

CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS
DEL
INSTITUTO POLITÉCNICO NACIONAL

DEPARTAMENTO DE CONTROL AUTOMÁTICO

**Identificación y Control de Procesos Químicos
con Múltiples Redes Neuronales**

TESIS QUE PRESENTA

Francisco Javier Pineda Ruelas*

PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS

EN LA ESPECIALIDAD DE

INGENIERÍA ELÉCTRICA

DIRECTOR DE TESIS:

Dr. Wen Yu Liu

México, D.F., 15 de Noviembre del 2000.

* Becario del CONACyT

Contenido

1	Introducción	11
1.1	Redes neuronales en procesos químicos	12
1.2	Enfoque multimodelo	13
1.3	Motivación	15
1.4	Definición del problema	15
1.5	Estructura del trabajo	17
2	Combinación de multimodelo y redes neuronales artificiales	19
2.1	Redes neuronales artificiales RNA's	19
2.1.1	RNA-Estáticas.	20
2.1.2	RNA-Dinámicas.	22
2.2	Enfoque multimodelo	27
2.2.1	Conceptos básicos	28
2.2.2	Identificación con un enfoque multimodelo	29
2.2.3	Control con un enfoque multimodelo	37
2.3	Selección de modelos	42
2.3.1	Algoritmo de selección estocástico	42
2.3.2	Algoritmo de selección determinístico	45
2.3.3	Algoritmo de selección con histéresis	46
2.4	Combinación del enfoque multimodelo y RNAD's	46
2.4.1	Identificación multimodelo con RNA dinámicas	46
2.4.2	Control multimodelo con RNA dinámicas	47

3	Identificación con un enfoque multimodelo neuronal	49
3.1	Identificación con una red neuronal dinámica	50
3.1.1	Análisis de estabilidad	52
3.2	Identificación con múltiples redes neuronales	61
3.2.1	Análisis de estabilidad del identificador multimodelo	64
4	Control con un enfoque multimodelo neuronal	71
4.1	Control con una red neuronal	71
4.1.1	Introducción al control neuronal	73
4.1.2	Control adaptable con redes neuronales dinámicas	76
4.2	Control con un enfoque multimodelo neuronal	89
4.2.1	Identificador neuronal y controlador multimodelo neuronal	90
4.2.2	Identificador multimodelo y un controlador neuronal	93
4.2.3	Identificador multimodelo y controlador multimodelo	94
5	Identificación y control de un proceso de neutralización de pH con múltiples RNA's dinámicas	100
5.1	Proceso de Neutralización de pH	103
5.2	Control de pH	110
5.2.1	Control PID de pH	111
5.2.2	Control PID no lineal	115
5.3	Identificación con un multimodelo neuronal dinámico	120
5.3.1	Identificación con un modelo neuronal	121
5.3.2	Identificación con un multimodelo neuronal	122
5.4	Control multimodelo neuronal de pH	124
5.4.1	Control de pH con un modelo neuronal dinámico	124
5.4.2	Control de pH con un multimodelo neuronal dinámico	129
5.4.3	Datos de simulación	132
6	Identificación y control de un proceso de fermentación con múltiples RNA's dinámicas	138

6.1	Dinámica de reactores biológicos	140
6.2	Identificación con un multimodelo neuronal dinámico	150
6.2.1	Identificación con un modelo neuronal	150
6.2.2	Identificación con un multimodelo neuronal	152
6.3	Control con un multimodelo neuronal dinámico	153
6.3.1	Control con un modelo neuronal	153
6.3.2	Control con un multimodelo neuronal	156
7	Conclusiones	164

Indice de Figuras

1-1	Un control y un modelo para todo el espacio de operación	15
1-2	Control multimodelo y selector para diferentes regiones de operación	16
2-1	Modelo de una neurona	20
2-2	Funciones de activación no lineales	21
2-3	Red neuronal estática multicapa	21
2-4	Modelo aditivo de una neurona	23
2-5	Red neuronal dinámica en su forma matricial	25
2-6	Red de Hopfield	26
2-7	RNA-dinámica tipo paralelo	27
2-8	Mecanismo de selección de modelos	30
2-9	Mapa auto organizado de Kohonen	34
2-10	Estructura multimodelo con expertos y selector	35
2-11	Esquema de identificación multimodelo con selector	36
2-12	Fases de localización y control en un control multimodelo	38
2-13	Esquema de control multimodelo	39
2-14	Ambientes S_i en el espacio de operación S	39
2-15	Esquema de un control neuronal multimodelo	41
2-16	Entrenamiento de una red neuronal controlador	42
2-17	Modelo clásico de un automata	43
2-18	Autómata interactuado con múltiples ambientes	44
2-19	Identificación multimodelo con redes neuronales dinámicas	47
2-20	Control multimodelo con redes neuronales dinámicas	48

3-1	Identificación de la dinámica de la planta	51
3-2	RNA-dinámica paralela	52
3-3	Identificación con un enfoque robusto	61
3-4	Identificación con un enfoque pasivo	62
3-5	Identificador multimodelo neuronal dinámico	62
3-6	Algoritmo de selección con histéresis	66
4-1	Control convencional	72
4-2	Estructura de control con modelo de referencia	75
4-3	Estructura de control con modelo interno	75
4-4	Estructura de control con una red neuronal dinámica	76
4-5	Red neuronal dinámica de una capa	77
4-6	Múltiples estimadores neuronales	90
4-7	Controlador con un identificador multimodelo neuronal	97
4-8	Generador de índices de desempeño	97
4-9	Lógica de selección	98
4-10	Un Identificador y un controlador multimodelo	98
4-11	Multiples identificadores y un control	99
4-12	Identificación multimodelo y de control multimodelo.	99
5-1	Curva típica del pH en forma de S	101
5-2	Curva típica de una base neutralizada por un ácido	104
5-3	Tanque de neutralización	108
5-4	Control de pH con un PID retroalimentado.	112
5-5	pH en la corriente de salida Q_4	112
5-6	Señal de control Q_3	113
5-7	Respuesta ante una perturbación en el pH de la mezcla.	114
5-8	Señal de control	115
5-9	Respuesta ante perturbaciones en el punto de ajuste	116
5-10	Señal de control	117
5-11	Perturbación en el flujo de alimentación Q_1	118

5-12	Respuesta ante perturbaciones en el flujo de alimentación Q_1	119
5-13	Señal de control	120
5-14	Perturbación en la concentración del flujo de alimentación Q_1	121
5-15	Respuesta ante perturbaciones en la concentración de Q_1	122
5-16	Señal de control	123
5-17	Relación del error entrada/salida de un PID no lineal	124
5-18	Estación de relación dentro del lazo de control	124
5-19	Estación de relación fuera del lazo de control	125
5-20	Estrategia de control anticipativo retroalimentado de pH	125
5-21	Comparación entre un control PID y un control anticipativo retroalimentado de pH	126
5-22	Control anticipativo retroalimentado ante perturbaciones en Q_1	127
5-23	Control anticipativo retroalimentado ante perturbaciones en la concentración de Q_1	128
5-24	Identificación con una red neuronal	129
5-25	Identificación con múltiples redes neuronales	130
5-26	Selección de modelos	131
5-27	Error de identificación con múltiples redes neuronales	132
5-28	Error de identificación una red neuronal	133
5-29	Estructura de control con un modelo neuronal	134
5-30	Respuestas con la red neuronal 1 y con la red neuronal 2	135
5-31	Estrucura de control con múltiples redes neuronales	135
5-32	Respuesta con un control multimodelo neuronal	136
5-33	Error de los modelos 1 y 2	136
5-34	Selector	137
6-1	Relación de la biotecnología con otras disciplinas	139
6-2	Fermentación y germinación de una célula de levadura	140
6-3	Reactor biológico en operación continua	143
6-4	Reactor biológico en operación por lotes con alimentación	143

6-5	Identificación con una red neuronal	151
6-6	Identificación con múltiples redes neuronales	153
6-7	Selección de modelos	154
6-8	Error de identificación en la concentración de microorganismos x_1	155
6-9	Error de identificación en la concentración de sustrato x_2	156
6-10	Control en 2 fases de un reactor biológico	157
6-11	Control con una red neuronal	158
6-12	Concentración de penicilina con una red neuronal	159
6-13	Concentración de sustrato en el reactor	160
6-14	Error de identificación con una red neuronal.	161
6-15	Comparación en la concentración de microorganismos	161
6-16	Comparación de la concentración de penicilina	162
6-17	Errores de identificación con un modelo y con un multimodelo neuronal	162
6-18	Selección de modelos	163
6-19	Errores de identificación multimodelo y de las redes neuronales 1 y 2.	163

Notación

$:=$ Significa: igual por definición

$x_t \in R^n$ Vector de estado de la planta

y_t Vector de salida de la planta

$u_t \in R^m$ Vector de entrada a la planta

$f(x_t, u_t, p)$ Función no lineal que representa la dinámica de la planta

$\hat{x}_t \in R^n$ Vector de estado de la red neuronal

\hat{y}_t Vector de salida de la red neuronal

$W_{1,t} \in R^{n \times n}$ Es la matriz de pesos retroalimentación de estado no lineal

$W_{2,t} \in R^{n \times n}$ Es la matriz de pesos de entrada

W_1^* y W_2^* Son valores fijos de las matrices de pesos $W_{1,t}$ y $W_{2,t}$

\bar{W}_1 y \bar{W}_2 Son cotas superiores de las matrices de pesos $W_{1,t}$ y $W_{2,t}$

\bar{W}_1 y \bar{W}_2 Son los estimados del error de las matrices de pesos $W_{1,t}$ y $W_{2,t}$

$A \in R^{n \times n}$ Es una matriz Hurwitz

$P = P^T < 0$ Es la solución de la ecuación de Riccati

$\sigma(\hat{x}_t) \in R^n$ Es una función vectorial cuyos elementos son funciones sigmoideas

$\phi(\cdot) \in R^{n \times n}$ Es una matriz diagonal $n \times n$

I_i Denota al i -ésimo modelo de la planta

$J_i(t)$ Denota el índice de desempeño del i -ésimo modelo de identificación

S_i Es la i -ésima región de operación de la planta

Δ Denota al error de identificación

Δ^* Denota al error de seguimiento

x^* Es la trayectoria deseada de la planta, solución del modelo de referencia.

$S(x_t) : R^n \rightarrow R^1$ Es la función de almacenamiento

\tilde{f} Es el error de modelado

\bar{d} Denota la cota superior de la dinámica no modelada

$\|\cdot\|$ Denota la norma euclidiana

σ Es el índice de selección del modelo de identificación

h Es la constante de histéresis

Capítulo 1

Introducción

El proceso de modernización tecnológica que ha experimentado la industria en los últimos años mejora las posibilidades de aplicar estrategias de control más sofisticadas, las cuales hace apenas 10 años eran difícil de implementar debido al estado de la tecnología de cómputo con que se contaba en ese tiempo [8]. Por otro lado, el diseño de plantas más avanzadas, el desarrollo de tecnología de información más eficiente y la imposición de restricciones ambientales y económicas más exigentes provocan que la ingeniería se enfrente a problemas de control y de modelado de procesos de mayor complejidad. De aquí que cada vez sea más común tener que controlar plantas altamente no lineales, que requieren del diseño de controladores que puedan operar con un buen desempeño en amplio intervalo del espacio de operación [28].

Si ya el problema de control de procesos químicos no lineales es una tarea interesante, el problema se dificulta cuando la planta experimenta cambios inducidos por variaciones en las condiciones de operación. En éste trabajo se propone un esquema de identificación y control el cual se aplica a dos procesos químicos: un proceso de neutralización de pH y proceso de fermentación. El primero presenta un comportamiento altamente no lineal debido a la naturaleza logarítmica y a la muy alta ganancia que lo caracteriza; y el segundo, presenta una dinámica compleja que experimenta variaciones cuando las condiciones de operación cambian.

En el caso de tener controladores convencionales, inicialmente sintonizados para operar bajo determinadas condiciones de operación, estos pueden experimentar un decremento en

su desempeño cuando cambia la región de operación de la planta. Si se cuenta con un modelo de identificación, este podría ser incapaz de representar la dinámica de la planta en alguna nueva región de operación.

La idea del enfoque multimodelos es sencilla: si el modelo de la planta cambia dependiendo de la región del espacio de operación en que se encuentre, entonces podemos pensar en dividir el espacio de operación en subespacios o regiones de operación (las cuales pueden representar diferentes condiciones de operación), y calcular la señal de control en función de la región de operación en que se encuentre operando la planta. De aquí la importancia de contar con múltiples modelos para representar la dinámica de la planta con la mayor exactitud posible.

1.1 Redes neuronales en procesos químicos

En la industria existen procesos químicos altamente no lineales y con dinámicas variantes en el tiempo debido a la complejidad de las reacciones químicas internas. Estas características hacen que sea muy difícil y costoso obtener un modelo analítico del proceso que sea razonablemente exacto en un intervalo de operación suficientemente grande. En el caso de contar con un modelo matemático, éste puede presentar errores de modelado debido a la presencia de dinámicas no modeladas, cambios en las condiciones de operación o fallas en el sistema que induzcan variaciones en los parámetros o en la estructura del modelo de la planta. En los últimos años las redes neuronales artificiales (RNAs) se han convertido en una herramienta importante en el modelado de procesos, debido a su gran capacidad de caracterizar funciones no lineales utilizando sólo la entrada y la salida de la planta. De aquí que esta herramienta proporcione una técnica de modelado de procesos confiable y a un bajo costo.

En algunas aplicaciones industriales se han obtenido modelos con RNA's (modelos neuronales), para proporcionar estimados de la concentración de biomasa en procesos de fermentación industrial, y para estimar la concentración del producto en columnas de destilación [74]. La aplicación de modelos neuronales para la estimación en línea de variables de proceso se considera en [42]. En algunas situaciones los modelos neuronales pueden ofrecer ventajas significativas; por ejemplo, si el modelo neuronal es lo suficientemente exacto éste podría utilizarse, al menos en teoría, en lugar de un analizador. De esta manera se pueden

obtener mediciones más frecuentes de las que actualmente se obtienen con instrumentación convencional. Además, el incremento en la frecuencia de las mediciones puede hacer posible la implementación de un control retroalimentado. Otra ventaja radica en la posibilidad de utilizar los modelos neuronales directamente en estrategias de control que se basan en el conocimiento del modelo de la planta [25].

La Ingeniería de control de procesos, la simulación, la supervisión, y la detección de fallas, son otras tareas que se basan en el conocimiento del "modelo" del proceso en cuestión. Este modelo puede construirse a partir de la experiencia directa con el proceso (modelo cualitativo), puede cuantificarse en términos de un modelo analítico de la planta, o a través de una combinación de ambos. Aunque las técnicas existentes para abordar estos problemas con este tipo de modelos pueden proporcionar soluciones aceptables, existen muchas situaciones en donde pueden fallar debido a la presencia de dinámicas no modeladas y a las no linealidades intrínsecas a muchos procesos químicos.

Por otro lado, uno de los principales obstáculos en el uso de técnicas de control y modelado avanzadas se tiene en el costo que implica el desarrollo y validación del modelo. Por esta razón la aplicación de RNA's puede ser de gran utilidad en el modelado de procesos no lineales de una manera confiable, a un bajo costo, y a partir de conocer solamente la entrada y salida de la planta[10] [11].

1.2 Enfoque multimodelo

Los primeros trabajos que utilizaron el enfoque multimodelo en aplicaciones de control se orientaron al estudio de sistemas de control lineal adaptables con la finalidad de mejorar la respuesta transitoria [49] [50] [51]. En el área de sistemas no lineales los esfuerzos se dirigieron a la identificación y control de sistemas dinámicos utilizando RNA's bajo la consideración de que sólo se tiene acceso a la entrada y a la salida de la planta [52] [35] [36]. Sin embargo, pronto se dieron cuenta que el enfoque multimodelo podría desarrollar todo su potencial cuando el sistema opera en el dominio no lineal.

Uno de los primeros intentos de aplicación se tiene en el control una nave espacial en donde se utilizaron múltiples controladores neuronales [53]. Los resultados en sistemas lineales se

extendieron al problema de control de un robot manipulador, en donde los trabajos de simulación y de control en tiempo real revelaron que podría alcanzarse un mejor desempeño utilizando el enfoque multimodelo [48]. Por otro lado, [34] [4] se han propuesto múltiples filtros de Kalman en problemas de control para mejorar la exactitud de la estimación de estado. Kaufman y colaboradores aplicaron este enfoque a problemas de control médico, mientras que Maybeck y Pogoda lo utilizaron en detección de fallas y control de naves espaciales. Bar y Shalom adaptaron este enfoque a problemas de control de tráfico aéreo.

El problema de selección de modelos asociado al enfoque multimodelo ha tomado importancia en el contexto de estabilización adaptiva y se estudió por primera vez en [39] con la finalidad de relajar las condiciones bajo las cuales sistemas lineales desconocidos podrían estabilizarse. En [45] se proponen algoritmos de selección con histéresis para evitar un número infinito de conmutaciones. También existen combinaciones aditivas de varias redes neuronales [70] [71], y propuestas de estructuras multimodelo aplicadas a modelos difusos [72] [23] [58].

Si se utiliza un control adaptable indirecto con sólo un modelo de identificación, cuando se presente un cambio en la región de operación de la planta el modelo deberá autoadaptarse para intentar ajustarse a las nuevas condiciones de operación antes de que se tome alguna acción de control. En sistemas lineales esta adaptación es posible, sin embargo, la lentitud de la adaptación puede resultar en grandes errores transitorios. En sistemas no lineales un sólo modelo de identificación puede no ser suficiente para identificar las diferentes regiones de operación por las que transite la dinámica de la planta, de aquí que un control basado en sólo un modelo de identificación tenga serias dificultades para controlar la planta. La figura 1-1 ilustra la dificultad de tener sólo un controlador y un modelo de identificación para todo el espacio de operación de la planta.

Por lo tanto, una estrategia de control que se base en múltiples modelos de identificación junto con un algoritmo de selección es una opción interesante, para contrarrestar adecuadamente cambios en la dinámica de la planta. La figura 1-2 intenta ilustrar la estrategia "divide y vencerás" en el control de un proceso que experimenta cambios en la región de operación.

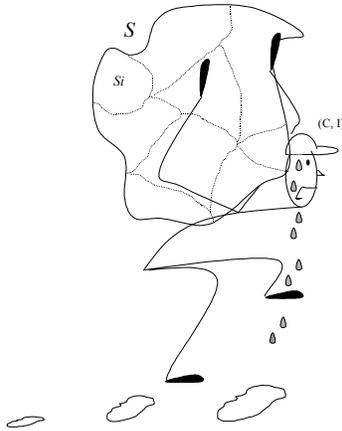


Figura 1-1: Un control y un modelo para todo el espacio de operación

1.3 Motivación

La característica no lineal, la presencia de dinámicas no modeladas y la operación en diferentes regiones de operación que experimentan algunos procesos químicos son problemas importantes que vistos en su conjunto han motivado a buscar esquemas de identificación y control que ayuden a contrarrestar estas dificultades. Esta es la motivación que nos ha llevado a combinar el enfoque multimodelo y las redes neuronales artificiales dinámicas (RNAD's), con la finalidad de construir un controlador multimodelo neuronal que permita mejorar el desempeño de procesos químicos no lineales que experimenten variaciones en las condiciones de operación.

El controlador multimodelo neuronal propuesto se prueba en dos procesos altamente no lineales: un proceso de fermentación y un proceso de neutralización de pH . Las dificultades que presentan estos procesos químicos representan un buen ejemplo para evaluar el desempeño del esquema de control propuesto.

1.4 Definición del problema

Un proceso químico que experimenta cambios en su región de operación que afectan la dinámica de la planta, puede verse como una planta cuyo modelo cambia cada vez su región de operación cambia.

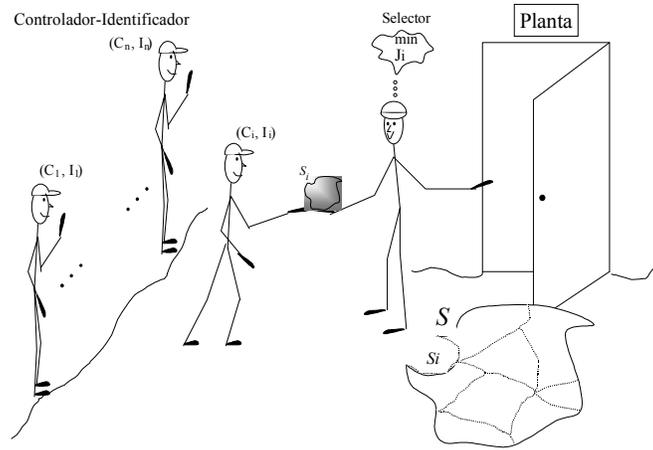


Figura 1-2: Control multimodelo y selector para diferentes regiones de operación

Considere el sistema descrito por un conjunto de ecuaciones diferenciales de la forma

$$\begin{aligned} \dot{x}_t &= f(x_t, u_t, p) \\ y_t &= x_t \end{aligned} \tag{1.1}$$

Estas ecuaciones contienen tanto a la planta como al ambiente externo.

Observación 1.1 *En el presente trabajo el interés principal es evaluar el desempeño de la técnica multimodelo en la identificación y el control de plantas altamente no lineales, por lo cuál se considera que el estado de la planta esta disponible.*

Cuando cambios en las condiciones de operación llevan al sistema a una región de operación diferente, entonces la planta (1.1) puede experimentar cambios ya sea en sus parámetros, si p cambia; o bien en su estructura, si f y h cambian. En sistemas no lineales, en donde diferentes ambientes o regiones de operación se describen por diferentes funciones f y h en (1.1), un sólo modelo puede ser insuficiente para representar los cambios que se presenten en el comportamiento del sistema.

En la definición del problema considere que la planta esta operando en una región de operación S_i del espacio de operación S , y que el identificador I_i asociado al controlador C_i ,

es capaz de controlar a la planta que esta operando en el ambiente S_i . La planta puede caer en cualquier región del espacio de operación S , y sólo se dispone de un número finito de identificadores. El objetivo es tener una estructura de identificación y control que permita modelar y controlar la planta independientemente de la región de operación S_i , utilizando un número finito de parejas (I_i, C_i) $i = 1, \dots, \sigma$.

1.5 Estructura del trabajo

En el capítulo 1 se plantea la necesidad de nuevas estructuras de identificación y control que permitan enfrentar el problema de modelado y de control de procesos químicos no lineales que experimenten variaciones en su dinámica ocasionadas por cambios en las condiciones de operación o en la región de operación de la planta. La hipótesis de partida propone combinar el enfoque multimodelo y las redes neuronales dinámicas para construir una nueva estructura que ayude a enfrentar el problema de identificación y control de procesos químicos complejos.

En el capítulo 2 se revisan los conceptos básicos de las RNA's y los diferentes enfoques que existen en la aplicación de multimodelos en tareas de identificación y control. Esta parte concluye con una primer estructura de un identificador y de un controlador multimodelo utilizando redes neuronales dinámicas.

El capítulo 3 tiene como objetivo estudiar la estructura del identificador multimodelo propuesto, para lo cuál, primero es necesario entender el proceso de identificación utilizando una red neuronal dinámica y su respectivo análisis de estabilidad. Luego se introduce el nuevo identificador multimodelo que utiliza redes neuronales dinámicas como modelos de identificación. Debido a que en este enfoque se tienen múltiples modelos de identificación y un algoritmo de selección, es necesario estudiar y asegurar la estabilidad del esquema de identificación multimodelo. Este capítulo concluye aportando una prueba de estabilidad del identificador multimodelo propuesto cuando se utiliza un algoritmo de selección con histéresis.

En el capítulo 4 inicia con una introducción al control neuronal y luego se proponen nuevas estructuras de control neuronal multimodelo, en donde el enfoque multimodelo se puede aplicar al estimador de la dinámica de la planta, al controlador neuronal, o a ambos.

Una de estas estructuras de control multimodelo neuronal se evalúa en dos procesos químicos altamente no lineales en los siguientes dos capítulos, y las estructuras de control restantes se evaluarán en trabajos futuros. La aplicación del controlador multimodelo neuronal propuesto resultó en un trabajo que se envió a evaluación en el *Journal of Process Control* [57].

Los capítulos 5 y 6 presentan la aplicación de los esquemas de identificación y control a dos procesos químicos altamente no lineales: un proceso de neutralización de pH y un proceso de fermentación. Las dificultades que presentan estos procesos los hacen buenos candidatos para evaluar las estructuras de identificación y control propuestas. Los resultados de aplicar el identificador multimodelo neuronal a estos procesos se presentaron en el CIE2000 [56], y la inclusión del estudio de estabilidad del identificador multimodelo produjo dos artículos más [54] [55].

El capítulo 5 inicia con una serie de conceptos que ayudan a entender los procesos de neutralización y las dificultades que existen en su control. También se presenta la simulación de dos estrategias clásicas para el control de pH propuestas por Shinskey. En la prueba del identificador multimodelo se comparan los errores de identificación de un identificador neuronal y del identificador multimodelo neuronal propuesto. Se concluye aplicando el nuevo esquema de control multimodelo neuronal al proceso de neutralización.

El capítulo 6 también inicia introduciendo en el conocimiento de los complejos procesos de fermentación. Luego se aplica el nuevo identificador multimodelo neuronal en un proceso de fermentación continua, en donde los resultados muestran una mejor aproximación de la dinámica de la planta con el identificador multimodelo propuesto. El capítulo concluye aplicando el controlador multimodelo neuronal a un proceso de fermentación para la producción de penicilina, en donde la concentración final de penicilina se toma como una manera sencilla de medir el desempeño de una estrategia de control.

En estos dos últimos capítulos se presentan pruebas de simulación. En el caso del pH se observa la superioridad del control neuronal respecto de las estrategias de control convencionales. Para los dos procesos químicos mencionados se obtiene un mejor desempeño del controlador multimodelo neuronal respecto del control neuronal adaptable indirecto que utiliza sólo un modelo de identificación.

En el capítulo 7 se presentan las conclusiones de este trabajo.

