pero con sistemas más complejos adquiere gran importancia. Si  $N_1$  lo hacemos mayor que el retardo, aumentaremos el ancho de banda del sistema de control. Por otra parte, si el horizonte de coincidencia no incluye la parte de la respuesta inversa mejoran las características dinámicas.

### Influencia del horizonte de predicción $N_2$

Los valores que vamos a tomar para el horizonte de predicción son 1, 2, 5 y 15, excepto para el sistema benchmark que con valores menores de 10 se hace inestable y, por tanto, tomaremos valores 10, 12, 15 y 20. En las Fig. 4.8, 4.9 y 4.10

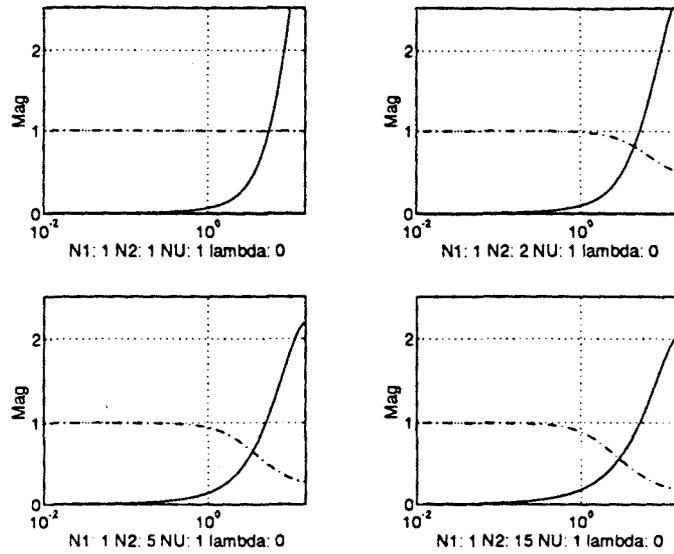


Figura 4.8: Funciones S y T para un sistema de primer orden variando  $N_2$

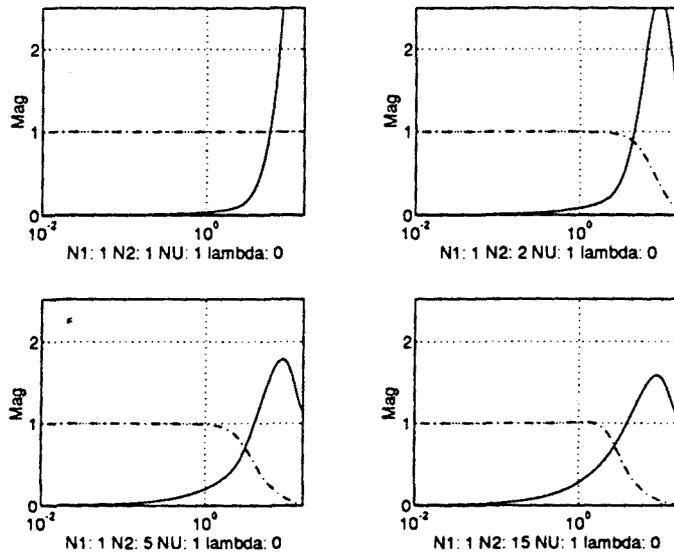


Figura 4.9: Funciones S y T para un sistema de segundo orden variando  $N_2$

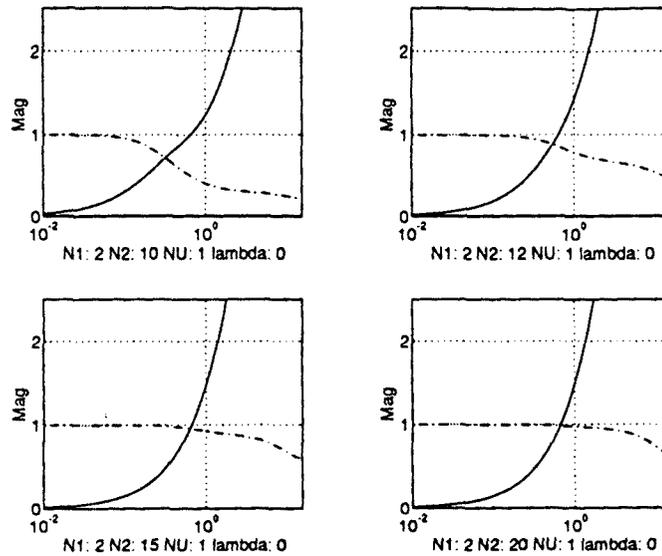


Figura 4.10: Funciones S y T para el sistema benchmark variando  $N_2$

se muestran las curvas de sensibilidad y sensibilidad complementaria para el sistema de primer orden, de segundo orden y el benchmark respectivamente.

Observamos en estas gráficas que el comportamiento de los sistemas de primer y segundo orden es muy distinto al del sistema benchmark. En los dos primeros casos vemos que al aumentar  $N_2$  la curva de sensibilidad atenúa más, con lo que rechazará mejor las perturbaciones. Por otro lado, la curva de sensibilidad complementaria también se ve afectada, y observamos que al aumentar el horizonte de predicción disminuimos el ancho de banda del sistema. Por tanto, el horizonte de predicción reduce el rendimiento del sistema y aumenta su robustez.

En el caso de sistema benchmark, la curva de sensibilidad apenas se ve afectada aunque se puede observar una ligera mejora en el rechazo a perturbaciones. Un cuanto a la sensibilidad complementaria, esta tiene un comportamiento opuesto al de los sistemas anteriores. Esto se puede explicar teniendo en cuenta que este sistema tiene respuesta inversa y al aumentar el horizonte de predicción esta parte

de la respuesta tiene menos importancia en la función de coste con lo que mejora la respuesta del sistema y su ancho de banda. Como conclusión, ante sistemas con retardos y respuesta inversa es conveniente escoger un horizonte de coincidencia que no incluya ni el retardo ni la respuesta inversa.

### **Influencia del horizonte de control $N_u$**

Los valores que vamos a tomar para el horizonte de control son 1, 2, 5 y 10. En las Fig. 4.11, 4.12 y 4.13 se muestran las curvas de sensibilidad y sensibilidad complementaria para el sistema de primer orden, de segundo orden y el benchmark respectivamente.

Obsevamos en las gráficas que al aumentar el horizonte de control la curva de sensibilidad crece a altas frecuencias, con lo que las perturbaciones de alta frecuencia se amplificaran más. Este es un comportamiento común a los tres sistemas.

Si observamos las curvas de sensibilidad complementaria vemos que al aumentar  $N_u$  se incrementa el ancho de banda del sistema, aunque en el sistema benchmark para  $N_u$  muy grandes tiene un comportamiento extraño. Conviene notar también que en el sistema de primer orden para valores mayores de  $N_u = 2$  no se observa ninguna mejora en el rendimiento. En conclusión, el incremento del horizonte de control aumenta el rendimiento, pero provoca una amplificación de las perturbaciones de alta frecuencia.

### **Influencia del factor de peso $\lambda$**

Los valores que vamos a tomar para el factor de peso son 0, 2, 10 y 50. En las Fig. 4.14, 4.15 y 4.16 se muestran las curvas de sensibilidad y sensibilidad complementaria para el sistema de primer orden, de segundo orden y el benchmark respectivamente.

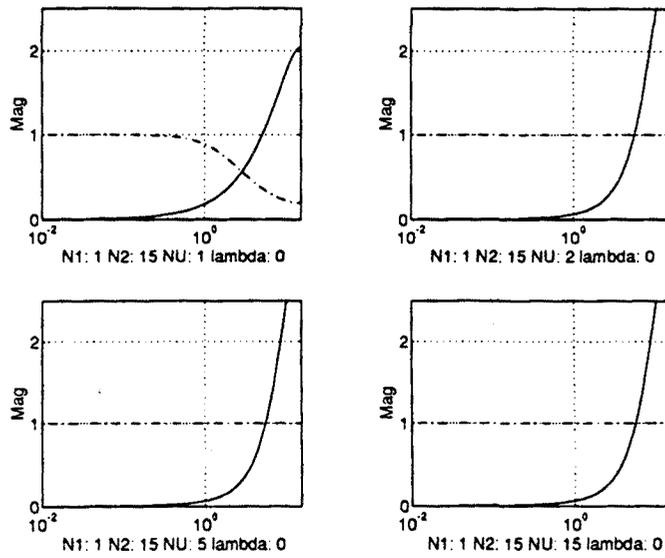


Figura 4.11: Funciones S y T para un sistema de primer orden variando  $N_u$

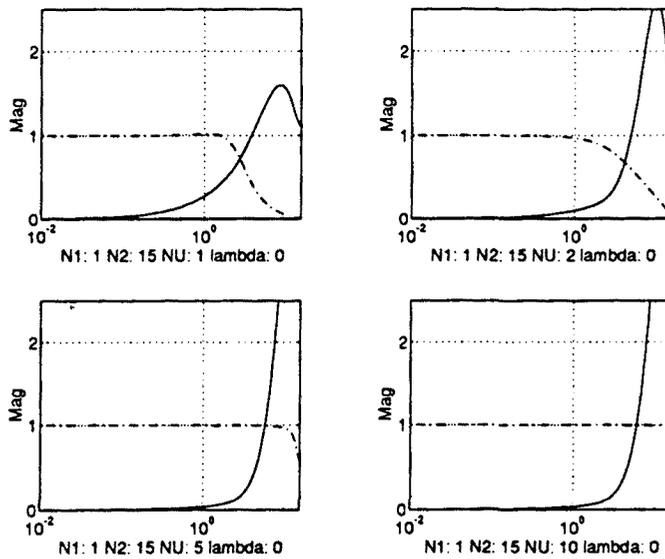


Figura 4.12: Funciones S y T para un sistema de segundo orden variando  $N_u$

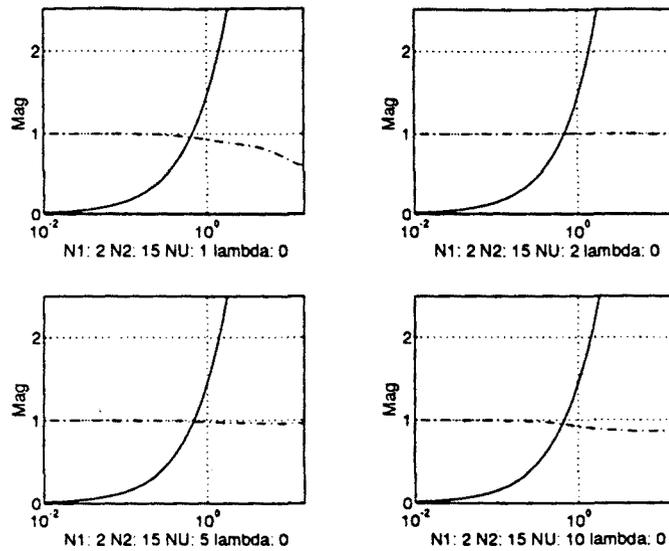


Figura 4.13: Funciones S y T para el sistema benchmark variando  $N_u$

En estas gráficas se observa que este parámetro tiene una gran y clara influencia sobre el rendimiento y la robustez. Respecto a la sensibilidad se ve claramente que al aumentar el factor de peso el sistema rechaza mejor las perturbaciones con lo que se consigue robustez.

En cuanto al rendimiento vemos también claramente que al aumentar  $\lambda$  se reduce el ancho de banda del sistema. Este es el parámetro que tiene una influencia más directa sobre las funciones de sensibilidad y sensibilidad complementaria. Sin embargo, escoger un valor en concreto no es inmediato, ya que el valor del factor de peso dependerá entre otras cosas de las unidades en que estén las variables. En estos ejemplos se ve que, dependiendo del sistema, un valor del factor de peso tiene más influencia o menos. Por tanto, la forma de sintonizar este parámetro será a partir de cero ir aumentándolo hasta conseguir un buen compromiso entre el rendimiento y la robustez del sistema.

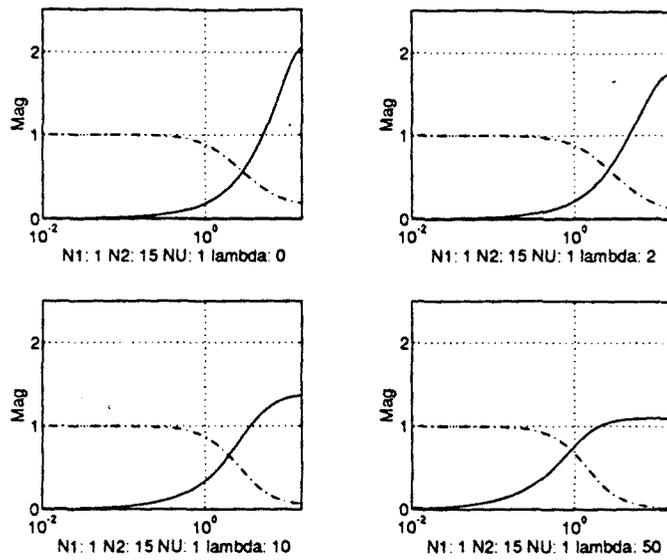


Figura 4.14: Funciones S y T para un sistema de primer orden variando  $\lambda$

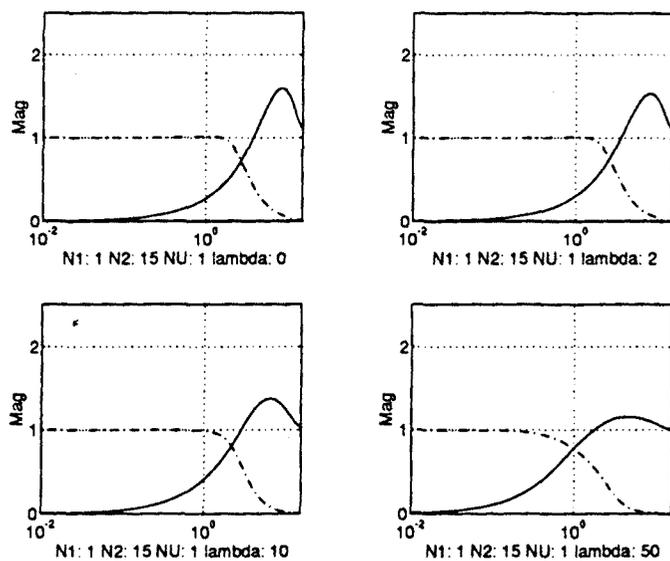


Figura 4.15: Funciones S y T para un sistema de segundo orden variando  $\lambda$

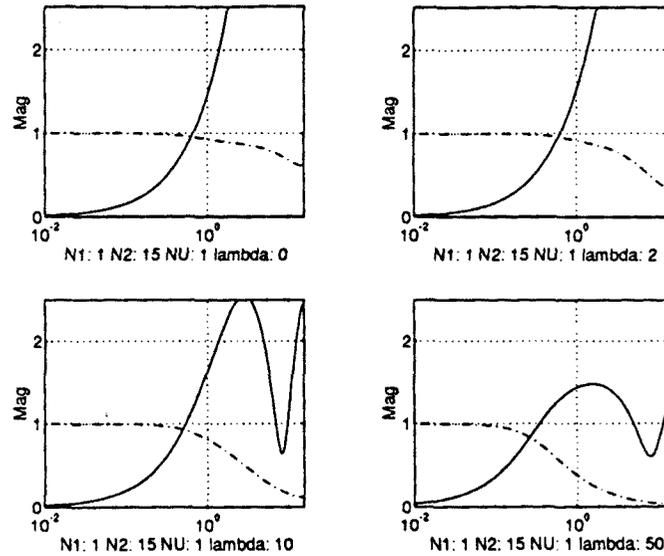


Figura 4.16: Funciones S y T para el sistema benchmark variando  $\lambda$

### Influencia del polinomio $T$

El último parámetro de diseño es el polinomio  $T$ . En la sección anterior vimos que en la función de transferencia en lazo cerrado no aparecía el polinomio  $T$ , por tanto, éste no influirá en el rendimiento. Donde si tiene una influencia muy importante es sobre la robustez. Vamos a ver como resultan las funciones de sensibilidad y sensibilidad complementaria para cuatro polinomios  $T$ . Estos polinomios serán:

$$\begin{aligned}
 T_1(q^{-1}) &= 1 \\
 T_2(q^{-1}) &= A(q^{-1}) \\
 T_3(q^{-1}) &= 1 - 0.9q^{-1} \\
 T_4(q^{-1}) &= A(q^{-1})(1 - 0.9q^{-1})
 \end{aligned}$$

donde  $A$  es el denominador de la función de transferencia del modelo nominal. En las Fig. 4.17, 4.18 y 4.19 se muestran las curvas de sensibilidad y sensibilidad complementaria para el sistema de primer orden, de segundo orden y el benchmark respectivamente.

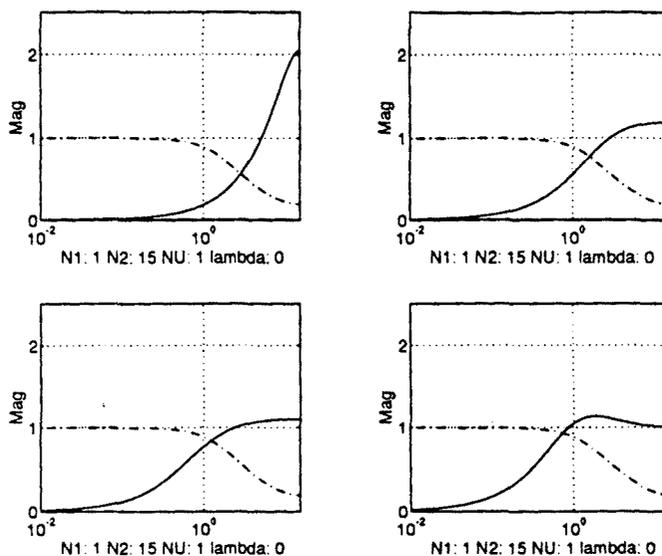


Figura 4.17: Funciones S y T para un sistema de 1<sup>er</sup> orden variando el polinomio T

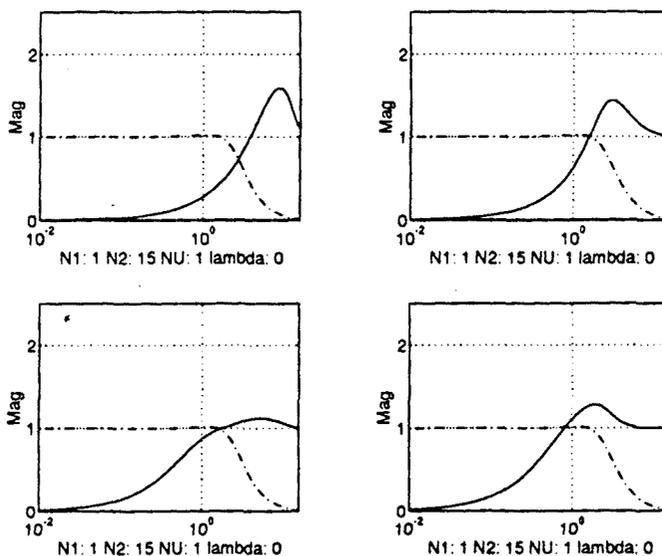


Figura 4.18: Funciones S y T para un sistema de 2<sup>o</sup> orden variando el polinomio T

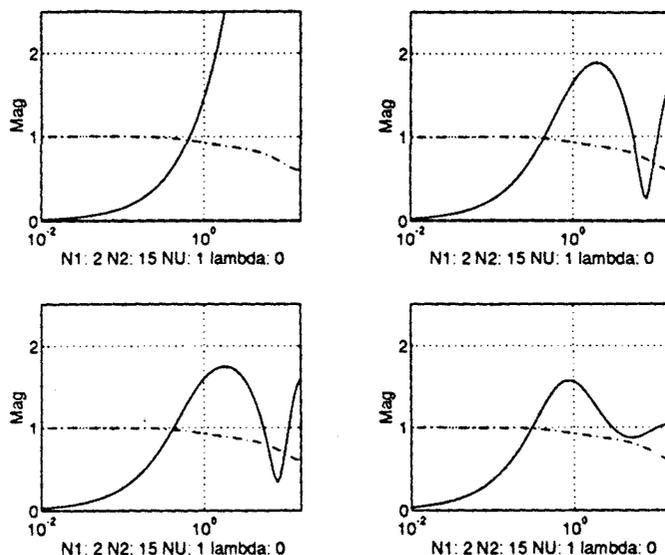


Figura 4.19: Funciones  $S$  y  $T$  para el sistema benchmark variando el polinomio  $T$  tivamente. En estas figuras, arriba a la izquierda aparecen las funciones  $S$  y  $T$  para  $T_1$ , arriba a la derecha para  $T_2$ , abajo a la izquierda para  $T_3$  y la última es para  $T_4$ .

Como hemos comentado, la sensibilidad complementaria no se ve afectada por el polinomio  $T$ . Esto tiene gran importancia, ya que sin reducir el rendimiento del sistema en lazo cerrado podemos aumentar la robustez. Observando las curvas de sensibilidad, vemos que es muy importante usar algún polinomio  $T$ , ya que en el caso que es igual a 1 el sistema amplifica mucho las perturbaciones y fácilmente puede llevar a la inestabilidad. Para  $T = A$  (es el caso DMC) el rechazo de perturbaciones es mucho mejor, pero no tanto como para los otros dos polinomios. Estos dos últimos polinomios consiguen un buen rechazo a perturbaciones, aunque para el sistema benchmark, el polinomio  $T_4$  es claramente el que más atenúa.

Así, el uso del polinomio  $T$  es fundamental, ya que no disminuye el ancho de banda y por otra parte si no se usa la amplificación de las perturbaciones puede tener resultados graves. Los polinomios que más atenúan las perturbaciones, y por

tanto proporcionan mayor robustez, son el  $T_3$  o el  $T_4$ .

### Conclusiones

En la Tabla 4.4 se presenta un resumen de como influyen los parámetros de sintonía cuando se aumentan sobre el rendimiento y la robustez. En la table no aparece el polinomio T ya que su influencia no se puede determinar en función de un aumento o disminución del polinomio.

Aumento	Rendimiento	Robustez
Horizonte de Coincidencia	↓	↑
Horizonte de Control	↑	↓
Factor de Peso	↓	↑

Tabla 4.4: Influencia de los parámetros de sintonía

## 4.4 Diseño Robusto de Controladores Predictivos

Las técnicas vistas en las secciones anteriores no conducen a una metodología de diseño de controladores predictivos robustos. Esta forma de diseño tiene la ventaja que si somos capaces de acotar la incertidumbre del modelo del proceso a controlar, podemos sintonizar el controlador de manera que funcione adecuadamente aunque varien los parámetros del proceso.

Esta técnica de diseño en general proporciona diseños muy conservadores, sobre todo si se pretende obtener rendimiento robusto. El método que se propone pretende que el controlador tenga estabilidad robusta y rendimiento nominal, de forma que se asegure la estabilidad para cualquier planta de la familia y un rendimiento

para el modelo nominal que se verá algo deteriorado cuando nos alejemos del modelo nominal.

El procedimiento de diseño de un controlador predictivo robusto se puede establecer en tres pasos:

1. Descripción de la familia de plantas para las que se han de cumplir los objetivos robustos. Para ello se escogerá una forma de representar la incertidumbre  $\Delta$  y se calculará una cota superior  $r$ .
2. Sintonía de los parámetros del controlador para que cumpla la condición de estabilidad robusta, que dependerá de la descripción de la incertidumbre que se haya usado. La forma de sintonizar el controlador será mediante el método de prueba y error pero apoyado en el estudio realizado en la sección anterior.
3. La condición de estabilidad nominal nos proporciona una caracterización frecuencial de la referencia  $W$  que se usará para filtrar las señales de referencia que entrel al sistema de manera que la trayectoria de referencia tenga un contenido frecuencial incluido en  $W$ .

Este procedimiento es iterativo y requiere de un gran número de cálculos, por lo que es necesario el uso de algún paquete de análisis y diseño como MATLAB que aporte un núcleo de funciones matemáticas básicas.

El método que se propone se va a desarrollar sobre el sistema benchmark (3.16) y el valor de los parámetros para el modelo nominal será:  $K = 1$ ,  $T_n = 1$ ,  $\omega_n = 15$  y un retardo de  $\tau = 0.2s$ . El sistema se ha muestreado con un periodo  $0.2s$ . Consideraremos que la incertidumbre asociada a estos parámetros viene dada por una variación máxima del 50% respecto del valor nominal. La descripción de esta incertidumbre será de tipo multiplicativo.

### 4.4.1 Cálculo de la incertidumbre

El objetivo de este primer punto es obtener una representación frecuencial de la incertidumbre bajo una forma multiplicativa. Es decir, expresar la familia  $\mathcal{F}$  de plantas generadas por las variaciones del 50% sobre los valores nominales de los parámetros en la forma expresada en (4.53), (4.54) y (4.55). Para poder especificar esta familia hemos de encontrar su correspondiente función cota  $r_m(\omega)$ .

Dado que el diseño del controlador predictivo se realiza a partir de modelos discretos, para cada modelo continuo resultante de la combinación de los posibles valores de los cuatro parámetros dentro de los considerados se ha de obtener su respectivo modelo discreto.

Cada una de estas discretizaciones será un miembro de la familia  $\mathcal{F}$ . Para cada discretización se calcula la incertidumbre multiplicativa asociada respecto al modelo nominal mediante la fórmula

$$\Delta_m = \left| \frac{A_0 B - A B_0}{A B_0} \right|$$

En la Fig. 4.20 podemos ver las curvas de incertidumbre para los miembros de la familia  $\mathcal{F}$  tomando tres valores de cada uno de los rangos de variación de los cuatro parámetros con incertidumbre.

El siguiente paso es encontrar la función cota  $r_m$  a partir de los diferentes  $\Delta_m(e^{j\omega})$  que hemos encontrado, es decir

$$|\Delta_m(e^{j\omega})| \leq r_m(\omega), \quad 0 \leq \omega \leq \pi \quad (4.59)$$

caracterizando así la familia  $\mathcal{F}$ .

Una vez que tenemos las gráficas de la incertidumbre el problema de encontrar la cota  $r_m$  resulta en un problema de diseño de filtros. Queremos que la ganancia de este filtro en las zonas de altas y bajas frecuencias verifique que

$$r_m(\approx 0) \geq 0.5 \quad (4.60)$$

$$r_m(\approx \infty) \geq 1.6 \quad (4.61)$$

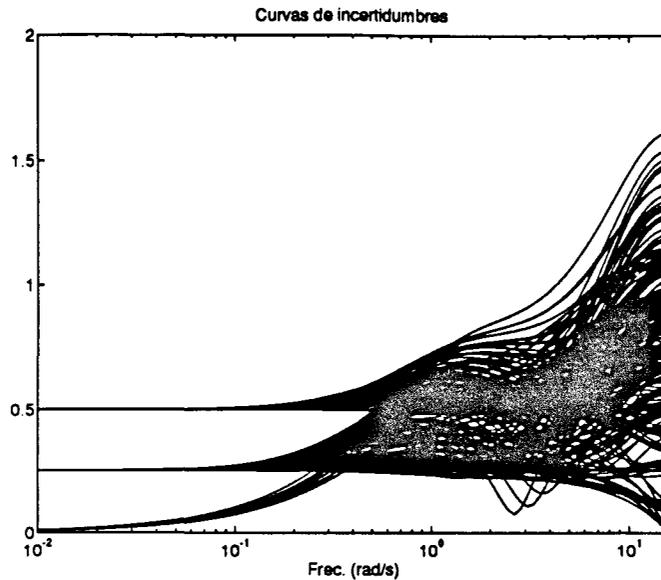


Figura 4.20: Curvas  $\Delta_m(e^{j\omega})$  para diferentes valores de  $K, T_n, \omega_n$  y  $\tau$

Hay muchos filtros que cumplen esta especificación, pero la idea es escoger un filtro sencillo que se ajuste lo más posible a las curvas de incertidumbre a altas frecuencias. A bajas frecuencias no es necesario que se ajuste mucho ya que es suficiente que  $1/r_m$  sea igual a 1 a estas frecuencias.

El filtro elegido que se ajusta a las condiciones de ganancia anteriores, es:

$$r_m(q^{-1}) = \frac{1.3q^{-1} - 0.5598q^{-2}}{1 - 0.0498q^{-1}} \quad (4.62)$$

el cual podemos ver representado una vez discretizado en la Fig. 4.21 junto con las curvas de incertidumbre.

Esta función  $r_m(\omega)$  nos servirá para establecer la función de peso a aplicar a la función de sensibilidad complementaria  $T_0$  a la hora de plantear la condición de estabilidad robusta.

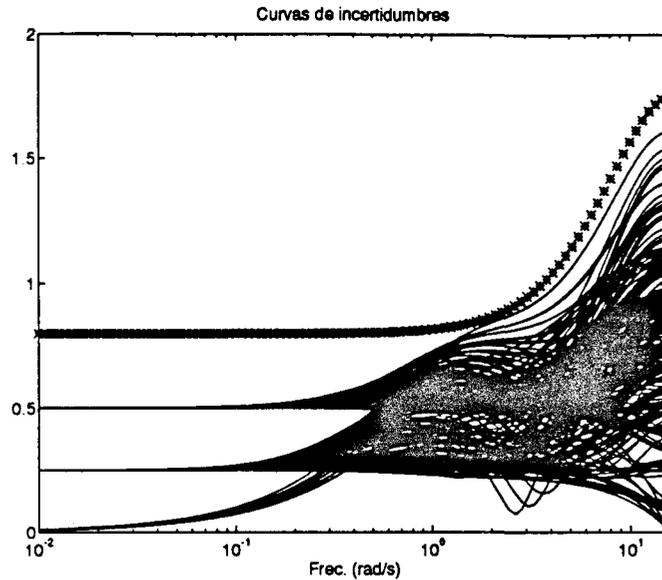


Figura 4.21: Curvas de incertidumbre  $\Delta_m$  y cota  $r_m$  (\*)

#### 4.4.2 Sintonía para estabilidad robusta

A partir de la información obtenida en el punto anterior, podemos plantear una especificación inicial para el controlador a sintonizar. Esta especificación hace referencia a la estabilidad robusta de sistema en lazo cerrado relativa al sistema nominal a través de una cota sobre la incertidumbre multiplicativa  $r_m(z)$  (4.62).

En términos del análisis robusto, el problema consiste en encontrar un conjunto de valores para  $N_1, N_2, N_u$  y  $\lambda$  de manera que se cumpla

$$\|T_0(e^{j\omega})r_m(e^{j\omega})\|_\infty < 1 \quad (4.63)$$

La forma de buscar los valores de los parámetros de sintonía es un proceso iterativo de prueba y error en el que a partir del estudio que se hizo sobre la influencia de estos parámetros en la forma de las funciones de sensibilidad y sensibilidad complementaria se va dando forma a la función de sensibilidad complementaria  $T_0$  hasta conseguir que se cumpla la especificación anterior.

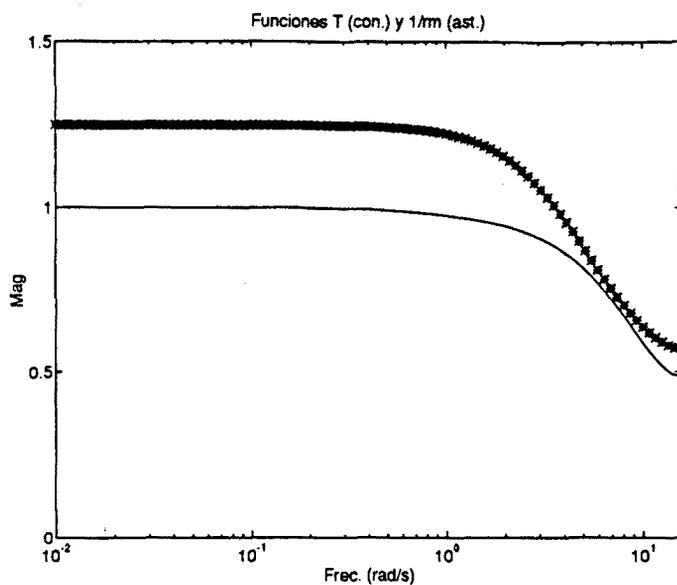


Figura 4.22: Función de sensibilidad complementaria para  $N_1 = 2$ ,  $N_2 = 20$ ,  $N_u = 1$  y  $\lambda = 1.5$  y función  $1/r_m$

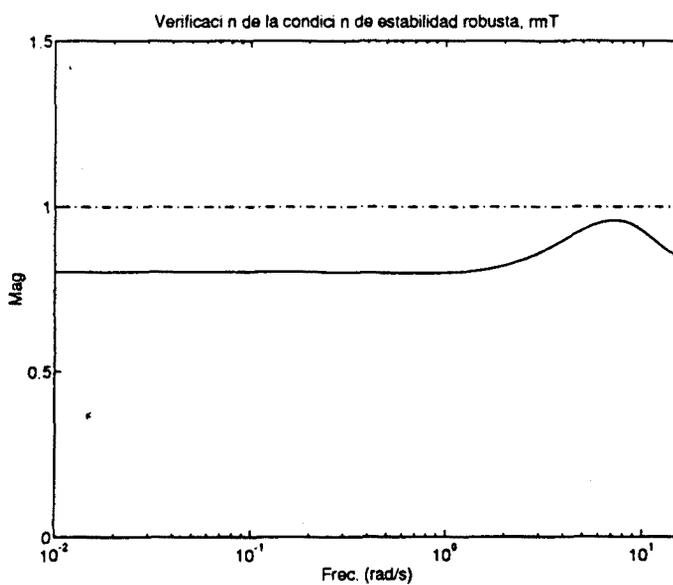


Figura 4.23: Verificación de la condición de análisis

Notar que el polinomio  $T$  no tiene influencia en la función de sensibilidad complementaria y, por tanto, no se usa en este punto. En la Fig. 4.22 se muestran la función de sensibilidad complementaria para los valores de los parámetros  $N_1 = 2$ ,  $N_2 = 20$ ,  $N_u = 1$  y  $\lambda = 1.5$  y se observa que siempre está por debajo de la función  $1/r_m$  y por tanto se cumplirá la condición de estabilidad robusta como se comprueba en la Fig. 4.23.

### 4.4.3 Caracterización frecuencial de la referencia

Junto con el requerimiento anterior hay que exigir un determinado rendimiento a través de una forma determinada de la función de sensibilidad. Recordemos la discusión llevada a cabo al principio del capítulo sobre el papel que juega la función de sensibilidad dentro del lazo de control donde deducíamos que contra más grande fuera el rango de (bajas) frecuencias en que podamos hacer  $S_0(e^{-j\omega}) \approx 0$ , mejor rendimiento conseguiremos. Entre otras características obtendremos:

- Aumento del rango de frecuencias en que tendremos una atenuación de las posibles perturbaciones presentes a la salida de la planta.
- Aumento del espectro de las señales de referencia para las cuales se puede conseguir  $e \approx 0$ .

Para darle forma a la función de sensibilidad el único parámetro que nos queda es el polinomio  $T$ . Mediante la elección de éste polinomio hemos de conseguir una buena atenuación de las perturbaciones y el espectro más grande posible de las señales de referencia. Según vimos en la sección anterior un polinomio que cumplía estas características es

$$T(q^{-1}) = A_0(q^{-1})(1 - 0.9q^{-1}) \quad (4.64)$$

El segundo requerimiento que vamos a exigir al sistema de control va a ser el de rendimiento nominal. Nos conformaremos con la condición de rendimiento

nominal, ya que la condición de rendimiento robusto suele producir controladores muy conservadores y, en general, con conseguir estabilidad robusta y un rendimiento nominal adecuado se consiguen buenos controladores. Así, el controlador tendrá que cumplir

$$\|W_S(e^{j\omega})S_0(e^{j\omega})\|_\infty < 1, \quad 0 \leq \omega \leq \pi \quad (4.65)$$

donde  $W_S$  se puede interpretar como la caracterización frecuencial de la clase de señales de referencia que vamos a aceptar. Esta caracterización frecuencial la vamos a especificar mediante un filtro de primer orden. Un filtro que cumple la especificación anterior es

$$W_S(q^{-1}) = \frac{0.1q^{-1}}{1.09q^{-1}} \quad (4.66)$$

En la Fig. 4.24 se muestra la función de sensibilidad para el polinomio  $T$  escogido y la función  $W_S^{-1}$  elegida de manera que siempre esté por encima de  $S_0$ . En la Fig. 4.25 se ve que efectivamente se verifica la condición de rendimiento nominal.

La función  $W_S$  que nos describe frecuencialmente las señales de referencia la podemos usar directamente como filtro de la referencia de manera que el espectro de la trayectoria de referencia no contenga componentes que no cumplan la condición de rendimiento nominal.

De esta forma hemos sintonizado todos los parámetros del controlador predictivo de forma sistemática y hemos conseguido que éste cumpla las especificaciones de estabilidad robusta y rendimiento nominal.

#### 4.4.4 Evaluación del diseño

Una vez encontrada una sintonía para el controlador predictivo vamos a pasar a evaluar su respuesta temporal. Los experimentos que se van a llevar a cabo consisten en estudiar las respuestas del sistema en los siguientes casos:

- Sistema nominal.

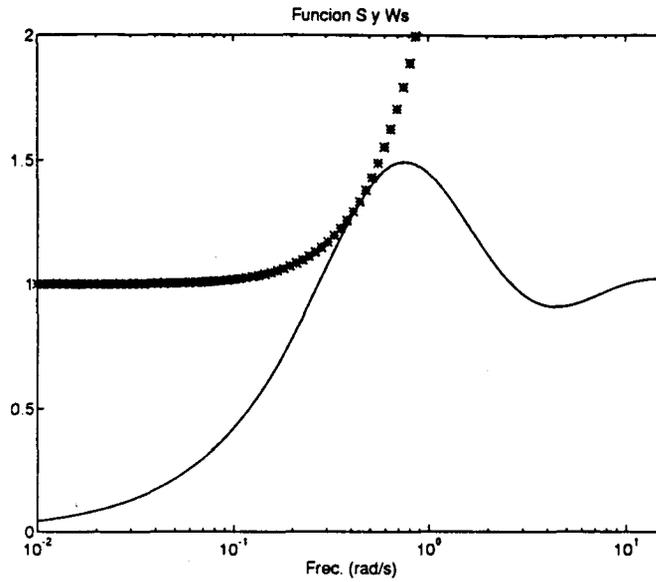


Figura 4.24: Función de sensibilidad para  $T(q^{-1}) = A_0(q^{-1})(1 - 0.9q^{-1})$  y función  $W_S^{-1}$

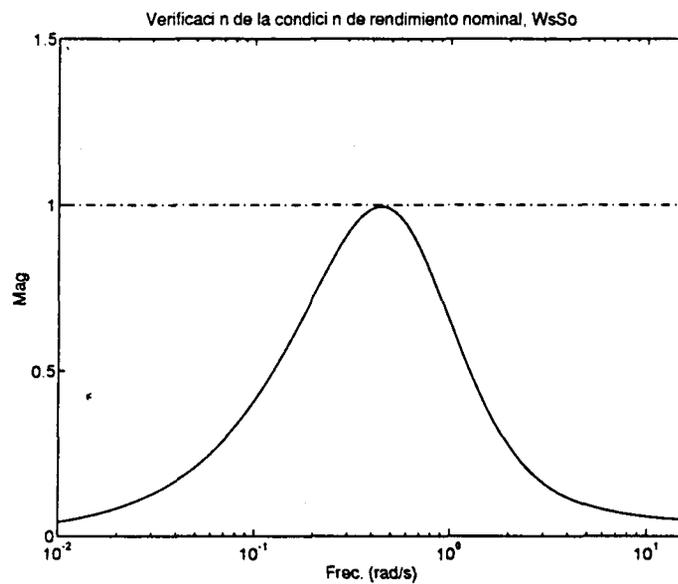


Figura 4.25: Verificación de la condición de análisis

- Sistema cuando los parámetros se han incrementado un 50%.
- Sistema cuando los parámetros se han disminuido un 50%.

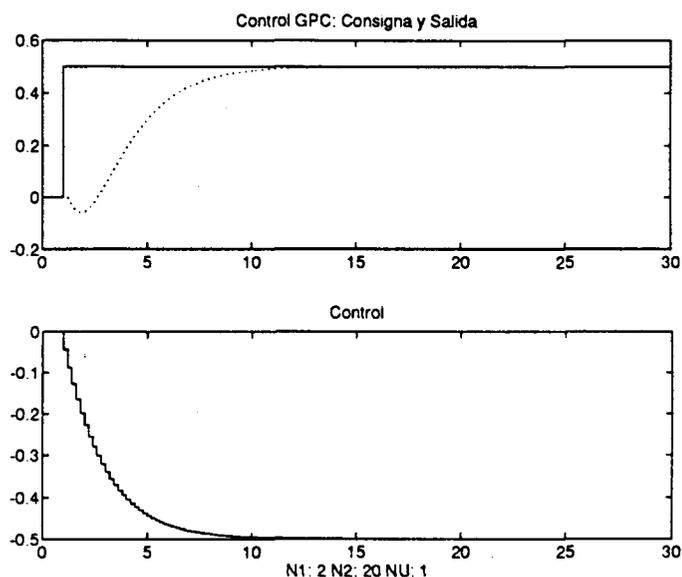


Figura 4.26: Respuesta del GPC sobre el sistema nominal

En la Fig. 4.26 vemos la respuesta del GPC para el sistema nominal. La respuesta es suave y sin grandes esfuerzos de control. El tiempo de respuesta es de menos de 10s. Para sistemas distintos del sistema nominal es obvio que el rendimiento será peor, aunque aceptable en cualquier caso. En las Fig. 4.27 y 4.28 vemos las respuestas del mismo GPC sobre dos sistemas cuando los valores de los parámetros, en un caso, se han aumentado un 50%, y en otro, se han disminuido un porcentaje igual. En ambos casos el tiempo de respuesta es algo mayor pero se puede considerar bueno y los esfuerzos de control son también pequeños. Así, el sistema de control funciona bien para los sistemas anteriores y tal como los hemos diseñado el controlador también funcionará bien para todos los sistemas reales tenidos en cuenta en el diseño.