Capítulo 2

Combinación de multimodelo y redes neuronales artificiales

2.1 Redes neuronales artificiales RNA's

Las RNA's pueden clasificarse como redes neuronales estáticas (RNE's), o como redes neuronales dinámicas RND's. La principal desventaja de las RNE's es que la actualización de los pesos no utiliza información de la estructura de datos local y que la aproximación de la función es sensible a los datos de entrenamiento [24]. Por otro lado, las RND's además de tener una representación más sencilla, pueden superar este inconveniente y tienen la ventaja de presentar un buen comportamiento cuando se tienen dinámicas no modeladas gracias a que su estructura incorpora retroalimentación.

Utilizando técnicas tradicionales de identificación es posible obtener un error de aproximación arbitrariamente pequeño para una amplia clase de funciones no lineales, sin embargo, no es posible garantizar la estabilidad del error de identificación [24]. Como se muestra en [32] [33] [65], algunos métodos parecidos al método de Lyapunov son una buena herramienta para obtener la ley de adaptación y para establecer las condiciones de estabilidad del error.

Los modelos que se obtienen con RNA's pueden utilizarse en lugar de los modelos convencionales. La exactitud con que un modelo neuronal puede representar a un proceso puede cambiar modificando la estructura de la red neuronal o bien sus parámetros internos. La

capacidad de representar procesos no lineales esta dada por la estructura de la red neuronal, pero principalmente por las funciones de procesamiento no lineales que se incluyen en su estructura y que se conocen como funciones de activación.

2.1.1 RNA-Estáticas.

En una RNA-estática los nodos en las diferentes capas de la red neuronal representan elementos de procesamiento a los que se les da el nombre de neuronas. El modelo de neurona más utilizado se basa en el modelo propuesto por McCulloch y Pitts en 1943. En este modelo cada señal de entrada x_1, \dots, x_n , se pondera por los pesos W_1, \dots, W_n . Una señal de ajuste en el nodo se representa por una entrada adicional constante con valor de 1, la cual se pondera por el peso W_0 . La salida y se obtiene sumando las entradas ponderadas a la neurona y evaluando el resultado de esta suma en una función de activación no lineal $f(\cdot)$. La figura 2-1 muestra el modelo neuronal básico de una neurona.

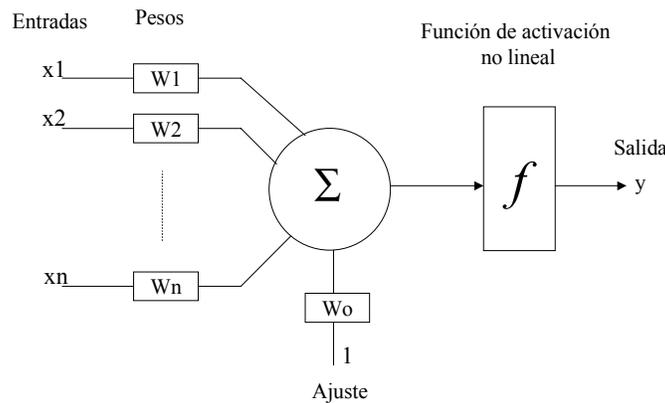


Figura 2-1: Modelo de una neurona

La función de activación no lineal f , comúnmente es alguna de las funciones que se muestran en la figura 2-2.

Normalmente se tiene una capa de entrada y una capa de salida, y el número de neuronas en estas capas depende del número de entradas y de salidas que se consideren. En las capas ocultas el número de neuronas puede variar en cada capa dependiendo de como se especifique

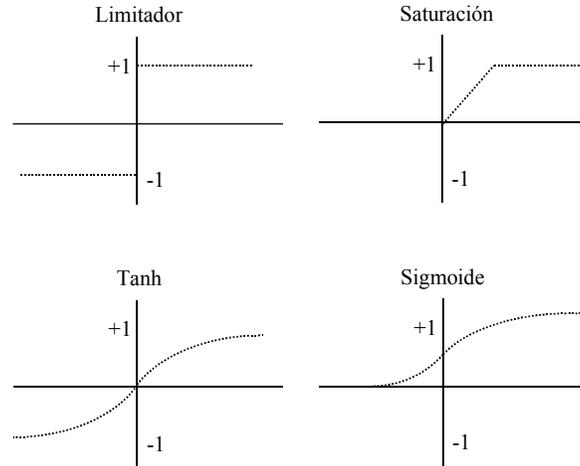


Figura 2-2: Funciones de activación no lineales

la red neuronal. La estructura de las capas ocultas es la que define en esencia a las RNA-estáticas.

Las neuronas de la capa de entrada no realizan funciones de procesamiento de datos, pues sólo son un medio por el cuál los datos se introducen a la red neuronal. Una vez que se introducen los datos éstos se propagan a través de las conexiones pasando por las capas ocultas hasta llegar a la capa de salida.

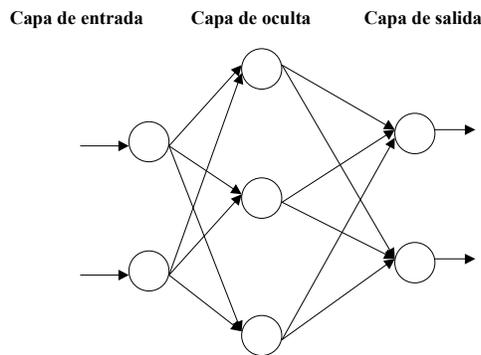


Figura 2-3: Red neuronal estática multicapa

De esta manera, con excepción de las neuronas de la capa de entrada, la entrada a cada neurona es una suma ponderada de las salidas de las neuronas de la capa anterior. Por ejemplo, si la información que proviene de la i -ésima neurona que pertenece a la capa $j - 1$

se denota como $I_{j-1,i}$, y entra a la k -ésima neurona que pertenece a la capa j , entonces la entrada total a la k -ésima neurona de la capa j es

$$\alpha_{j,k} = d_{j,k} + \sum_{i=1}^n w_{j-1,i,k} I_{j-1,i}$$

en donde $d_{j,k}$ es el ajuste que se conoce como umbral y que esta asociado a cada interconexión. La salida de cada nodo se obtiene al aplicar una función no lineal o función de activación (figura 2-2), a la entrada total $\alpha_{j,k}$ de la k -ésima neurona. Esta función no lineal normalmente es una función sigmoideal, sin embargo, es posible utilizar cualquier función cuya derivada sea acotada. Es interesante mencionar que el comportamiento que describe la función no lineal sigmoideal también se observa en el comportamiento de las neuronas del ser humano.

2.1.2 RNA-Dinámicas.

La introducción de retroalimentación en la estructura de la red neuronal produce una red dinámica con varios estados de equilibrio. La ecuación general de este tipo de redes es la siguiente

$$\dot{x} = f(x_t, u_t, \theta) \tag{2.1}$$

En donde x representa el estado de la red neuronal, u las entradas externas, y θ los parámetros de la red. f es una función que representa la estructura de la red.

Modelo dinámico aditivo

Considere la neurona que se muestra en la figura 2-4.

En términos físicos los pesos sinápticos $w_{j,1}, w_{j,2}, \dots, w_{j,N}$ representan conductancias y las N entradas respectivas $x_1(t), x_2(t), \dots, x_N(t)$ potenciales. Las entradas se aplican a un punto suma de corrientes.

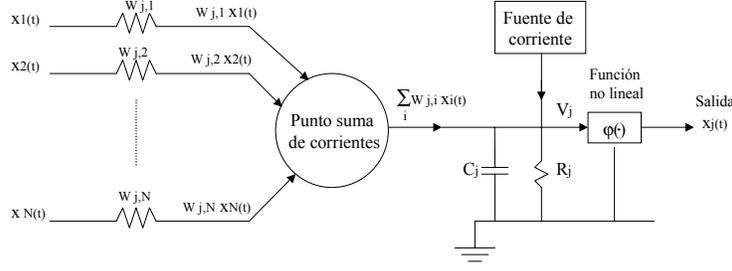


Figura 2-4: Modelo aditivo de una neurona

La corriente total I_{Ti} que fluye hacia el nodo de entrada de la función no lineal $\varphi(\cdot)$ es

$$I_{Ti} = \sum_{i=1}^n w_{j,i} x_i(t) + I_j$$

en donde el primer término de la sumatoria se debe a los estímulos $x_1(t), x_2(t), \dots, x_N(t)$, que actúan sobre los pesos sinápticos $w_{j,1}, w_{j,2}, \dots, w_{j,N}$ respectivamente, y el segundo término proviene de la fuente de corriente I_j que representa el ajuste externo.

Si $v_j(t)$ es el campo inducido en la entrada de la función de activación no lineal $\varphi(\cdot)$, entonces la corriente total que se aleja I_{To} del nodo de entrada es

$$I_{To} = \frac{v_j(t)}{R_j} + C_j \frac{dv_j(t)}{dt}$$

en donde el primer término se debe a la resistencia de fuga R_j , y el segundo a la capacitancia de fuga C_j . Aplicando la ley de corrientes de Kirchhoff al nodo de entrada de la función $\varphi(\cdot)$ en la figura 2-4, se tiene

$$C_j \frac{dv_j(t)}{dt} + \frac{v_j(t)}{R_j} = \sum_{i=1}^n w_{j,i} x_i(t) + I_j \quad (2.2)$$

El término de capacitancia $C_j \frac{dv_j(t)}{dt}$ en la ecuación (2.2) es la forma más sencilla de incluir dinámica al modelo de una neurona. Dado el campo inducido $v_j(t)$, es posible determinar la

salida de la neurona j utilizando la relación no lineal $\varphi(\cdot)$

$$x_j(t) = \varphi(v_j(t)) \quad (2.3)$$

El modelo RC que describe la ecuación (2.2) se conoce como modelo aditivo y constituye la base de las neuronas dinámicas clásicas. Una característica de este modelo es que la señal $x_i(t)$ que se aplica a la neurona j es una función del tiempo que varía lentamente.

Ahora considere una red compuesta de la interconexión de N neuronas, cada una de las cuales tiene el mismo modelo matemático que describen las ecuaciones (2.2) (2.3)

$$C_j \frac{dv_j(t)}{dt} = -\frac{v_j(t)}{R_j} + \sum_{i=1}^n w_{j,i} x_i(t) + I_j, \quad j = 1, 2, \dots, N \quad (2.4)$$

la cual tiene la misma forma que la ecuación general (2.1).

Considerando que la constante de tiempo $\tau_j = R_j C_j$ es la misma para todo j , normalizando el tiempo t con respecto a esta constante de tiempo, y normalizando los pesos $w_{i,j}$ e I_j con respecto a R_j [24], la ecuación anterior puede escribirse como:

$$\frac{dv_j(t)}{dt} = -v_j(t) + \sum_i w_{j,i} \varphi(v_i(t)) + I_j, \quad j = 1, 2, \dots, N$$

en este modelo aditivo los campos inducidos $v_1(t), v_2(t), \dots, v_N(t)$ de las neuronas individuales constituyen al vector de estado de la red neuronal. El diagrama a bloques de la forma matricial de esta red en donde se aprecia la presencia de retroalimentación de estado se muestra en la figura 2-5, en donde W es la matriz de pesos sinápticos, $v(t)$ es el vector de campos inducidos, $x_j(t)$ es el vector de salidas, y $x_i(t)$ es el vector de entrada.

Red de Hopfield

La red de Hopfield tiene dos capas: la capa de entrada y la capa de Hopfield. Cada nodo de la capa de entrada se conecta directamente a sólo un nodo de la capa de Hopfield. La capa de Hopfield consiste de un conjunto de neuronas y un conjunto de elementos de retardo que forman un sistema retroalimentado de varios lazos como se observa en la figura 2-6.

En esta red el número de lazos de retroalimentación es igual al número de neuronas. La

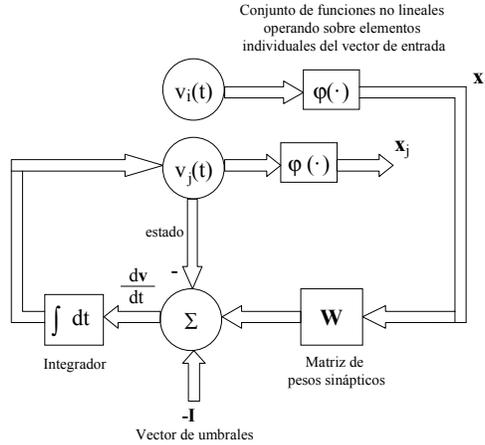


Figura 2-5: Red neuronal dinámica en su forma matricial

salida de cada neurona se pondera y retroalimenta a través de un elemento de retardo a cada una de las neuronas de la red, excepto a la misma neurona. En el estudio de la dinámica de la red de Hopfield se puede utilizar el modelo aditivo de una neurona.

Durante el entrenamiento normalmente se requiere que la salida de la red sea igual a la entrada. La intensidad de las conexiones se debilitan, decrementando los pesos correspondientes, si la salida de una neurona es diferente de la entrada, y se fortalecen en caso contrario. La red entrenada se utiliza aplicando un patrón de entrada a la red. Luego, las salidas de la red se retroalimentan a través de pesos hasta que se cumple con un criterio convergencia, comúnmente cuando no hay cambios en los nodos de salida de la red. Esta es la salida final de la red para el patrón de entrada.

La presencia de retroalimentación en esta red la hace interesante para aplicaciones de modelado y control de sistemas dinámicos [52].

Red neuronal dinámica paralela

Cosidere la siguiente RND con estructura paralelo cuya estructura se muestra en la figura 2-7.

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)\gamma(u_t) \quad (2.5)$$

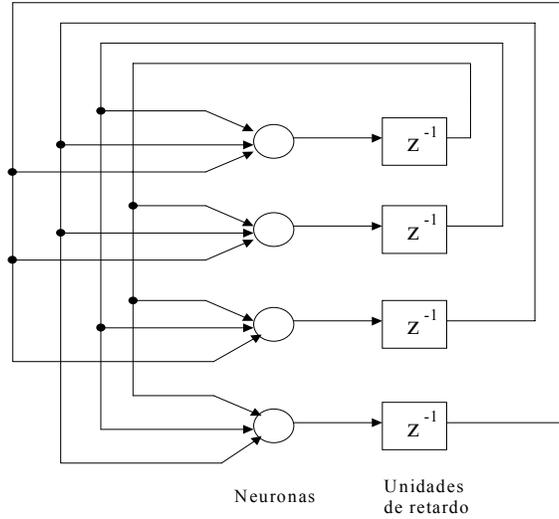


Figura 2-6: Red de Hopfield

Donde $\hat{x}_t \in R^n$ es el estado de la RN, $u_t \in R^m$ es la entrada a la planta, $W_{1,t} \in R^{n \times n}$ es la matriz de retroalimentación de estado no lineal, $W_{2,t} \in R^{n \times n}$ es la matriz de entrada, y $A \in R^{n \times n}$ es una matriz Hurwitz. La función vectorial $\sigma(\hat{x}_t) \in R^n$ se considera de dimensión n con elementos incrementando monótonicamente, $\gamma(u_t)$ es una función vectorial de dimensión n , y como $u_t \in R^m$, entonces en $\gamma(u_t)$, $n - m$ elementos son cero. La función matricial $\phi(\cdot)$ es una matriz diagonal $n \times n$, esto es

$$\phi(\hat{x}_t) = \text{diag}(\phi(\hat{x}_1), \dots, \phi(\hat{x}_n))$$

Los elementos $\sigma_i(\cdot)$ y $\phi_i(\cdot)$ representan funciones sigmoideas de la forma general

$$\sigma_i(x_i) = \frac{a_i}{(1 + \exp(-b_i x_i))} - c_i$$

en donde a_i representa el valor máximo, b_i la pendiente de la curvatura de la función sigmoideal, y c_i es una constante que desplaza verticalmente a la sigmoide.

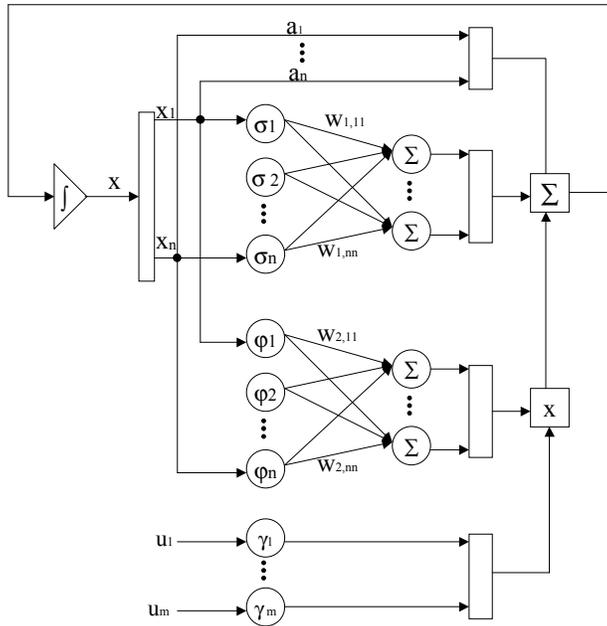


Figura 2-7: RNA-dinámica tipo paralelo

2.2 Enfoque multimodelo

El objetivo de esta sección es establecer algunos de los conceptos básicos del enfoque multimodelo y hacer una revisión de los esquemas multimodelo existentes en tareas de identificación y control. También se presentan algunos esquemas de identificación y control utilizando redes neuronales.

La técnica multimodelo se utiliza para afrontar el problema del cambio en la región de operación que experimentan muchas plantas no lineales desconocidas. El fin de esta metodología es diseñar esquemas de identificación y control con capacidad de modelar y controlar la dinámica de una planta independientemente de la región de operación en que se encuentre.

El enfoque multimodelo utiliza múltiples modelos para describir el comportamiento de un sistema que opera en distintos ambientes o regiones de operación, o bien, el comportamiento de sistemas con dinámicas muy complejas. Este enfoque inicia como un esfuerzo para mejorar el desempeño transitorio de sistemas lineales con grandes incertidumbres paramétricas y ha evolucionado gradualmente hacia el diseño de sistemas de control adaptables que en el curso

del tiempo puedan actuar con rapidez y exactitud en el control de procesos no lineales con un alto grado de incertidumbres.

2.2.1 Conceptos básicos

Modelo

Un modelo puede considerarse como una representación de las partes esenciales de un sistema en una forma conveniente y puede tener diferentes formas dependiendo del problema a resolver. El método más utilizado para representar el comportamiento de un sistema es a través de un modelo matemático. En sistemas muy complejos puede ser necesario utilizar modelos heurísticos y matemáticos bajo diferentes consideraciones para poder mejorar la representación del sistema.

Ambiente o región de operación

La dinámica de un sistema puede cambiar en distintas regiones de operación. Considere que el sistema se describe por un conjunto de ecuaciones diferenciales de la forma

$$\begin{aligned}\dot{x}_t &= f(x_t, u_t, p) \\ y_t &= x_t\end{aligned}\tag{2.6}$$

Estas ecuaciones contienen tanto a la planta como a las condiciones de operación externas. Si f y h se consideran fijos, diferentes condiciones de operación pueden expresarse con diferentes valores del vector de parámetros constante p , o bien con diferentes funciones $(f_i, h_i), i = 1, 2, \dots$. De esta manera fallas en el sistema, fallas en sensores y actuadores, perturbaciones externas y variaciones en los parámetros pueden llevar al sistema a diferentes regiones de operación.

Múltiples modelos

Muchos sistemas complejos operan en distintos ambientes. Cuando las condiciones de operación cambian las característica entrada-salida del sistema puede cambiar rápidamente o incluso discontinuamente. Si se utiliza sólo un modelo de identificación, éste tendrá que adaptarse al nuevo ambiente antes de que se tome alguna acción de control. En sistemas

lineales esta adaptación puede ser posible, sin embargo, la lentitud de la adaptación puede producir grandes errores transitorios. En sistemas no lineales en donde diferentes ambientes se describen por diferentes funciones f y h en la ecuación (2.6), un modelo puede no ser suficiente para identificar los cambios en la dinámica del sistema. De aquí la necesidad de múltiples modelos tanto para identificar como para controlar adecuadamente los diferentes ambientes que se presenten.

Sin embargo, la necesidad de múltiples modelos va más lejos ya que en algunos ambientes puede ser necesario disponer de diferentes modelos cuya exactitud dependa de la región del espacio de estado en donde caen las trayectorias del sistema.

2.2.2 Identificación con un enfoque multimodelo

Los modelos matemáticos son necesarios para el control de procesos, predicción de comportamiento, detección de fallas, estimación de variables no medibles, así como para entender mejor el comportamiento del proceso. Sin embargo, muchos procesos reales involucran una gran cantidad de variables, son no lineales, variantes en el tiempo, el conocimiento de los fenómenos físicos y químicos involucrados a menudo es incompleto, algunas variables críticas pueden no ser medibles y algunos parámetros físicos pueden ser desconocidos.

Existen métodos para la obtención de modelos a partir de datos observados. Dentro de los métodos que se han desarrollado para identificación de sistemas no lineales están el NARMAX, Hammerstein, Wiener y el Hammerstein-Wiener. Sin embargo, con estas estructuras a menudo es difícil representar el comportamiento del sistema en todo su intervalo de operación. Por lo anterior, son necesarios nuevos métodos de identificación de sistemas que permitan obtener modelos útiles en todo el intervalo de operación de la planta.

La identificación de sistemas con un enfoque multimodelo, también conocida como identificación basada en la región de operación, puede proporcionar una solución a éste problema. El principio fundamental de la identificación con un enfoque multimodelo consiste en dividir el espacio de entrada en regiones de operación y aproximar cada una de estas regiones con un modelo local que puede ser lineal o no lineal. Finalmente la salida del modelo global se obtiene interpolando o conmutando entre los modelos locales.

Un multimodelo describe una estructura que consiste de un conjunto de submodelos y de un mecanismo (figura 2-8) que se encarga de combinar las salidas de estos submodelos. Por ejemplo, si \hat{y}_i denota el mapeo del submodelo i , y la validez relativa del submodelo i se expresa por medio de una función de interpolación g_i , en donde $g_i \in [0, 1]$; entonces, si el submodelo i es muy exacto para una cierta región de operación, el valor de la función g_i será muy cercano a uno [29]. Considerando que el multimodelo consiste de N submodelos, la salida del multimodelo \hat{y} puede escribirse de la siguiente manera

$$\hat{y} = F[(\hat{y}_1, g_1), (\hat{y}_2, g_2), \dots, (\hat{y}_N, g_N)] \quad (2.7)$$

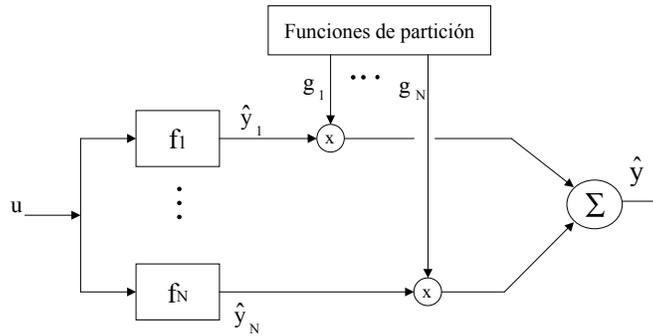


Figura 2-8: Mecanismo de selección de modelos

Ventajas del enfoque multimodelo

En [17], se plantean algunas ventajas del enfoque multimodelo con respecto a la capacidad de aproximación que se puede lograr con una red neuronal:

- Puede ser muy difícil construir un modelo global a partir de un conjunto de datos de entrada y salida del proceso
- El enfoque multimodelo permite combinar diferentes técnicas de modelado, es decir, modelos lineales y no lineales pueden integrarse en la estructura del multimodelo. Conocimiento a *priori* (que exista sólo para ciertas regiones de operación) puede utilizarse para describir el sistema en dichas regiones.

- La estructura del modelo puede interpretarse tanto cualitativamente, en términos de las regiones de operación, como cuantitativamente en términos de modelos individuales.
- Las variables de entrada relevantes para la identificación del proceso pueden variar entre regiones de operación. En el enfoque multimodelo es posible utilizar diferentes entradas en cada región de operación. Esto es una ventaja en procesos multivariados en donde distintas variables son importantes en diferentes regiones de operación. La mayoría de los procesos químicos son multivariados por naturaleza.
- La implementación de estrategias multimodelo en un modelo de control predictivo es ventajoso ya que una menor cantidad de parámetros del modelo serán relevantes en un instante de tiempo dado.

Entrenamiento de un identificador multimodelo

El entrenamiento para un identificador multimodelo que utiliza modelos fijos para representar las regiones de operación de la planta consiste de dos partes [17]:

- Primero se debe obtener (aprender) la partición desconocida del dominio de entrada. Esta es una tarea típica de las técnicas de aprendizaje no supervisado, en donde se tiene que decidir como dividir el dominio de entrada en distintas regiones de operación sin conocer la respuesta correcta, es decir, sin la ayuda de un tutor. La división del espacio de entrada también puede basarse en el conocimiento del proceso en lugar de utilizar un algoritmo de aprendizaje no supervisado.
- En segundo lugar se debe aprender el mapeo del dominio de entrada al espacio de salida. En este caso las salidas del proceso se conocen y por tanto se utiliza un método de aprendizaje supervisado. La salida del proceso y se compara con la predicción del modelo \hat{y} con la finalidad de definir una medida del error. Los parámetros de los modelos que representan a las regiones de operación se ajustan tomando como base esta medida del error.

Las dos etapas de éste método de entrenamiento (explorar una exacta descomposición del espacio de entrada y aprender el mapeo entrada salida), pueden ocurrir en paralelo o en forma secuencial.

Estructuras de Identificación con un enfoque multimodelo.

Existen diferentes enfoques en la aplicación del enfoque multimodelo en identificación [17]. La primer estructura utiliza conocimiento *a priori* del proceso para determinar los submodelos, la segunda estructura utiliza un algoritmo de agrupamiento como una etapa de preprocesamiento, el tercer enfoque consiste en utilizar una red neuronal para establecer las fronteras de las diferentes regiones de operación, y finalmente la cuarta estructura utiliza múltiples modelos y un algoritmo de selección.

1) Partición basada en conocimiento apriori.