Es obvio que para el sistema nominal se puede encontrar una sintonía que obtenga una mejor respuesta. En la Fig. 4.29 se muestra la respuesta de un GPC

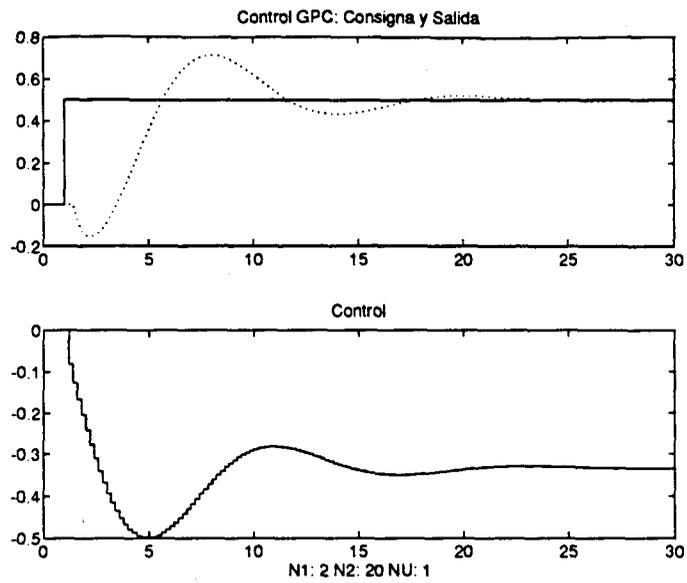


Figura 4.27: Respuesta del GPC habiendo aumentado un 50% el valor de los parámetros del sistema

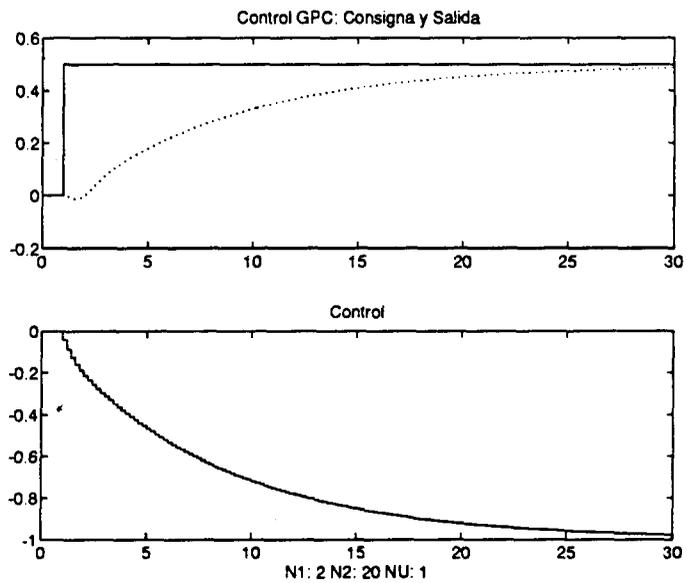


Figura 4.28: Igual que la gráfica anterior pero disminuyendo un 50%

sintonizado con los siguientes valores de los parámetros:  $N_1 = 3, N_2 = 15, N_u = 2, \lambda = 0, T = 1$  y sin ningún filtro en la referencia. La respuesta es muy rápida aunque los esfuerzos de control son también muy grandes. Sin embargo, si el sistema real es distinto del sistema nominal el rendimiento se deteriora rápidamente, llegando a inestabilizarse. En la Fig. 4.30 se muestra la respuesta de este mismo controlador sobre un sistema cuyos valores de los parámetros se han incrementado un 50% sobre el nominal. En este caso el sistema se hace inestable.

En esta sección se ha presentado un procedimiento de aplicación de las técnicas del control robusto a CPBM. La teoría del control robusto se caracteriza por una postura muy conservadora dado que el controlador se diseña para el peor de los casos. No obstante es un conservadurismo bien empleado, ya que la poca información que se tiene en cuenta se emplea de forma eficiente. En general, será más interesante tener un rendimiento aceptable para un conjunto de plantas alrededor de la planta nominal que un rendimiento óptimo para la planta nominal, pero que haya riesgo de inestabilidad cuando la planta real difiere de la nominal.

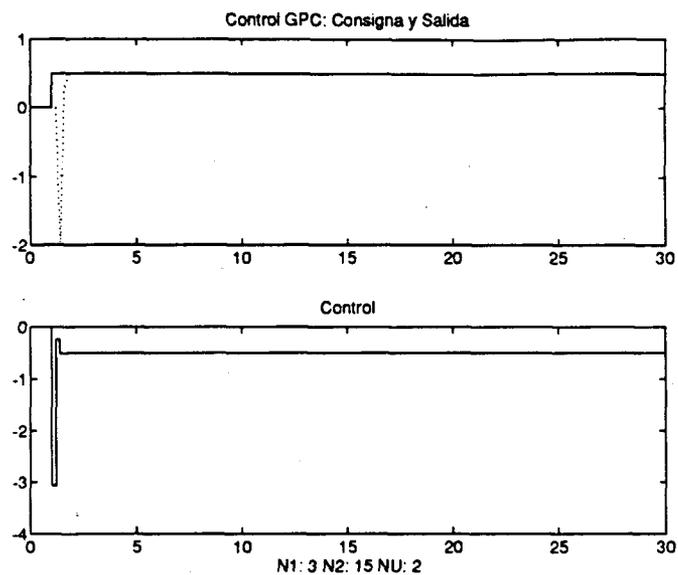


Figura 4.29: Respuesta de un GPC sobre el sistema nominal

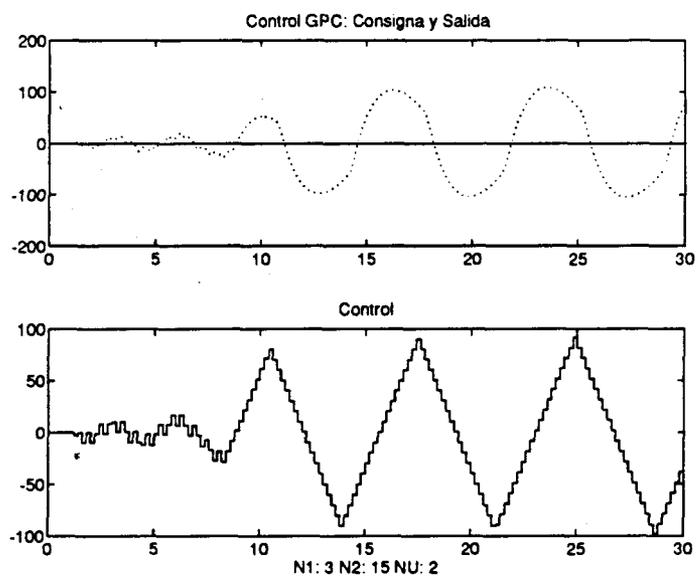


Figura 4.30: Respuesta de un GPC habiendo aumentado un 50% el valor de los parámetros del sistema

## Capítulo 5

# Robustez en CPBM y Modelo del Ruido

Otra forma de abordar el análisis y diseño robusto de controladores predictivos es a través del modelo del ruido. Cuando usamos modelos lineales simples para describir la dinámica de procesos más complejos, el modelo del ruido no trata de modelizar únicamente la dinámica entre la perturbación y la salida, sino que es una descripción que engloba todas las dinámicas que no se han decrito con el modelo de la planta.

Así, la elección del modelo del ruido es un aspecto fundamental para conseguir un buen rendimiento, p.e., suponer que la perturbación es constante o un movimiento browniano conduce automáticamente a que haya un integrador en la ley de control y, por tanto, el sistema no tenga error en estado estacionario. Su influencia sobre la robustez del sistema es el objeto de este capítulo. El capítulo comienza siguiendo la línea de estudio propuesta en [Clarke 92].

## 5.1 Predicción y Modelo del Ruido

Hay muchos modelos en los que se basan los controladores predictivos: respuesta impulso, respuesta escalón, variables de estado, función de transferencia, etc. Vamos a realizar este estudio sobre un modelo general de respuesta impulso infinita, ya que cualquier modelo lineal se puede transformar en uno de este tipo. Después particularizaremos estos resultados para controladores concretos como el GPC o DMC. Consideremos pues que la planta está descrita por modelo lineal general:

$$y(t) = M(q^{-1})u(t) + N(q^{-1})\xi(t) \quad (5.1)$$

donde

- $M(q^{-1})$  : Dinámica de la variable manipulada.
- $N(q^{-1})$  : Dinámica de la perturbación.
- $\xi(t)$  : Ruido blanco.

La predicción de la salida  $j$  instantes después basándonos en este modelo será:

$$y(t+j) = M(q^{-1})u(t+j) + N(q^{-1})\xi(t+j) \quad (5.2)$$

Dado que desconocemos como será el ruido  $\xi(t+j)$  en el futuro, descompondremos el término de ruido en una parte futura y otra pasada que si podremos estimarla. En forma polinomial quedará,

$$\begin{aligned} N(q^{-1}) &= 1 + n_1q^{-1} + \dots + n_{j-1}q^{-j+1} + q^{-j}(n_j + n_{j+1}q^{-1} + \dots) \\ &= N_j^*(q^{-1}) + q^{-j}N_j(q^{-1}) \end{aligned} \quad (5.3)$$

Así, la salida futura se puede escribir como

$$y(t+j) = M(q^{-1})u(t+j) + N_j^*(q^{-1})\xi(t+j) + N_j(q^{-1})\xi(t) \quad (5.4)$$

El primer término de la perturbación está compuesto únicamente de componentes futuros del ruido, mientras el segundo término incluye  $\xi(t)$  que se puede estimar a partir de datos pasados mediante:

$$\xi(t) = \frac{y(t) - M(q^{-1})u(t)}{N(q^{-1})} \quad (5.5)$$

La predicción de varianza mínima la obtenemos haciendo la suposición que la mejor predicción de los términos de ruido futuro es que son nulos. Por tanto,

$$\hat{y}(t+j) = M(q^{-1})u(t+j) + \frac{N_j(q^{-1})}{N(q^{-1})}y(t) - \frac{M(q^{-1})N_j(q^{-1})}{N(q^{-1})}u(t) \quad (5.6)$$

A partir de las definiciones de  $N_j$  y  $N_j^*$  y dividiendo la expresión (5.3) por  $N$  obtenemos

$$1 = \frac{N_j^*(q^{-1})}{N(q^{-1})} + q^{-j} \frac{N_j(q^{-1})}{N(q^{-1})} \quad (5.7)$$

Y la predicción de varianza mínima queda definitivamente

$$\begin{aligned} \hat{y}(t+j) &= M(q^{-1})(u(t+j) - \frac{N_j(q^{-1})}{N(q^{-1})}u(t)) + \frac{N_j(q^{-1})}{N(q^{-1})}y(t) \\ \hat{y}(t+j) &= M(q^{-1})\frac{N_j^*(q^{-1})}{N(q^{-1})}u(t+j) + \frac{N_j(q^{-1})}{N(q^{-1})}y(t) \end{aligned} \quad (5.8)$$

Ahora bien, como el segundo sumando de (5.7) está retardado  $j$  posiciones, comparando los coeficientes de las potencias de  $q^{-i}$  de cada sumando, implica que los primeros  $j$  términos del polinomio  $N_j^*/N$  deben ser

$$1 + 0q^{-1} + \dots + 0q^{-j+1} \quad (5.9)$$

para que se pueda cumplir la igualdad. Esto significa que la componente de la predicción de la salida que depende de valores futuros de control (la respuesta forzada) no se ve afectada por la elección del modelo del ruido  $N(q^{-1})$ . El modelo del ruido sólo afecta a la predicción de la respuesta libre de la planta.

### Predicción en el DMC

Vamos a aplicar la ecuación de predicción anterior al caso del DMC el cual usa un modelo de ruido tipo movimiento browniano

$$y(t) = G(q^{-1})\Delta u(t) + \frac{\xi(t)}{\Delta} \quad (5.10)$$

para el cual

$$\begin{aligned} M(q^{-1}) &= G(q^{-1})\Delta \\ N(q^{-1}) &= \frac{1}{\Delta} \end{aligned}$$

donde  $G(q^{-1})$  es la respuesta escalón del proceso. Expandiendo el término de ruido en forma polinomial obtenemos:

$$\begin{aligned} N_j(q^{-1}) &= 1 + q^{-1} + q^{-2} + \dots \\ \frac{N_j^*(q^{-1})}{N(q^{-1})} &= \frac{1 + q^{-1} + \dots + q^{-j+1}}{1 + q^{-1} + q^{-2} + \dots} = 1 - q^{-j} \end{aligned}$$

Por tanto, la predicción óptima en el DMC es

$$\hat{y}(t+j) = G(q^{-1})\Delta u(t+j) - G(q^{-1})\Delta u(t) + y(t) \quad (5.11)$$

### Predicción en el GPC

El modelo del GPC es un CARIMA (4.37) y para este modelo tenemos

$$\begin{aligned} M(q^{-1}) &= \frac{B(q^{-1})}{A(q^{-1})} = G(q^{-1})\Delta \\ N(q^{-1}) &= \frac{T(q^{-1})}{A(q^{-1})\Delta} \end{aligned}$$

Recordemos que la parte forzada de la predicción es independiente del modelo del ruido  $N(q^{-1})$ , ésta sólo dependerá de  $A/B$  que es exactamente el mismo modelo de la planta que se usa en el DMC. La predicción de la respuesta libre si que dependerá del modelo del ruido y en particular del polinomio  $T(q^{-1})$ . Sin embargo, si se elige  $T = A$  tendremos que el modelo del ruido también es exactamente igual al del DMC. Queda patente que el DMC es un caso particular del GPC.

La división del término de ruido nos conduce como ya vimos a la solución de una ecuación diofántica

$$N(q^{-1}) = \frac{T(q^{-1})}{A(q^{-1})\Delta} = E_j(q^{-1}) + q^{-j} \frac{F_j(q^{-1})}{A(q^{-1})\Delta} \quad (5.12)$$

La predicción óptima en el GPC resulta

$$\hat{y}(t+j) = \frac{E_j(q^{-1})B(q^{-1})}{T(q^{-1})}\Delta u(t+j) + \frac{F_j(q^{-1})}{T(q^{-1})}y(t) \quad (5.13)$$

## 5.2 Error de Predicción

El modelo nominal que se usa para hacer las predicciones normalmente no será exacto, siempre tendrá una cierta dosis de incertidumbre. Vamos a estudiar como influye el modelo del ruido en el error de predicción que cometemos al ser el modelo inexacto y haber perturbaciones. Supongamos que la representación exacta de la planta viene dada por

$$y(t) = M_0(q^{-1})u(t) + x(t) \quad (5.14)$$

donde  $x(t)$  es un término general de perturbación. La predicción exacta usando este modelo es

$$y(t+j) = M_0(q^{-1})u(t+j) + x(t+j) \quad (5.15)$$

Por tanto, el error de predicción que cometeremos será la diferencia entre la predicción exacta y la predicción calculada con (5.8)

$$y(t+j) - \hat{y}(t+j) = (M_0(q^{-1}) - \frac{N_j^*(q^{-1})}{N(q^{-1})}M(q^{-1}))u(t+j) + x(t+j) - \frac{N_j(q^{-1})}{N(q^{-1})}y(t) \quad (5.16)$$

Ahora sustituyendo  $y(t)$  por (5.14) y

$$\frac{N_j(q^{-1})}{N(q^{-1})} = q^j \left(1 - \frac{N_j^*(q^{-1})}{N(q^{-1})}\right) \quad (5.17)$$

operando se puede deducir la siguiente expresión

$$y(t+j) - \hat{y}(t+j) = \frac{N_j^*(q^{-1})}{N(q^{-1})} [(M_0(q^{-1}) - M(q^{-1}))u(t+j) + x(t+j)] \quad (5.18)$$

Así, los errores de predicción se verán afectados por los errores de modelado y las perturbaciones a través del filtro

$$L(q^{-1}) = \frac{N_j^*(q^{-1})}{N(q^{-1})} \quad (5.19)$$

Si somos capaces de conseguir un filtro  $L$  que atenue las frecuencias a las que se producen los errores de modelado y las perturbaciones conseguiremos un sistema más robusto.

### Error de predicción en el DMC

El error de predicción que cometemos con el DMC tendrá la expresión

$$y(t+j) - \hat{y}(t+j) = \frac{N_j^*(q^{-1})}{\Delta} [(M_0(q^{-1}) - M(q^{-1}))u(t+j) + x(t+j)] \quad (5.20)$$

Debido al término  $\Delta$ , el error de predicción será cero para perturbaciones y errores de modelado constantes. Cuando no sean constantes, estos estarán filtrados a través de

$$L(q^{-1}) = \frac{N_j^*(q^{-1})}{\Delta} = \frac{1 + q^{-1} + \dots + q^{-j+1}}{1 - q^{-1}} = 1 - q^{-j} \quad (5.21)$$

Observar que el filtro cambia con los valores de  $j$  y por tanto, la selección del horizonte de coincidencia  $N_1 - N_2$  influirá sobre el error de predicción. De hecho, en el DMC, éstos son los únicos parámetros de diseño que influyen.

En la Fig. 5.1 podemos ver la respuesta en frecuencia de este filtro para un rango de  $j$  entre  $N_1 = 2$  y  $N_2 = 15$  y para el caso del ejemplo benchmark (3.16). No están dibujadas las respuestas para todos los valores de  $j$ , únicamente para los valores inicial y final ya que para el resto de valores las respuestas están contenidas entre estos valores límite.

Podemos observar que los errores de baja frecuencia se filtran, pero cuando  $j$  se va incrementando, el rango de frecuencias que atenúa se va reduciendo. Así, si podemos estimar el contenido frecuencial de la perturbación  $x(t)$ , podremos escoger el horizonte de coincidencia más adecuado para que no se vea reflejado en la predicción. En cuanto a los errores de modelado el objetivo será evitar los cambios rápidos de la señal de control, ya que el filtro  $L$  amplifica las altas frecuencias y provocará grandes errores de predicción. Existen diversos métodos para limitar las altas frecuencias de la señal de control que dan robustez al sistemas y que las veremos más adelante.

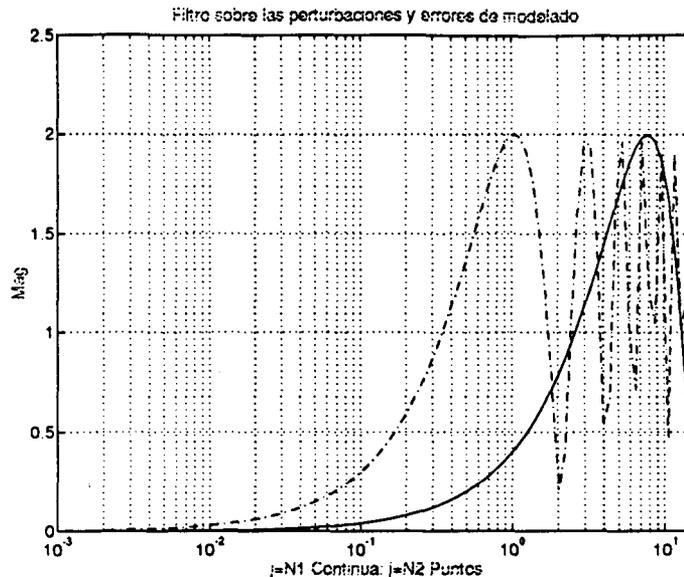


Figura 5.1: Respuesta en frecuencia del filtro  $L$  en el DMC para  $N_1 = 2$  y  $N_2 = 15$ .

### 5.3 Error de predicción en el GPC: El polinomio $T$

El estudio del error de predicción en el GPC es más interesante, ya que dispone de otro parámetro de diseño del filtro  $L$ , el polinomio  $T$ . Según vimos en (5.12) la solución de la ecuación diofántica para el GPC nos daba que  $N_j^* = E_j$  y la expresión del filtro  $L$  queda

$$L(q^{-1}) = \frac{E_j(q^{-1})A(q^{-1})\Delta}{T(q^{-1})} \quad (5.22)$$

Observar en esta expresión que el único término de diseño es  $T$ . Una selección apropiada de este polinomio puede reducir los errores de predicción y proporcionar al sistema robustez. Sin embargo, la selección no es sencilla, ya que  $E_j$  también depende de  $T$  de una forma complicada.

Al igual que en el DMC, el término  $\Delta$  del filtro eliminará perturbaciones o errores de modelado constantes. También, el horizonte de coincidencia tendrá la misma influencia que en el DMC. Observar por último que si elegimos  $T = A$  el filtro

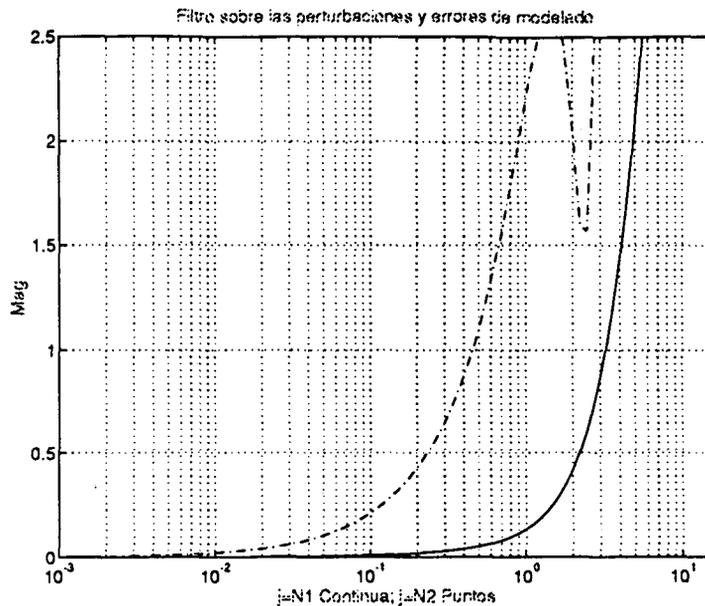


Figura 5.2: Respuesta en frecuencia del filtro  $L$  en el GPC para  $T = 1$ .

resultante es el mismo que el del DMC, lo cual es lógico ya que el DMC es un caso particular del GPC.

En las Fig. 5.2 y 5.3 podemos ver las respuestas en frecuencia del filtro  $L$  para  $T = 1$  y  $T = 1 - 0.9q^{-1}$  respectivamente. Observamos que para  $T = 1$  el sistema será menos robusto, ya que amplifica mucho las altas frecuencias y cualquier perturbación o error de modelado a estas frecuencias será magnificado en el error de predicción llevando al sistema posiblemente a la inestabilidad. El filtro  $L$  que resulta para  $T = 1 - 0.9q^{-1}$  tiene características más robustas que el anterior y que el del DMC ya que amplifica menos las altas frecuencias. Para ver el efecto del polinomio  $T$  en el error de predicción vamos a hacer dos tipos de experimentos: uno en que introduciremos un error de modelado y otro en que tendremos una perturbación. Los experimentos se han llevado a cabo sobre el ejemplo benchmark (3.16).

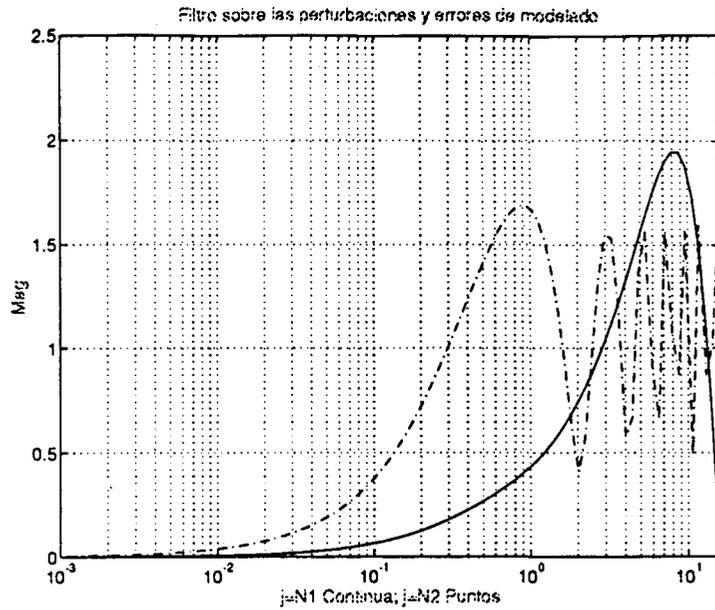


Figura 5.3: Respuesta en frecuencia del filtro  $L$  en el GPC para  $T = 1 - 0.9q^{-1}$ .

### Errores de modelado

En estos primeros experimentos se ha cambiado el modelo de la planta fijando la constante del polo simple a  $T_n = 2$  y la frecuencia natural a  $\omega_0 = 10$ , y se ha mantenido el mismo modelo nominal con el que se realizan las predicciones. El controlador GPC se ha sintonizado con los siguientes valores:  $N_1 = 2$ ,  $N_2 = 15$ ,  $N_u = 2$  y no se han usado ni factor de peso, ni filtro en la señal de consigna.

Se han realizado tres experimentos con distintos valores del polinomio  $T$ .

En la Fig. 5.4 vemos la respuesta del GPC con  $T = 1$ . El sistema es inestable debido a los errores de modelado. La respuesta del GPC con  $T = 1 - 0.9q^{-1}$  (Fig. 5.5) es mucho mejor debido a las mejores características del filtro  $L$  que no amplifica los errores de modelado. Esta respuesta también es mejor que para  $T = A$  (DMC) que aunque estabiliza el sistema es algo más oscilatoria (Fig. 5.6). En la Fig. 5.7 se muestra el espectro de densidad de potencia del error de modelado para este último

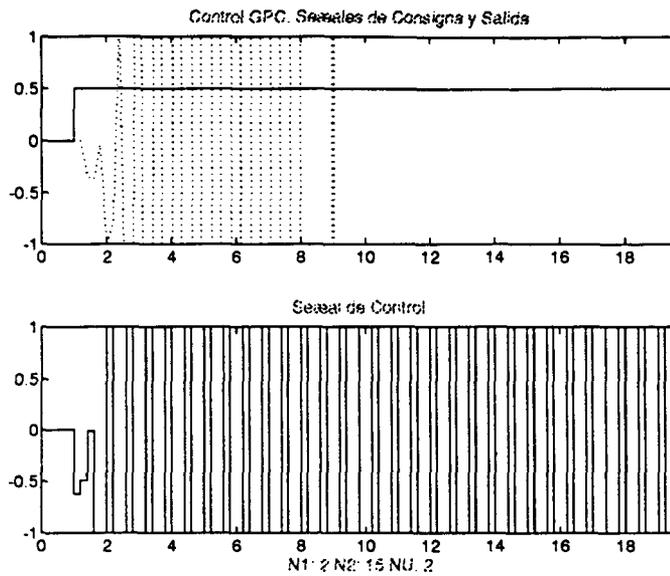


Figura 5.4: Respuesta del GPC con  $T = 1$ .

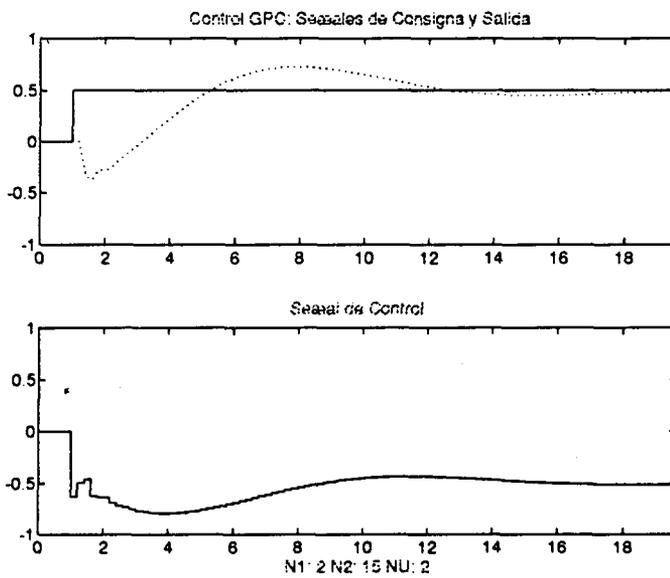


Figura 5.5: Respuesta del GPC con  $T = 1 - 0.9q^{-1}$ .

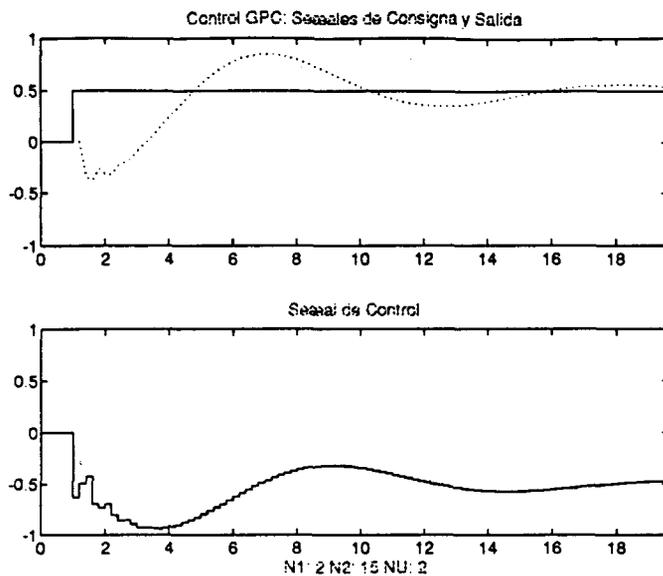


Figura 5.6: Respuesta del GPC con  $T = A$  (DMC).

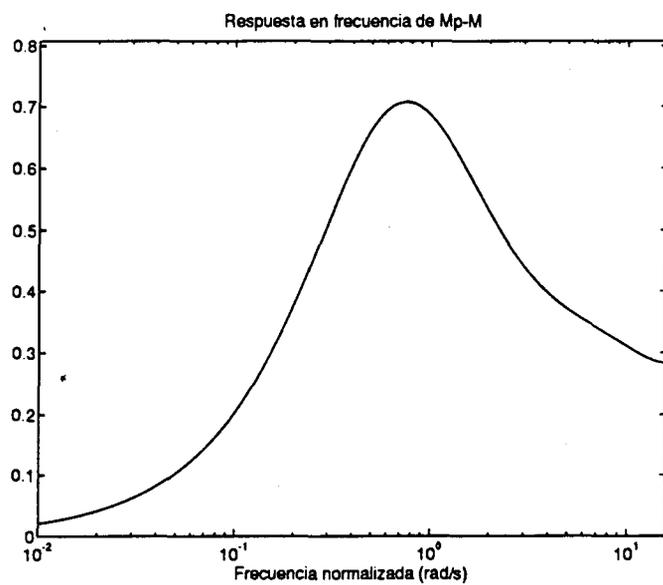


Figura 5.7: Respuesta en frecuencia del error de modelado  $M_p - M$ .

caso. Este espectro se ha calculado aplicando la transformada discreta de Fourier a la secuencia de errores producida al hacer la diferencia entre la salida de la planta real y la salida del modelo nominal.

### Perturbaciones

Los siguientes experimentos que se han llevado a cabo son similares a los anteriores. En estos no hay errores de modelado, pero se ha añadido un ruido coloreado a la salida,

$$r(t) = \frac{.6}{1 - 0.4q^{-1}}\xi(t) \quad (5.23)$$

donde  $\xi(t)$  es un ruido blanco de media cero y varianza 0.3. El espectro de este ruido se ha calculado mediante la transformada discreta de Fourier y se puede ver en la Fig. 5.8.

En estos experimentos se han usado los mismos parámetros de sintonía que antes y se han realizado tres simulaciones con los tres polinomios  $T$  usados, también, anteriormente. En la Fig. 5.9 vemos la respuesta en lazo cerrado del GPC con  $T = 1$ . Al igual que antes los errores debidos al ruido son amplificados y la respuesta del sistema es muy mala.

En la Fig. 5.11 vemos la respuesta de un DMC ( $T = A$ ) que es relativamente buena, ya que no amplifica apenas el ruido, pero peor que la de un GPC con  $T = 1 - 0.9q^{-1}$  que consigue una menor varianza de la salida ya que éste filtra mejor el ruido y, por tanto, afecta menos a la predicción y consecuentemente a la señal de control que en este caso es prácticamente constante.

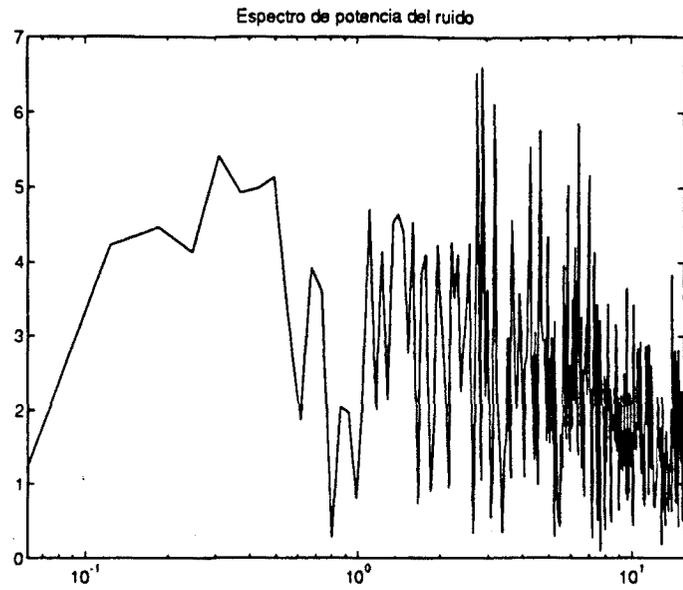


Figura 5.8: Espectro de densidad de potencia del ruido aditivo.

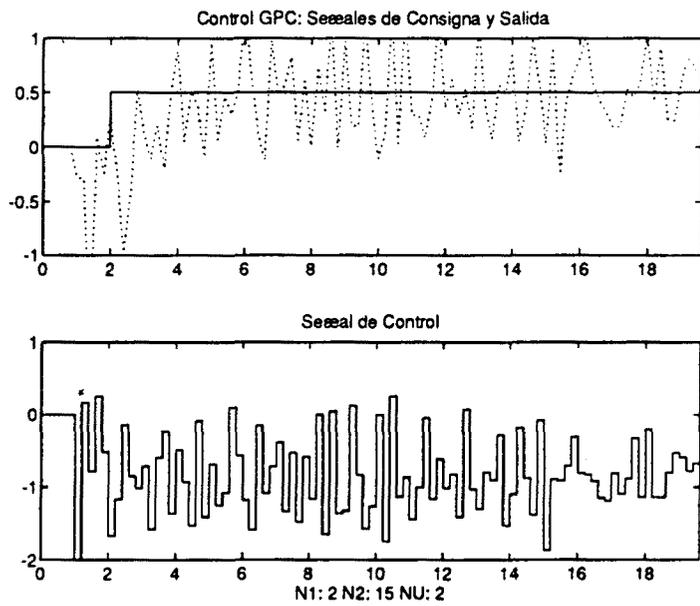


Figura 5.9: Respuesta del GPC con  $T = 1$ .

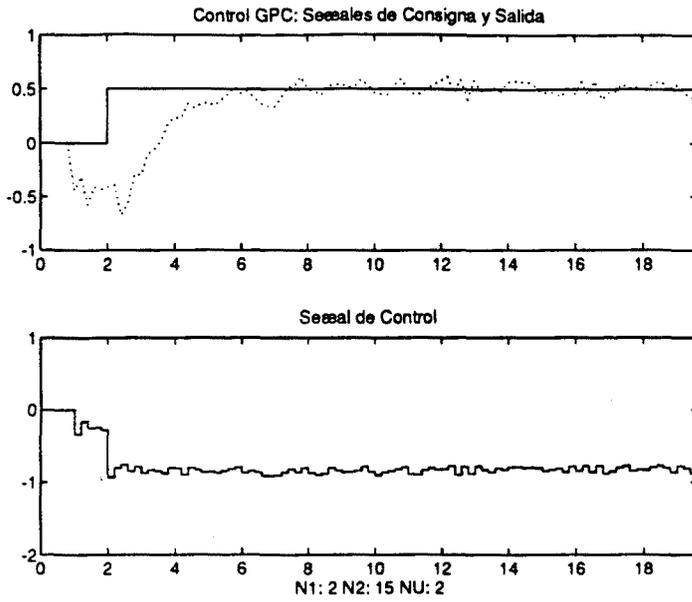


Figura 5.10: Respuesta del GPC con  $T = 1 - 0.9q^{-1}$ .

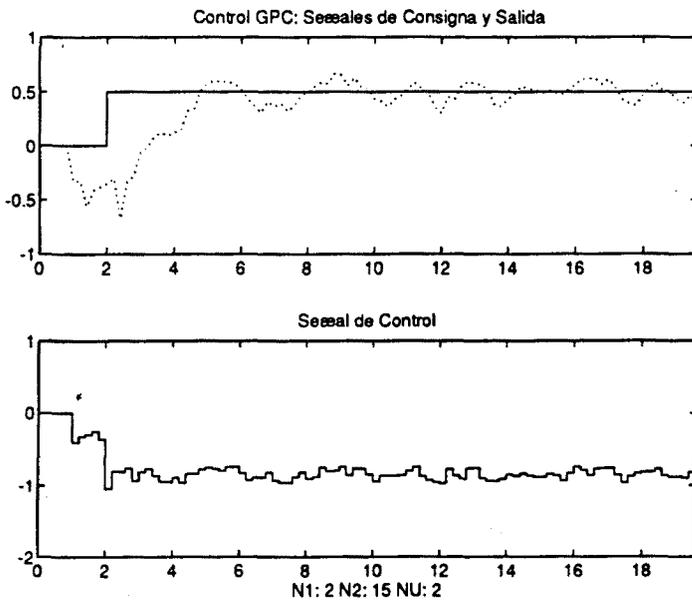


Figura 5.11: Respuesta del GPC con  $T = A$  (DMC).

## 5.4 Métodos para Incrementar la Robustez

A veces no es posible obtener el grado de robustez necesario únicamente sintonizando el polinomio  $T$  y el horizonte de coincidencia. Una estrategia clásica para aumentar la robustez del sistema de control es desintonizar el lazo hasta llegar a un compromiso entre la reducción del rendimiento y el aumento de la robustez.

Vamos a estudiar esta estrategia bajo la óptica de la teoría que hemos visto en las dos primeras secciones. La influencia de los errores de modelado y las perturbaciones venía descrita por la siguiente ecuación,

$$y(t+j) - \hat{y}(t+j) = L(q^{-1})[(M_0(q^{-1}) - M(q^{-1}))u(t+j) + x(t+j)] \quad (5.24)$$

El filtro  $L$  se puede diseñar para que no amplifique las altas frecuencias, pero es imposible conseguir que las atenue completamente. Así, las perturbaciones de alta frecuencia se verán reflejadas en el error de predicción inevitablemente.

En el caso de errores de modelado, dependiendo del tipo de incertidumbre que tengamos puede ser que  $M_e = M_0 - M$  tenga una característica pasa-altas, amplificando el error de predicción y en este caso tendremos problemas de robustez. La solución obvia es limitar el contenido frecuencial de la señal de control eliminando las altas frecuencias de manera que no se amplifiquen. El resultado es una reducción del ancho de banda del sistema y, por tanto, del rendimiento a costa de aumentar la robustez del sistema de control.

Esta limitación del contenido frecuencial de la señal de control se puede realizar de varias formas. Se presentarán tres formas: aumentar el factor de peso, limitar los incrementos de la señal de control y filtrar la señal de control.

Antes de nada conviene hacer un estudio de como influyen la incertidumbre de los distintos parámetros del modelo (3.16) en el error de modelado  $M_e$ . Recordar que los parámetros en los que podíamos tener incertidumbre son la ganancia  $K$ , la constante de tiempo del polo simple  $T_n$ , la frecuencia natural  $\omega_0$  y el retardo  $\tau$ . Los

valores de estos parámetros para el modelo nominal son  $K = 1, T_n = 1, \omega_0 = 15$  y  $\tau = 0.2$ .

En la Fig. 5.12 vemos la respuesta en frecuencia de  $M_e$  para varios valores de la ganancia que podría tener el modelo real. Observamos que a bajas frecuencias la magnitud de  $M_e$  es igual a la diferencia de la ganancia del modelo real menos la ganancia del modelo nominal, y a altas frecuencias atenua. Este es el resultado que podíamos esperar.

Para eliminar la influencia de este error de modelado de nada sirve limitar el contenido frecuencial de la señal de control a altas frecuencia, ya que este es un error de modelado de bajas frecuencias. En principio ha de ser posible eliminar su influencia diseñando el filtro  $L$ .

En las Fig. 5.13 y 5.14 vemos que para incertidumbres en la constante de tiempo y la frecuencia natural el error de modelado presenta una característica pasa-bandas. En ningún caso llega a amplificar, pero podemos observar que las frecuencias a las que menos atenua dependen del valor de los parámetros. A mayor frecuencia natural y menor constante de tiempo vemos que el diagrama se desplaza hacia las altas frecuencias. Por tanto, la efectividad de la limitación frecuencial de la señal de control dependerá de la incertidumbre. Mediante el diseño del filtro  $L$  y esta limitación frecuencial de la señal de control podremos conseguir que el error de modelado no influya en el error de predicción.

Por último, en la Fig. 5.15 vemos la respuesta en frecuencia de  $M_e$  para varios valores del retardo. Observamos que para valores mayores del retardo, más amplifica el error de modelado y este se da a altas frecuencias. Así, para este tipo de incertidumbre la limitación frecuencial de la señal de control es la única solución para eliminar la influencia del error de modelado.