La división del espacio de entrada en regiones de operación puede hacerse si se tienen un buen conocimiento del proceso. Si se define al conjunto de puntos de operación como Z , entonces un punto de operación $z \in Z$ es un vector de variables que caracterizan el comportamiento del sistema. Las regiones de operación están definidas como $Z_i \subset Z$, y pueden entenderse como una vecindad de puntos de operación que caracterizan el comportamiento de la planta. Por ejemplo, en un proceso de neutralización la variable pH caracteriza el comportamiento del sistema, de aquí que z representa a la variable pH . En general un proceso de neutralización puede dividirse en tres regiones de operación: neutra, ácida y alcalina; entonces la región de operación neutra puede definirse como la vecindad de puntos de operación tal que, $6.5 < pH < 7.5$.

Sea \mathbf{x} el vector de entrada al modelo que consta de m variables distintas $\mathbf{x} = (x_1, x_2, \dots, x_m)$, la salida del proceso y , y la predicción del modelo \hat{y} . En muchos casos el vector \mathbf{x} es un subconjunto del punto de operación z , ya que la salida del proceso y no está incluida en el vector de entrada al modelo \mathbf{x} .

En este método la descomposición del espacio de entrada se basa en el conocimiento *a priori* del proceso a identificar. Diferentes regiones de operación pueden asignarse, tomando como base que se conoce muy bien el proceso, por medio de una definición de umbrales de las variables que caracterizan al proceso x_{iTR} , para cada variable x_i ($i \in [1, \dots, m]$). La salida del multimodelo depende del criterio de selección o de las funciones de partición que produzcan particiones suaves o bruscas.

Divisiones suaves del espacio de entrada permiten que el modelo de una región pueda verse influenciado por regiones de operación colindantes, así uno o más submodelos pueden ajustarse durante la fase de entrenamiento. De acuerdo con esto, uno o más submodelos contribuyen a la predicción global del multimodelo en cada instante de tiempo. Por otro lado, si se utilizan particiones bruscas del espacio de operación, se tendrá que sólo un submodelo será responsable de la predicción del multimodelo en cada región de operación.

2) **Partición basada en entrenamiento no supervisado**

En este caso el problema de identificación de la planta se descompone en subproblemas utilizando un algoritmo de aprendizaje no supervisado, y la identificación de los subproblemas o regiones de operación se basa en un algoritmo de aprendizaje supervisado tal como el gradiente descendiente. Alguno de los algoritmos no supervisados: el algoritmo adaptivo de agrupamiento "k-centros" [14], o el mapa auto organizado (SOM) [31], pueden representar el mecanismo de partición.

La tarea de los algoritmos de agrupamiento es descomponer el dominio de entrada (espacio de operación) en k-regiones, y luego encontrar vectores centro que representen de una manera óptima a los vectores de entrada en cada región. El algoritmo de agrupamiento "k-centros" tradicional sólo puede asegurar optimalidad local, lo cual depende de la localización inicial de los vectores centro de cada grupo (vectores de referencia). El algoritmo adaptivo de agrupamiento "k-centros", supera ésta desventaja aproximando una solución de agrupamiento óptima. Cada grupo forma una región de operación para un submodelo, los cuales pueden ser identificados por modelos de redes neuronales.

En el segundo tipo de algoritmo no supervisado las regiones de operación pueden clasificarse por medio de un mapa auto organizado. A diferencia del algoritmo adaptivo de agrupamiento "k-medios", el mapa auto organizado define una vecindad de adaptación. Mientras que el algoritmo de agrupamiento sólo ajusta al vector de referencia ganador $c_k(t)$, el mapa auto organizado de Kohonen además ajusta a todos los vectores de referencia que están dentro de la vecindad de adaptación. La figura 2-9 ilustra un mapa auto organizado en la clasificación de regiones de operación.

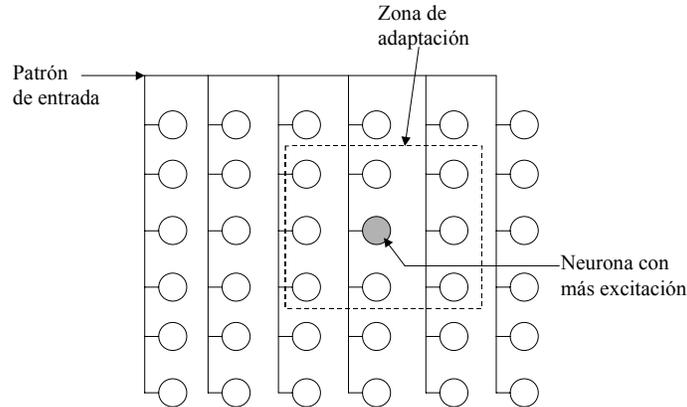


Figura 2-9: Mapa auto organizado de Kohonen

El entrenamiento consiste en aplicar un vector de entrada a la red (punto de operación z), que es recibien todas las neuronas y se calcula la salida de cada neurona. Se permite que las neuronas interactuen unas con otras y se localiza a la neurona que responda más al patrón de entrada. Sólo a esta neurona y a sus vecinas, dentro de cierta distancia, se les permite ajustar sus pesos para que sean más sensibles al patrón de entrada en particular. De esta manera el entrenamiento tiene el efecto de organizar un "mapa" (espacio de puntos de operación Z), de tal forma que diferentes áreas del mapa correspodan a diferentes regiones de operación y a la vez a diferentes patrones de entrada.

3) Partición con redes de selección no lineales

Este enfoque fué introducido por Weigend [77]. La estructura de expertos seleccionados con una red neuronal no lineal y el algoritmo de entrenamiento tienen un enfoque probabilístico. El modelo esta formado por un cierto número de expertos y un mecanismo de selección. Cada experto, en la forma de una red neuronal, corresponde a un submodelo. La división del espacio de entrada se realiza con una red neuronal. La red neuronal que trabaja como selector proporciona una salida a cada submodelo. La estructura del modelo de expertos con una red de selección no lineal se muestra en la figura 2-10.

Las entradas a los submodelos y las entradas a la red de selección consisten de un vector de variables medibles del dominio de entrada. Los expertos y la red selectora pueden compartir las mismas entradas o puede utilizarse un conjunto distinto de entradas.

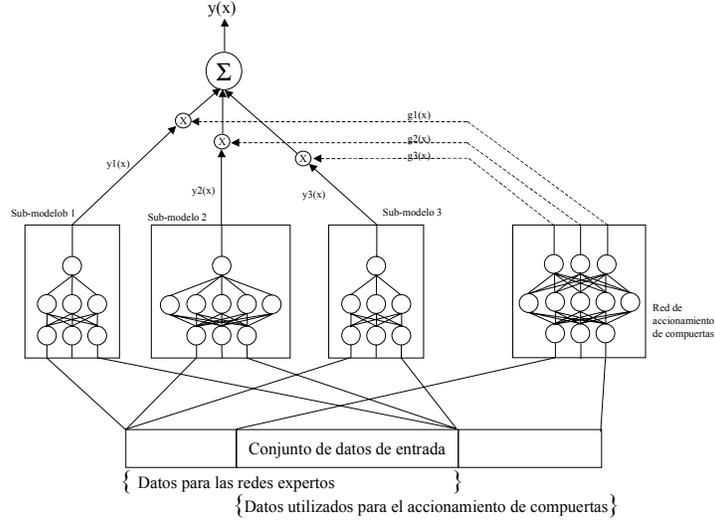


Figura 2-10: Estructura multimodelo con expertos y selector

La red de selección se implementa con una red neuronal directa, y sus salidas g_1 , g_2 y g_3 , se combinan con las salidas de los expertos para obtener la salida del multimodelo. Debido a que las salidas de la red de selección se tratan como probabilidades estas tienen valores en el intervalo $[0, 1]$. Las neuronas de salida de la red de selección combinan la activación de las unidades ocultas ξ , con los pesos correspondientes para formar una activación intermedia s_j , de acuerdo con la siguiente expresión

$$s_j = \sum_h w_{jh} \xi_h + b_j \quad j = (1, 2, \dots, K) \quad (2.8)$$

Donde el subíndice h representa el h -ésimo nodo oculto. Entonces la activación intermedia es exponenciada y normalizada, lo que resulta en la salida final de la red selectora:

$$g_j = \frac{e^{s_j}}{\sum_{l=1}^K e^{s_l}} \quad (2.9)$$

Las salidas de la red de selección están restringidas por $\sum_{j=1}^K g_j = 1$. La competencia entre los submodelos queda establecida por esta restricción. La partición utiliza fronteras

suaves ya que g_j puede tomar valores en el rango $[0, 1]$. Cada vector de datos de entrada puede ser asignado suavemente a varios submodelos. Como se muestra en la figura 2-10, la salida del j -ésimo experto, es ponderada por la j -ésima salida de la red de selección. Los expertos se pueden implementar con redes neuronales directas o recurrentes. Finalmente, no puede aplicarse un método de aprendizaje sencillo ya que no se conocen las fronteras de la regiones de operación. Este problema puede tratarse con la herramienta estadística de máxima probabilidad [17].

4) Identificación con un multimodelo y selector

Esta metodología utiliza múltiples modelos y un selector para modelar una planta que opera en distintas regiones de operación [49].

Debido a que la planta p puede operar en cualquier punto del espacio operación es necesario determinar cual de los identificadores I_i debe seleccionarse y en que instante debe ser seleccionado. En cualquier problema de identificación y control pueden utilizarse modelos múltiples de identificación, pero sólo una señal de control puede aplicarse a la planta. Por esta razón, el criterio de selección debe establecerse en función de los errores de identificación.

La estructura de un sistema capaz, al menos en teoría, de identificar la región de operación actualmente presente se muestra en la figura 2-11.

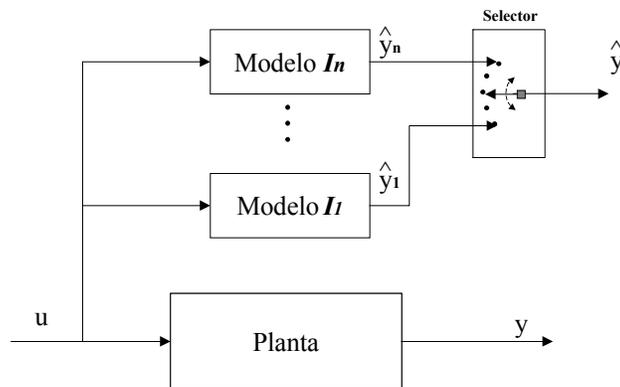


Figura 2-11: Esquema de identificación multimodelo con selector

El esquema introducido en [49] muestra N modelos de identificación denotados por $\{I_i\}_{i=1}^N$, con salidas correspondientes $\{\hat{y}_i\}_{i=1}^N$. Cada modelo de identificación representa a la

planta en una región de operación diferente. En cada instante se determina una medida de los errores de identificación $e_i \triangleq \hat{y}_i - y$ a través de un índice de desempeño $J_i(t), i = 1, 2, \dots, N$, y se selecciona el modelo de identificación que tenga el menor error $\min_i \{J_i(t)\}$ para representar a la planta. La construcción del índice de desempeño se verá en la sección 2.3.

Cuando la planta se encuentre en la región de operación S_i , entonces el modelo I_i tendrá el menor índice de desempeño y su salida será seleccionada. Uno de los problemas de este enfoque se presenta cuando un cambio en la región de operación produce un comportamiento de la planta que no este considerado dentro de la estructura del multimodelo.

2.2.3 Control con un enfoque multimodelo

El diseño de controladores con un enfoque multimodelo puede ser de utilidad para contrarrestar cambios en las condiciones de operación que modifiquen la dinámica de la planta. Un control multimodelo depende de la representación de un proceso a través de múltiples modelos, y utiliza diferentes modelos para generar una señal de control que logre que el sistema siga una trayectoria deseada.

Fases de un control multimodelo

Una estrategia de control con enfoque multimodelo involucra dos fases: la fase de localización y la fase de control.

a) Fase de localización

En la fase de localización se hace una comparación entre los distintos modelos que integran al multimodelo para buscar al modelo que mejor representa al sistema. Se debe establecer un criterio para evaluar la calidad de los modelos. La decisión de que modelo representa mejor al proceso se la llama localización.

b) Fase de control

La fase de control inicia después de localizar el modelo que mejor represente al proceso. Los modelos se utilizan para construir una señal de control que permita al proceso seguir a las variables de referencia deseadas. Esta fase requiere de dos pasos: generar la señal de control básica para cada modelo por separado y la síntesis de la señal de control final que se aplicará al proceso. Esta señal de control final puede obtenerse seleccionando la señal de

control básica del modelo que mejor represente al proceso según el criterio de localización, o combinando las señales de control básicas de los distintos modelos disponibles. La figura 2-12 muestra las fases de localización y de control.

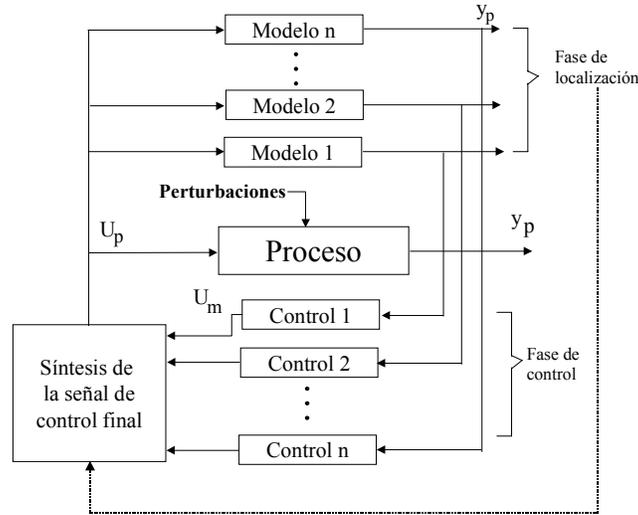


Figura 2-12: Fases de localización y control en un control multimodelo

Control multimodelo con selector

Esta estructura multimodelo se ilustra en la figura 2-13, y fué propuesta por Narendra [48] quien plantea el problema de control de la siguiente manera:

Considere que la planta y que todos los modelos de identificación pueden parámetroizarse de la misma forma, y sea S un conjunto cerrado y acotado que representa a un espacio de parámetros de dimensión finita. Suponga que el vector de parámetros de la planta p , y que el vector de parámetros del i -ésimo modelo \hat{p}_i , pertenecen a S (figura 2-14).

Para cada vector de parámetros \hat{p}_i existe una vecindad $S_i \subset S$, que se nombra i -ésimo ambiente y que se caracteriza como sigue: para todo $p \in S_i$, el controlador C_i , produce un error de seguimiento (entre la salida de la planta y la salida deseada) el cuál, de acuerdo con algún criterio J_c , es más pequeño que una constante ϵ_2 . ϵ_2 puede considerarse como el máximo error que resultará de utilizar el controlador fijo C_i , cuando $p \in S_i$.

Si $p \in S_i$, entonces es posible minimizar el criterio de error J_c ajustando los parámetros del controlador. Al final del transitorio el controlador puede proporcionar las condiciones

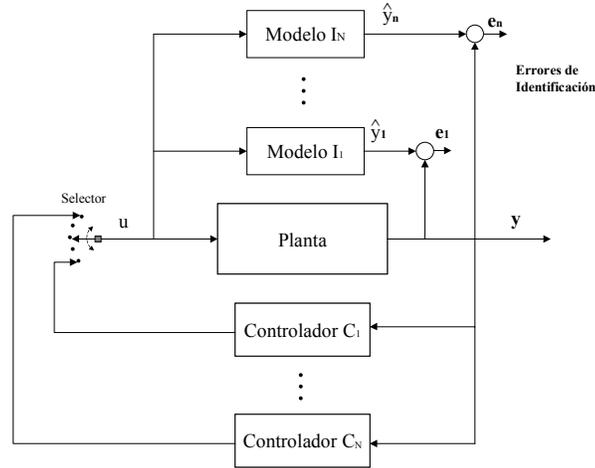


Figura 2-13: Esquema de control multimodelo

iniciales para un controlador adaptable. El objetivo de una etapa de adaptación es obtener un controlador C_i^* , de tal forma que el error de seguimiento sea más pequeño que una constante $\epsilon_1 \ll \epsilon_2$. De esta manera ϵ_2 puede considerarse como el máximo error transitorio y ϵ_1 como el máximo error en estado estacionario. El controlador fijo C_i define el máximo error transitorio ϵ_2 , y el controlador adaptable C_i^* define el máximo error en estado estacionario ϵ_1 , cuando se sabe *a priori* que $p \in S_i$.

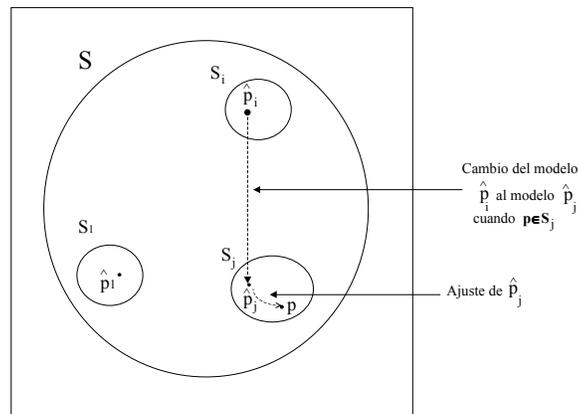


Figura 2-14: Ambientes S_i en el espacio de operación S .

Debido a que el vector de parámetros de la planta p , puede caer en cualquier lugar de S es necesario determinar cuál de los controladores C_i debe seleccionarse y en que momento

cambiar de un controlador a otro.

Si se presenta un nuevo vector de parámetros de la plana p que no este dentro de los ambientes anticipados, es decir $p \notin \{S_i\}_{i=1}^N$, entonces nos enfrentamos a una situacion no conocida, es decir, no anticipable. En este caso es necesario realizar un aprendizaje en linea de un nuevo modelo de identificación I_{N+1} (descrito por el vector \hat{p}_{N+1}), y por lo tanto, el ambiente correspondiente S_{N+1} tiene que aprenderse en linea. Después de esto se continua con el proceso de adaptación hasta que el error en estado estacionario sea menor a ϵ_1 . Una vez que se determina el nuevo modelo \hat{p}_{N+1} , el ambiente S_{N+1} pasa a ser un ambiente anticipado.

Red neuronal controlador multimodelo

En un controlador multimodelo con modelos conocidos *a priori* (figura 2-13), cuando el proceso experimenta una perturbación que lleva su salida a un un valor próximo a la salida de un modelo específico, se aplica la señal de control correspondiente a dicho modelo para llevar a la salida de regreso al valor deseado.El problema se presenta cuando el proceso no se comporta como uno de los modelos preestablecidos. En este caso la señal de control que se aplica es la que corresponde al modelo que más se apróxima al sistema. Sin embargo, esta acción no llevará al sistema a su trayectoria deseada. Una solución a este problema puede ser incrementar el número de modelos en el intervalo de operación del proceso, pero este enfoque es poco exacto e incrementa el costo de implementación del controlador.

Otra opción consiste en utilizar una red neuronal para implementar tanto la etapa de localización como la de control [1]. El algoritmo de aprendizaje se encarga de calcular el cambio necesario en los pesos con la finalidad de minimizar algún criterio de error. La fase de aprendizaje inicia presentando un vector de entrada en la capa de entrada de la red neuronal y propagando este vector hacia la capa de salida, luego se compara con el valor deseado y se calcula un error. Este error se propaga en sentido opuesto a través de la red neuronal para calcular el cambio necesario en los pesos para la minimización del error.

La red neuronal se entrena para controlar el proceso de tal forma que éste siga a la respuesta deseada en presencia de perturbaciones. En caso de que el proceso se desvie de los modelos preestablecidos, la red neuronal se adapta así misma y produce la señal de control apropiada ya que la red neuronal "tiene su propia inteligencia inherente" [1].

La estructura este controlador neuronal se muestra en la figura 2-15. Se tienen dos

modelos del sistema para simular el proceso en los límites superior e inferior en donde el proceso podría operar cuando se presente una perturbación o una falla.

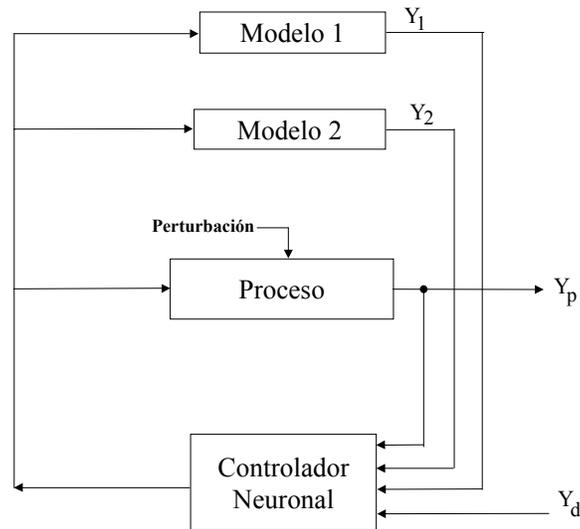


Figura 2-15: Esquema de un control neuronal multimodelo

La red neuronal controlador tiene cuatro entradas: la salida actual del proceso Y_P , la salida deseada Y_d , y las salidas de los modelos Y_1 y Y_2 ; y una salida: la entrada al proceso. El entrenamiento de esta red neuronal se ilustra en la figura 2-16.

En el entrenamiento la salida de la red neuronal se compara con una señal de control óptima del proceso. Entonces se introduce una perturbación para llevar al proceso a su límite superior al mismo tiempo que se selecciona la señal de control óptima correspondiente al modelo que representa al proceso en su límite superior. Después, se introduce otra perturbación que lleve al proceso al límite inferior y se selecciona la entrada de control óptima correspondiente a este modelo. Así la red neuronal se entrena para producir la entrada adecuada que llevará al proceso a la respuesta deseada, aún cuando se presenten perturbaciones o fallas en el sistema que traten de llevar al proceso a un punto de operación diferente. La ventaja de este esquema es que la red neuronal intentará llevar al sistema a la respuesta deseada, a pesar de que perturbaciones en el sistema traten desviarlo a un modelo distinto de los preestablecidos [1].

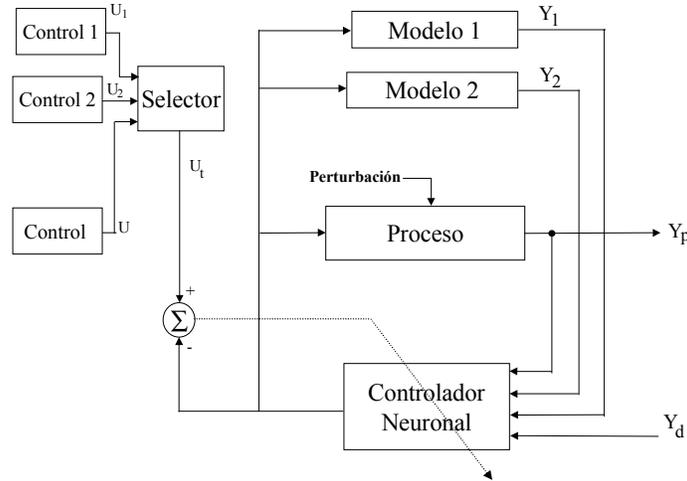


Figura 2-16: Entrenamiento de una red neuronal controlador

2.3 Selección de modelos

En la literatura existen esquemas de selección directos e indirectos. En los esquemas directos la selección de un nuevo controlador se basa directamente en la salida de la planta, mientras que en el esquema indirecto se utilizan múltiples modelos para determinar en que momento y cuál modelo se debe seleccionar. Los algoritmos de selección se encargan de localizar al modelo o modelos que representan mejor a la planta.

Dentro de los esquemas de selección indirectos se tienen algoritmos de selección estocásticos y determinísticos. Los primeros se basan en la teoría de aprendizaje automático la cual utiliza una medida probabilística de los modelos y ha sido ampliamente desarrollada para sistemas discretos. Los selectores determinísticos no consideran los efectos de ruido en el ambiente y su algoritmo de selección se basa en una medida del índice de desempeño de cada uno de los modelos.

2.3.1 Algoritmo de selección estocástico

Para el caso de sistemas discretos el problema de selección de modelos puede plantearse desde un enfoque probabilístico. La teoría de aprendizaje automático tiene un gran potencial en diversos problemas de ingeniería no lineales con un alto nivel de incertidumbres [46] [61].

La teoría de automatas se basa en la teoría de aprendizaje. Los sistemas de aprendizaje son sistemas de procesamiento de información cuya estructura y comportamiento se inspiran en los sistemas biológicos. Los organismos nacen con poca información inicial y aprenden acciones apropiadas por medio de ensayo y error, el aprendizaje es evidenciado por la modificación de su comportamiento. Las estrategias de control adaptable mejoran ligeramente el comportamiento del sistema en cada período de muestreo, estimando los parámetros en tiempo real (parámetros del modelo o de la ley de control), para alcanzar el objetivo de control deseado.

Dentro de la teoría de automatas se consideran tres tipos de aprendizaje automata: determinístico, de estructura fija y de estructura variable. En particular el automata de estructura variable es de interés en ambientes no estacionarios debido a que su estructura puede cambiar cuando el ambiente cambia. Un ambiente es estacionario si sus probabilidades de penalización son constantes, de otra forma es no estacionario.

Un automata interactua con un ambiente y aprende la acción óptima que ofrece el ambiente, su decisión se basa en información obtenida seleccionando acciones y observando si la acción es recompensada y luego actualiza recursivamente su distribución de probabilidad p_n . El modelo clásico de un sistema de aprendizaje consiste de un automata conectado a través de un lazo retroalimentado a un ambiente aleatorio (figura 2-17).

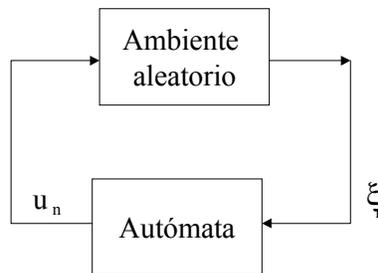


Figura 2-17: Modelo clásico de un automata

Los sistemas de aprendizaje son máquinas adaptables que interactúan con un ambiente y que pueden aprender dinámicamente la acción óptima que el ambiente ofrece. El papel del ambiente es establecer la relación entre las acciones del automata u_n , y su respuesta ξ_n . El ambiente incluye todas las condiciones e influencias externas y produce respuestas aleatorias

cuyas propiedades estadísticas dependen de los estímulos actuales. En control automático el ambiente corresponde al proceso a ser controlado.

Un automata operando en múltiples ambientes se muestra en la figura 2-18. El esquema de ambientes múltiples puede asociarse con el enfoque multimodelo en el sentido de que cada ambiente corresponde a un modelo de identificación de la planta, y la función $F(\cdot)$ representa la estrategia de selección.