El estudio de los tres métodos de limitar las altas frecuencias de la señal de control lo vamos a hacer sobre un modelo del proceso que tiene los siguientes parámetros:  $K = 2., T_n = 0.2$  y  $\tau = 0.6$ . El controlador utilizado es un GPC con

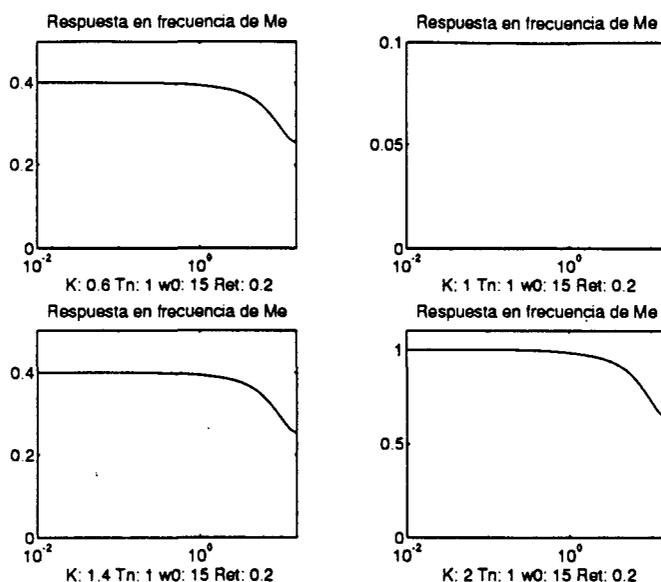


Figura 5.12: Respuesta en frecuencia de  $M_e$  para varios valores de la ganancia.

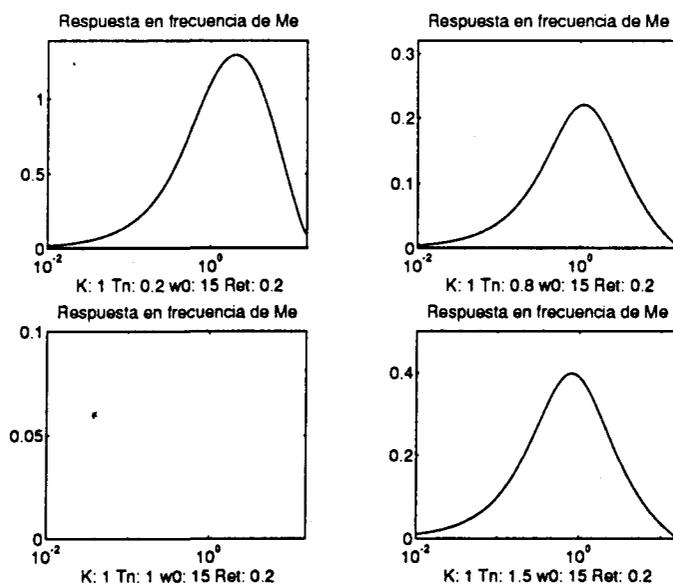


Figura 5.13: Respuesta en frecuencia de  $M_e$  para varios valores de la constante de tiempo del polo simple.

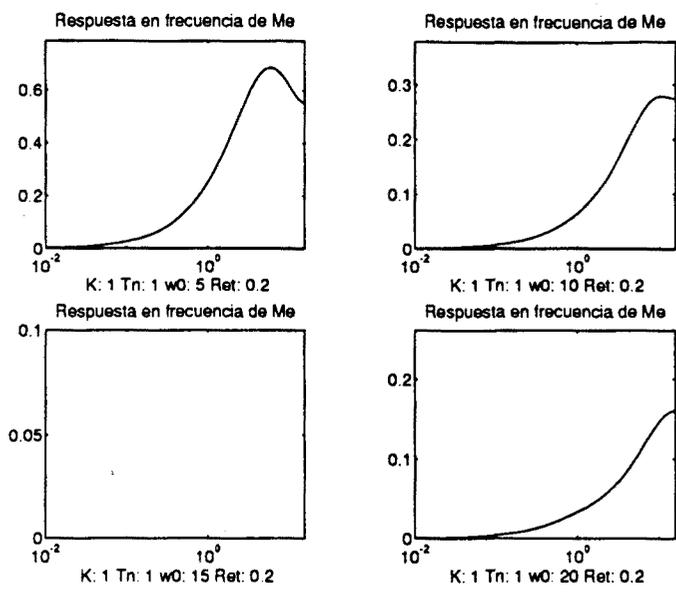


Figura 5.14: Respuesta en frecuencia de  $M_e$  para varios valores de la frecuencia natural.

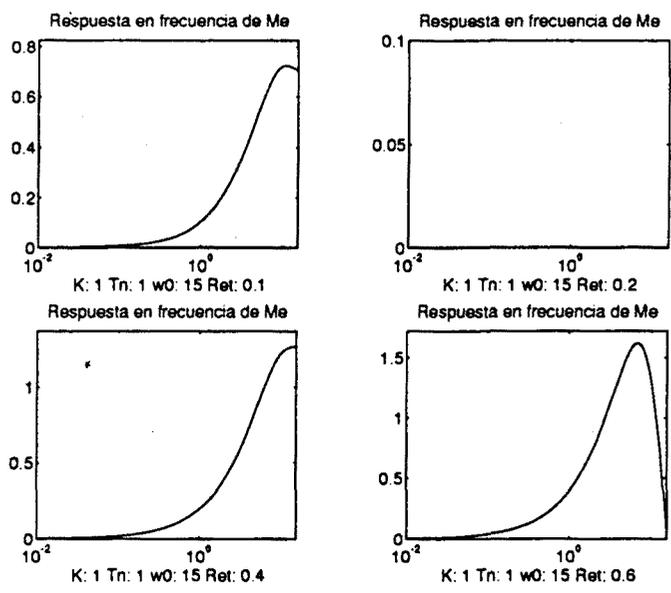


Figura 5.15: Respuesta en frecuencia de  $M_e$  para varios valores de retardo.

$N_1 = 2$ ,  $N_2 = 15$ ,  $N_u = 1$  y  $T = 1 - 0.9q^{-1}$ . La respuesta de este sistema la podemos ver en la Fig. 5.16 y observamos que es inestable.

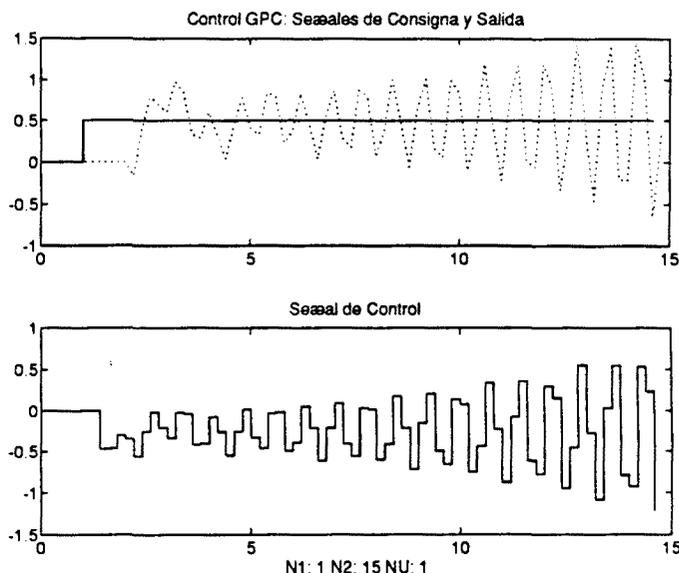


Figura 5.16: Respuesta del sistema con un controlador GPC.

#### 5.4.1 Aumentar el factor de peso $\lambda$

Ya vimos en el capítulo anterior que ésta es una forma de aumentar la robustez. Desde la perspectiva en que estamos analizando el problema en este capítulo podemos llegar a la misma conclusión. El factor de peso hace que los incrementos de control sean más pequeños, por tanto las variaciones en la señal de control serán más suaves. Contra más grande sea el factor de peso, más suave será la señal de control, con lo cual estaremos eliminando más efectivamente altas frecuencias. En la Fig. 5.17 vemos la respuesta de un GPC con factor de peso 10 y el resto de los parámetros igual que en el experimento anterior.

En este caso el sistema es estable. El método para buscar el factor de peso adecuado es de prueba y error, ya que a priori no existe una relación directa entre éste

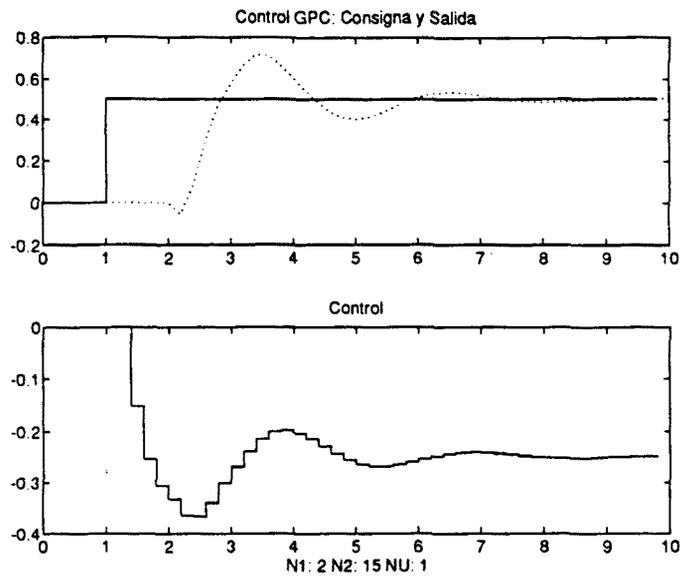


Figura 5.17: Respuesta de un GPC con factor de peso 10.

y las frecuencias de la señal de control que elimina. En consecuencia, el factor de peso nos proporciona robustez como hemos visto en este experimento que estabiliza el sistema, pero no tenemos un método para su elección.

### 5.4.2 Limitar los incrementos de la señal de control $\Delta u$

La segunda forma de eliminar altas frecuencias de la señal de control es limitar los incrementos de ésta. Al limitar los incrementos estamos forzando a que la señal de control varíe más lentamente y, por tanto, tenga una forma más suave. Aparentemente, deberíamos estar eliminando altas frecuencias de la señal de control.

Un problema que se plantea es que estamos introduciendo una no linealidad en el controlador (ya que estamos haciendo una minimización con restricciones) que dificulta su análisis. Esta no linealidad puede provocar que el sistema entre en un ciclo límite y que la señal de control cambie en cada instante de muestreo entre el

límite máximo y el mínimo con lo que tendríamos una señal de alta frecuencia.

Así, limitar los incrementos de la señal de control no nos garantiza la eliminación de un cierto rango de altas frecuencias, aunque si confiere cierta robustez al sistema de control. Se ha realizado un experimento similar a los anteriores pero limitando los incrementos a 0.05 y el resultado se puede ver en la Fig. 5.18.

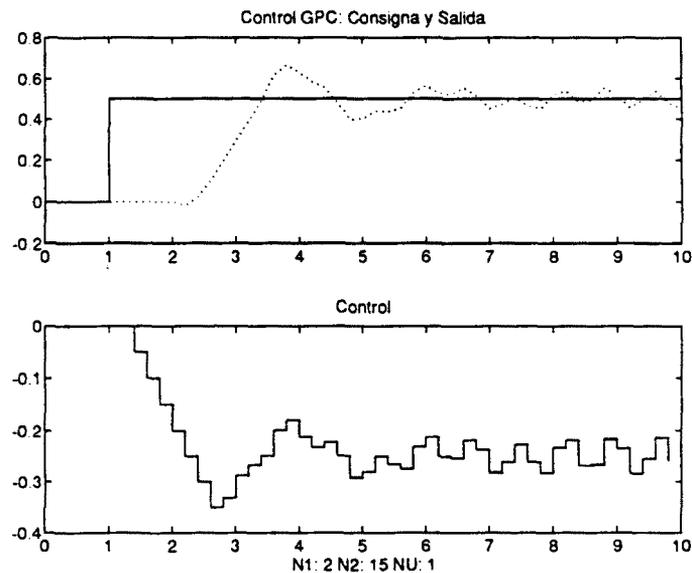


Figura 5.18: Respuesta de un GPC limitando los incrementos de control a 0.05.

Podemos observar que el sistema ya no se inestabiliza, pero cuando llega a la consigna entra en un ciclo límite. Por tanto, este método de ganar robustez no es bueno, ya que depende de la planta que se esté controlando el que vaya bien o no.

### 5.4.3 Filtrar la señal de control

Este último método consiste en eliminar mediante un filtro las componentes frecuenciales de la señal de control que nos interesen. Es obvio que este es el método más directo de abordar el problema y el que va a dar mejores resultados. En la Fig.

5.19 vemos la respuesta en frecuencia del error de modelado y observamos que a partir de la frecuencia  $0.1\text{rad/s}$  empieza a crecer la magnitud, lo que provocará un error de predicción considerable.

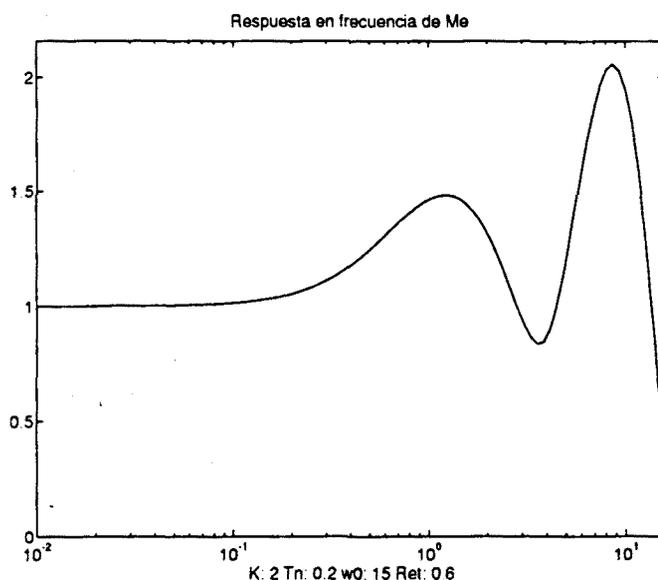


Figura 5.19: Respuesta en frecuencia del  $M_e$ .

Para evitar que la señal de control se vea amplificada por el error de modelado, pasaremos ésta por un filtro que elimine componentes frecuenciales de la señal de control a partir de la frecuencia  $0.1\text{rad/s}$ . Un filtro con estas características frecuenciales podría ser uno de primer orden con la siguiente función de transferencia,

$$F(q^{-1}) = \frac{0.15}{1 - 0.85q^{-1}}$$

y cuya respuesta en frecuencia se puede ver en la Fig. 5.20.

La respuesta del sistema total la podemos ver en la Fig. 5.21. Observamos que al haber eliminado las altas frecuencias de la señal de control el sistema responde de forma suave y sin oscilaciones. De las tres formas propuestas de aumentar la robustez, ésta parece la mejor, ya que por un lado se consigue la mejor respuesta

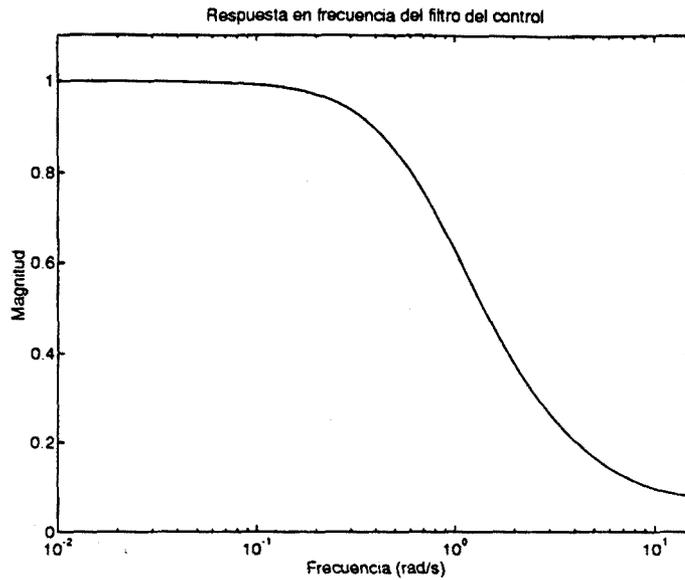


Figura 5.20: Respuesta en frecuencia del filtro  $F(q^{-1}) = \frac{0.15}{1-0.85q^{-1}}$ .

dinámica y por el otro la forma de diseñar el filtro del control es inmediata a partir de la observación de la respuesta en frecuencia del error de modelado  $M_e$ .

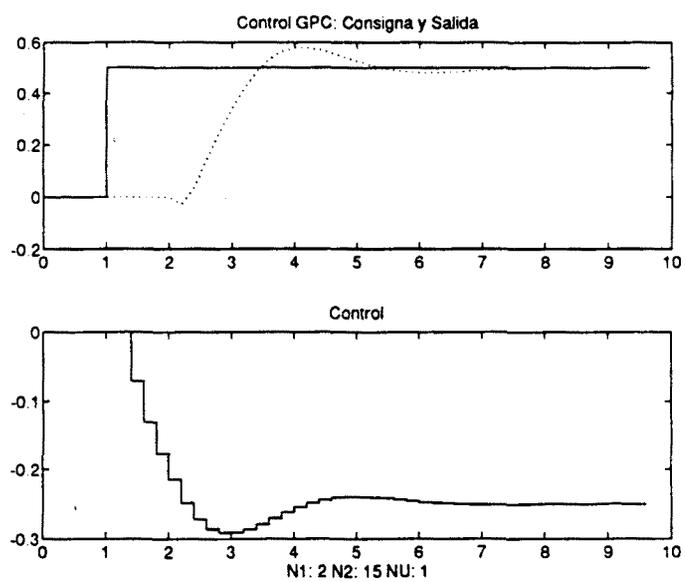


Figura 5.21: Respuesta de un GPC con el filtro  $F$ .

# Capítulo 6

## Conclusiones

En el presente trabajo se han estudiado diversos aspectos de implementación de controladores predictivos y se ha llevado a cabo un análisis de robustez de estos controladores. Los puntos principales tratados se exponen a continuación.

Se han presentado los elementos comunes a los CPBM. Para cada uno de estos elementos se dispone de varias opciones, dependiendo de la elección que se haga obtenemos un método u otro de control predictivo.

El primer aspecto de implementación tratado es referente al DMC. Si la deducción de la ley de control de este controlador la hacemos a partir de un modelo de respuesta escalón truncado, en vez de uno infinito, en la respuesta del sistema nos encontramos con comportamientos indeseables que a veces pasan desapercibido ya que se suelen producir una vez que se ha estacionado la salida. Otro problema encontrado es que la respuesta escalón ha de ser lo suficientemente grande para que los últimos  $N_2$  elementos sean prácticamente iguales; sino, se cometen errores en la predicción de la respuesta libre que se traducen en un comportamiento anómalo de la respuesta del sistema.

Se ha estudiado la forma de la función de coste en función de los parámetros

de diseño. Un factor de peso grande o un horizonte de coincidencia cercano produce mínimos de la función de coste más definidos, lo que tiene interés en problemas con restricciones donde se han de usar métodos numéricos, ya que éstos suelen tener problemas para encontrar el mínimo cuando la función a minimizar tiene grandes valles.

Se ha propuesto una implementación de sistemas multivariables mediante controladores SISO con compensación feedforward. Esta estrategia nos proporciona un rendimiento mejor que un controlador SISO con un coste computacional mucho menor que en el caso totalmente multivariable. Además, esta forma de implementación permite abordar el tema de las restricciones mediante una estructura override, de manera que no es necesario recurrir a la programación cuadrática, siempre más costosa computacionalmente, para tratar este problema. Los experimentos se han llevado a cabo sobre reactor químico exotérmico encamisado que es multivariable y presenta interacción entre sus lazos.

Se ha presentado una forma de analizar la robustez de controladores predictivos LTI basado en las técnicas del control robusto. El estudio se ha realizado sobre un controlador concreto, el GPC, de cual se ha deducido su forma como controlador de dos grados de libertad. Se ha llevado a cabo un estudio para establecer patrones de como influyen los parámetros de diseño sobre las funciones de sensibilidad y sensibilidad complementaria, o lo que es lo mismo, sobre la estabilidad y el rendimiento. Esto nos ha permitido proponer un método de diseño de controladores predictivos que tiene en cuenta las incertidumbres sobre la planta y, por tanto, tiene un comportamiento robusto.

Se ha presentado otra forma de abordar el tema de la robustez a través del modelo del ruido que usan los controladores predictivos. El modelo del ruido tiene una influencia directa en el error de predicción que se comete debido a las diferencias entre el modelo nominal usado para las predicciones y el modelo real. En el caso del GPC, el polinomio  $T$  es el elemento básico del modelo del ruido y nos permite diseñarlo para mejorar la robustez. Si el modelo del ruido no proporciona la suficiente

robustez se proponen tres formas de aumentarla. La más eficiente y que mejores resultados proporciona consiste en usar un filtro a la salida del controlador que filtre las frecuencias a las que se producen los errores de modelado.

Como resultado de estos estudios se aporta una librería de funciones escritas para el paquete MATLAB que permiten el análisis, diseño e implementación de controladores predictivos. Los experimentos realizados con el reactor químico se han realizado sobre ACSL.

### Lineas futuras

En el futuro se pretende abordar varios aspectos que han quedado abiertos. El primero hace referencia al controlador predictivo no lineal. Dada la actual metodología de diseño de sistemas de control en la cual se parte de un modelo bastante exacto de la planta sobre el cual se prueban los distintos diseños, parece lógico usar este mismo modelo en el controlador para predecir el comportamiento del sistema como hace el controlador propuesto. El problema que surge al usar un controlador de este tipo es que si para controladores predictivos lineales aún quedan muchas lagunas en la teoría, para controladores no lineal no hay prácticamente ninguna. El estudio de las propiedades de estabilidad, rendimiento y robustez de este tipo de controladores es un punto que nos proponemos investigar.

En esta misma línea, el uso de modelos no lineales paramétricos (Hammersstein, bilineal, ...) en controladores predictivos puede facilitar el estudio de controladores no lineales, a parte de las ventajas que presentan *per se*.

En el análisis de robustez y la librería de funciones que lo implementan se han tenido en cuenta el DMC y el GPC. Un tercer controlador muy popular y con características propias (uso de funciones base para estructurar la ley de control) es el PFC. Se pretende completar la librería con las funciones necesarias para realizar el análisis de robustez de este controlador.

Otro tema de investigación futura puede ser la ampliación del análisis de robustez basado en la teoría del control robusto al análisis y diseño de controladores multivariables, ya que esta teoría ya proporciona las herramientas. El problema consistirá fundamentalmente en como formularlo adecuadamente.

Muchos de los aspectos tratados en esta tesis no tienen un carácter completamente general. Este es un problema común en la mayoría de los resultados que se presentan en CPBM. Nos proponemos seguir trabajando en la obtención de una formulación general válida para el análisis y diseño de controladores predictivos basados en modelos.

# Apéndice A

## Reactor Químico Exotérmico Encamisado

Los reactores químicos son elementos muy comunes en la industria de procesos. El funcionamiento de un reactor químico exotérmico encamisado es el siguiente. En un tanque de reacción entra un flujo de líquido que contiene disuelta una sustancia A con una determinada concentración inicial. En el interior del reactor la sustancia A reacciona transformándose una parte en B. Esta reacción es exotérmica, o sea, libera una cierta cantidad de energía en forma de calor. Por tanto, es necesario enfriar el reactor. Esto se hace haciendo pasar un líquido refrigerante por una cámara (*camisa*) que envuelve el tanque.

El objetivo de control es mantener una determinada concentración de la sustancia B y conseguir que la temperatura en el interior del tanque sea la adecuada para que se produzca la reacción. Un esquema de un reactor químico se puede ver en la Fig. A.1.

En el tanque entra un flujo de líquido  $F_I$  con una concentración inicial de sustancia A  $C_{a0}$ . El volumen de líquido en el reactor es constante  $V$ , por tanto el flujo de líquido que entra es igual al que sale. Las concentraciones de las sustancias A y B a

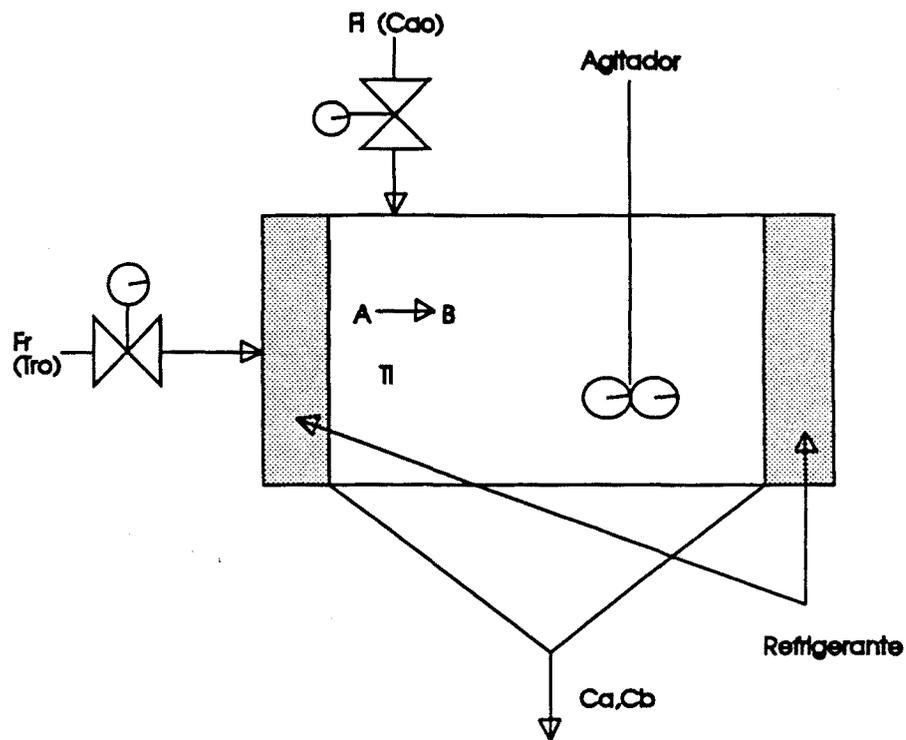


Figura A.1: Esquema de un reactor químico exotérmico encamisado

la salida del tanque son  $C_a$  y  $C_b$  respectivamente. A la cámara de refrigeración entra un flujo de líquido refrigerante  $F_r$  a una temperatura  $T_{r0}$ . Finalmente, la temperatura en el interior del tanque es  $T_l$ .

## A.1 Modelo del Proceso

Para poder hacer un estudio en simulación del reactor químico se ha desarrollado un modelo no lineal del proceso que nos proporcione las dinámicas del mismo. Para obtener el modelo se han aplicado principios fundamentales y las ecuaciones diferenciales y algebraicas obtenidas se han programado en el lenguaje de simulación ACSL (*Advanced Continuous Simulation Language*).

**Balance másico**

Aplicamos la ley de conservación de la masa a las sustancias A y B. Esto es, la variación de masa dentro del tanque será igual a la masa que entra más la que se genera, menos la masa que se consume y menos la que sale.

$$V \frac{dC_a}{dt} = F_l C_{a0} - V k C_a - F_l C_a \quad (\text{A.1})$$

$$V \frac{dC_b}{dt} = V k C_a - F_l C_b \quad (\text{A.2})$$

donde

$F_l$  : Flujo (volumétrico) de entrada al tanque ( $m^3/h$ ).

$C_a$  : Concentración de A en el tanque ( $Kgmol/m^3$ ).

$C_b$  : Concentración de B en el tanque ( $Kgmol/m^3$ ).

$V$  : Volumen del tanque ( $m^3$ ).

$k$  : Velocidad de reacción ( $1/h$ ).

$C_{a0}$ : Concentración de A en el flujo de entrada al tanque ( $Kgmol/m^3$ ).

La velocidad de reacción ( $k$ ) es la velocidad con que se transforma la sustancia A en B. Esta velocidad depende de la temperatura a la que se produce la reacción de la siguiente forma,

$$k = \alpha e^{-\frac{\beta}{T_l + 273}} \quad (\text{A.3})$$

$$\beta = \frac{Ea}{R} \quad (\text{A.4})$$

donde

$\alpha$  : Coeficiente que depende del tipo de reacción ( $1/h$ ).

$T_l$  : Temperatura en el interior del tanque ( $^{\circ}C$ ).

$E_a$  : Energía de activación de la reacción ( $kJ/Kgmol$ ).

$R$  : Constante de los gases ideales ( $kJ/Kgmol \text{ } ^\circ K$ ).

### Balance energético

Ahora aplicamos la ley de la conservación de la energía en el interior del tanque y en el interior de la cámara de refrigeración. Esto es, la variación de energía será igual a la energía que entra más la que se genera, menos la energía que sale menos la consumida.

$$V \rho_l C_{e_l} \frac{dT_l}{dt} = F_l \rho_l C_{e_l} (T_{l0} - T_l) - U S (T_l - T_r) + V k C_a H \quad (A.5)$$

$$V \rho_r C_{e_r} \frac{dT_r}{dt} = F_r \rho_r C_{e_r} (T_{r0} - T_r) + U S (T_l - T_r) \quad (A.6)$$

donde

$\rho_l$  : Densidad del líquido del tanque ( $Kg/m^3$ ).

$\rho_r$  : Densidad del refrigerante ( $Kg/m^3$ ).

$C_{e_l}$  : Calor específico del líquido del tanque ( $kJ/Kg \text{ } ^\circ C$ ).

$C_{e_r}$  : Calor específico del refrigerante ( $kJ/Kg \text{ } ^\circ C$ ).

$T_{l0}$  : Temperatura del líquido a la entrada del tanque ( $^\circ C$ ).

$U$  : Coeficiente de transmisión de calor global ( $kJ/h \text{ } m^2 \text{ } ^\circ C$ ).

$S$  : Superficie de intercambio de calor ( $m^2$ ).

$T_{r0}$  : Temperatura del refrigerante a la entrada de la cámara ( $^\circ C$ ).

$T_r$  : Temperatura del refrigerante en la cámara ( $^\circ C$ ).

$H$  : Calor de reacción ( $kJ/Kgmol$ ).

$F_r$  : Flujo de refrigerante ( $m^3/h$ ).

### Constantes del proceso

Las cinco ecuaciones planteadas forman el modelo matemático del proceso. A continuación se dan valores a las constantes que aparecen en el modelo. Estas constantes no corresponden a ningún proceso físico real. Se han elegido de manera que el proceso tenga una dinámica similar a la de un reactor químico estándar.

$$V = 125 \text{ m}^3$$

$$C_{a0} = 8 \text{ Kg/mol/m}^3$$

$$\alpha = 6700 \text{ 1/h}$$

$$E_a = 25000 \text{ kJ/Kg/mol}$$

$$R = 8.314 \text{ kJ/Kg/mol}^{\circ}\text{K}$$

$$\rho_l = 800 \text{ Kg/m}^3$$

$$\rho_r = 1000 \text{ Kg/m}^3$$

$$C_{e_l} = 3 \text{ kJ/Kg}^{\circ}\text{C}$$

$$C_{e_r} = 4.1868 \text{ kJ/Kg}^{\circ}\text{C}$$

$$T_{l0} = 30^{\circ}\text{C}$$

$$U = 4300 \text{ kJ/h m}^2^{\circ}\text{C}$$

$$S = 25 \text{ m}^2$$

$$T_{r0} = 20^{\circ}\text{C}$$

$$H = 20000 \text{ kJ/kgmol}$$

## A.2 Respuesta en Lazo Abierto

La respuesta en lazo abierto del proceso tiene un doble interés. Por una parte conocer como influyen las variables manipuladas en las variables de control, con el objeto de establecer el emparejamiento de las variables. Por otro parte, decidir

el periodo de muestreo que se va a usar en los controladores, ya que estos son digitales.

Primero hemos de elegir el punto de trabajo en que va a operar el reactor y determinar el estado estacionario a partir del cual realizaremos las simulaciones.

El punto de trabajo del reactor será  $C_b = 4 \text{ kgmol/m}^3$  y  $T_l = 60^\circ\text{C}$ . El valor del resto de variables para que el proceso esté en estado estacionario se calcula como es sabido igualando las derivadas a cero. Nos quedan los siguientes valores:

$$\begin{aligned}k &= 0.8024 \text{ 1/h} \\C_a &= 4 \text{ kgmol/m}^3 \\F_l &= 100.3052 \text{ m}^3/\text{h} \\F_r &= 5.8870 \text{ m}^3/\text{h} \\T_r &= 52.5393^\circ\text{C}\end{aligned}$$

Intuitivamente, el emparejamiento de lazos lógico sería controlar la concentración de B con el flujo de A y la temperatura del tanque con el flujo de refrigerante. En las Fig. A.2 y A.3 podemos observar como responde el proceso ante escalones unitarios en el flujo de A y en el flujo del refrigerante respectivamente.

Vemos que la concentración de B se ve afectada prácticamente igual en ambos casos. Sin embargo, el flujo de refrigeración afecta mucho a la temperatura del tanque que el flujo de A. Por tanto, los emparejamientos  $F_l - C_b$  y  $F_r - T_l$  parecen los más adecuados.

Estos experimentos también nos sirven para determinar el periodo de muestreo de los controladores. Se puede observar que la variable más rápida ( $C_b$ ) tiene un tiempo de respuesta de unas 15 horas. Por tanto, una elección adecuada del periodo de muestreo será de 1 hora.

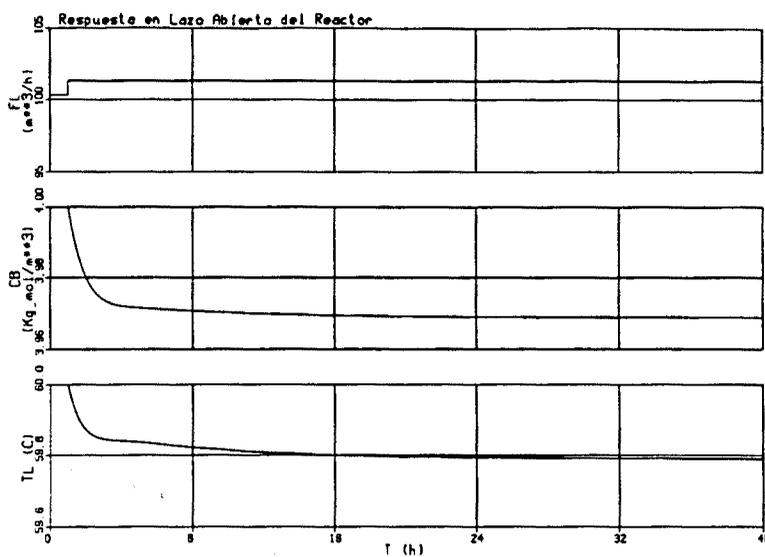


Figura A.2: Respuesta ante un salto en  $F_L$

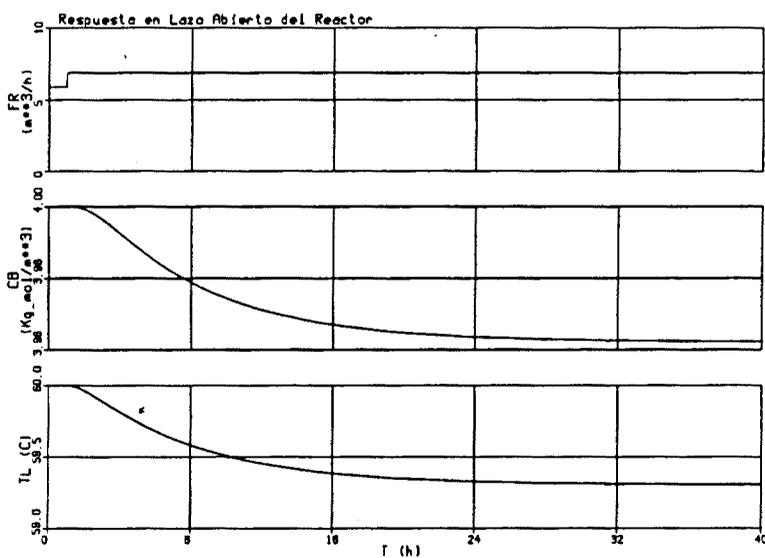


Figura A.3: Respuesta ante un salto en  $F_T$

## **Apéndice B**

# **Toolbox de Análisis y Diseño de CPBM Robusto**

En el presente apéndice se muestran los listados de los programas y funciones MATLAB que componen la librería usada para realizar los experimentos llevados a cabo en la tesis.

Algunos programas no se han incluido ya que eran prácticamente iguales que alguno de los que si se muestran. En cualquier caso, todos los programas y funciones describen su objetivo y funcionamiento pudiendo ser consultado con el comando help.

## B.1 Definición de Modelos

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                               bm.m                               %
% %
% Programa para definir el modelo nominal (numm/denm) y el modelo %
% real (nump/denp) de una planta. Por defecto el resto de programas %
% de esta libreria toma estos modelos para hacer los calculos. Ademas, %
% hace una grafica de la respuesta escalon de ambos modelos.      %
% %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                               Ejemplo Benchmark
%
K=1.;
Tn=1.;
w0=15.;
xi=1.;
retardo=0.2;
pm1=.2;
ruido=.0;

num=[K*Tn*w0^2 -K*w0^2];
den=conv([Tn 1],[1 2*xi*w0 w0^2]);
[numr,denr]=pade(retardo,4);
numerador=conv(num,numr);
denominador=conv(den,denr);
[a,b,c,d]=tf2ss(num,den);
[ad1,bd1,cd1,dd1]=c2dt(a,b,c,pm1,retardo);

```

```

[numm,denm]=ss2tf(ad1,bd1,cd1,dd1,1);

Kp=input('Ganancia del sistema (1.): ');if isempty(Kp), Kp=1.;end;
Tnp=input('Constante de tiempo del polo simple(1.): ');if isempty(Tnp),
                                                    Tnp=1.;end;
w0p=input('Frecuencia natural (15.):');if isempty(w0p), w0p=15;end;;
xip=1.;
retardop=input('Retardo (.2):');if isempty(retardop), retardop=0.2;end
num1=[Kp*Tnp*w0p^2 -Kp*w0p^2];
den1=conv([Tnp 1],[1 2*xip*w0p w0p^2]);
[a,b,c,d]=tf2ss(num1,den1);

[ad2,bd2,cd2,dd2]=c2dt(a,b,c,pm1,retardop);
[nump,denp]=ss2tf(ad2,bd2,cd2,dd2,1);

t1=0:pm1:6;
nt1=length(t1);
respp=dstep(nump,denp,nt1);
respm=dstep(numm,denm,nt1);
figure(1);clg;
plot(t1,respp,t1,respm);
title('Respuesta escalon del proceso (verde) y del modelo');

bden=conv(nump,denm);nbden=length(bden);
anum=conv(denp,numm);nanum=length(anum);
aden=conv(denp,denm);
if nbden > nanum
anum=[anum,zeros(1,nbden-nanum)];
else
bden=[bden, zeros(1,nanum-nbden)];

```

```
end
numem=anum-bden;

figure(2);
wnyquist=pi/pm1;
w=logspace(-2,log10(wnyquist),100);
[mag,fase]=dbode(numem,aden,pm1,w);
my=max(mag)+.1;
semilogx(w,mag);axis([.01 wnyquist 0 my]);
title('Respuesta en frecuencia de Me');
xlabel(['K: ',num2str(Kp),' Tn: ',num2str(Tnp),' w0: ',num2str(w0p),
        ' Ret: ',num2str(retardop)]);
```

## B.2 Controladores

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%          function du = dmc(g,uold,yold,n1,n2,nu,lambda,w,k)    %
%                                                                 %
%          Dynamic Matrix Controller                             %
%          -----                                              %
%          du = dmc (g,uold,yold,n1,n2,nu,lambda,w)            %
%                                                                 %
%          % Calcula el siguiente incremento de control para un DMC con: %
%          %                                                                 %
%          %          g:          Respuesta escalon de la planta          %
%          %          uold, yold: Valores pasados de entrada y salida      %
%          %          n1, n2:     Horizonte de coincidencia                %
%          %          nu:        Horizonte de control                     %
%          %          lambda:     Factor de peso                           %
%          %          w:          Vector de consignas futuras: [w(t+1)....w(t+n2)] %
%          %          k :         vecor de ganancias                       %
%          %          Devuelve el incremento de control actual du          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ng=length(g);

ddu1=zeros(1,ng);
for i=1:ng-1,
    ddu1(ng-i) = uold(i+1) - uold(i) ;           % u's filtradas
end;

```

```
% Calculo de la respuesta libre
for j=1:n2,
    f(j) = yold + (g(j+2:ng)-g(2:ng-j))*ddu1(1:ng-j-1)';
end;

% Calculo del vector de controles futuros y se toma el primero
e0=(w(n1:n2) - f(n1:n2))' ;
du = k*e0;

return;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
function [du,dyaux,duaux] = gpc(b,a,t,uold,yold,n1,n2,nu,lambda,w,
                                dyaux1,duaux1)
%
%          Generalised Predictive Controller
%          -----
%[du,dyaux,duaux]=gpc(b,a,t,uold,yold,n1,n2,nu,lambda,w,dyaux,duaux)
%
% Compute the next-control increment for a GPC controller with:
%   b, a:      the plant b(z)/a(z) polynomials
%   t         prefilter
%   uold, yold: old values of input,output
%   of length nb-1, na with u(t-1), y(t) as LAST elements
%   n1, n2:    output costing horizons
%   nu:        size of future-control-increment vector
%   lambda:    control costing
%   w:         future-set-point vector: [w(t+1).....w(t+n2)]
%
%   return the current control du
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

a1 = fliplr(a(2:length(a))); %miss a0=1
b1 = fliplr(b(2:length(b))); %miss b0=0 and b1 (not used in free)
%if b1==[], b1=0;end;
if t==[1],
    t1=0;
else
    t1 = fliplr(t(2:length(t))); %miss t0=1
end