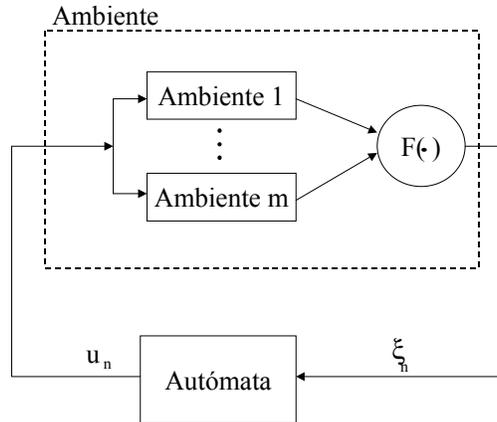


Figura 2-18: Autómata interactuando con múltiples ambientes

Un esquema de refuerzo actualiza el vector p_{n+1} en función de la respuesta de los ambientes $\xi_n^j, j = 1, \dots, m$, la acción del automata u_n y el vector de probabilidad p_n .

$$p_{n+1} = F(p_n, u_n, \xi_n) \quad (2.10)$$

La respuesta ξ_n del ambiente puede corresponder a la respuesta del ambiente que tenga la máxima probabilidad dentro de vector p_{n+1} . Otra posibilidad es que la función $F(\cdot)$ represente un selector suave en donde la salida ξ_n , se construya con las respuestas de todos los ambientes.

Observación 2.1 *Un esquema de refuerzo puede compararse con el procedimiento de estimación recursiva que se utiliza en control adaptable. El algoritmo de refuerzo actualiza la distribución de probabilidad p_{n+1} a partir de la distribución de probabilidad p_n . (2.10). Diversos algoritmos para ajustar las probabilidades después de cada período de muestreo (iteración*

con el ambiente) se basan en cambios incrementales en las probabilidades. El esquema de refuerzo es el corazón del automata.

Observación 2.2 *El algoritmo de selección puede ser determinístico o estocástico. En particular los algoritmos determinísticos puede ser riesgosos ya que un modelo puede tener el menor índice de desempeño por un largo período de tiempo, lo cuál puede hacer que se pierda información importante del ambiente.*

2.3.2 Algoritmo de selección determinístico

Este algoritmo de selección consiste en monitorear continuamente el índice de desempeño $J_j(t)$ de cada uno de los modelos I_j que integran al multimodelo. Cada índice de desempeño $J_j(t)$ $j = 1, \dots, N$ se construye con los errores de identificación e_j , de aquí que el índice de desempeño sea una medida de la exactitud con la que los modelos de identificación I_j representan a la planta. El algoritmo se encarga de seleccionar el controlador que corresponde al modelo que tenga el valor más pequeño del índice de desempeño $J_j(t)$.

Esta estrategia se basa en el hecho de que pequeños errores de identificación producen pequeños errores de seguimiento [50] [51]. La elección del índice de desempeño se basa en observaciones empíricas, de donde se deduce que se requieren medidas de error instantáneo y medidas del error a lo largo de toda la trayectoria para estimar de una manera confiable la exactitud de los modelos de identificación I_j [50].

Un índice de desempeño que incorpora estas características tiene la siguiente forma

$$J_j(t) = \alpha e_j^2(t) + \beta \int_0^t e^{-\lambda(t-\tau)} e_j^2(\tau) d\tau \quad (2.11)$$

En donde $\alpha \geq 0, \beta > 0$ y $\lambda > 0$ son parámetros de diseño, α y β determinan, respectivamente, el peso que tienen el error instantáneo y el error a lo largo de toda la trayectoria, mientras que λ determina la memoria del índice de desempeño a lo largo de toda la trayectoria.

2.3.3 Algoritmo de selección con histéresis

Para prevenir una conmutación arbitrariamente rápida puede utilizarse el algoritmo de histéresis introducido en [45]. Este algoritmo se basa en el cálculo del índice de desempeño de cada modelo de identificación como se presentó en el punto anterior. La idea de incluir una constante de histéresis es que a veces dos modelos de identificación pueden ser suficientemente buenos, de tal forma que un selector convencional puede conmutar con una gran velocidad entre uno y otro.

El algoritmo de selección con histéresis puede establecerse de la siguiente manera: si el modelo de identificación y su controlador respectivo (I_j, C_j) se están utilizando en el tiempo t , y en ese instante el modelo I_k aproxima mejor a la planta, de tal forma que $J_k(t) = \min_i \{J_i(t)\}$, entonces el par (I_j, C_j) debe retenerse si $J_j(t) \leq J_k(t) + \delta$ y se debe seleccionar el par (I_k, C_k) en cualquier otro caso. La constante δ es llamada constante de histéresis.

2.4 Combinación del enfoque multimodelo y RNAD's

En esta estructura se propone combinar el enfoque multimodelo con las técnicas de identificación y control utilizando RNA dinámicas. Por un lado, el enfoque multimodelo permite modelar un sistema en diferentes regiones de operación, mientras que las RNA dinámicas tienen una gran capacidad de aproximar sistemas no lineales, además de que presentan un buen comportamiento en presencia de dinámicas no modeladas.

2.4.1 Identificación multimodelo con RNA dinámicas

El identificador multimodelo propuesto consiste de un número de redes neuronales dinámicas que están continuamente identificando a la planta en línea. Los errores de identificación $(\hat{y}_i - y)$ se están evaluando en forma continua y al mismo tiempo que se calcula el índice de desempeño J_i de cada identificador neuronal I_i . La estructura incluye un selector con un algoritmo de histéresis que se encarga de buscar a la red neuronal que aproxime mejor la dinámica de la planta. La figura 2-19 ilustra el esquema de identificación que utiliza redes

neuronales dinámicas como modelos de identificación.

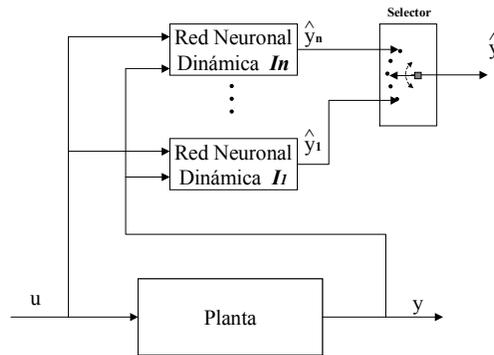


Figura 2-19: Identificación multimodelo con redes neuronales dinámicas

La calidad de cada modelo de identificación I_1, \dots, I_n se mide con un índice de desempeño como (2.11). Tomando como base este esquema se calcula la señal de control que lleve al proceso a seguir una respuesta deseada.

El utilizar redes neuronales dinámicas como modelos de identificación puede disminuir el número de modelos que se requieren para modelar una planta no lineal compleja. Esto se debe a que una red neuronal dinámica puede aproximar bien a la planta inclusive en diferentes regiones de operación, pero tal vez no en todo el espacio de operación o en presencia de grandes cambios en las condiciones de operación. En esta estructura multimodelo todas las redes neuronales están tratando de aproximar a la planta independientemente de la región de operación en que se encuentre.

2.4.2 Control multimodelo con RNA dinámicas

Tomando como base la estructura de identificación multimodelo anterior se propone un controlador neuronal que utiliza la dinámica de la red neuronal que mejor aproxima a la planta para, junto con un modelo de referencia, calcular la señal de control necesaria para que la planta siga una respuesta deseada. La figura 2-20 muestra una de las estructuras de control multimodelo neuronal propuestas. Otras estructuras se presentan en el capítulo 4.

En la figura 2-20 se puede apreciar sólo un controlador, el cual calcula la señal de control que corresponde al modelo de identificación seleccionado, en vez de tener que calcular n

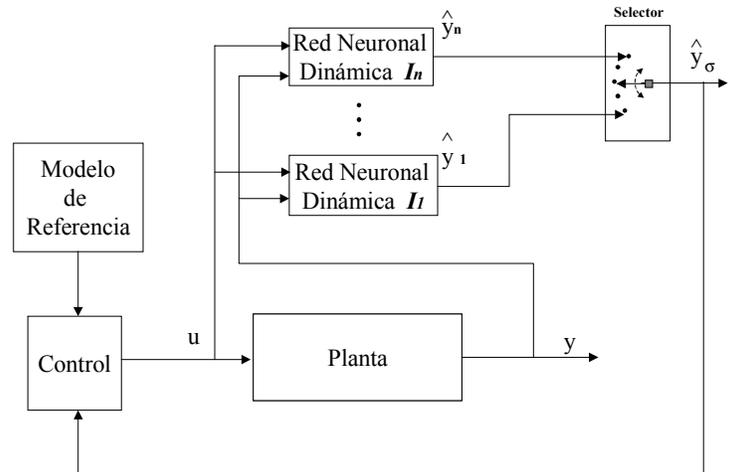


Figura 2-20: Control multimodelo con redes neuronales dinámicas

controles como en otros enfoques. La disminución en el número de controladores y en el número de identificadores puede hacer que el costo computacional sea menor.

Capítulo 3

Identificación con un enfoque multimodelo neuronal

La mayoría de los procesos químicos pueden caracterizarse como no lineales y no estacionarios. Un sistema dinámico es no estacionario si las propiedades estadísticas de las salidas del sistema se modifican con el tiempo. Una subclase importante de los procesos no estacionarios es aquella que conmuta entre diferentes regiones de operación. Estos sistemas dinámicos exhiben comportamientos muy diferentes bajo distintas condiciones de operación. Este comportamiento no estacionario o estacionario por partes es la motivación para la aplicación del enfoque multimodelo al control e identificación de sistemas.

Por otro lado, la identificación de sistemas no lineales desconocidos o "cajas negras" utilizando redes neuronales es una herramienta importante que no requiere de un modelo analítico, y se basa en las bondades de aproximación de las redes neuronales. Debido a que muchos procesos químicos son altamente no lineales e involucran reacciones químicas complejas, una red neuronal dinámica con todo y su buen comportamiento ante dinámicas no modeladas, no puede modelar con exactitud la dinámica de este tipo de plantas, y menos aún si el proceso conmuta entre diferentes regiones de operación.

La aplicación del enfoque multimodelo a procesos altamente no lineales o con dinámicas complejas es una buena alternativa para resolver el problema de identificación de procesos químicos. El primer enfoque multimodelo [34], utilizó múltiples filtros de Kalman para mejorar la exactitud de estimaciones de estado. La introducción de selectores en la estructura

multimodelo se inició en [39], con la finalidad de relajar las condiciones bajo las cuales sistemas lineales desconocidos podrían estabilizarse con esquemas de control adaptable. En [47] se utiliza un esquema multimodelo con selector para mejorar la respuesta transitoria de sistemas de control adaptable.

A continuación se presenta la técnica de identificación utilizando una red neuronal, luego se estudia la estabilidad del proceso de identificación desde un enfoque de pasividad. Finalmente se introduce la estructura del identificador multimodelo neuronal propuesta y su prueba de estabilidad.

3.1 Identificación con una red neuronal dinámica

El empleo de redes neuronales parece ser una herramienta efectiva para identificar sistemas no lineales complejos cuando no se tiene suficiente información del modelo de la planta, o cuando se considera a la planta como una caja negra. Los identificadores neuronales pueden clasificarse dependiendo del tipo de red neuronal que se utiliza para su construcción, de aquí que existan identificadores neuronales estáticos y dinámicos. Las redes neuronales dinámicas tienen una gran capacidad de representar sistemas no lineales, y un buen comportamiento en presencia de dinámicas no modeladas gracias a que su estructura incorpora retroalimentación [32][52].

Modelo neuronal directo

El procedimiento para entrenar una red neuronal con la finalidad de representar la dinámica de la planta se le conoce como modelado directo. La estructura para obtener este modelo se ilustra en la figura 3-1.

En este esquema de identificación la red neuronal se conecta en paralelo con la planta y el error de identificación, entre la salida de la red y la salida de planta, es la señal que se utiliza en el algoritmo de aprendizaje de la red neuronal para ajustar sus pesos. Esto con la finalidad de que la salida de la red \hat{y} se aproxime a la salida de la planta y .

Esta es una estructura de aprendizaje supervisado clásica en donde el tutor, en este caso la planta, proporciona los valores deseados (que se representan por la salida de la planta), directamente en el sistema coordinado del aprendizaje o red neuronal.

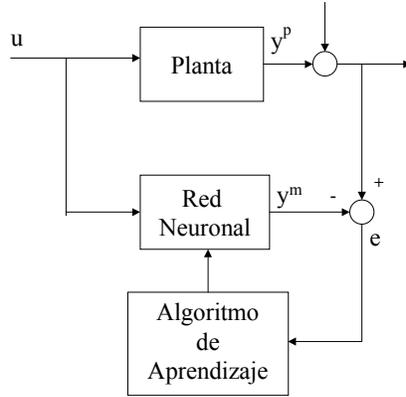


Figura 3-1: Identificación de la dinámica de la planta

RNA dinámica

Considere la siguiente red neuronal dinámica paralelo cuya estructura se muestra en la figura 3-2

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)\gamma(u_t)$$

Donde $\hat{x}_t \in R^n$ es el estado de la RNA, $u_t \in R^m$ es la entrada a la planta, $W_{1,t} \in R^{n \times n}$ es la matriz de retroalimentación de estado no lineal, $W_{2,t} \in R^{n \times n}$ es la matriz de entrada, y $A \in R^{n \times n}$ es una matriz Hurwitz. La función vectorial $\sigma(\hat{x}_t) \in R^n$ se considera de dimensión n cuyos elementos se incrementan monótonicamente. La función $\phi(\cdot)$ es una matriz diagonal $n \times n$: $\phi(\hat{x}_t) = \text{diag}(\phi(\hat{x}_1), \dots, \phi(\hat{x}_n))$; $\gamma(u_t)$ es una función vectorial de dimensión n , pero como $u_t \in R^m$, entonces los últimos $n - m$ elementos de $\gamma(u_t)$ son igual a cero. Los elementos $\sigma_i(\cdot)$ y $\phi_i(\cdot)$ comúnmente representan funciones sigmoideas con la forma general

$$\sigma_i(x_i) = \frac{a_i}{(1 + \exp(-b_i x_i))} - c_i \quad (3.1)$$

en donde a_i representa el valor máximo, b_i representa la pendiente de la curvatura de la función sigmoideal y c_i es una constante que desplaza verticalmente a la sigmoide.

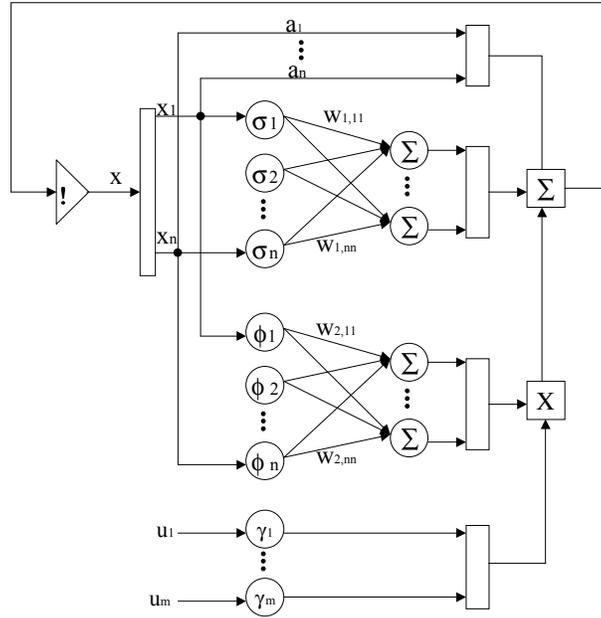


Figura 3-2: RNA-dinámica paralela

3.1.1 Análisis de estabilidad

Desde el punto de vista de la Ingeniería es importante asegurar estabilidad antes de implementar una aplicación en un sistema real. A pesar de que se han reportado aplicaciones exitosas, no existen muchos resultados sobre análisis de estabilidad de redes neuronales. La estabilidad asintótica global (EAG) de las redes neuronales dinámicas se ha estudiado durante la última década. El método de análisis *entrada para estado estable* (EEE) [69], es una herramienta efectiva para estudiar a las redes neuronales dinámicas. En [64] se establece que si los pesos son suficientemente pequeños entonces la red neuronal es EEE, y si la entrada es cero, entonces se tiene EAG.

Muchos investigadores estudian la estabilidad del error de identificación y del error de seguimiento de las redes neuronales [59]. En [26] se estudiaron las condiciones de estabilidad cuando se utilizan perceptrones multicapa para identificar y controlar un sistema no lineal. Un análisis similar al análisis de Lyapunov es adecuado para estudiar la estabilidad de las redes neuronales dinámicas. El caso de una red neuronal dinámica de una capa se discute en [62] [79], y las redes multicapa en [60]. Debido a que las redes neuronales no pueden represen-

tar con exactitud a un sistema no lineal desconocido, deben hacerse algunas modificaciones [21] para incorporar robustez en el algoritmo del gradiente y en el de retropropagación [62] [79]. Una de las ventajas del enfoque de identificación de sistemas desconocidos o "cajas negras", es que las incertidumbres pueden considerarse dentro de la caja negra, de tal manera que el algoritmo del gradiente sin modificación alguna puede tener propiedades de robustez.

En esta sección se presenta el análisis de estabilidad del identificador neuronal desde el enfoque de la teoría de pasividad [80][81]. La teoría de pasividad puede ayudar al estudio de sistemas no lineales pobremente definidos, y ofrece soluciones para la prueba de estabilidad absoluta. Esta teoría puede conducirnos a conclusiones generales sobre estabilidad utilizando sólo la característica entrada/salida. Las propiedades de pasividad de perceptrones multicapa se examinan en [15]. Uno de los objetivos es mostrar como una ley de aprendizaje similar al gradiente puede hacer al error de identificación estable, asintóticamente estable, o EEE.

Considere la clase de sistemas no lineales descritos por

$$\dot{x}_t = f(x_t, u_t) \quad (3.2)$$

$$y_t = x_t \quad (3.3)$$

en donde $x_t \in R^n$ es el estado, $u_t \in R^m$ el vector de entrada, $y_t \in R^m$ el vector de salida, la función $f : R^n \times R^m \rightarrow R^n$ es localmente Lipschitz, y la función de salida $h : R^n \times R^m \rightarrow R^n$ es continua. También se considera que para cualquier condición inicial $x^0 = x_0 \in R^n$, la salida $y_t = h(\phi(t, x^0, u))$ del sistema (3.2) es tal que, $\int_0^t |u_s^T y_s| ds < \infty$, para todo $t \geq 0$, es decir, la energía almacenada en el sistema (3.2) esta acotada.

Considere las siguientes propiedades de pasividad y de estabilidad de sistemas pasivos.

Definición 3.1 *Se dice que un sistema como (3.2) es pasivo si existe una función C^r no negativa $S(x_t) : R^n \rightarrow R$, (función de almacenamiento), tal que, para todo u_t , todas condiciones iniciales x^0 , y todo $t \geq 0$, la siguiente desigualdad es cierta:*

$$\dot{S}(x_t) \leq u_t^T y_t - \varepsilon u_t^T u_t - \delta y_t^T y_t - \rho \psi(x_t), \quad (x_t, u_t) \in R^n \times R^m$$

donde ε , δ y ρ son constantes no negativas, $\psi(x_t)$ es una función semidefinida positiva de x_t , tal que $\psi(0) = 0$. $\psi(x_t)$ se conoce como velocidad de disipación del estado.

Además se dice que el sistema es estrictamente pasivo si existe una función definida positiva $V(x_t) : R^n \rightarrow R$, tal que,

$$\dot{S}(x_t) = u_t^T y_t - V(x_t)$$

Propiedad 1. Si la función de almacenamiento $S(x_t)$ es diferenciable y el sistema dinámico (3.2) es pasivo, entonces la función de almacenamiento $S(x_t)$ satisface $\dot{S}(x_t) \leq u_t^T y_t$.

Definición 3.2 Un sistema como (3.2), se dice que es globalmente "entrada para estado estable" (EEE) si existe una función-K $\gamma(s)$ (continua y estrictamente creciente, con $\gamma(0) = 0$), y una función KL $\beta(s, t)$ (función-K y para cada $s_0 \geq 0$ fijo, $\lim_{t \rightarrow \infty} \beta(s_0, t) = 0$), tal que, para cada $u_t \in L_\infty$ ($\sup \{\|u(t)\|, t \geq 0\} < \infty$) y para cada estado inicial $x^0 \in R^n$, se cumple que:

$$\|x(t, x^0, u_t)\| \leq \beta(s, t)(\|x^0\|, t) + \gamma(s) \|u(t)\|$$

para todo $t \geq 0$.

Propiedad 2. Si un sistema es "entrada para estado estable" (EEE), el comportamiento del sistema debe permanecer acotado cuando sus entradas son acotadas.

Análisis de estabilidad

Considere el sistema no lineal

$$\dot{x}_t = f(x_t, u_t, t), \quad x_t \in R^n, \quad u_t \in R^m \quad (3.4)$$

que será identificado con la siguiente red neuronal dinámica paralela

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)\gamma(u_t) \quad (3.5)$$

donde $\hat{x}_t \in R^n$ es el estado de la RNA, $u_t \in R^m$ es la entrada a la planta, $W_{1,t} \in R^{n \times n}$ es la matriz de retroalimentación de estado no lineal, $W_{2,t} \in R^{n \times n}$ es la matriz de entrada, y $A \in R^{n \times n}$ es una matriz Hurwitz. La función vectorial $\sigma(\hat{x}_t) \in R^n$ se considera de dimensión n cuyos elementos se incrementan monótonicamente. La función $\phi(\cdot)$ es una matriz diagonal $n \times n$: $\phi(\hat{x}_t) = \text{diag}(\phi(\hat{x}_1), \dots, \phi(\hat{x}_n))$; $\gamma(u_t)$ es una función vectorial de dimensión n , pero como $u_t \in R^m$, entonces los últimos $n-m$ elementos de $\gamma(u_t)$ son igual a cero. La función $\gamma(\cdot)$ se selecciona como $\|\gamma(u_t)\|^2 \leq \bar{u}$ y los elementos de $\sigma(\cdot)$ y $\phi(\cdot)$ normalmente son funciones sigmoidales como 3.1.

Observación 3.1 *Las redes neuronales dinámicas se han discutido en diversos trabajos: [11],[7],[9] y [16]. En donde se muestra que el modelo de Hopfield es un caso especial de estas redes con $A = \text{diag}\{a_i\}$, $a_i := -1/R_i C_i$, $R_i > 0$ y $C_i > 0$. R_i y C_i son, respectivamente, la resistencia y la capacitancia en el i –ésimo nodo de la red .*

Definamos el error de identificación como

$$\Delta_t := \hat{x}_t - x_t$$

debido a que $\sigma(\cdot)$ y $\phi(\cdot)$ se seleccionan como funciones sigmoidales, éstas satisfacen las siguientes suposiciones.

A1: Las funciones $\sigma(\cdot)$ y $\phi(\cdot)$ satisfacen la condición de Lipschitz generalizada

$$\tilde{\sigma}^T \Lambda_1 \tilde{\sigma} \leq \Delta_t^T D_\sigma \Delta_t, \quad \text{y} \quad (\tilde{\phi}_t u_t)^T \Lambda_2 (\tilde{\phi}_t u_t) \leq \bar{u} \Delta_t^T D_\phi \Delta_t$$

donde: $\tilde{\sigma} := \sigma(\hat{x}_t) - \sigma(x_t)$, $\tilde{\phi} := \phi(\hat{x}_t) - \phi(x_t)$, y $\Lambda_1, \Lambda_2, D_\sigma$ y D_ϕ son constantes positivas conocidas.

Generalmente una red neuronal dinámica como (3.5) no es capaz de seguir exactamente al sistema no lineal (3.4). Esto puede escribirse de la siguiente manera

$$\dot{x}_t = Ax_t + W_1^* \sigma(x_t) + W_2^* \phi(x_t) \gamma(u_t) - \tilde{f}_t \tag{3.6}$$

en donde las matrices W_1^* y W_2^* no se conocen a *priori* y se supone que sólo se conocen cotas superiores de estas matrices de pesos

$$W_1^* \Lambda_1^{-1} W_1^{*T} \leq \bar{W}_1, \quad W_2^* \Lambda_2^{-1} W_2^{*T} \leq \bar{W}_2 \quad (3.7)$$

en donde las matrices \bar{W}_1 y \bar{W}_2 se conocen a *priori* y la función vectorial \tilde{f}_t puede considerarse como el error de modelado y/o perturbaciones. La dinámica del error de identificación se obtiene de (3.5) y (3.6)

$$\dot{\Delta}_t = A \Delta_t + \tilde{W}_{1,t} \sigma(\hat{x}_t) + \tilde{W}_{2,t} \phi(\hat{x}_t) \gamma(u_t) + W_1^* \tilde{\sigma} + W_2^* \tilde{\phi} \gamma(u_t) + \tilde{f}_t \quad (3.8)$$

en donde: $\tilde{W}_{1,t} := W_{1,t} - W_1^*$, y $\tilde{W}_{2,t} := W_{2,t} - W_2^*$

Si se define:

$$R := \bar{W}_1 + \bar{W}_2, \quad Q := D_\sigma + \bar{u} D_\phi + Q_0 \quad (3.9)$$

y las matrices A y Q_0 se seleccionan de tal modo que satisfagan las siguientes condiciones:

- (1) el par $(A, R^{1/2})$ es controlable y el par $(Q^{1/2}, A)$ es observable.
- (2) la condición local de frecuencia [75] satisface:

$$A^T R^{-1} A - Q \geq 1/4 [A^T R^{-1} - R^{-1} A] R [A^T R^{-1} - R^{-1} A]^T \quad (3.10)$$

entonces puede considerarse la siguiente suposición.

A2: Existe una matriz estable A y una matriz estrictamente definida positiva Q_0 , tal que la ecuación matricial de Riccati

$$A^T P + P A + P R P + Q = 0 \quad (3.11)$$

tiene una solución positiva $P = P^T > 0$. Esta suposición se satisface fácilmente si seleccionamos A como una matriz diagonal estable.

El siguiente teorema establece la ley de aprendizaje del identificador neuronal.

Teorema 3.1 Si los pesos $W_{1,t}$ y $W_{2,t}$ se actualizan de la siguiente manera

$$\begin{aligned}\dot{W}_{1,t} &= -K_1 P \Delta_t \sigma^T(\hat{x}_t) \\ \dot{W}_{2,t} &= -K_2 P \phi(\hat{x}_t) \gamma(u_t) \Delta_t^T\end{aligned}\tag{3.12}$$

donde P es la solución de la ecuación de Riccati (3.11), entonces la dinámica del error de identificación (3.8) es estrictamente pasiva desde \tilde{f}_t hasta el error de identificación $2P\Delta_t$.

Prueba. Seleccione la función de Lyapunov (función de almacenamiento) como:

$$S_t = \Delta_t^T P \Delta_t + \text{tr}\{\tilde{W}_{1,t}^T K_1^{-1} \tilde{W}_{1,t}\} + \text{tr}\{\tilde{W}_{2,t}^T K_2^{-1} \tilde{W}_{2,t}\}\tag{3.13}$$

en donde $P \in R^{n \times n}$ es una matriz definida positiva. Utilizando (3.8), tenemos que la derivada de la función de almacenamiento es

$$\begin{aligned}\dot{S}_t &= \Delta_t^T (PA + A^T P) \Delta_t + 2\Delta_t^T P \tilde{W}_{1,t} \sigma(\hat{x}_t) + 2\Delta_t^T P \tilde{W}_{2,t} \phi(\hat{x}_t) \gamma(u_t) \\ &\quad + 2\Delta_t^T P \tilde{f}_t + 2\Delta_t^T P [W_1^* \tilde{\sigma}_t + W_2^* \tilde{\phi} \gamma(u_t)] \\ &\quad + 2\text{tr}\{\tilde{W}_{1,t}^T K_1^{-1} \tilde{W}_{1,t}\} + 2\text{tr}\{\tilde{W}_{2,t}^T K_2^{-1} \tilde{W}_{2,t}\}\end{aligned}$$

Como $\Delta_t^T P W_1^* \tilde{\sigma}_t$ es un escalar, utilizando **A1** y la siguiente desigualdad matricial

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y\tag{3.14}$$

donde X, Y y $\Lambda \in R^{n \times k}$, para cualquier matriz definida positiva $\Lambda = \Lambda^T > 0$ se tiene

$$\begin{aligned}2\Delta_t^T P W_1^* \tilde{\sigma}_t &\leq \Delta_t^T P W_1^* \Lambda_1^{-1} W_1^{*T} P \Delta_t + \tilde{\sigma}_t^T \Lambda_1 \tilde{\sigma}_t \\ &\leq \Delta_t^T (P \bar{W}_1 P + D_\sigma) \Delta_t\end{aligned}\tag{3.15}$$

y

$$2\Delta_t^T P W_2^* \tilde{\phi}_t \gamma(u_t) \leq \Delta_t^T (P \bar{W}_2 P + \bar{u} D_\phi) \Delta_t$$

Utilizando estas desigualdades es posible escribir \dot{S}_t como:

$$\begin{aligned}\dot{S}_t &\leq \Delta_t^T [PA + A^T P + P(\bar{W}_1 + \bar{W}_2)P + (D_\sigma + \bar{u}D_\phi + Q_0)]\Delta_t \\ &\quad + 2tr\{\tilde{W}_{1,t}^T K_1^{-1}\tilde{W}_{1,t}\} + 2\Delta_t^T P\tilde{W}_{1,t}\sigma(\hat{x}_t) + 2\Delta_t^T P\tilde{f}_t \\ &\quad + 2tr\{\tilde{W}_{2,t}^T K_2^{-1}\tilde{W}_{2,t}\} + 2\Delta_t^T P\tilde{W}_{2,t}\phi(\hat{x}_t)\gamma(u_t) - \Delta_t^T Q_0\Delta_t\end{aligned}$$

ya que $\dot{\tilde{W}}_{1,t} = \dot{W}_{1,t}$, si utilizamos la ley de actualización (3.12) y **A2**, se obtiene:

$$\dot{S}_t \leq -\Delta_t^T Q_0\Delta_t + 2\Delta_t^T P\tilde{f}_t \quad (3.16)$$

De la definición 1, si definimos la entrada como \tilde{f}_t y la salida como $2P\Delta_t$, entonces el sistema es estrictamente pasivo con $V_t = \Delta_t^T Q_0\Delta_t \geq 0$. ■

Observación 3.2 Debido a que la velocidad de adaptación esta dada por $K_i P$ ($i = 1, 2$), y K_i puede ser cualquier matriz positiva, el proceso de aprendizaje (3.12) de la red neuronal dinámica no depende de la solución P de la ecuación de Riccati (3.11).

Corolario 3.1 Si sólo se presentan incertidumbres paramétricas, esto es $\tilde{f}_t = 0$, entonces la ley de adaptación (3.12), puede hacer al error de identificación Δ_t , asintoticamente estable

$$\lim_{t \rightarrow \infty} \Delta_t = 0, \quad W_{1,t} \in L_\infty, \quad W_{2,t} \in L_\infty, \quad \lim_{t \rightarrow \infty} \dot{W}_{1,t} = 0, \quad \lim_{t \rightarrow \infty} \dot{W}_{2,t} = 0 \quad (3.17)$$

Prueba. Debido a que la dinámica del error de identificación (3.8) es pasiva, de acuerdo con la propiedad 1, la función de almacenamiento satisface

$$\dot{S}_t \leq \tilde{f}_t^T 2P\Delta_t = 0 \quad (3.18)$$

El hecho de que la función S_t sea definida positiva, implica que $\Delta_t, W_{1,t}$, y $W_{2,t}$ esten acotados. De aquí que la dinámica del error de identificación (3.8) $\dot{\Delta}_t \in L_\infty$.

De las ecuaciones 3.16 y 3.18, se puede escribir

$$\frac{d}{dt} S_t \leq -\Delta_t^T Q_0\Delta_t \leq 0$$

Integrando

$$\int_0^\infty \|\Delta_t\|_{Q_0} \leq S_0 - S_\infty < \infty$$

De esta manera $\Delta_t \in L_2 \cap L_\infty$, y utilizando el lema de Barlat se obtiene (3.17).

Como $u_t, \sigma(\hat{x}_t), \phi(\hat{x}_t)$ y P están acotados

$$\lim_{t \rightarrow \infty} \dot{W}_{1,t} = 0, \quad \text{y} \quad \lim_{t \rightarrow \infty} \dot{W}_{2,t} = 0.$$

■

Observación 3.3 *Para el caso de acoplamiento de modelo $\tilde{f}_t = 0$, con un análisis similar al de Lyapunov, se puede obtener el mismo resultado que se obtiene en los corolarios 3.1 y 3.2 [79]. Sin embargo, en el caso en que el error de modelado \tilde{f}_t sea diferente de cero, se tiene una nueva conclusión sobre el identificador neuronal: el algoritmo del gradiente (3.12) también es robusto respecto a dinámicas no modeladas, a perturbaciones acotadas y a ruido estocástico.*

Teorema 3.2 *Utilizando la ley de actualización (3.12), la dinámica del error de identificación (3.8) es "entrada para estado estable" (EEE).*

Prueba. Utilizando la desigualdad matricial (3.14), tenemos

$$2\Delta_t^T P \tilde{f}_t \leq \Delta_t^T P \Lambda_f P \Delta_t + \tilde{f}_t^T \Lambda_f^{-1} \tilde{f}_t$$

con lo cual podemos escribir (3.16) como

$$\begin{aligned} \dot{S}_t &= -\Delta_t^T Q_0 \Delta_t + 2\Delta_t^T P \tilde{f}_t \\ &\leq \lambda_{\min}(Q_0) \|\Delta_t\|^2 + \Delta_t^T P \Lambda_f P \Delta_t + \tilde{f}_t^T \Lambda_f^{-1} \tilde{f}_t \\ &\leq -\alpha_{\|\Delta_t\|} \|\Delta_t\| + \beta_{\|\tilde{f}_t\|} \|\tilde{f}_t\| \end{aligned}$$

donde $\alpha_{\|\Delta_t\|} := [\lambda_{\min}(Q_0) - \lambda_{\max}(P \Lambda_f P)] \|\Delta_t\|$, y $\beta_{\|\tilde{f}_t\|} := \lambda_{\max}(\Lambda_f^{-1}) \|\tilde{f}_t\|$. Podemos

seleccionar una matriz definida positiva Λ_f , tal que

$$\lambda_{\max}(P\Lambda_f P) \leq \lambda_{\min}(Q_0) \quad (3.19)$$

De esta manera α y β son funciones K_∞ , y S_t es una función de Lyapunov EEE.

Utilizando el teorema 1 de [69], se tiene que la dinámica del error de identificación (3.8) es EEE. ■

Corolario 3.2 *Si el error de modelado \tilde{f}_t esta acotado, entonces la ley de actualización (3.12) puede hacer el procedimiento de identificación estable*

$$\Delta_t \in L_\infty, \quad W_{1,t} \in L_\infty, \quad W_{2,t} \in L_\infty$$

Prueba. *A partir de la propiedad 2 sabemos que EEE significa que el comportamiento de la red neuronal dinámica debe permanecer acotado cuando su entrada este acotada.* ■

Observación 3.4 *Debido a que las variables de estado y de salida estan físicamente acotadas se puede considerar que el error de modelado \tilde{f}_t también esta acotado [26] [60] [62]. La condición (3.19) puede establecerse si Λ_f es una matriz constante suficientemente pequeña. A diferencia de las leyes adaptables robustas, como por ejemplo banda muerta [60] y modificación- σ [32], no es necesario conocer la cota superior de las incertidumbres.*

Observación 3.5 *Las incertidumbres estructurales producen variaciones en los parámetros de un control adaptable, de tal manera que se debe utilizar una modificación robusta en el algoritmo de aprendizaje para hacer el proceso de identificación estable [21]. Los métodos adaptables robustos pueden extenderse directamente a la teoría de neuro-identificadores [26] [60] [62]. Sin embargo, un identificador neuronal es una especie de identificador de "caja-negra", en el sentido de que no es necesaria información estructural y todas las incertidumbres estan dentro de la "caja negra". Aunque algoritmos adaptables robustos son apropiados para identificación neuronal, no son los más sencillos. Por medio de la técnica de pasividad se obtiene que el algoritmo puro del gradiente es robusto con respecto a todos los tipos de incertidumbres acotadas.*

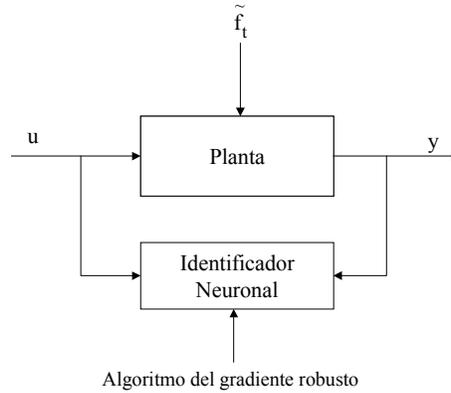


Figura 3-3: Identificación con un enfoque robusto

Observación 3.6 Cuando se tienen incertidumbres paramétricas es suficiente el algoritmo del gradiente en la ley de adaptación de pesos. Sin embargo, el problema es más complejo cuando se tienen incertidumbres estructurales, es decir cuando existe un error de modelado $\Delta f \neq 0$. Cuando se estudia la robustez del proceso de identificación desde el un enfoque de un análisis de Lyapunov se considera a la dinámica no modelada Δf , como una perturbación externa (fig.3-3) y se utiliza el algoritmo del gradiente modificado para incorporar robustez al proceso de identificación. Para el mismo caso ($\Delta f \neq 0$), pero desde el enfoque de pasividad, se considera a la dinámica no modelada no como una perturbación, sino como una entrada más al sistema (fig. 3-4) y se prueba que el algoritmo del gradiente puro es suficiente para proporcionar robustez al proceso de identificación.

3.2 Identificación con múltiples redes neuronales

El enfoque multimodelo y las redes neuronales, de manera independiente, proporcionan beneficios interesantes en las tareas de identificación, y su combinación puede resultar en una herramienta muy poderosa para enfrentar el problema de identificación de sistemas no lineales complejos. En esta sección se combinan estas dos herramientas efectivas en la identificación de sistemas y se propone un identificador multimodelo neuronal dinámico, en donde cada identificador es una red neuronal dinámica que aproxima en línea la dinámica de la planta. En la figura 3-5 se ilustra el esquema de identificación propuesto y se observan

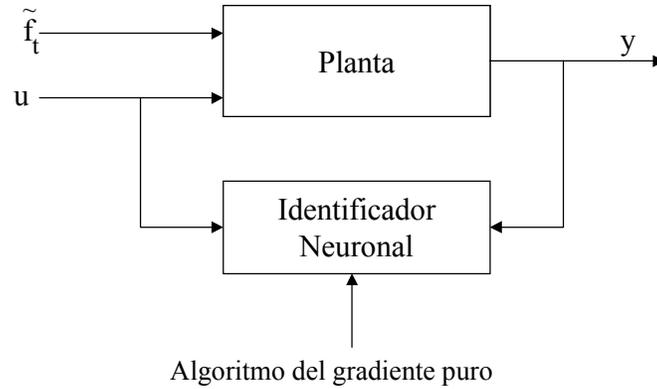


Figura 3-4: Identificación con un enfoque pasivo

los índices de desempeño de cada RNA-dinámica, y un bloque etiquetado con J_{\min} que se encarga de buscar el $\sigma \in \{1, \dots, N\}$.

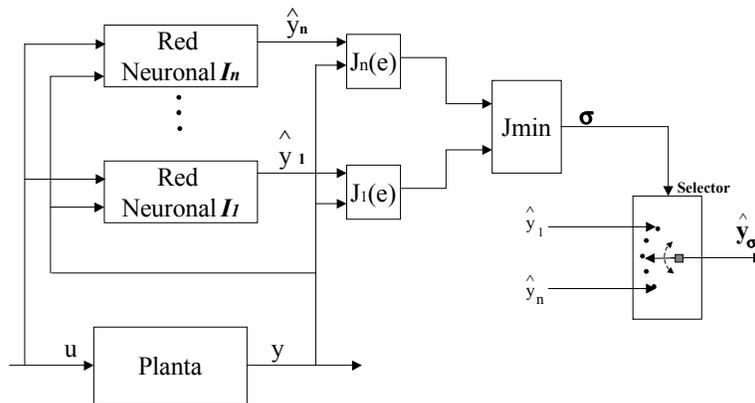


Figura 3-5: Identificador multimodelo neuronal dinámico

En esta estructura multimodelo una vez que se calcula el índice de desempeño de cada red dinámica se procede a localizar el identificador que tiene el menor índice de desempeño $J_{\min} = J_\sigma$. En esta fase, se comparan todos los índices de desempeño para localizar el índice σ , dentro del conjunto de N identificadores, correspondiente al identificador neuronal que aproxima mejor la dinámica de la planta. Luego se pasa la señal de mando σ al selector que se encarga de activar la σ -ésima entrada del selector. Para hacer el trabajo de selección se utiliza un algoritmo de selección con histéresis el cual, al mismo tiempo que mantiene activa

solamente a la mejor red neuronal, evita que se presenten conmutaciones innecesarias cuando la diferencia en la calidad de aproximación de dos redes neuronales es muy pequeña.

Si con una red neuronal de la forma

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)\gamma(u_t) \quad (3.20)$$

la planta se puede escribir como

$$\dot{x}_t = Ax_t + W_{1,t}\sigma(x_t) + W_{2,t}\phi(x_t)\gamma(u_t) - \tilde{f}_t$$

utilizando un identificador multimodelo con N identificadores neuronales de la forma (3.20) y un algoritmo de selección (con el índice σ denotando el identificador neuronal seleccionado), la planta no lineal podrá escribirse como

$$\dot{x}_t = A_\sigma x_t + W_{1,t}^\sigma \sigma_\sigma(x_t) + W_{2,t}^\sigma \phi_\sigma(x_t) \gamma_\sigma(u_t) - \tilde{f}_t$$

en donde los valores de A_σ , $W_{1,t}^\sigma$, $W_{2,t}^\sigma$, σ_σ , ϕ_σ , y γ_σ caracterizan la dinámica de la red neuronal que mejor aproxima la dinámica de la planta y $\sigma \in M = \{1, 2, \dots, m\}$ es la señal de mando al selector para activar la salida del identificador I_σ .

El esquema de selección o conmutación consiste en monitorear los índices de desempeño $J_j(t)$ de cada red neuronal dinámica I_j . Tales índices de desempeño se basan en los errores de identificación e_j respectivos; después, se selecciona el identificador correspondiente al modelo que tenga el valor más pequeño del índice de desempeño. La elección del índice de desempeño es motivada por observaciones empíricas, las cuales revelan que se requieren tanto medidas del error instantáneo como medidas del error a lo largo de toda la trayectoria para estimar, con confiabilidad, la exactitud de los modelos de identificación [50]. Un índice de desempeño que incorpora estas características tiene la siguiente forma

$$J_j(t) = \alpha e_j^2(t) + \beta \int_0^t e^{-\lambda(t-\tau)} e_j^2(\tau) d\tau$$

En donde $\alpha \geq 0, \beta > 0$ y $\lambda > 0$ son parámetros de diseño. α y β determinan el peso que tienen el error instantáneo y el error a lo largo de toda la trayectoria respectivamente; mientras que λ determina la memoria del índice de desempeño a lo largo de toda la trayectoria.

3.2.1 Análisis de estabilidad del identificador multimodelo

El análisis de estabilidad para una red neuronal se presentó en la sección 3.1.1, ahora es necesario estudiar la estabilidad del identificador multimodelo neuronal cuando cada modelo se elige con un algoritmo de selección con histéresis.

Algoritmo de selección con histéresis

La selección del modelo de identificación en la estructura multimodelo se decide en función del modelo que mejor represente la dinámica de la planta. Se elige al identificador que tiene el valor más pequeño del índice de desempeño $J_k(t) = \min\{J_i(t)\}$.

En el proceso de selección se puede presentar el caso en que dos modelos de identificación tengan un valor muy pequeño del índice de desempeño de tal forma que el selector este conmutando indefinidamente entre estos dos modelos en un pequeño intervalo de tiempo. El algoritmo de selección con histéresis [45] ayuda a corregir esta excesiva conmutación. Sin embargo, su implementación no debe afectar la estabilidad del proceso de identificación.

Suponga que el identificador neuronal I_j se esta utilizando en el tiempo t , y en ese instante el identificador I_k logra tener el menor índice de desempeño $J_k(t) = \min\{J_i(t)\}$, entonces si $J_j(t) \leq J_k(t) + \delta$ el identificador I_j debe seguir representando la dinámica de la planta, pero si $J_j(t) > J_k(t) + \delta$, entonces se debe seleccionar al identificador I_k . En éste caso δ es la constante de histéresis.

Análisis de estabilidad

Sea $\{f_1, f_2, \dots, f_m\}$ una familia de m funciones $f_i : R^n \times R_+ \rightarrow R^n$ donde $R_+ = [0, \infty)$. Considere el sistema dinámico:

$$\dot{x} = f_\sigma(x, t), \quad x(0) = x_0 \quad (3.21)$$

donde $\sigma \in M = \{1, 2, \dots, m\}$ es la entrada de selección. En (3.21) se introduce un

algoritmo para seleccionar σ de los valores de M , de tal forma que logremos que las salidas de las correspondientes funciones de prueba:

$$\delta_i(t) = g_i(x, t, \sup_{\tau \in [0, t)} d_i(x(\tau), \tau)), \quad i \in M \quad (3.22)$$

tengan ciertas propiedades deseadas. Cada una de estas funciones de prueba representan a cada uno de los índices de desempeño de los m modelos de identificación que conforman al multímodelo neuronal. g_i, d_i son funciones que mapean $R^n \times R_+ \times R_+ \rightarrow R_+$ y $R^n \times R_+ \rightarrow R_+$, respectivamente.

Para definir el algoritmo sea $\rho : R_+^m \rightarrow M$ la función cuyo valor en $\delta = [\delta_1, \delta_2, \dots, \delta_m]' \in R_+^m$, denotado como $\rho(\delta)$, es el menor entero $i \in M$, para el cual $\delta_i \leq \delta_j, j \in M$.

Sea h un número real positivo que se conoce como constante de histéresis y definamos la función de transición $\phi : M \times R_+^m$, como:

$$\phi(i, \delta) = \begin{cases} i, & \text{si } \delta_i < \delta_{\rho(\delta)} + h \quad (i \text{ no cambia}) \\ \rho(\delta), & \text{si } \delta_i \geq \delta_{\rho(\delta)} + h \quad (i \text{ cambia a } \rho(\delta)) \end{cases} \quad (3.23)$$

de esta manera, para $i \in M$ y $\delta \in R_+^m$, $\phi(i, \delta) = i$ si $\delta_i < \delta_j + h$ para todo $j \in M$, de otro modo $\phi(i, \delta)$ es igual al menor $j \in M$ para el cual $\delta_j \leq \delta_k$, para todo $k \in M$.

La función σ en la ecuación (3.21) esta definida recursivamente a lo largo de una solución de (3.21), de cuerdo con la siguiente ecuación:

$$\sigma(t) = \phi(\sigma^-(t), \delta(t)), \quad \sigma^-(0) = i_0 \quad (3.24)$$

donde $\delta \in R_+^m$, i_0 es la condición inicial en M y $\sigma^-(t)$ es el límite por abajo, es decir: $\sigma^-(t) = \lim_{\tau \rightarrow t} \sigma(\tau)$. Así, arrancando en un estado inicial i_0 en $t = 0^-$, σ permanece en este estado hasta un tiempo $t_1 \geq 0$, tal que para algún $j \in M$, $\delta_i \geq \delta_j + h$, en cuyo caso σ cambia al estado $\rho(\delta)$ el cual es el menor valor $i \in M$ para el cual $\delta_i \leq \delta_j, j \in M$.

Supongamos que cada f_i, g_i y d_i es al menos localmente Lipschitz en x y continua por

pedazos en t . Note que $\sigma(0)$ esta bien definida por (3.24). También observe que debido a la constante de histéresis h y a la suavidad de f_i , debe de existir un intervalo de duración máxima $(0, t_1)$ en donde σ permanezca constante. Este intervalo puede ser el de máxima existencia de x , o de no ser así x estará acotada en $[0, t_1)$. Si lo último es cierto, debe ocurrir un cambio en t_1 y de nuevo debido a la constante h , la continuidad de x , y la suavidad de f_i , debe existir in intervalo (t_1, t_2) de duración máxima en el cual σ permanezca constante.

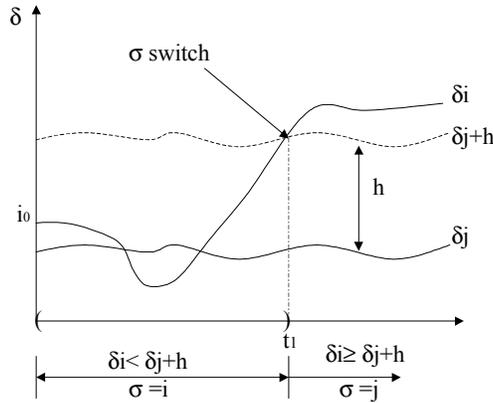


Figura 3-6: Algoritmo de selección con histéresis

Continuando con éste razonamiento, concluimos que debe existir un intervalo $[0, T)$ de duración máxima en el que existe un único par (x, σ) con x continua y σ constante por pedazos, el cual satisface (3.21-3.24). Además en cada subintervalo estrictamente propio $[0, \tau) \subset [0, T)$, σ debe cambiar a lo mucho un número finito de veces. Para estudiar el comportamiento del sistema (3.21-3.24) conforme $t \rightarrow T$, es necesario hacer las siguientes consideraciones.

Sea \mathcal{S} la clase de todas las funciones continuas por pedazos $s : [0, \infty) \rightarrow M$. En lo sucesivo para cada $s \in \mathcal{S}$, T_s denota la duración del máximo intervalo de existencia de las ecuaciones:

$$\dot{x} = f_{s(t)}(x, t), \quad x(0) = x_0$$

y x_s es la solución correspondiente.

Suposición 3.1 Para cada $s \in \mathcal{S}$ y cada $j \in M$, la función de prueba

$$\delta_j(t) = g_j(x_s(t), t, \sup_{\tau \in [0, t)} d_j(x_s(\tau), \tau))$$

tiene un límite (que puede ser infinito) conforme $t \rightarrow T_s$.

Observación 3.7 Que la función de prueba $\delta_j(t)$, $j \in M$ tenga un límite cuando $t \rightarrow T$, indica que cuando t se aproxima a T , entonces δ_j se aproxima a un valor bien definido b . En el contexto multimodelo decimos que para cada modelo neuronal el error de identificación e converge. Es decir que la función de prueba $\delta_j(t) = J(e)$ no cambia si $T \rightarrow \infty$.

Suposición 3.2 Existe al menos un entero $\mu \in M$ tal que para cada $s \in \mathcal{S}$, la función $\delta_\mu(t) = g_\mu(x_s(t), t, \sup_{\tau \in [0, t)} d_\mu(x_s(\tau), \tau))$ esta acotada en $[0, T_s)$.

Observación 3.8 En el contexto multímodelo decimos que existe al menos un modelo neuronal en la estructura multimodelo tal que su índice de desempeño esta acotado debido a que $\delta(t) = \int \|e\|^2 dt$ y e esta acotado.

Esta condiciones nos ayudan al probar el siguiente teorema

Teorema 3.3 (1) Si utilizamos la Red Neuronal Dinámica Multimodelo

$$\dot{x} = f_\sigma(x, t), \quad x(0) = x_0$$

con M y σ como se definen en (3.21), y el algoritmo de selección con histéresis que se describe en (3.23) y (3.24), entonces existe un tiempo $T^* \leq T$ después del cual σ es constante, y además $\delta_{\sigma(T^*)}$ esta acotada en $[0, T)$.

Prueba. Para un estado inicial fijo (x_0, i_0) , sea (x, σ) la única solución a (3.21 - 3.24), y $[0, T)$ el intervalo de duración máximo en el que ésta solución esta definida; sean $\delta_i, i \in M$ las salidas correspondientes de la función de prueba a lo largo de ésta solución. Particione M en dos subconjuntos disjuntos M_b y M_u , de tal forma que δ_i es acotada en $[0, T)$, para $i \in M_b$, y δ_i es no acotada en $[0, T)$, para $i \in M_u$.