```

```

na1 = length(a1); nb1 = length(b1); nt1 = length(t1);
N=length(dyaux1);

dy = zeros(1,N+n2); du = zeros(1,N+n2);    % delta y's u's

dy(1:N)=dyaux1;
du(1:N)=duaux1;

nyold=length(yold);

dy(N) = yold(nyold) - yold(nyold-1) - t1*dy(N-nt1:N-1)';

nuold=length(uold);
uaux=[uold uold(nuold)*ones(size(dy))];
for i=1:n2,
    du(N+i-2) = uaux(nuold+i-1) - uaux(nuold+i-2) - t1*du(N+i-nt1-2:N+i-3)';
end;

for i=1:n2,
    dy(N+i) = -a1*dy(N+i-na1:N+i-1)' + b1*du(N+i-nb1:N+i-1)'; % free response
end;

dyaux=dy(2:N+1);
duaux=du(2:N+1);
%Calculo de la respuesta libre f

f = [yold(nyold) zeros(1,n2)];           %initialise with current y(t)
%f(1) = yold(nyold) + dy(N) + t1*dy(N-nt1:N-1)';
for i=1:n2,
    f(i+1) = f(i) + dy(N+i) + t1*dy(N+i-nt1:N+i-1)'; %y(t), y(t+1)... y(t+n2)

```

```
end;
f= f(2:n2+1);

% now compute the future control vector, and take first element
k=gainm(b,a,n1,n2,nu,lambda);
e0=(w(n1:n2) - f(n1:n2))' ;
du=k*e0;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
function [du,dyaux,duaux] = gpcres(b,a,t,uold,yold,n1,n2,nu,lambda,w,
                                dyaux1,duaux1)
%                                                                 %
%   Generalised Predictive Controller                             %
%   -----                                                       %
%[du,dyaux,duaux]=gpcres(b,a,t,uold,yold,n1,n2,nu,lambda,w,dyaux1,duaux1)
%                                                                 %
% Compute the next-control increment for a GPC controller with:  %
%   b, a:   the plant b(z)/a(z) polynomials                       %
%   t       prefilter                                             %
%   uold, yold: old values of input,output                       %
%               of length nb-1, na with u(t-1), y(t) as LAST elements%
%   n1, n2: output costing horizons                               %
%   nu:     size of future-control-increment vector              %
%   lambda: control costing                                       %
%   w:      future-set-point vector: [w(t+1).....w(t+n2)]        %
%                                                                 %
%   return the current control du                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

a1 = fliplr(a(2:length(a))); %miss a0=1
b1 = fliplr(b(3:length(b))); %miss b0=0 and b1 (not used in free)
if t==[1],
    t1=0;
else
    t1 = fliplr(t(2:length(t))); %miss t0=1
end
na1 = length(a1); nb1 = length(b1); nt1 = length(t1);

```

```

N=length(dyaux1);

dy = zeros(1,N+n2); du = zeros(1,N+n2);    % delta y's u's

dy(1:N)=dyaux1;
du(1:N)=duaux1;

nyold=length(yold);

dy(N) = yold(nyold) - yold(nyold-1) - t1*dy(N-nt1:N-1)' ;

nuold=length(uold);
uaux=[uold uold(nuold)*ones(size(dy))];
for i=1:n2,
    du(N+i-1) = uaux(nuold+i-1) - uaux(nuold+i-2) - t1*du(N+i-nt1-1:N+i-2)';
end;

for i=1:n2,
    dy(N+i) = -a1*dy(N+i-na1:N+i-1)' + b1*du(N+i-nb1:N+i-1)'; % free response
end;

dyaux=dy(2:N+1);
duaux=du(2:N+1);
%sum up to get the POSITIONAL free response f

f = [yold(nyold) zeros(1,n2)];           %initialise with current y(t)
%f(1) = yold(nyold) + dy(N) + t1*dy(N-nt1:N-1)';
for i=1:n2,
    f(i+1) = f(i) + dy(N+i) + t1*dy(N+i-nt1:N+i-1)'; %y(t), y(t+1)... y(t+n2)
end;
f= f(2:n2+1);

```

```
% now compute the future control vector, and take first element
```

```
global E0 FF G N1 N2 NU LAMBDA U UMIN UMAX YMIN YMAX
```

```
ustep = ones(1,n2+1);           % step-response
```

```
g = filter(b,a,ustep);         % into g: note direct link in filter
```

```
% now construct the G matrix with g's down the columns
```

```
G = zeros(n2,nu);
```

```
for i=1:nu,                     %nu columns
```

```
    G(:,i) = [zeros(1,i-1) g(2:n2-i+2)]'; %banded
```

```
end;
```

```
E0=(w(n1:n2) - f(n1:n2))';
```

```
FF=f;
```

```
uinci=zeros(size(1:nu));
```

```
opciones=[];
```

```
N1=n1;
```

```
N2=n2;
```

```
NU=nu;
```

```
LAMBDA=lambda;
```

```
nbb=length(uold);
```

```
U=uold(nbb);
```

```
% Se establecen las restricciones
```

```
UMIN=-10.;
```

```
UMAX=10.;
```

```
YMIN=-10.;
```

```
YMAX=10.;
```

```
dumin=-.05;
```

```
dumax=.05;
```

```
% Se llama a la rutina de minimizacion con restricciones
uinc = constr('fc', uinci, opciones, dumin, dumax);
du = uinc(1);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
    function [f,res]=fc(uinc)
%                                                                 %
% Rutina que calcula el valor de la funcion de coste y fija las   %
% restricciones actuales para la rutina de minimizacion         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
global E0 FF G N1 N2 NU LAMBDA U UMIN UMAX YMIN YMAX
GG = G(N1:N2,:); %slice top off
```

```
f=E0'*E0-uinc*(GG'*E0)-E0'*GG*uinc'+uinc*(GG'*GG + LAMBDA*eye(NU))*uinc' ;
u=U;
```

```
for i=1,NU;
res(i)=-u-uinc(i)+UMIN;
u=u+uinc(i);
end;
u=U;
```

```
NU2=NU+NU;
for i=NU+1,NU2;
res(i)=u+uinc(i-NU)-UMAX;
u=u+uinc(i-NU);
end;
```

```
NU2N2=NU2+N2;
for i=NU2+1,NU2N2;
res(i)=YMIN - FF(i-NU2)-G(i-NU2,:)*uinc';
```

```
end;  
NU2N22=NU2N2+N2;  
for i=NU2N2+1,NU2N22;  
res(i)=FF(i-NU2N2)+G(i-NU2N2,:)*uinc'-YMAX;  
end;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                               dodmc.m                               %
%
% Programa para simular un DMC. Los calculos se realizan en cada periodo%
% de muestreo como si se trabajara en tiempo real.                    %
%
% Al usuario se le pide que introduzca el modelo de la planta real,    %
% el modelo nominal, el periodo de muestreo, as\ 'Y como los parametros %
% de sintonia del DMC. Todas las peticiones tienen valores por defecto, %
% aunque en el caso de los modelos y el periodo de muestreo tienen que %
% haber sido almacenados previamente en las variable nump y denp para %
% el modelo de la planta real, en numm y denm para el modelo nominal y %
% en pm1 el periodo de muestreo.                                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp('Simulacion del DMC');

a=input('Denominador de la planta real (denp): ');
if isempty(a), a=denp; end;
b=input('Numerador de la planta real (nump): ');
if isempty(b), b=nump; end;
den=input('Denominador del modelo nominal (denm): ');
if isempty(den), den=denm; end;
num=input('Numerador del modelo nominal (numm): ');
if isempty(num), num=numm; end;
pm=input('Periodo de muestreo (pm1): ');
if isempty(pm), pm=pm1; end;

na = length(a); nb = length(b); mab = max(na,nb-1);
zs = zeros(1,mab-1);           %Inicializacion del estado del filtro

```

```

nt = input('How many samples do you want to use? (31):');
if isempty(nt), nt=31; end;

ng=40;
t=1:nt; %Definicion del vector de tiempos
w=0.5*sign(sin(t*(2*pi/200))); %Definicion de la consigna
u=zeros(1,ng+nt+1); y=zeros(size(t)); %Inicializacion
w(1:5)=zeros(1,5);

n1 = input('N1? (1): '); if isempty(n1), n1=1; end;
n2 = input('N2? (10): '); if isempty(n2), n2=10; end;
nu = input('NU? (1): '); if isempty(nu), nu=1; end;
lambda=0;

% Calculo de la respuesta escalon
ustep=ones(1,ng);
g=filter(b,a,ustep);

k=gainm(b,a,n1,n2,nu,lambda);

for i = mab:nt,
    [y(i) zs] = filter(b(2:nb),a,u(ng+i-1),zs); % Salida del proceso
    du = dmc(g,u(i:ng+i-1),y(i),n1,n2,nu,lambda,w(i)*ones(1,n2),k);
    u(ng+i) =min(10.,max(-10.,u(ng+i-1) + du));
end;

subplot(211);
t=0:pm:pm*(nt-1);
plot(t,y,':g');
hold on; stairs(t,w); hold off;
title(['Control DMC: Consigna y Salida']);

```

```
subplot(212);  
stairs(t,u(ng+1:ng+nt));  
title('Control');  
xlabel(['N1: ',num2str(n1),' N2: ',num2str(n2),' NU: ',num2str(nu)]);  
  
subplot(111);
```

### B.3 Análisis

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
function [du,dyaux,duaux] = gpcgfc(dib,b,a,t,uold,yold,n1,n2,nu,lambda,
                                w,dyaux1,duaux1)
%
%       Generalised Predictive Controller
%       -----
% [du,dyaux,duaux] = gpcgfc(dib,b,a,t,uold,yold,n1,n2,nu,lambda,w,
%                           dyaux1,duaux1)
%
% Compute the next-control increment for a GPC controller with:
%   b, a:      the plant b(z)/a(z) polynomials
%   t         prefilter
%   uold, yold: old values of input,output
%   of length nb-1, na with u(t-1), y(t) as LAST elements
%   n1, n2:   output costing horizons
%   nu:       size of future-control-increment vector
%   lambda:   control costing
%   w:       future-set-point vector: [w(t+1).....w(t+n2)]
%
%   Dibuja la funcion de coste
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ustep = ones(1,n2+1);           % step-response
g = filter(b,a,ustep);         % into g: note direct link in filter

% now construct the G matrix with g's down the columns
G = zeros(n2,nu);

```

```

for i=1:nu,                                %nu columns
    G(:,i) = [zeros(1,i-1) g(2:n2-i+2)]'; %banded
end;
G = G(n1:n2,:);                            %slice top off

a1 = fliplr(a(2:length(a))); %miss a0=1
b1 = fliplr(b(3:length(b))); %miss b0=0 and b1 (not used in free)
if t==[1],
    t1=0;
else
    t1 = fliplr(t(2:length(t))); %miss t0=1
end
na1 = length(a1); nb1 = length(b1); nt1 = length(t1);
N=5;
dy = zeros(1,N+n2+na1); du = zeros(1,N+n2+nb1); % delta y's u's

for i=1:N,
    dy(i)=dyaux1(i);
    du(i)=duaux1(i);
end;
for i=1:na1,
    dy(N+i) = yold(i+1) - yold(i) - t1*dy(N+i-nt1:N+i-1)'; % filtered y's
end;
uaux=[uold uold(length(uold))*ones(size(dy))];
for i=1:n2,
    du(N+i) = uaux(i+1) - uaux(i) - t1*du(N+i-nt1:N+i-1)'; % filtered u's
end;

for i=1:n2,
    dy(N+i+na1) = -a1*dy(N+i:N+na1+i-1)' + b1*du(N+i:N+nb1+i-1)';
end;

```

```

dyaux=dy(2:N+1);
duaux=du(2:N+1);

%Calculo de la respuesta libre f
f = [yold(na1+1) zeros(1,n2)];           %initialise with current y(t)
for i=1:n2,
    f(i+1) = f(i) + dy(N+i+na1) + t1*dy(N+i+na1-nt1:N+i+na1-1)';
end;
f= f(2:n2+1);

% Calculo de la funcion de coste y grafico
aux1=inv(G'*G + lambda*eye(nu));
aux2=G'*G + lambda*eye(nu);
e0=(w(n1:n2) - f(n1:n2))' ;
ddu = aux1*G'*e0;
du = ddu(1);
if dib==1,
    v=0:1.:20.;
    u1=-5.:5:5.;
    u2=u1;
    [uinc1,uinc2]=meshgrid(u1,u2);
    nnn=length(uinc1);
    j=zeros(uinc1,uinc2);
    for i=1:nnn,
        for k=1:nnn,
            j(i,k)=(e0'*e0-[uinc1(i,k) uinc2(i,k)]*(G'*e0)-e0'*G*[uinc1(i,k) uinc2(i,k)
                +[uinc1(i,k) uinc2(i,k)]*aux2*[uinc1(i,k) uinc2(i,k)]')');
        end;
    end;
end;
mesh(u1,u2,j);view(-15,5);

```

```
title(['N1: ',num2str(n1),' N2: ',num2str(n2),' Lambda: ',num2str(lambda)])  
end;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
function [Ej,Fj]=diofan(T,V,j);
%                                                                 %
% Funcion que resuelve ecuaciones diofanticas:                  %
%                                                                 %
%                               -j                               %
%                               T = Ej*V + q *Fj                 %
%                                                                 %
% function [Ej,Fj]=diofan(T,V,j);                               %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

nt=length(T)-1;
nv=length(V)-1;
if nv==0,
    ne=min(j-1,nt);
    nf=max(0.,nt-i);
    if j>=nt+1;
        Ej=T/V(1);
        Fj=0.;
    else
        Ej=T(1:j)/V(1);
        Fj=T(j+1:nt+1);
    end;
else
    ne=j-1;
    nf=max(nt-j,nv-1);
    dim=ne+nf+2;
    TT=[T zeros(1,dim-nt-1)]';
    Ej=zeros(1,ne+1);

```

```
Fj=zeros(1,nf+1);
EF=[Ej Fj];
Q=zeros(dim,dim);
for i=1:j,
    Q(:,i)=[zeros(1,i-1) V zeros(1,dim-nv-i)]';
end;
Q(j+1:dim,j+1:dim)=diag(diag(ones(nf+1,nf+1)));
EF=inv(Q)*TT;
Ej=EF(1:j)';
Fj=EF(j+1:dim)';
end;
end;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
function K= gainm(b,a,n1,n2,nu,lambda)
%
%          Calculo de la matriz de ganancias inv(G'*G + lambda*I)G'
%          -----
%          K= gainm(b,a,n1,n2,nu,lambda)
%
%
%          b, a:      the plant b(z)/a(z) polynomials
%          n1, n2:    output costing horizons
%          nu:        size of future-control-increment vector
%          lambda:    control costing
%
%          return the first column of tha matrix
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ustep = ones(1,n2+1);          % step-response
g = filter(b,a,ustep);        % into g: note direct link in filter

% now construct the G matrix with g's down the columns
%global G
G = zeros(n2,nu);
for i=1:nu,                    %nu columns
    G(:,i) = [zeros(1,i-1) g(2:n2-i+2)]'; %banded
end;
G = G(n1:n2,:);               %slice top off

% now compute the gain matrix

```

```
TT=inv(G'*G + lambda*eye(nu))*G';  
K=TT(1,:);
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
    function pc=polcar(b,a,r,s);
%                                                                 %
% Rutina que calcula el polinomio caracteristico de un controlador de %
% dos grados de libertad. %
%                                                                 %
% pc = A*R + B*S %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

pc1 = conv(r,a); pc2 = conv(b,s);
npc1 = max(size(pc1)); npc2 = max(size(pc2));
if npc1 > npc2, pc2 = [pc2 zeros(1,npc1-npc2)]; end;
if npc2 > npc1, pc1 = [pc1 zeros(1,npc2-npc1)]; end;
pc = pc1+pc2; root = abs(roots(pc)); n = length(root);
for i = 1:n,
    if root(i) > 1,
        disp('closed-loop unstable');
        roots(pc),
        return;
    end;
end;

np = max(size(pc)); np1 = np;
for i = np1:-1:1;
    if abs(pc(i)) < pc(1)*sqrt(eps), np = np-1; end;
end;
pc = pc(1:np);
return

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
function [numS,denS,numT,denT]=sensib(b,a,r,s,t,pm);
%                                                                 %
% Rutina que hace un grafico de las funciones de sesibilidad y de la %
% sensibilidad complementaria y devuelve las funciones de transferencias%
% de estas curvas.                                                                 %
%                                                                 %
%      [numS,denS,numT,denT]=sensib(b,a,r,s,t,pm)                                                                 %
%                                                                 %
%      b,a      : Numerador y denominador del modelo nominal                                                                 %
% r,s,t : Polinomios de un controlador de dos grados de libertad%
%      pm      : Periodo de muestreo                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

pc=polcar(b,a,r,s);
ar=conv(a,r);
bt=conv(b,t);
ws=logspace(-2,log10(pi/pm),100);
nar=length(ar);npc=length(pc);
if npc<nar, pc=[pc zeros(1,nar-npc)]; end;
[mags,fass]=dbode(ar,pc,pm,ws);
numS=ar;
denS=pc;
npc=length(pc);
nbt=length(bt);

[magt,fast]=dbode(bt,pc,pm,ws);
numT=bt;
denT=pc;
retardo=0;

```

```
i=1;
while bt(i) == 0,
retardo=retardo+1;
i=i+1;
end;
bbt=bt(retardo+1:nbt);nbbt=length(bbt);
if npc>nbbt, bbt=[bbt zeros(1,npc-nbbt)]; end;
semilogx(ws,mags,ws,magt,'-.')%,wti,magti);
title('Funciones S (Continua) y T (Discontinua)');
ylabel('Mag');
xlabel('Frecuencia normalizada (rad/s)');
axis([.01 (pi/pm) 0 2.5]);
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
function incert(b,a,num,den,pm);
%                                                                 %
%      Funcion que dibuja la curva frecuencial de la incertidumbre de %
%      un modelo (lm).                                                                 %
%                                                                 %
%      incert(b,a,num,den,pm);                                                                 %
%                                                                 %
%      b,a      : Numerador y denominador del proceso real.                                                                 %
%      num,den  : Numerador y denominador del modelo nominal.                                                                 %
%      pm      : Periodo de muestreo.                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ba=conv(b,a);
baden=conv(ba,den);nbaden=length(baden);
aa=conv(a,a);
aanum=conv(aa,num);naanum=length(aanum);

if nbaden>naanum, aanum=[aanum zeros(1,nbaden-naanum)];end;
if naanum>nbaden, baden=[baden zeros(1,naanum-nbaden)];end;

numerador=aanum-baden;nnumerador=length(numerador);
if nnumerador>nbaden, baden=[baden zeros(1,nnumerador-nbaden)];end;

[mag,fas,w]=dbode(numerador,baden,pm);
loglog(w,mag,'r');

```

## B.4 Diseño Robusto

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%
%                               bmincert.m                               %
%
%
% Programa para calcular la incertidumbre de una familia de plantas   %
% definidas a partir del modelo nominal correspondiente al modelo     %
% benchmark.                                                            %
% Para cada familia de plantas que queramos controlar es necesario   %
% definir un m-file como este.                                         %
% El programa calcula la matriz svlm que es usada en el siguiente paso %
% por el programa rmincert.                                            %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Valor nominal de los par\ss{}metros
K=1.;
Tn=1.;
w0=15.;
xi=1.;
retardo=0.2;
pm1=.2;
percent=50.;
N=3;

% Rango de frecuencias en el que calcular las incertidumbres
wnyquist=pi/pm1;
w=logspace(-2,log10(wnyquist),100);

%Calculo del modelo nominal

```

```

num=[K*Tn*w0^2 -K*w0^2];
den=conv([Tn 1],[1 2*xi*w0 w0^2]);
[numr,denr]=pade(retardo,4);
numerador=conv(num,numr);
denominador=conv(den,denr);
[a,b,c,d]=tf2ss(num,den);
[ad1,bd1,cd1,dd1]=c2dt(a,b,c,pm1,retardo);
[numm,denm]=ss2tf(ad1,bd1,cd1,dd1,1);

[svo,fase]=dbode(numm,denm,pm1,w);

% Definicion de los intervalos de variacion de los parametros a partir
% del porcentaje deseado.
Kmin=K*(1-percent/100);
Kmax=K*(1+percent/100);
Tnmin=Tn*(1-percent/100);
Tnmax=Tn*(1+percent/100);
w0min=w0*(1-percent/100);
w0max=w0*(1+percent/100);
retardomin=retardo*(1-percent/100);
retardomax=retardo*(1+percent/100);

% Definimos los incrementos de los parametros y la dimension de la
% matriz.
Kstep=(Kmax-Kmin)/(N+1);
Tnstep=(Tnmax-Tnmin)/(N+1);
w0step=(w0max-w0min)/(N+1);
retardostep=(retardomax-retardomin)/(N+1);
maxj=(N+1)^4;
svlm=zeros(maxj,100);