Hacemos $T_s = T$. Tomando en cuenta la condición 2, el conjunto M_b es no vacío, así que para $i \in M_b$, tenemos:

$$\delta_i(t) \leq \beta < \infty, \quad i \in M_b, \quad t \in [0, T) \quad (3.25)$$

De (3.23), tenemos que $\phi(i, \delta) = k$, cuando $\delta_i \geq \delta_k + h$ para algun $k \in M$. Si $k \in M_b$ entonces por (3.7) $\delta_i \geq \beta + h$. De aquí que, si $i \in M_u$ entonces $\delta_i > \beta + h$. Ahora podemos considerar que existe un tiempo t_1 en el que:

$$\delta_i(t) > \beta + h, \quad i \in M_u, \quad t \in [t_1, T) \quad (3.26)$$

y

$$|\delta_i(t) - \delta_i(t_1)| < h/4, \quad i \in M_b, \quad t \in [t_1, T) \quad (3.27)$$

(Note que t_1 debe existir por la condición 1).

Observación 3.9 *La situación opuesta a (3.26) nos permite establecer la condición para $i \in M_b$, en $t \in [t_1, T)$, mientras que (3.27) nos dice que para $i \in M_b$, $\delta_i(t)$ presenta pequeñas variaciones en $t \in [t_1, T)$, que no rebasan una cuarta parte del valor de h .*

Así que de (3.26) y (3.27), podemos decir que:

$$i \in M_b \iff \exists t_2 \in [t_1, T), \text{ tal que } \delta_i(t_2) \leq \beta + h \quad (3.28)$$

i) Para verificar que δ_{σ^*} esta acotada en $[0, T)$, suponga que σ no cambia en t_1 y que en $t \in [t_1, T)$ no ocurren más cambios, i.e., $T^* = t_1$ y $\sigma^* = \sigma(T^*)$. Como σ no cambia:

$$\delta_{\sigma^*} < \delta_j + h, \quad j \in M, \quad t \in [T^*, T)$$

De aquí que si $j \in M_b$, entonces por (3.25), $\delta_{\sigma^*} < \beta + h, t \in [T^*, T)$, de tal forma que δ_{σ^*} esta acotada en $[0, T)$, tal como se esperaba.

Observación 3.10 *Es razonable suponer que $j \in M_b$, ya que si $j \in M_u$ entonces δ_j no estaría acotada por lo que no tendría posibilidad de competir con δ_{σ^*} . En otras palabras, sólo los δ_j , con $j \in M_b$ tienen posibilidad de tener un mejor desempeño que δ_{σ^*} .*

ii) Ahora suponga que σ cambia en $t_2 \in [t_1, T)$ y que después de éste cambio σ permanece constante en $t \in [t_2, T)$. Hacemos $T^* = t_2$ y $\sigma^* = \sigma(T^*)$. En vista de la definición del selector en (3.23) y (3.24), tenemos:

$$\delta_{\sigma^*}(t_2) \leq \delta_j(t_2), \quad j \in M \quad (3.29)$$

Como M_b no es vacío, de (3.25) tenemos:

$$\delta_{\sigma^*}(T^*) \leq \beta < \beta + h$$

Por (3.28), δ_{σ^*} esta acotada en $[t_1, T)$ y $\sigma^* \in M_b$.

Ahora revisaremos los siguientes casos: $j \in M_b$ y $j \in M_u$

a) Para cualquier $t_2 \in [t_1, T)$, y cualquier $j \in M_b$, escribimos:

$$\delta_{\sigma^*}(t) - \delta_j(t) = (\delta_{\sigma^*}(t) - \delta_{\sigma^*}(t_2)) + (\delta_{\sigma^*}(t_2) - \delta_j(t_2)) + (\delta_j(t_2) - \delta_j(t))$$

De la ec. (3.29), tenemos que $(\delta_{\sigma^*}(t_2) - \delta_j(t_2)) \leq 0$, entonces:

$$\begin{aligned} \delta_{\sigma^*}(t) - \delta_j(t) &\leq (\delta_{\sigma^*}(t) - \delta_{\sigma^*}(t_2)) + (\delta_j(t_2) - \delta_j(t)) \\ &\leq \delta_{\sigma^*}(t) - \delta_{\sigma^*}(t_1) + \delta_{\sigma^*}(t_1) - \delta_{\sigma^*}(t_2) \\ &\quad + \delta_j(t_2) - \delta_j(t_1) + \delta_j(t_1) - \delta_j(t) \\ &< |\delta_{\sigma^*}(t) - \delta_{\sigma^*}(t_1)| + |\delta_{\sigma^*}(t_1) - \delta_{\sigma^*}(t_2)| \\ &\quad + |\delta_j(t_2) - \delta_j(t_1)| + |\delta_j(t_1) - \delta_j(t)| \end{aligned}$$

Como σ^* y j estan en M_b , se tiene a partir de (3.27) que:

$$\delta_{\sigma^*}(t) - \delta_j(t) < h, \quad t \in [t_2, T), \quad j \in M_b$$

b) Por otro lado, para $j \in M_u$ las ecuaciones (3.25) y (3.26) implican que:

$$\begin{aligned}\delta_{\sigma^*}(t) - \delta_j(t) &\leq \beta - \delta_j(t) < \beta - (\beta + h) < -h \\ &< h, \quad j \in M_u, \quad t \in [t_2, T), \quad h > 0\end{aligned}$$

Por a) y b) se tiene que para todo $j \in M$, $\delta_{\sigma^*}(t) - \delta_j(t) < h$, $t \in [t_2, T)$, lo que de acuerdo con (3.23) implica que σ es constante en $[T^*, T)$ y no cambia más hasta $t = T$.

Observación 3.11 *Del inciso i) se concluye que δ_{σ^*} esta acotada en $[0, T)$, y como δ es una función del error de identificación, entonces podemos decir que el proceso de identificación es estable. Por otro lado, del inciso ii) se tiene que σ es constante en $[T^*, T)$, lo que significa que el modelo se mantiene o no cambia en dicho intervalo.*

■

Capítulo 4

Control con un enfoque multimodelo neuronal

4.1 Control con una red neuronal

El progreso alcanzado en el campo de la teoría de control ha generado nuevas ideas para el control de procesos complejos, una de estas ideas es el control de procesos con un enfoque multimodelo.

El diseño de controladores con un enfoque multimodelo puede ser de utilidad para contrarrestar cambios en las condiciones de operación que modifiquen la dinámica de la planta. Un control multimodelo depende de la representación de un proceso a través de múltiples modelos y utiliza diferentes modelos para generar una señal de control que logre que el sistema siga una trayectoria deseada.

El enfoque multimodelo es poderoso para sistemas expuestos a altos valores de perturbaciones [1], ya que el sistema puede cambiar su región de operación bajo la acción de una perturbación. En [2] se ha demostrado la utilidad de esta técnica en estudios de simulación, y en el control de un proceso térmico no lineal en presencia de perturbaciones ambientales [5][6].

Los controladores convencionales pueden no tener el desempeño esperado debido a cambios en las condiciones de operación y a la presencia de no linealidades. Por ejemplo, con-

sidere el esquema de control de la figura 4-1, en donde los parámetros del controlador se sintonizan de tal forma que la salida $y(t)$, siga a la trayectoria deseada $y_d(t)$. El proceso puede ser lineal o no lineal, y puede estar sujeto a grandes perturbaciones que pueden llevar al sistema a una región de operación diferente. La perturbación puede ser una falla en el proceso, un cambio en los parámetros del proceso provocados por cambios ambientales, puede ser de tipo estocástica, o de otro tipo.

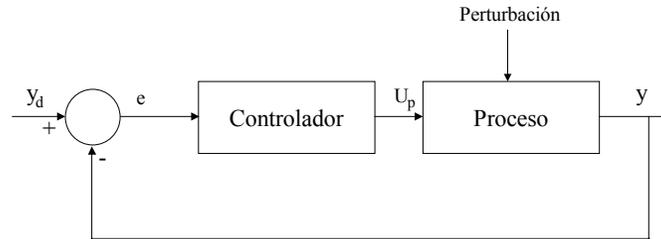


Figura 4-1: Control convencional

Un controlador para este proceso debe proporcionar una señal de control adecuada de tal manera que el proceso siga la trayectoria deseada aún después de que se presenten perturbaciones o cambios en las condiciones de operación. Los controladores convencionales, los cuales normalmente se diseñan para un cierta región de operación y para cierto tipo de perturbaciones no pueden contrarrestar grandes e inesperadas perturbaciones. Por otro lado, un controlador multimodelo puede contrarrestar diferentes perturbaciones si se construyen diferentes modelos del sistema que describan el comportamiento de la planta en distintas regiones de operación o en distintas condiciones de falla.

El control retroalimentado de sistemas no lineales es un gran reto para la ingeniería, especialmente cuando la dinámica de la planta es altamente no lineal o muy compleja, lo que dificulta la obtención de un modelo analítico. Una alternativa es identificar a la planta no lineal con una red neuronal y luego diseñar un controlador adaptable retroalimentado con ayuda del identificador neuronal. Esta idea se puede extender al diseño de controladores neuronales con un enfoque multimodelo.

El diseño de controladores neuronales es una herramienta prometedora para el control de sistemas parcial o totalmente desconocidos, debido a que esta técnica no requiere de un

modelo analítico de la planta a controlar.

4.1.1 Introducción al control neuronal

Los modelos de sistemas dinámicos y sus inversas que se obtienen a partir de redes neuronales tienen una utilidad inmediata en el diseño de esquemas de control. En la literatura se han propuesto y utilizado una gran cantidad de estructuras de control utilizando una red neuronal. A continuación se presenta una introducción a las estructuras clásicas de control utilizando redes neuronales.

Control adaptable neuronal

Las definiciones clásicas de control adaptable directo e indirecto [47] se retoman, pero ahora incluyendo a las redes neuronales como herramientas para realizar la identificación y el control de sistemas no lineales. Así, de la misma manera como en el enfoque clásico se tienen dos tipos de estructuras para resolver el problema de control adaptable: control adaptable directo e indirecto; cuando se utilizan RNA's también es posible establecer dos enfoques de control neuronal adaptable:

Control adaptable neuronal directo

En el control adaptable neuronal directo, no se realiza la identificación de la planta sino que el control se lleva a cabo directamente por una red neuronal.[52].

Control adaptable neuronal indirecto

En este enfoque de control neuronal, la planta se identifica en línea con una red neuronal y la dinámica del controlador se ajusta con la ayuda del modelo de identificación neuronal de la planta [37].

En este trabajo se utiliza el enfoque de control adaptable indirecto, utilizando, primero una, y luego múltiples redes neuronales dinámicas (modelos neuronales), para identificar el comportamiento de la planta en sus diferentes regiones de operación.

Control supervisado

Existen muchos casos en donde seres humanos realizan acciones de control retroalimentado en una tarea en particular o en donde se ha tenido dificultad en el diseño de controladores automáticos con técnicas de control convencionales, por ejemplo, en casos en donde

es imposible obtener un modelo analítico de la planta a controlar. Por lo anterior, es deseable diseñar un controlador automático que imite las acciones de control de un ser humano [82]. Entrenar una red neuronal es equivalente a obtener un modelo directo de un sistema, tal como se describe en la sección 3.1. La diferencia es que, ahora las entradas a la red corresponden a la información sensorial que recibe el humano. Las salidas deseada de la red que se utilizan en el aprendizaje corresponden a la entrada de control del humano hacia el sistema. Un control neuronal supervisado puede ser capaz de imitar las acciones de un ser humano [38].

Control inverso directo

El control inverso directo utiliza un modelo inverso de la dinámica de la planta. El modelo inverso se conecta en cascada a la planta a controlar con la finalidad de que el sistema compuesto de por resultado un mapeo identidad entre la respuesta deseada (la entrada a la red neuronal) y la salida del sistema a controlar. De esta manera la red neuronal trabaja directamente como controlador. Este esquema de control es común en aplicaciones de robótica [41].

Este enfoque depende en gran medida de la calidad del modelo inverso que se utiliza como controlador. Una de los principales inconvenientes que presenta es la pobre robustez, lo cual puede atribuirse principalmente a que no tiene retroalimentación.

Control con modelo de referencia

En este caso el desempeño deseado del sistema en lazo cerrado se especifica a través de un modelo de referencia estable M , el cuál se define por el par $\{r(t), y^r(t)\}$. El sistema de control tiene como objetivo lograr que la salida de la planta $y^p(t)$ se aproxime asintóticamente a la salida del modelo de referencia, es decir

$$\lim \|y^r(t) - y^p(t)\| \leq \epsilon \quad (4.1)$$

para alguna constante $\epsilon \geq 0$.

En esta estructura [52], el error (4.1) se utiliza para entrenar a la red neuronal que esta trabajando como controlador. Este enfoque esta relacionado con el entrenamiento del modelo

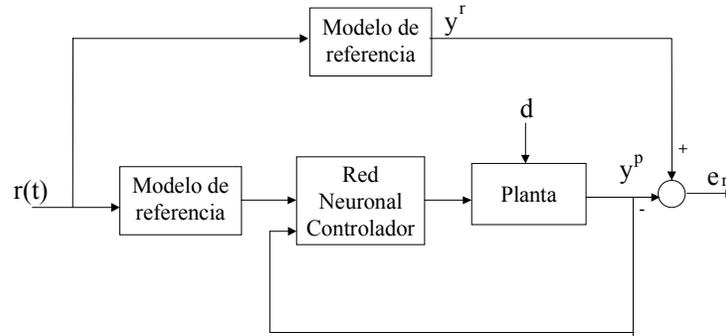


Figura 4-2: Estructura de control con modelo de referencia

inverso de la planta en donde el controlador neuronal se conecta en cascada con la planta. Cuando el modelo de referencia es un mapeo identidad ambos enfoques coinciden. En general el entrenamiento forzará al controlador neuronal a ser un modelo inverso "des-sintonizado", en el sentido definido por el modelo de referencia.

Control con modelo interno

En este caso se utilizan los modelos directo e inverso de la planta dentro del lazo de retroalimentación (figura 4-3). El modelo (directo) del sistema se conecta en paralelo con el sistema real y la diferencia entre la salida del modelo y del sistema se utiliza como señal de retroalimentación [43]. Esta señal de retroalimentación se procesa por un subsistema controlador que esta relacionada con el modelo inverso de la planta.

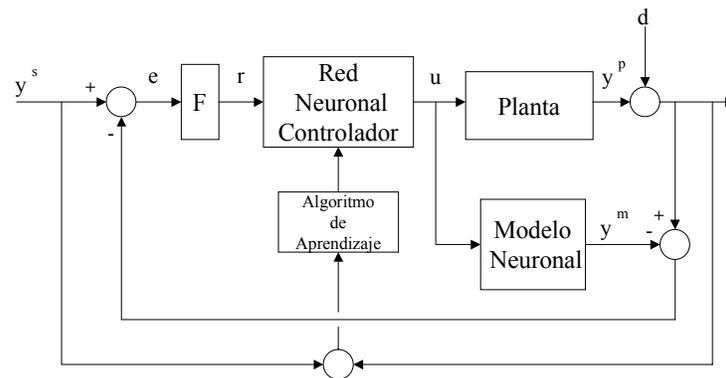


Figura 4-3: Estructura de control con modelo interno

Una de las ventajas de este enfoque es que su construcción es directa si se dispone de los modelos neuronales directo e inverso. El entrenamiento del controlador neuronal se basa en el esquema de aprendizaje inverso especializado [25], en donde la entrada del controlador es una señal de entrenamiento que recorre el espacio de salida deseado (señal de referencia). La señal de error para el algoritmo de entrenamiento es la diferencia entre la señal de entrenamiento (referencia) y la salida del sistema. Cuando se tiene una planta con ruido se puede tomar la diferencia entre la señal de entrenamiento y la salida del modelo directo. El subsistema F normalmente es un filtro lineal diseñado para introducir propiedades de robustez y seguimiento al sistema en lazo cerrado. Es importante notar que la implementación de esta estructura de control se limita a sistemas estables en lazo abierto. Esta técnica se ha utilizado ampliamente en el control de procesos.

4.1.2 Control adaptable con redes neuronales dinámicas

La estructura de control con una red neuronal dinámica se presenta en la figura 4-4.

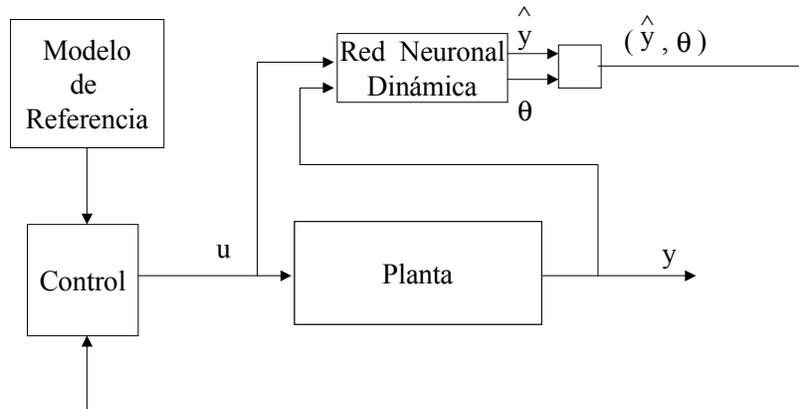


Figura 4-4: Estructura de control con una red neuronal dinámica

En esta figura se puede observar como la red neuronal dinámica además de proporcionar un estimado de la salida de la planta \hat{y} , también manda información sobre su dinámica interna θ_t para el cálculo de la señal de control. El controlador utiliza esta información de la red neuronal, así como el modelo de referencia para hacer que la salida de la planta y , siga la trayectoria deseada.

Sea el sistema no lineal a controlar representado por

$$\dot{x}_t = f(x_t, u_t, t), \quad x_t \in R^n, \quad u_t \in R^m, \quad n \geq m \quad (4.2)$$

y considere la siguiente estructura paralelo de la red neuronal dinámica [79].

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + u_t \quad (4.3)$$

en donde $\hat{x}_t \in R^n$ es el estado de la red neuronal, $W_{1,t} \in R^{n \times n}$ es la matriz de retroalimentación de estado no lineal, y $A \in R^{n \times n}$ es una matriz estable. Se considera que las funciones vectoriales $\sigma(\cdot) \in R^n$ sus elementos crecen monótonicamente. Dichos elementos $\sigma_i(\cdot)$ normalmente son funciones sigmoideas.

$$\sigma_i(x_i) = \frac{a_i}{1 + e^{-b_i x_i}} - c_i$$

La estructura de esta red neuronal dinámica se muestra en la figura (4-4).

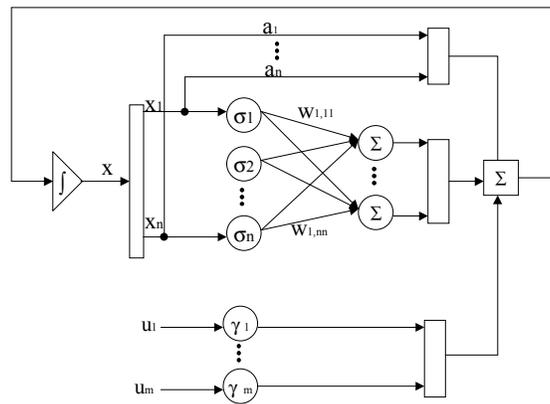


Figura 4-5: Red neuronal dinámica de una capa

Observación 4.1 *Esta red neuronal dinámica es una de las más sencillas ya que no tiene capas ocultas. Se utiliza esta red neuronal porque se desea hacer al controlador neuronal más confiable. Este modelo puede incrementar la dinámica no modelada, sin embargo, es posible compensarla. Los pesos de esta red neuronal dinámica no convergen a alguna constante, como en el perceptron multicapa, pero es posible probar que están acotados. El controlador se adaptará en línea de tal manera que no es necesario entrenar a la red neuronal fuera de línea.*

Si se define el error de identificación como:

$$\Delta_t := x_t - \hat{x}_t$$

Si $\sigma_i(\cdot)$ se seleccionan como funciones sigmoideas, se satisface la siguiente consideración.

A1 : La función $\sigma(\cdot)$ satisface la condición de sector:

$$\tilde{\sigma}_t^T \Lambda_\sigma \tilde{\sigma}_t \leq \Delta_t^T D_\sigma \Delta_t$$

donde $\tilde{\sigma}_t := \sigma(x_t) - \sigma(\hat{x}_t)$, Λ_σ y D_σ son matrices constantes definidas positivas y conocidas.

En el caso general cuando la red neuronal (4.3) no es capaz de modelar exactamente al sistema no lineal (4.2), entonces el sistema no lineal puede representarse de la siguiente manera:

$$\dot{x}_t = Ax_t + W_1^* \sigma(x_t) + u_t + \Delta f(x_t, u_t, t) \quad (4.4)$$

en donde $\Delta f(x_t, u_t, t)$ es el error de modelado, y W_1^* es cualquier matriz constante, acotada por

$$W_1^* \Lambda_\sigma^{-1} W_1^{*T} \leq \bar{W}_1$$

en donde Λ_σ y \bar{W}_1 son matrices conocidas a priori.

A2 Considere que la dinámica no modelada $\Delta f(x_t, u_t, t)$ esta acotada de la siguiente manera

$$\|\Delta f(x_t, u_t, t)\| \leq \|\eta_\sigma\|_{\Lambda_1} = \eta_\sigma^T \Lambda_1 \eta_\sigma \leq \bar{\eta}$$

en donde la matriz $\eta_\sigma \in R^n$ es una matriz diagonal y $\bar{\eta}$ es la cota superior del error de modelado.

Ahora, si la matriz A es estable, el par $(A, R^{1/2})$ es controlable, el par $(Q, R^{1/2})$ es observable, y la condición de frecuencia local satisface la siguiente expresión [62]:

$$A^T R^{-1} A - Q \geq \frac{1}{4} [A^T R^{-1} - R^{-1} A] R [A^T R^{-1} - R^{-1} A]^T \quad (4.5)$$

entonces la siguiente ecuación matricial de Riccati

$$A^T P + PA + PRP + Q = 0 \quad (4.6)$$

tiene una solución positiva $P = P^T > 0$.

A3 Lo anterior permite considerar que existe una matriz estrictamente definida positiva Q_1 , tal que si las matrices R y Q se definen como

$$R := \bar{W}_1 + \Lambda_1^{-1}, \quad Q := Q_1 + D_\sigma$$

la ecuación matricial de Riccati (4.6) tiene una solución positiva. Estas condiciones se satisfacen si se selecciona A como una matriz diagonal estable.

Si la ley de aprendizaje dinámico está representada por la siguiente ecuación diferencial

$$\dot{W}_{1,t} = -K_1 P \Delta_t \sigma^T \quad (4.7)$$

en donde $K_1 \in R^{n \times n}$ es una matriz definida positiva, P es la solución de la ecuación matricial de Riccati (4.6), y $W_{1,0} = W_1^*$ son las condiciones iniciales.

Observación 4.2 *Esta ley de aprendizaje puede garantizar la estabilidad asintótica global del error debido a que, como se ilustra en el siguiente teorema, esta ley se obtiene con el enfoque de Lyapunov. Así que, el problema de mínimo local relacionado con el aprendizaje de redes neuronales estáticas no se presenta en este caso.*

Control Adaptable robusto

El diseño del controlador que se utiliza en este trabajo se enmarca dentro del concepto de control adaptable neuronal indirecto, en donde la ley de control se construye con un modelo de identificación de la planta que se obtiene con una red neuronal dinámica como (4.3).

De (4.4) se sabe que el sistema no lineal (4.2) se puede modelar como:

$$\dot{x}_t = Ax_t + W_1^* \sigma(x_t) + u_t + \Delta f(x_t, u_t, t) \quad (4.8)$$

sumando y restando $W_{1,t} \sigma(\hat{x}_t)$ y $W_1^* \sigma(x_t)$, es posible escribir (4.8) de la siguiente manera

$$\dot{x}_t = Ax_t + W_{1,t} \sigma(\hat{x}_t) + u_t + \Delta f + \tilde{W}_{1,t} \sigma(\hat{x}_t) + W_1^* \tilde{\sigma} \quad (4.9)$$

Si la ley de adaptación es $\dot{W}_{1,t} = -K_1 P \Delta_t \sigma^T$, entonces por el corolario 3.2, $W_{1,t}$ está acotada. Además si se utiliza A1 y A2, se tiene que

$$\Delta f + \tilde{W}_{1,t} \sigma(\hat{x}_t) + W_1^* \tilde{\sigma} \quad (4.10)$$

también está acotado.

Por lo anterior es posible escribir (4.9) de la siguiente manera

$$\dot{x}_t = Ax_t + W_{1,t}\sigma(\hat{x}_t) + u_t + d_t \quad (4.11)$$

en donde

$$d_t = \Delta f(x_t, u_t, t) + \tilde{W}_{1,t}\sigma(\hat{x}_t) + W_1^* \tilde{\sigma} \quad (4.12)$$

esta acotado por $\bar{d} = \sup \|d_t\|$.

El objetivo de un control adaptable es forzar al sistema no lineal (4.2) a seguir una trayectoria óptima $x_t^* \in R^r$, la cual se considera suficientemente suave. Esta trayectoria es la solución del modelo de referencia no lineal

$$\dot{x}_t^* = \varphi(x_t^*, t), \quad x^*(0) = x_0^* \quad (4.13)$$

con condición inicial fija x_0^* . Si la trayectoria tiene puntos de discontinuidad en algunos instantes fijos, es posible sustituirla por alguna trayectoria aproximada suficientemente suave.

Observación 4.3 *La consideración implícita en la formulación del modelo de referencia es que el diseñador conoce suficientemente la planta de tal forma que puede especificar el comportamiento deseado de la planta en términos de la salida del modelo de referencia. Si estamos frente a un problema de regulación, entonces el modelo de referencia estará definido por: $\varphi(x_t^*, t) = 0$, y $x^*(0) = c$, en donde c es una constante que indica el punto de ajuste de la planta. Si el problema es de seguimiento, entonces la solución del modelo de referencia (4.13) x^* , definirá la trayectoria deseada de la respuesta de la planta.*

Si se define el error de trayectoria de estado como

$$\Delta_t^* = x_t - x_t^*$$

de las ecuaciones (4.11) y (4.13), se tiene la dinámica del error de trayectoria

$$\dot{\Delta}_t^* = Ax_t + W_{1,t}\sigma(\hat{x}_t) + u_t + d_t - \varphi(x_t^*, t) \quad (4.14)$$

Ahora considere que la ley de control u_t , es la suma de dos controles

$$u_t = u_{1,t} + u_{2,t} \quad (4.15)$$

en donde $u_{1,t} \in R^n$ es la parte de linealización directa y $u_{2,t} \in R^n$ se encarga de la compensación de la dinámica no modelada d_t .

Como se dispone de $\varphi(x_t^*, t)$, x_t^* y $W_{1,t}\sigma(\hat{x}_t)$, se puede seleccionar $u_{1,t}$ de la siguiente forma

$$u_{1,t} = \varphi(x_t^*, t) - Ax_t^* - W_{1,t}\sigma(\hat{x}_t) \quad (4.16)$$

Al sustituir $u_{1,t}$ en la ecuación (4.14), se tiene

$$\dot{\Delta}_t^* = A\Delta_t^* + u_{2,t} + d_t \quad (4.17)$$

La segunda parte de la ley de control $u_{2,t}$ compensará la dinámica no modelada d_t , para lo cual se pueden aplicar cuatro algoritmos robustos:

(a) **Compensación exacta.**

Restando la ecuación (4.3) de (4.11), tenemos la dinámica del error de identificación

$$\dot{\Delta}_t = A\Delta_t + d_t$$

entonces, la dinámica no modelada toma la forma

$$d_t = (\dot{x}_t - \dot{\hat{x}}_t) - A(x_t - \hat{x}_t)$$

Si se dispone de la dinámica del estado \dot{x}_t , entonces se puede seleccionar $u_{2,t}^a$, de la forma

$$u_{2,t}^a = -d_t = A(x_t - \hat{x}_t) - (\dot{x}_t - \dot{\hat{x}}_t) \quad (4.18)$$

sustituyendo en (4.17) se tiene

$$\dot{\Delta}_t^* = A\Delta_t^* \quad (4.19)$$

Si A es una matriz estable se tiene que Δ_t^* es estable global y asintoticamente, es decir

$$\lim_{t \rightarrow \infty} \Delta_t^* = 0$$

Observación 4.4 *Para observar la dinámica de la planta en lazo cerrado (cuando $u_t = u_{1,t} + u_{2,t}$ se han sustituido en la ecuación (4.11)) en relación con el modelo de referencia y con los controles $u_{1,t}$ y $u_{2,t}$, apliquemos primero el control $u_{1,t}$, lo cual resulta en:*

$$\dot{x}_t = \dot{x}_t^* + A\Delta_t^* + u_{2,t} + d_t$$

en este caso, si $u_{2,t} = 0$, entonces un error de modelado d_t considerable puede hacer que el comportamiento de la planta tenga un pobre desempeño respecto del comportamiento deseado, es decir, $u_{1,t}$ puede no ser capaz de proporcionar un buen control.

Ahora considere que se aplica todo el control, es decir: $u_t = u_{1,t} + u_{2,t}$. Entonces el sistema en lazo cerrado descrito por la ecuación 4.19 toma la forma:

$$\dot{x}_t = \dot{x}_t^* - A(x_t - x_t^*)$$

esto es la dinámica en lazo cerrado de la planta, en donde se puede observar que la dinámica de la planta \dot{x}_t será igual a la dinámica del modelo de referencia \dot{x}_t^ , en la medida en que el error de seguimiento $(x_t - x_t^*)$ sea pequeño. La matriz A actúa como un factor de amplificación del error de seguimiento por lo que es deseable valores pequeños de A . Por otro lado, de la observación anterior podemos decir que si la dinámica de la planta es importante, entonces en la misma medida también lo será $u_{2,t}$.*

(b) **Compensación aproximada.**

Si no se dispone de la derivada del estado \dot{x}_t , se puede tomar una aproximación de la

derivada de x_t , de la siguiente forma

$$\dot{x}_t = \frac{x_t - x_{t-\tau}}{\tau} + \delta_t$$

donde $\delta_t > 0$, es el error de aproximación de la derivada.

Si se selecciona el control $u_{2,t}^b$ como

$$\begin{aligned} u_{2,t}^b &= A(x_t - \hat{x}_t) - \left(\frac{x_t - x_{t-\tau}}{\tau} - \dot{\hat{x}}_t \right) \\ &= A(x_t - \hat{x}_t) - (\dot{x}_t - \dot{\hat{x}}_t) + \delta_t \end{aligned} \quad (4.20)$$

entonces $u_{2,t}^b = u_{2,t}^a + \delta_t$, y la dinámica del error de trayectoria es

$$\dot{\Delta}_t^* = A\Delta_t^* + \delta_t$$

Se define una función de Lyapunov como

$$V_t = \Delta_t^{*T} P_2 \Delta_t^*, \quad P_2 = P_2^T > 0$$

y se obtiene su derivada

$$\dot{V}_t = \Delta_t^{*T} (A^T P_2 + P_2 A) \Delta_t^* + 2\Delta_t^{*T} P_2 \delta_t \quad (4.21)$$

utilizando la desigualdad 3.14, se puede escribir $2\Delta_t^{*T} P_2 \delta_t$, de la siguiente manera

$$2\Delta_t^{*T} P_2 \delta_t \leq \Delta_t^{*T} P_2 \Lambda P_2 \Delta_t^* + \delta_t^T \Lambda^{-1} \delta_t$$

en donde Λ es cualquier matriz definida positiva. De esta manera la ecuación (4.21) toma la forma

$$\dot{V}_t = \Delta_t^{*T} (A^T P_2 + P_2 A + P_2 \Lambda P_2 + Q_2) \Delta_t^* + \delta_t^T \Lambda^{-1} \delta_t - \Delta_t^{*T} Q_2 \Delta_t^*$$

en donde Q_2 es cualquier matriz definida positiva. Como A es una matriz estable, existen

Λ y Q_2 , tal que la ecuación de Riccati

$$A^T P_2 + P_2 A + P_2 \Lambda P_2 + Q_2 = 0 \quad (4.22)$$

tiene una solución positiva $P_2 = P_2^T > 0$.

Definiendo la siguiente seminorma

$$\|\Delta_t^*\|_{Q_2}^2 = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \Delta_t^{T*} Q_2 \Delta_t^* dt \quad (4.23)$$

en donde $Q_2 = Q_2^T > 0$ es una matriz de pesos dada, el seguimiento de trayectoria de estado puede formularse como el siguiente problema de optimización

$$J_{\min} = \min_{u_t} J, \quad J = \|x_t - x_t^*\|_{Q_2}^2$$

Note que

$$\lim_{T \rightarrow \infty} \frac{1}{T} \Delta_0^{T*} Q_2 \Delta_0^* = 0$$

Utilizando la red neuronal (4.3), la ley de control (4.20), y el control (4.16), es posible hacer que el error de seguimiento de trayectoria satisfaga la siguiente propiedad

$$\|\Delta_t^*\|_{Q_2}^2 \leq \|\delta_t\|_{\Lambda^{-1}}^2$$

en donde una elección adecuada de Λ y Q_2 pueden hacer que la ecuación de Riccati (4.22) tenga una solución positiva y que $\|\Delta_t^*\|_{Q_2}^2$ sea suficientemente pequeño si τ es suficientemente pequeño.

(c) **Compensación con la técnica de modo deslizante**

Si no se dispone de la derivada del estado, es posible aplicar la técnica de modo deslizante.

Considere una función tipo Lyapunov de la forma

$$V_t = \Delta_t^{*T} P_3 \Delta_t^* \quad (4.24)$$

en donde P_3 es la solución de la ecuación de Lyapunov

$$A^T P_3 + P_3 A = -I \quad (4.25)$$

derivando (4.24) y sustituyendo la derivada del error de trayectoria (4.17) se tiene

$$\dot{V}_t = \Delta_t^{*T} (A^T P_3 + P_3 A) \Delta_t^* + 2\Delta_t^{*T} P_3 u_{2,t} + 2\Delta_t^{*T} P_3 d_t \quad (4.26)$$

De cuerdo con la técnica de modo deslizante, se puede seleccionar $u_{2,t}$ como

$$u_{2,t}^c = -k P_3^{-1} \text{sgn}(\Delta_t^*) \quad (4.27)$$

en donde k es una constante positiva, y

$$\text{sgn}(\Delta_t^*) = [\text{sgn}(\Delta_{1,t}^*), \dots, \text{sgn}(\Delta_{n,t}^*)]^T \in R^n$$

Sustituyendo (4.25) y (4.27) en (4.26), se tiene

$$\begin{aligned} \dot{V}_t &= -\|\Delta_t^*\|^2 - 2k \|\Delta_t^*\| + 2\Delta_t^{*T} P_3 d_t \\ &\leq -\|\Delta_t^*\|^2 - 2k \|\Delta_t^*\| + 2\lambda_{\max}(P_3) \|\Delta_t^*\| \|d_t\| \\ &= -\|\Delta_t^*\|^2 - 2\|\Delta_t^*\| (k - \lambda_{\max}(P_3) \|d_t\|) \end{aligned}$$

si se selecciona

$$k > \lambda_{\max}(P_3) \bar{d}$$

en donde \bar{d} se define como en (4.12), entonces $\dot{V}_t < 0$, y por consiguiente

$$\lim_{t \rightarrow \infty} \Delta_t^* = 0$$

(d) **Control óptimo local**

Este enfoque puede aplicarse si \dot{x}_t no esta disponible y no se aproxima como en el inciso (b). Con la finalidad de estudiar la estabilidad del error de seguimiento se introduce la siguiente función de Lyapunov

$$V_t(\Delta_t^*) = \Delta_t^{T*} P_4 \Delta_t^*, \quad P_4 = P_4^T > 0 \quad (4.28)$$

Derivando (4.28) y sustituyendo la derivada del error de seguimiento (4.17), se tiene

$$\dot{V}_t = 2\Delta_t^{T*} P_4 \dot{\Delta}_t^* = \Delta_t^{T*} (A^T P_4 + P_4 A) \Delta_t^* + 2\Delta_t^{T*} P_4 u_{2,t} + 2\Delta_t^{T*} P_4 d_t \quad (4.29)$$

el termino $2\Delta_t^{T*} P_4 d_t$, puede estimarse utilizando la desigualdad (3.14)

$$2\Delta_t^{T*} P_4 d_t \leq \Delta_t^{T*} P_4 \Lambda_4^{-1} P_4 \Delta_t^* + d_t^T \Lambda_4 d_t \quad (4.30)$$

Sustituyendo (4.30) en (4.29), y sumando y restando los terminos $\Delta_t^{T*} Q_4 \Delta_t^*$ y $u_{2,t}^{dT} R_4 u_{2,t}^d$, en donde $Q_4 = Q_4^T > 0$ y $R_4 = R_4^T > 0$, podemos escribir

$$\begin{aligned} \dot{V}_t \leq & \Delta_t^{T*} (A^T P_4 + P_4 A + P_4 \Lambda_4 P_4 + Q_4) \Delta_t^* + 2\Delta_t^{T*} P_4 u_{2,t}^d \\ & + u_{2,t}^{dT} R_4 u_{2,t}^d + d_t^T \Lambda_4^{-1} d_t - \Delta_t^{T*} Q_4 \Delta_t^* - u_{2,t}^{dT} R_4 u_{2,t}^d \end{aligned} \quad (4.31)$$

Como A es una matriz estable, existen matrices Λ_4 y Q_4 , tal que la ecuación matricial de Riccati

$$A^T P_4 + P_4 A + P_4 \Lambda_4 P_4 + Q_4 = 0 \quad (4.32)$$

tiene una solución definifda positiva $P_4 = P_4^T$.

con lo cual, la ecuación (4.31) se reduce a

$$\dot{V}_t \leq -(\|\Delta_t^*\|_{Q_4}^2 + \|u_{2,t}^d\|_{R_4}^2) + \Psi(u_{2,t}^d) + d_t^T \Lambda_4^{-1} d_t \quad (4.33)$$

$$\text{en donde } \Psi(u_{2,t}^d) = 2\Delta_t^{T*} P_4 u_{2,t}^d + u_{2,t}^{dT} R_4 u_{2,t}^d \quad (4.34)$$

Si escribimos (4.33), de la siguiente manera

$$\|\Delta_t^*\|_{Q_4}^2 + \|u_{2,t}^d\|_{R_4}^2 \leq \Psi(u_{2,t}^d) + d_t^T \Lambda_4^{-1} d_t - \dot{V}_t$$

entonces, integrando cada termino de $0 \rightarrow T$, dividiendo cada termino por T , y tomando el límite del supremo de estas integrales, cuando $T \rightarrow \infty$, se obtiene

$$\begin{aligned} & \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T \Delta_t^{T*} Q_4 \Delta_t^* dt + \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_{2,t}^{dT} R_4 u_{2,t}^d \\ & \leq \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T d_t^T \Lambda_4^{-1} d_t + \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T (\Psi(u_{2,t}^d)) dt + \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T (-\dot{V}_t) dt \end{aligned}$$

Tomando las definiciones de seminorma (4.23), se tiene

$$\|\Delta_t^*\|_{Q_4}^2 + \|u_{2,t}^d\|_{R_4}^2 \leq \|d_t\|_{\Lambda_4^{-1}}^2 + \overline{\lim}_{T \rightarrow \infty} \frac{1}{T} \int_0^T (\Psi(u_{2,t}^d)) dt$$

lo cual fija un nivel de tolerancia para el error de trayectoria de seguimiento. De esta manera, ahora el objetivo de control es minimizar $\Psi(u_{2,t}^d)$ y $\|d_t\|_{\Lambda_4^{-1}}^2$. Para minimizar $\|d_t\|_{\Lambda_4^{-1}}^2$, se debe de minimizar Λ_4^{-1} . De la ecuación (4.5), si seleccionamos Q_4 , tal que (4.32) tenga solución, entonces se puede seleccionar el mínimo Λ_4^{-1} , como

$$\Lambda_4^{-1} = A^{-1T} Q_4 A^{-1}$$

Para minimizar $\Psi(u_{2,t}^d)$, considere que en un tiempo $t > 0$, $x^*(t)$ y $\hat{x}(t)$ ya existen y no dependen de $u_{2,t}^d$. Entonces la solución de este problema de optimización esta dada por

$$\Psi(u_{2,t}^d) = 2\Delta_t^{T*} P_4 u_{2,t}^d + u_{2,t}^{dT} R_4 u_{2,t}^d \quad (4.35)$$

$$\text{sujeto a } A_0(u_{1,t} + u_{2,t}^d) \leq B_0$$

Cuando no se tienen restricciones $u_{2,t}^d$ se selecciona de acuerdo con ley de control óptimo cuadrática lineal.

$$u_{2,t}^d = -2R_4^{-1}P_4\Delta_t^*$$

La cual al sustituirla en en la ecuación (4.35) hace que $\Psi(u_{2,t}^d) = 0$. A $u_{2,t}^d$ se le llama control óptimo local debido a que se calcula utilizando información local.

Observación 4.5 *Los enfoques (a) y (c) son compensaciones exactas de d_t . El enfoque (a), necesita información de \dot{x}_t . Debido a que el enfoque (c) utiliza un control en modo deslizante, $u_{2,t}^c$ se inserta en el sistema en lazo cerrado y puede presentarse vibración en la entrada de control lo cual puede excitar dinámicas no modeladas de alta frecuencia.*

4.2 Control con un enfoque multimodelo neuronal

Un controlador multimodelo general consiste de cuatro subsistemas

1. Múltiples estimadores neuronales (ver Cap. 3). Este subsistema puede producir la mejor estimación de la dinámica de la planta, de tal manera que la salida del estimador multimodelo puede seguir a la salida de la planta real con un mínimo error de identificación. La estructura de este subsistema se muestra en la figura 4-6.
1. Controlador multimodelo neuronal. Este subsistema es un controlador que se basa en un estimador multimodelo neuronal en donde el estimador proporciona información de la red neuronal dinámica que mejor aproxima al proceso y con una técnica de control apropiada se puede producir el mejor control para la planta. La estructura de este subsistema se muestra en la figura 4-7.
1. Generador de índices de desempeño. Este subsistema coleccionará información del estimador multimodelo, como por ejemplo, los errores de identificación e_i , $i = 1, \dots, N$, y generará una nueva señal $\sigma \in \{1, \dots, N\}$, que será fácilmente utilizada por el selector. La estructura de este subsistema se muestra en la figura ??.

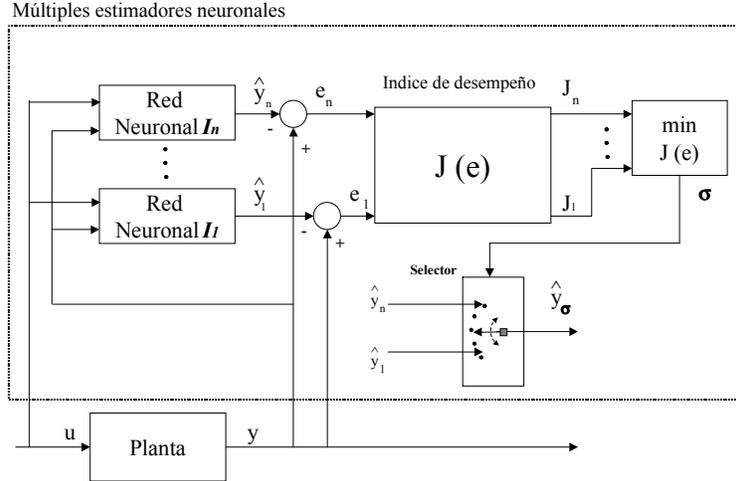


Figura 4-6: Múltiples estimadores neuronales

2. Lógica de selección. Este subsistema puede utilizar el índice de desempeño y junto con una lógica interna, puede proporcionar una señal de mando que decida que modelo debe seleccionarse. La estructura de este subsistema se muestra en la figura 4-9.

La motivación detrás del esquema de control con un estimador multimodelo es la siguiente: *utilizando un índice de desempeño adecuado, se espera que el mejor estimador de la planta realice el mejor trabajo en el control del proceso.*

Se pueden considerar tres esquemas diferentes en un controlador multimodelo neuronal:

4.2.1 Identificador neuronal y controlador multimodelo neuronal

La estructura se muestra en la figura 4-10.

En la ecuación (4.11) se establece que el sistema no lineal puede escribirse como

$$\dot{x}_t = Ax_t + W_{1,t}\sigma(\hat{x}_t) + u_t + \tilde{f} \quad (4.36)$$

en donde \tilde{f} es la dinámica no modelada de la planta y la ley de actualización de pesos está dada por la ecuación (5.10). Considerando que la dinámica no modelada está acotada

podemos escribir la ecuación anterior como:

$$\dot{x}_t = Ax_t + W_{1,t}\sigma(\hat{x}_t) + u_t + d_t$$

en donde d_t esta acotada de acuerdo con la ecuación (4.10). Si d_t esta acotada y la ley de actualización de pesos es como (5.10), entonces por el corolario 3.2 el procedimiento de identificación es estable y $W_{1,t}$ esta acotada.

Se debe de forzar al sistema no lineal a que siga una trayectoria óptima $x_t^* \in \mathfrak{R}^r$, la cuál se asume que debe ser suficientemente suave. Esta trayectoria es la solución del modelo de referencia

$$\dot{x}_t^* = \varphi(x_t^*, t), \quad x^*(0) = x_0^* \quad (4.37)$$

con condición inicial fija. Si la trayectoria tiene puntos de discontinuidad en algunos instantes fijos, es posible utilizar una trayectoria aproximada que sea suave.

Observación 4.6 *El modelo de referencia especifica como la salida del proceso debe responder (idealmente) a la señal de comando r_t . En un control retroalimentado la mayoría de las veces esta señal de comando es la trayectoria deseada de la respuesta de la planta. En este trabajo se considera que se conoce dicha trayectoria la cual esta definida por la solución del modelo de referencia (4.37).*

Si se define el error de seguimiento de trayectoria como

$$\Delta_t^* = x_t - x_t^*$$

y el error de identificación es

$$\Delta_t = \hat{x}_t - x_t$$

La dinámica del error de seguimiento es

$$\dot{\Delta}_t^* = Ax_t + W_{1,t}\sigma(\hat{x}_t) + u_t + d_t - \varphi(x_t^*, t) \quad (4.38)$$

Ahora considere que la acción de control u_t consta de dos partes:

$$u_t = u_{1,t} + u_{2,t}$$

en donde $u_{1,t} \in \mathfrak{R}^n$ es la parte de linealización directa y $u_{2,t} \in \mathfrak{R}^n$ es la señal de control que se encarga de la compensación de la dinámica no modelada d_t .

Como $\varphi(x_t^*, t)$, x_t^* y $W_{1,t}\sigma(\hat{x}_t)$ están disponibles se puede seleccionar $u_{1,t}$ como:

$$u_{1,t} = \varphi(x_t^*, t) - Ax_t - W_{1,t}\sigma(\hat{x}_t)$$

y sustituyendo $u_{1,t}$ en la dinámica del error de seguimiento dada en la ecuación (4.38), se tiene:

$$\dot{\Delta}_t^* = A\Delta_t + u_{2,t} + d_t$$

El diseño de $u_{2,t}$ se puede hacer con el enfoque multimodelo con la finalidad de cancelar de una manera efectiva a la dinámica no modelada d_t . En la compensación de d_t se pueden aplicar dos técnicas: un control de modos deslizantes y un control óptimo local.

El control de modos deslizantes utiliza una alta ganancia y una técnica de estructura variable. Puede compensar incertidumbres con cualquier exactitud, sin embargo, es sensible al ruido y la posibilidad de vibraciones limita su aplicación. Por otro lado, el control óptimo local es un control suave, pero su exactitud no es muy buena.

Si se utiliza el enfoque multimodelo en el diseño del controlador podemos tomar las ventajas de los dos tipos de control mencionados y superar sus desventajas. De esta manera la lógica de selección tiene dos fases:

- Si el error de identificación $\|\Delta_t\| \geq S$, ($S > 0$ es una constante que opera como el umbral de selección), entonces $u_{2,t}$ se calcula a partir la ley de control óptimo local:

$$u_{2,t} = -2R^{-1}P\Delta_t^*$$

que es la solución del problema de optimización sin restricciones establecido en la ecuación

(4.35).

- Si el error de identificación $\|\Delta_t\| < S$, entonces la ley de control $u_{2,t}$ se calcula con la técnica de modos deslizantes

$$u_{2,t} = -kP^{-1}sgn(\Delta_t^*), \quad k > 0$$

4.2.2 Identificador multimodelo y un controlador neuronal

Este control es un control adaptable indirecto en donde se utilizan múltiples redes neuronales dinámicas para estimar el comportamiento de la planta no lineal. Esta estructura se muestra en la figura 4-11

El identificador multimodelo neuronal tiene la forma

$$\dot{\hat{x}}_t = A_\sigma \hat{x}_t + W_{1,t}^\sigma \sigma_\sigma(\hat{x}_t) + u_t \quad (4.39)$$

en donde $\sigma \in M = \{1, 2, \dots, m\}$ es la señal de mando al selector. Se selecciona el índice de desempeño de cada identificador como

$$\delta_i(t) = \alpha \Delta_i^2(t) + \beta \int_0^t \Delta_i^2(\tau) d\tau$$

en donde $\alpha \geq 0, \beta > 0$ y $\lambda > 0$ son parámetros de diseño, y $\Delta_i(t)$ es el error de identificación de i -ésimo modelo. La lógica de selección es el siguiente algoritmo de histéresis

$$\phi(i, \delta) = \begin{cases} i & \text{Si } \delta_i < \delta_{\rho(\delta)} + h \quad (i \text{ permanece}) \\ \rho(\delta) & \text{Si } \delta_i \geq \delta_{\rho(\delta)} + h \quad (i \text{ cambia a } \rho(\delta)) \end{cases}$$

en donde h es la constante de histéresis.

De acuerdo con la ecuación (4.11) la planta no lineal puede escribirse como:

$$\dot{x}_t = A_\sigma x_t + W_{1,t}^\sigma \sigma_\sigma(\hat{x}_t) + u_t + d_t$$

en donde d_t es la dinámica no modelada de la planta y σ es un índice que señala al mejor identificador dentro del multimodelo.

De la ecuación (4.12) se sabe que d_t esta acotada y la acción de control u_t tiene dos partes:

$$u_t = u_{1,t} + u_{2,t}$$

en donde $u_{1,t} \in R^n$ es la parte de linealización directa y $u_{2,t} \in R^n$ es la compensación de la dinámica no modelada d_t .

$$u_{1,t} = \varphi(x_t^*, t) - A_\sigma x_t^* - W_{1,t}^\sigma \sigma_\sigma(\hat{x}_t)$$

aquí, A_σ , $W_{1,t}^\sigma$ y $\sigma_\sigma(\hat{x}_t)$ se toman del mejor identificador neuronal σ , de tal forma que Δ_t^*

$$\dot{\Delta}_t^* = A_\sigma \Delta_t^* + u_{2,t} + d_t$$

$u_{2,t}$ se diseña de la siguiente forma para compensar d_t

$$u_{2,t} = A(x_t - \hat{x}_t) - \left(\frac{x_t - x_{t-\tau}}{\tau} - \dot{\hat{x}}_t \right).$$

4.2.3 Identificador multimodelo y controlador multimodelo

Esta estructura es más flexible en el diseño de un controlador multimodelo neuronal, ya que es una combinación de los dos enfoques anteriores. La estructura se muestra en la figura 4-12.

La planta no lineal puede escribirse como

$$\dot{x}_t = A_\sigma x_t + W_{1,t}^\sigma \sigma_\sigma(\hat{x}_t) + u_t + d_t$$

en donde d_t es la dinámica no modelada de la planta y σ es un índice que señala al mejor identificador dentro del multimodelo.