```

```

% Calculamos en el espacio de parametros la incertidumbre asociada
% a cada modelo.
j=0;
for Kj=Kmin:Kstep:Kmax
    for Tnj=Tnmin:Tnstep:Tnmax
        for w0j=w0min:w0step:w0max
            for retardoj=retardomin:retardostep:retardomax
j=j+1;
num1=[Kj*Tnj*w0j^2 -Kj*w0j^2];
den1=conv([Tnj 1],[1 2*xi*w0j w0j^2]);
[a,b,c,d]=tf2ss(num1,den1);
[ad2,bd2,cd2,dd2]=c2dt(a,b,c,pm1,retardoj);
[nump,denp]=ss2tf(ad2,bd2,cd2,dd2,1);
% Calculo de G-Go.
bden=conv(nump,denm);nbden=length(bden);
anum=conv(denp,numm);nanum=length(anum);
aden=conv(denp,denm);
if nbden > nanum
    anum=[anum,zeros(1,nbden-nanum)];
else
    bden=[bden, zeros(1,nanum-nbden)];
end
numem=anum-bden;
% Calculo de la incertidumbre y respuesta en frecuencia
[mag,fase]=dbode(numem,aden,pm1,w);
svlm(j,:)=mag' ./svo';
end;end;end;end;

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                               rmincert.m                               %
%
% Programa para calcular la cota superior de una familia de plantas. %
% Antes se ha de haber generado la matriz svlm que contiene las    %
% incertidumbres de la familia de plantas.                          %
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

subplot(111);

sino='s';
while sino=='s',
for j=1:maxj
semilogx(w,svlm(j,:));
axis([.01 wnyquist 0 2]);
hold on;
end;
title('Curvas de incertidumbres');
xlabel('Frec. (rad/s)')

lbf=input('Magnitud a bajas frecuencias: ');
laf=input('Magnitud a altas frecuencias: ');
pin=input('Punto de inflexion: ');
num=[laf lbf*pin];
den=[1 pin];
[a,b,c,d]=tf2ss(num,den);
[ad1,bd1]=c2d(a,b,pm1);
[numrm,denrm]=ss2tf(ad1,bd1,c,d,1);
[mag,fase]=dbode(numrm,denrm,pm1,w);

```

```
semilogx(w,mag,'*r')  
hold off;  
sino=input('Continuar? (s/n) ');if isempty(sino),sino='s';end;  
end;
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                               disgpc.m                          %
%                                                                 %
% Rutina iterativa para buscar los parametros del GPC de        %
% manera que cumpla la condicion de estabilidad robusta.        %
% Antes se ha de haber ejecutado el programa que calcula la cota %
% superior de la incertidumbre (numrm/denrm).                   %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp('Diseño de un GPC robusto');
figure(1);
sino='s';
while sino=='s',
    n1 = input('N1? (1): '); if isempty(n1), n1=1; end;
    n2 = input('N2? (15): '); if isempty(n2), n2=15; end;
    nu = input('NU? (1): '); if isempty(nu), nu=1; end;
    lambda = input('Factor de peso? (0): '); if isempty(lambda),
                                                lambda=0; end;
    tt = input('Modelo del ruido T(z) (1): ');
    if isempty(tt), tt=1; end;

    [mag,fase]=dbode(denrm,numrm,pm1,w);
    clg;
    semilogx(w,mag,'*');hold on;

    [r,s,ttt]=gpcrst(numm,denm,tt,n1,n2,nu,lambda);
    rdelta=conv(r,[1 -1]);
    [numS,denS,numT,denT]=sensib(numm,denm,rdelta,s,ttt,pm1);
    title('Funciones T (con.) y 1/rm (ast.)');
end

```

```

    xlabel('Frec. (rad/s)');
    hold off;
    sino=input('Continuar? (s/n) ');if isempty(sino),sino='s';end;
end;

figure(2);
numrmT=conv(numrm,numT);
denrmT=conv(denrm,denT);
[mag,fase]=dbode(numrmT,denrmT,pm1,w);
linea=ones(size(w));
semilogx(w,mag,w,linea,'-.'); axis([.01 wnyquist 0 1.5]);
title('Verificacion de la condicion de estabilidad robusta, rmT');
xlabel('Frec. (rad/s)');
ylabel('Mag');

figure(1)
sino='s';
while sino=='s',
    [mag,fase]=dbode(numS,denS,pm1,w);
    semilogx(w,mag);
    axis([.01 wnyquist 0 2.]);
    hold on;
    title('Funcion S y Ws');
    xlabel('Frec. (rad/s)')

fc=input('Frecuencia de corte de un filtro de primer orden: ');
num=[1];
den=[1/fc 1];
[a,b,c,d]=tf2ss(num,den);
[ad1,bd1]=c2d(a,b,pm1);
[numws,denws]=ss2tf(ad1,bd1,c,d,1);

```

```
[mag,fase]=dbode(denws,numws,pm1,w);
semilogx(w,mag,'*r')
hold off;
sino=input('Continuar? (s/n) ');if isempty(sino),sino='s';end;
end;

figure(2);
numwsS=conv(numws,numS);
denwsS=conv(denws,denS);
[mag,fase]=dbode(numwsS,denwsS,pm1,w);
linea=ones(size(w));
semilogx(w,mag,w,linea,'-.'); axis([.01 wnyquist 0 1.5]);
title('Verificacion de la condicion de rendimiento nominal, WsSo');
xlabel('Frec. (rad/s)');
ylabel('Mag');
```

# Bibliografía

- [Abu 90] Abu El Ata-Doss, S., Estival, J.L., *Predictive Functional Control Technique (PFC)*, Workshop on "Computer integrated Design of Controlled Industrial Systems", París, 26-27 Abril, 1990.
- [Aström 73] Aström, K.J., Wittenmark, B., *On self-tuning regulators*, Automatica, Vol. 9, 1973, pp. 185-199.
- [Berlin 93] Berlin, F., Frank, P.M., *Design and realization of a MIMO predictive controller for a 3-tank system*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Bitmead 90] Bitmead, R., Gevers, M., Wertz, V., *Adaptive Optimal Control*, Prentice Hall, Australia, 1990.
- [Bruijn 84] Bruijn, P.M., Verbruggen, H.B., *Model algorithmic control using impulse response models*, Journal A, No. 25, 1984, pp. 69-74.
- [Bruijn 86] Bruijn, P.M., Verbruggen, H.B., *Predictive control: A comparison and simple implementation*, IFAC Low Cost Automation, Valencia, 1986.
- [Camacho 93] Camacho, E., *Constrained Generalized Predictive Control*, IEEE Trans. on Automatic Control, Vol. 38, No. 2, 1993, pp. 327-332.
- [Campo 86] Campo, P.J., Morari, M.,  *$\infty$ -Norm Formulation of Model Predictive Control Problems*, Proc. American Control Conf., Seattle, WA, 1986, pp. 339-343.

- [Chang 83] Chang, T.s., Seborg, D.E., *A Linear programming Approach to Multi-variable Feedback Control with Inequality Constraints*, Int. J. of Control, Vol. 37, 1983, pp. 583–597.
- [Charos 90] Charos, G.N., *Model Predictive Constrained Control: Development, Implementation and decentralization*, PhD. Tesis, Georgia Institute of Tech., USA, 1990.
- [Chen 82] Chen, C.C., Shaw, L., *On receding horizon feedback control*, Automatica, Vol. 18, 1982, pp. 349–352.
- [Clarke 87] Clarke, D.W, Mohtadi, C., Tuffs, P.S, *Generalized Predictive Control*, Parts 1 and 2. Automatica, Vol. 23, No. 2, 1987, pp. 137–160.
- [Clarke 89] Clarke, D.W, Mohtadi, C., *Properties of Generalized Predictive Control*, Automatica, Vol. 25, No. 6, 1989, pp. 859–875.
- [Clarke 90] Clarke, D.W., *Model Based Predictive Control: Research and Industrial Applications*, Workshop on Computer Integrated Design of Controlled Industrial Systems, EEC/ESPRIT, Brussels, 1990.
- [Clarke 91] Clarke, D.W., Scattolini, R., *Constrained receding-horizon predictive control*, IEE Proc. Pt. D. 138(4), pp. 347–354.
- [Clarke 92] Clarke, D.W., *Implementation Aspects and Design Parameters in MBPC*, CIM-Europe, CIDIC Course, Madrid, 1992.
- [Clarke 93] Clarke, D.W., *Generalized Predictive Control and its Application*, University of Oxford, Oxford, 1993.
- [Clarke 93a] Clarke, D.W., *Advances in model-based predictive control*. Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Cutler 80] Cutler, C.R., Ramaker, B.L., *Dynamic Matrix Control- a computer control algorithm*, Proc. JACC, San Francisco, USA, 1980.

- [Cutler 87] Cutler, C.R., Hawkins, R.B., *Constrained Multivariable Control of a Hydrocracker Reactor*, Proc. American Control Conf., Minneapolis, MN, 1987, pp. 1014-1020.
- [DeKeyser 85a] DeKeyser, R.M.C., van Cauwenberghe, A.R., *Extended Prediction Self-Adaptive Control*, IFAC Symposium on Identification and System Parameter Estimation, York, UK, 1985, pp. 1255-1269.
- [DeKeyser 85b] DeKeyser, R.M.C., VandeVelde, A.R., Dumortier, A.G., *A Comparative Study of Self-Adaptive Long-Range Predictive Control Methods*, IFAC Symposium on Identification and System Parameter Estimation, York, UK, 1985, pp. 1317-1322.
- [Dion 91] Dion, J.M., Dugard, L., Franco, A., Nguyen Minh Tri, Rey, D., *MIMO Adaptive Constrained Predictive Control Case Study: An Environmental Test Chamber*, Automatica, Vol. 27, No. 4, 1991, pp. 611-626.
- [Douas 92] Douas, M., Fernández, G., Martín, J.M., *Adaptive Predictive Control of the First Spanish Installation for the Official Approval of Nuclear Equipment*, Cambridge Workshop on "Industrial Applications of Model Based Predictive Control". 6-7 Julio, 1992.
- [Doyle 92] Doyle, J.C., Francis, B.A., Tannenbaum, A.R., *Feedback Control Theory*, Macmillan Publishing Company, New York, 1992.
- [Francis 87] Francis, B.A., *A course in  $H_\infty$  control theory*, Springer-Verlag, 1987.
- [García 84] García, C.E., Morshedi, A.M., *Quadratic programming solution of dynamic matrix control*, Eng. Chem. Commun., 1984, 46, pp. 073-087.
- [García 86] García, C.E., Prett, D.E., *Advances in Industrial Model- Predictive Control*, Chemical Process Control, CPCIII CACHE and Elsevier, Amsterdam, 1986, pp. 249-294.
- [García 89] García, C.E., Prett, D.M., Morari, M., *Model Predictive Control: Theory and Practice-a Survey*, Automatica, Vol. 25, No. 3, 1989, pp. 335-348.

- [Hangos 93] Hangos, K.M., Csáki, Zs., Varga, E.I., *Use of qualitative models for the choice of design parameters of MBPC*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Kaynak 89] Kaynak, O., Hayer, H., *Predictive Control of a Robotic Arm*, In: U. Rembold (Ed.) IFAC Robot Control' 88, Pergamon Press, Oxford, 1989, pp. 213–218.
- [Keerthi 86] Keerthi, S.S., Gilbert, E.G., *Moving horizon approximation for a general class of optimal nonlinear infinite horizon discrete time systems*, Proc. 20th Ann. conf. on info. Sci. and Systems, Princeton Univ., 1986, pp. 301–306.
- [Kouvaritakis 92] Kouvaritakis, B., Rossiter, J.A., Chang, A.O.T., *Stable generalised predictive control: an algorithm with guaranteed stability*, IEE Proceedings-D, Vol. 139, No. 4, Oxford, Julio 1992, pp. 349–362.
- [Kouvaritakis 93] Kouvaritakis, B., Rossiter, J.A., *Multivariable stable generalised predictive control*, IEE Proceeding-D, Vol. 140, No. 5, Oxford, Septiembre 1993.
- [Krauss 93] Krauss, P., Dass, K., Chotai, A., *Model-based predictive controller with Kalman filtering for state estimation*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Kuntze 86] Kuntze, H.B., Jacobasch, A., Richalet, J., Arber, Ch., *On the predictive functional control of an elastic industrial robot*, 25th Conference on Decision and Control, 1986.
- [Kuznetsov 93] Kuznetsov, A.G, Clarke, D.W., *Application of constrained GPC for improving performance of controlled plants*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.

- [Kwon 77] Kwon, W.H., Pearson, A.E., *A modified quadratic cost problem and feedback stabilization of a linear systems*, IEEE Trans. Automatic Control, Vol. 22, 1977, pp. 838-842.
- [Kwon 83] Kwon, W.H., Bruckstein, A.N., Kailath, T., *Stabilizing state-feedback design via the moving horizon method*, Int. J. of Control, Vol. 37, 1983.
- [Lambert 87] Lambert, E.P., *Process control applications of long-range prediction*. D.Phil thesis, Department of Engineering Science, Oxford University, 1987.
- [Lee 94] Lee, J.H., Morari, M., García, C.E., *State-space Interpretation of Model Predictive Control*, Automatica, Vol. 30, No. 4, 1994, pp. 707-717.
- [Lemos 85] Lemos, J.M., Mosca, E., *A multi-predictor based LQ-self-tuning controller*, IFAC Conference on Identification and System Parameter Estimation, York, Pergamon Press, Oxford, 1985.
- [Madrid 93] Madrid, A.P., Santos, M., Dormido, S., Morilla, F., *Constrained generalized predictive control with dynamic programming*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Martín 86] Martín, g.D., Caldwell, J.M., *Predictive Control Applications for the Petroleum Refining Industry*, Japan Petroleum institute, Petroleum Refining Conference, ToKyo, Japan, 1986.
- [Martín-Sánchez 84] Martín-Sánchez, J.M. Shah, S.L., *Multivariable adaptive predictive control of binary distillation column*, Automatica, Vol. 20, No. 5, 1984.
- [Matsko 85] Matsko, T.N., *Internal Model Control for Chemical Recovery*. Chem. Eng. Progress, 81, No. 12, pp. 46-51.
- [Mayne 90] Mayne, D.Q., Michalska, H., *Receding horizon control of nonlinear systems*, IEEE Trans. Automatic Control, Vol. 35, 1990, pp. 814-824.

- [Megías 94] Megías, D., *Control DMC de un Reactor Químico. Aspectos de Implementación*, Proyecto Final de Carrera, Departamento de Informática, UAB, 1994.
- [Mehra 80] Mehra, R.K., Rouhani, R., *Theoretical Considerations on Model Algorithmic Control for Nonminimum Phase Systems*, Proc. Joint automatic Control Conf, San Francisco, 1980.
- [Mehra 82] Mehra, R.K., Rouhani, R., Eterno, J., Richalet, J., Rault, A., *Model Algorithmic Control for Nonminimum Phase Systems*, Eng. Foundation Conference on Chemical Process Control II, Sea Island, GA, 1982, pp. 287-310
- [Mohtadi 88] Mohtadi, C., *On the role of prefiltering in parameter estimation and control*, Workshop on Adaptive Control Strategies for Industrial Use. Banff, Canadá, 1988.
- [Mohtadi 89] Mohtadi, C., *On the role of prefiltering in parameter estimation and control*, In: S. Shah, G. Dumont "Adaptive control strategies for industrial use", Springer-Verlag, Berlín, 1989, pp. 121-144.
- [Morshedi 85] Morshedi, A.M., Cutler, C.r., Skrovanek, T.A., *Optimal Solution of Dynamic Matrix Control with Linear programming Techniques (LDMC)*. Proc. American Control Conf., Boston, MA, 1985, pp. 199-208.
- [Morari 89] Morari, M., Zafrou, E., *Robust Process Control*, Prentice-Hall International, Englewood-Cliffs, N.J. 1989.
- [Moreno 94] Moreno, R., *Estimación de Estados y Control predictivo del proceso de Fangos Activados*, Tesis Doctoral, Universidad Autónoma de Barcelona, 1994.
- [Mosca 84] Mosca, E., Zappa, G., Manfredi, C., *Multistep Horizon Self-tuning controllers: the MUSMAR Approach*, 9th IFAC World Congress, Pergamon Press, Oxford, 1984, pp. 155-159.

- [Mosca 92] Mosca, E., Zhang, J., *Stable redesign of predictive control*, Automatica, Vol. 28, 1992, pp. 1229-1233.
- [Nicolao 93] Nicolao, G. de., Scattolini, R., *Stability and output terminal constraints in predictive control*, Advances in Model-Based Predictive Control. Conference proceedings, University of Oxford, Oxford, 1993.
- [Ordys 92] Ordys, A.W., Clarke, D.W., *A State-Space Description for GPC Controllers*, Report No. OUEL 1934/92, University of Oxford, Oxford, 1992.
- [Peterka 84] Peterka, V., *Predictor-Based Self-Tuning Control*, Automatica, Vol. 20, 1984, pp. 39-50.
- [Plo 94] Plo, J., *Control Predictivo No Lineal de un Manipulador de Tres Grados de Libertad*, Proyecto Final de Carrera, Departamento de Informática. UAB, 1994.
- [Prada 93] Prada, C., Serrano, J., Vega, P., Piera, M.A., *A Comparative Study of DMC and GPC Controllers*, Advances in Model-Based Predictive Control. Conference proceedings. University of Oxford, Oxford, 1993.
- [Prett 79] Prett, D.M., Gillette, R.D., *Optimization and constrained multivariable control of a catalytic cracking unit*, Proc. JACC, San Francisco, USA, 1979.
- [Propoi 63] Propoi, A.I., *use of LP Methods for Synthesizing Sampled-Data Automatic Systems*, Automation and Remote Control, No. 24, 1963.
- [Rake 87] Rake, H., Hoffmann, U., Bredderman, R., *Self-tuning on-off Control*, In M. Singh, Encyclopedia of Systems and Control, Vol. 6. pp. 4209-4211.
- [Rake 90] Rake, H., *Model Based Predictive On-Off Control*, ESPRIT-CIM Workshop "Computer integrated Design of Controlled Industrial Systems", 1990.

- [Richalet 78] Richalet, J., Rault, A., Testud, J.L., Papon, J., *Model Predictive Heuristic Control: Applications to Industrial Processes*, Automatica, Vol. 14, 1978, pp. 413-428.
- [Richalet 93] Richalet, J., *Industrial Applications of Model Based Predictive Control*, Automatica, Vol. 29, No. 5, 1993, pp. 1251-1274.
- [Robinson 91] Robinson, B.D., Clarke, D.W., *Robustness effects of a prefilter in generalized predictive control*, IEE Proceedings-D, Vol. 138, pp. 1-8.
- [Rouhani 82] Rouhani, R., Mehra, R.K., *Model algorithmic control (MAC); basic theoretical properties*, Automatica, Vol. 18, 1982, pp. 401-414.
- [Sachin 93] Sachin C. Patwardhan, Madhavan, K.P., *Nonlinear Model Predictive Control Using Second-Order Model Approximation*, Ind. Eng. Chem. Res., 1993, 32, pp. 334-344.
- [Serrano 94] Serrano, J., Megías, D., Prada, C., *Control DMC de un Reactor Químico: Aspectos de Implementación*, Seminario de Automática y Electrónica Industrial. Universidad Rovira Virgili, Tarragona, 1994.
- [Sommer 93] Sommer, St., *Model-based predictive control methods based on non-linear and bilinear parametric system descriptions*, Advances in Model-Based Predictive Control, Conference proceedings, University of Oxford, Oxford, 1993.
- [Tan 93] Tan, Y., Keyser, R., *Neural network based adaptive predictive control*, Advances in Model-Based Predictive Control, Conference proceedings. University of Oxford, Oxford, 1993.
- [Tae-Woong 93] Tae-Woong Yoon, Clarke, D.W., *Prefiltering in Receding-Horizon Predictive Control*, Report No. OUEL 1995/93, University of Oxford, Oxford, 1993.
- [Tsang 88] Tsang, T.C.T., Clarke, D.W., *Generalized predictive control with input constraints*, IEE Proceedings-D, Vol. 135, pp. 451-460.

- [Vilanova 94] Vilanova, R., *Aplicació del Control Robust a processos de tipus biològic Control  $H^\infty$  del procés de Fangs Activats*, Tesina. Dept. Informàtica, UAB, Bellaterra, Mayo 1994.
- [Warwick 88] *Implementation of self-tuning controllers*, Peter Peregrinus Ltd., London, U.K. 1988.
- [Ydstie 84] Ydstie, B.E., *Extended Horizon Adaptive Control*, 9th World Congress of IFAC, Budapest, 1984.
- [Zadeh 62] Zadeh, L.A., Whalen, B.H., *On Optimal Control and Linear Programming*, IRE Trans. Autom. Control, 7, 1962, No. 4.



Universitat Autònoma de Barcelona

Servei de Biblioteques

Reg. 1500489337

Sig. T 405/3128

Ref. 12500