La ley de control u_t tiene dos partes

$$u_t = u_{1,t} + u_{2,t}$$

en donde $u_{1,t} \in R^n$ es la parte de linealización directa y $u_{2,t} \in R^n$ es la compensación de la dinámica no modelada d_t .

$$u_{1,t} = \varphi(x_t^*, t) - A_\sigma x_t^* - W_{1,t}^\sigma \sigma_\sigma(\hat{x}_t)$$

Para el diseño de $u_{2,t}$ se aplica un enfoque multimodelo con la finalidad de cancelar de una manera efectiva d_t . La lógica de control de nuevo consta de dos fases:

- Si el error de identificación $\|\Delta_t\| \geq S$, ($S > 0$ es una constante que opera como el umbral de selección), entonces $u_{2,t}$ se calcula a partir la ley de control óptimo local:

$$u_{2,t} = -2R^{-1}P\Delta_t^*$$

que es la solución del problema de optimización sin restricciones (ecuación (4.35)).

- Si el error de identificación $\|\Delta_t\| < S$, entonces la ley de control $u_{2,t}$ se calcula con la técnica de modos deslizantes

$$u_{2,t} = -kP^{-1}\text{sgn}(\Delta_t^*), \quad k > 0$$

Observación 4.7 *El enfoque de multiples controladores neuronales esta dentro del campo del control adaptable indirecto. La filosofia de operación es la siguiente: en primer lugar, es necesario obtener un modelo del subsistema de estimación neuronal, entonces en base en dicho modelo se diseña la ley de control adaptable. El control esta conformado de dos partes: la parte de linealización directa y la parte de compensación de incertidumbres.*

- **Identificador neuronal y controlador multimodelo neuronal.** *En el primer enfoque la parte de linealización directa se toma del identificador neuronal, mientras*

que la compensación de incertidumbres utiliza la técnica multimodelo. Este enfoque puede superar una pobre respuesta transitoria ocasionada por el compensador.

- **Identificador multimodelo y un controlador neuronal.** En el segundo enfoque el controlador hace la linealización directa en base a la información del mejor modelo de identificación que le proporciona el multimodelo, mientras que para la compensación de incertidumbres utiliza una técnica de control clásica. Este enfoque puede mejorar la pobre respuesta transitoria producida cuando se utiliza sólo un identificador neuronal en procesos no estacionarios.
- **Identificador multimodelo y controlador multimodelo.** En el tercer enfoque el control de linealización directa se calcula en base al identificador multimodelo neuronal, y el control que se encarga de la compensación de incertidumbres también utiliza un enfoque multimodelo. Este enfoque puede superar la pobre respuesta transitoria producida tanto por el compensador como por utilizar sólo un modelo de identificación. Sin embargo, el proceso de diseño puede ser complejo.

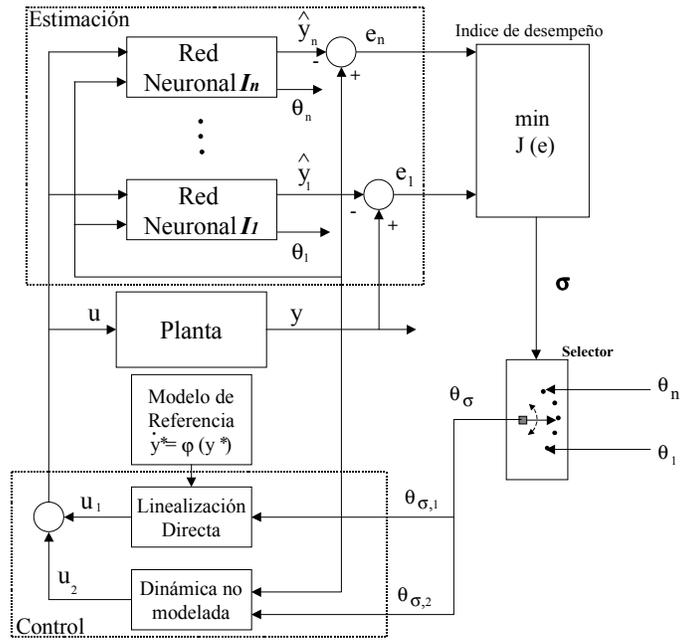


Figura 4-7: Controlador con un identificador multimodelo neuronal

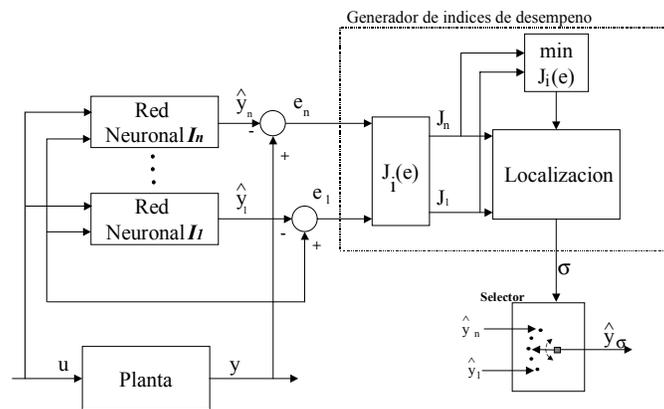


Figura 4-8: Generador de índices de desempeño

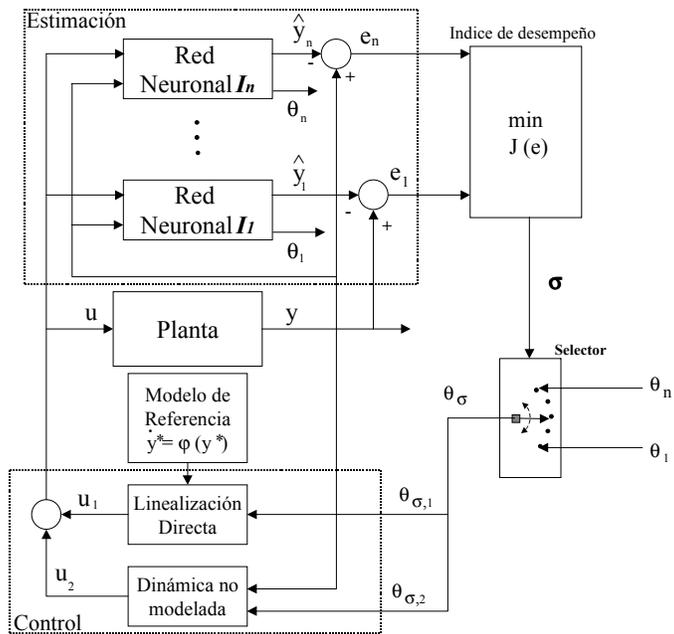


Figura 4-11: Múltiples identificadores y un control

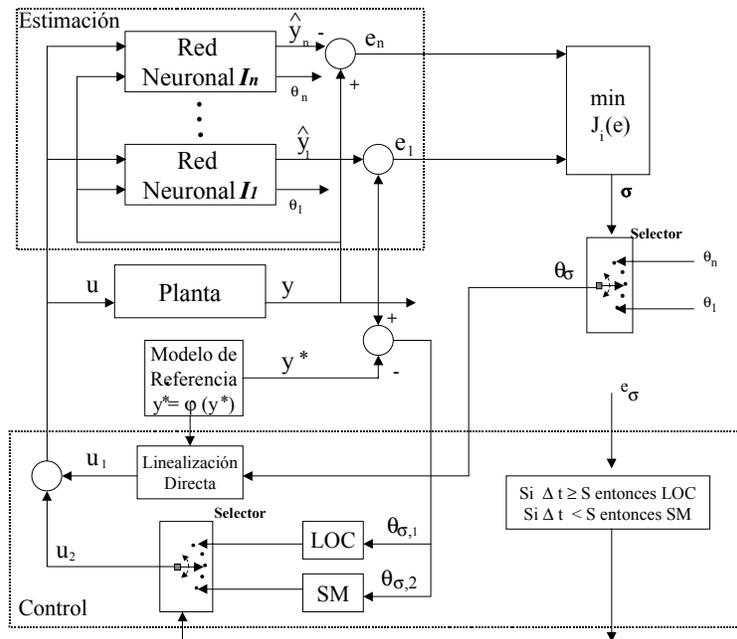


Figura 4-12: Identificación multimodelo y de control multimodelo.

Capítulo 5

Identificación y control de un proceso de neutralización de pH con múltiples RNA's dinámicas

Los procesos de neutralización de pH pueden agruparse de la siguiente manera:

- En primer lugar se puede hablar de los procesos de neutralización de pH que operan satisfactoriamente. Sin embargo, siempre existe la posibilidad de mejorar el desempeño de estas plantas con un diseño de la planta y del sistema de control menos costoso y más sencillo.
- También existen procesos de neutralización de pH que experimentan problemas de control, como por ejemplo, lenta recuperación ante perturbaciones y oscilaciones sostenida.
- Finalmente, en un tercer grupo se tienen aquellas plantas que aparentan estar operando bien, pero en realidad su desempeño es pobre, por la falta de una instrumentación adecuada.

Los problemas en el control de pH se producen principalmente por dos tipos de perturbaciones: perturbaciones en el flujo de alimentación y perturbaciones en la composición del flujo de alimentación.

En la práctica industrial las perturbaciones en el flujo de alimentación se contrarrestan por medio de un control anticipativo: midiendo las variaciones en el flujo y compensandolas con un control de balance de materia, es decir, con un control de relación. Por otro lado, las perturbaciones en la composición del flujo de alimentación son más difícil de manejar, esto se debe a que la ganancia del proceso puede cambiar en ordenes de magnitud tan sólo en unos cuantos segundos, (ver figura 5-1).

El concepto de pH definido como el logaritmo negativo de la concentración de iones hidrógeno, a menudo origina que la descripción del sistema incluya relaciones altamente no lineales.

En una solución no amortiguada, esto es, una solución que contenga sólo ácidos y bases fuertes la curva característica del pH , respecto del ácido o base que se adiciona (reactante), es una curva en forma de S en donde la pendiente varia en órdenes de magnitud, como se observa en la figura 5-1.

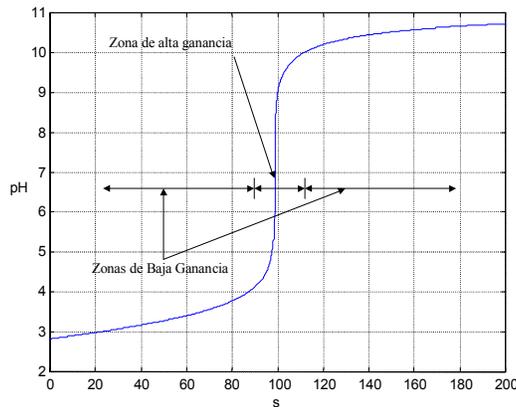


Figura 5-1: Curva típica del pH en forma de S .

Observación 5.1 *Se dice que una solución es amortiguada cuando a ésta se le adiciona una sustancia amortiguadora (buffer), que tiene la función de moderar la rapidez de los cambios en la concentración de un ión determinado. La extremada gran sensibilidad del pH cerca del punto neutro ($pH = 7$), proporciona al sistema propiedades muy específicas que dificultan el control del pH cerca de su punto neutro.*

Shinsky [67], ilustra la extrema sensibilidad que existe en soluciones de ácidos y bases fuertes con el siguiente ejemplo: considere los límites de $pH = 7 \pm 0.5$ como un blanco de $15cm$ de diámetro, entonces, llevar una corriente de desecho de un $pH = 4$ a un $pH = 7$ puede compararse con disparar al blanco a una distancia de $30m$. Pero, si se parte de un $pH = 2$, es equivalente a disparar a una distancia de $3Km$; mientras que partir de un $pH = 0$ equivale a disparar a una distancia de $300Km$ del blanco.

Otro problema importante se tiene en los elementos de medición y en los actuadores, los cuales introducen constantes de tiempo y tiempos muertos pequeños que pueden generar efectos dinámicos con repercusiones importantes en el sistema de control de pH [73]. Se sabe que introducir capacitancias (retardos de primer orden) en sistemas lineales ayuda a atenuar las oscilaciones, lo cuál permite incrementar la ganancia del controlador. Sin embargo, en sistemas de control de pH la inclusión de estas capacitancias adquiere menor importancia que la que tiene en sistemas lineales [73].

La forma en S de la curva de titulación figura 5-1, significa que la ganancia del proceso puede variar considerablemente[66]. En la práctica esta relación frecuentemente excede $1000 : 1$. La gran variación en la ganancia del proceso produce dificultades en los sistemas de control de pH , por ejemplo; sí en un sistema de control retroalimentado la ganancia del controlador se ajusta lo suficientemente baja para asegurar estabilidad alrededor del punto neutro, entonces el controlador tendrá un pobre desempeño lejos de éste punto y la recuperación a grandes perturbaciones será extremadamente lenta. Por otro lado, un controlador de ganancia fija que tenga un buen desempeño en presencia de grandes perturbaciones puede ocasionar oscilaciones continuas alrededor del punto neutro debido a que la ganancia del controlador es muy alta para mantener estabilidad en éste punto.

De esta manera, puede considerarse que el problema principal en el control continuo de pH puede ser provocado por fuertes variaciones en la ganancia del sistema. De aquí que un control con un buen desempeño en una situación puede ser de poco valor en otra situación.

Las dificultades para contrarrestar perturbaciones con un controlador retroalimentado surgen principalmente de las no linealidades de los sistemas de pH , en combinación con la segura existencia de constantes de tiempo y tiempos muertos. Comunmente éstos efectos dinámicos pequeños, introducidos por el elemento de medición y el actuador, puedan tener

un fuerte impacto en el desempeño de un control de pH .

La teoría de control de invariantes de reacción es otro enfoque en el control de pH [20]. Utilizando los conceptos de invariantes de reacción es posible obtener un modelo del sistema de pH [20]. En este caso, utilizar una retroalimentación de ganancia lineal corresponde, visto desde el punto de vista del pH , a utilizar un controlador de ganancia no lineal [73]. Este enfoque utiliza un control PID lineal, pero la diferencia con el enfoque tradicional en donde la variable que se retroalimenta es el pH , es que aquí se retroalimenta una variable llamada "invariante de reacción". El invariante de reacción se calcula a partir de mediciones de pH y de un modelo del sistema [20]. El principal obstáculo de aplicar el enfoque de invariantes de reacción, es obtener información relevante lo suficientemente rápido acerca de como cambian los invariantes de reacción cuando cambia la capacidad de amortiguamiento.

5.1 Proceso de Neutralización de pH

En la industria la variable pH se utiliza ampliamente para medir el estado que guarda una reacción. Una de las principales características de esta variable es su naturaleza logaritmico. Desde el punto de vista de control el pH es una problema difícil, debido a que se tiene una medición severamente no lineal. Esta característica impone demandas excepcionales en el sistema de control.

Una sustancia base, es aquella que tiene una muy alta concentración de iones OH^- , respecto de la concentración de iones H^+ , mientras que una sustancia ácida es aquella que tiene una concentración muy alta de iones H^+ , respecto de la concentración de iones OH^- .

La curva de pH es la propiedad que caracteriza a todo sistema ácido-base. La figura 5-2, es una curva típica de una base a la que se le adiciona un ácido. La forma de la curva se relaciona con las constantes de equilibrio, con la ionización del ácido y de la base, así como con las concentraciones de los iones.

La base del sistema coordinado es logaritmica debido a que el pH esta definido como el logaritmo negativo de la concentración de iones de hidrógeno, es decir:

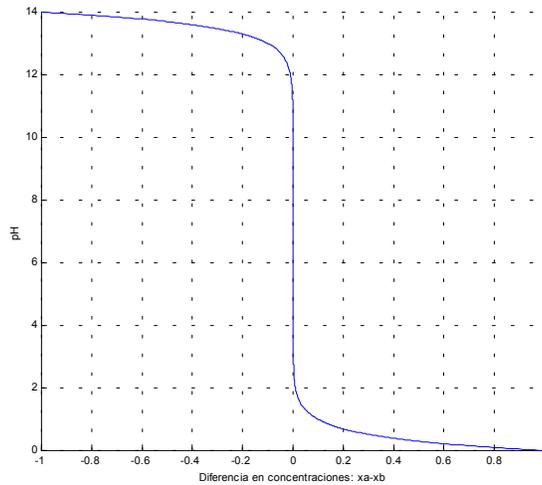
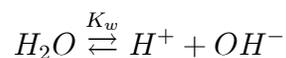


Figura 5-2: Curva típica de una base neutralizada por un ácido

$$pH = -\log[H^+] \text{ o bien } [H^+] = 10^{-pH}$$

de la expresión anterior podemos observar que si el $pH = 0$, entonces la concentración de iones de hidrogeno es máxima, es decir; $[H^+] = 1$, lo que significa que se tiene un ácido fuerte. En la medida en que el valor del pH se incrementa la concentración de iones de hidrogeno disminuye en forma logaritmica. Otro punto importante de la curva caracterísitica se observa cuando una mezcla tiene un $pH = 7$, en cuyo caso se considera que la sustancia es neutra, esto sucede cuando la concentración iones hidrogeno $[H^+]$ y la concentración de iones hidróxilo $[OH^-]$ es la misma. En el otro extremo, cuando el $pH = 14$, la concentración de iones hidroxilo es máxima y entonces se tiene una base fuerte.

Una relación importante es la que se obtiene de la ionización del agua pura, la cual se ioniza en iones hidrógeno e hidroxilo a igual concentración:



La constante de equilibrio (a $25^{\circ}C$) para la ionización del agua es

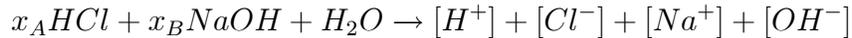
$$K_w = [H^+][OH^-] = 10^{-14} \quad (5.1)$$

En donde $[H^+]$, denota la concentración de iones de hidrógeno y $[OH^-]$ la concentración de iones hidroxilo. Esta relación permite definir la concentración del ion hidroxilo de cualquier solución acuosa cuyo pH se conozca. De la definición de pH y de la ecuación (5.1), se tiene

$$[OH^-] = 10^{pH-14}$$

Como el punto neutro se tiene cuando las concentraciones de los iones de hidrógeno e hidroxilo son iguales, esto sucede cuando $pH = 7$.

Los ácidos y bases fuertes se ionizan completamente, esto significa que todos los grupos hidrógeno e hidroxilo aparecen en forma ionizada en la solución. Por ejemplo, considere la adición de ácido clorhídrico HCl y sosa cáustica $NaOH$ a agua, cuyas concentraciones se denotan como x_A y x_B , respectivamente.



La solución resultante debe ser balanceada, esto es:

$$[H^+] + [Na^+] = [Cl^-] + [OH^-]$$

Como el ácido y la base se ionizan completamente, entonces todos los grupos hidrógeno e hidroxilo aparecen en forma ionizada:

$$[H^+] + x_B = x_A + [OH^-]$$

Sustituyendo las concentraciones de iones hidrógeno e hidroxilo, se obtiene una expresión que relaciona la concentración del ácido, de la base, y el pH de la solución.

$$x_A - x_B = 10^{-pH} - 10^{pH-14} \quad (5.2)$$

Observese que cuando la diferencia de concentraciones $(x_A - x_B) = 0$, entonces la solución es neutra y el $pH = 7$.

De la diferencia de concentraciones $(x_A - x_B)$, dependen los cambios en el pH de la solución. Cada cambio en una unidad en el pH será consecuencia de un cambio en la diferencia entre concentraciones aproximadamente en un factor de 10.

A partir de la expresión (5.2) se obtiene la siguiente tabla en donde se observa la relación entre la diferencia de concentraciones del ácido y la base, y el valor de pH correspondiente.

pH	$(x_A - x_B), N$	pH	$(x_A - x_B), N$
0	1.0	8	-0.99×10^{-6}
2	1.0×10^{-2}	10	-1.0×10^{-4}
4	1.0×10^{-4}	12	-1.0×10^{-2}
6	0.99×10^{-6}	14	-1.0
7	0		

(5.3)

Tabla 5.1 Relación entre concentraciones y pH

De la tabla anterior se puede observar que controlar en el intervalo neutro de pH (de 6 a 8) es particularmente difícil cuando sólo se tienen ácidos y bases fuertes, debido a que una muy pequeña diferencia en sus concentraciones producirá un gran cambio en el valor del pH .

Considere por ejemplo una solución de ácido fuerte con un $pH = 2$, la cuál debe neutralizarse con sosa caústica. Para llevar a la solución a valores de pH entre 6 y 8, debe adicionarse sosa caústica hasta igualar la concentración del ácido de $10^{-2}N$, con una tolerancia de $\pm 10^{-6}N$, o bien con una exactitud de 1 parte en 10,000.

Las válvulas de control y los medidores de flujo no pueden construirse con esta exactitud, por lo que es necesario contar con un sistema de control eficiente, un tanque bien mezclado, y con un buen sistema de medición que permita alcanzar la precisión necesaria [66].

Descripción del proceso y modelo matemático

Se considera un complejo proceso de neutralización de pH , diseñado por Hall y Seborg [22]. Este es un proceso altamente no lineal, que por su naturaleza logarítmica experimenta cambios bruscos en su región de operación, razón por la cual se seleccionó con la finalidad de utilizarlo en la prueba del controlador multimodelo neuronal que se propone en este trabajo.

El tanque de neutralización con agitador tiene 3 corrientes de entrada y una corriente de salida.

- Una corriente ácida: $Q_1(HNO_3)$
- Una corriente de moderadora (buffer): $Q_2(NaHCO_3)$.
- Una corriente base $Q_3(NaOH)$ con trazas de $NaHCO_3$
- La corriente de salida Q_4

El nivel del líquido en el tanque se deja variar libremente.

Observación 5.2 *En la práctica, sería muy difícil reproducir este proceso bajo las mismas condiciones experimentales. Esta situación produciría variaciones en la dinámica de la planta en cada experimento. Dichas variaciones pueden verse como cambios en la región de operación de la planta. Por lo anterior, el enfoque de identificación multimodelo también puede ser de gran ayuda en el desarrollo experimental.*

El modelo físico fué desarrollado por Hall [22], bajo las siguientes consideraciones:

- Mezclado perfecto

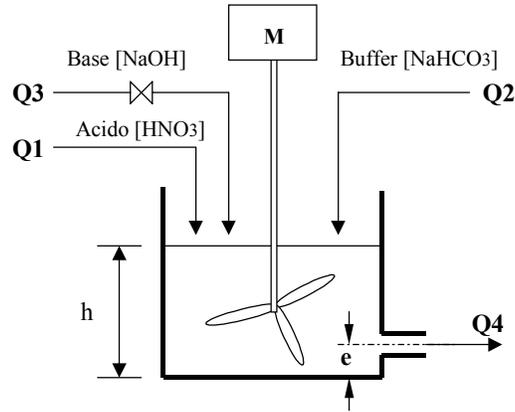
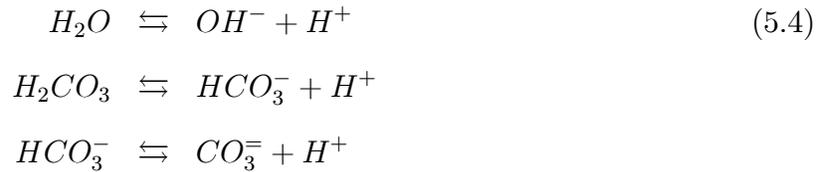


Figura 5-3: Tanque de neutralización

- Densidad constante
- Los iones involucrados son perfectamente solubles
- No entran o salen sustancias del sistema, excepto a través de las corrientes de flujo.

Las reacciones químicas que se involucran en el proceso son:



En [20], se definen dos invariantes de reacción para la i -ésima corriente

$$\begin{aligned}
 Wa_i &= [H^+]_i - [OH^-]_i - [HCO_3^-]_i - 2[CO_3^{=}]_i \\
 Wb_i &= [H_2CO_3]_i + [HCO_3^-]_i + [CO_3^{=}]_i
 \end{aligned}
 \tag{5.5}$$

Las cantidades Wa y Wb se llaman invariantes de reacción porque son independientes de la duración de la reacciones (5.4).

Considerando que las reacciones son lo suficientemente rápidas para que el sistema éste en equilibrio, se pueden utilizar las relaciones de equilibrio para determinar la concentración del ión hidrógeno a partir de los invariantes de reacción.

Las constantes de equilibrio están dadas por:

$$\begin{aligned} K_{a1} &= \frac{[HCO_3^-][H^+]}{[H_2CO_3]} \\ K_{a2} &= \frac{[CO_3^{2-}][H^+]}{[HCO_3^-]} \\ K_w &= [H^+][OH^-] \end{aligned} \quad (5.6)$$

Combinando las ecuaciones (5.5) y (5.6), se obtiene la siguiente relación

$$W_a = [H^+] - \frac{K_w}{[H^+]} - W_b \frac{\frac{K_{a1}}{[H^+]} + \frac{2K_{a1}K_{a2}}{[H^+]^2}}{1 + \frac{K_{a1}}{[H^+]} + \frac{K_{a1}K_{a2}}{[H^+]^2}} \quad (5.7)$$

El modelo dinámico del proceso de neutralización es el siguiente:

a) **Balance de masa en el tanque**

$$A \frac{dh}{dt} = Q_1 + Q_2 + Q_3 - Q_4$$

b) **Relación cabezal-flujo de la corriente de salida**

$$Q_4 = c(h + e)^d$$

Donde e denota la distancia vertical del fondo del tanque al centro de la salida del tanque, h es el nivel de líquido en el tanque, y c y d son constantes.

c) **Balances de componentes.**

Los balances de componentes en función de los invariantes de reacción están dados por:

$$\begin{aligned}\frac{hAdW_{aT}}{dt} &= Q_1(W_{a1} - W_{aT}) + Q_2(W_{a2} - W_{aT}) + Q_3(W_{a3} - W_{aT}) \\ \frac{hAdW_{bT}}{dt} &= Q_1(W_{b1} - W_{bT}) + Q_2(W_{b2} - W_{bT}) + Q_3(W_{a3} - W_{bT})\end{aligned}\quad (5.8)$$

En donde el subíndice T , denota las concentraciones en el tanque. Estas ecuaciones permiten determinar los invariantes de reacción en el tanque W_{aT} , y W_{bT} como funciones del tiempo. Finalmente el pH en el tanque puede calcularse a partir de la ecuación (5.7) y los invariantes W_{aT} , y W_{bT} que se obtienen de la ecuación (5.8).

Observación 5.3 *Se dice que una solución es amortiguada cuando a ésta se le adiciona una sustancia amortiguadora (buffer), que tiene la función de moderar la rapidez de los cambios en la concentración de un ión determinado. La extremada gran sensibilidad del pH cerca del punto neutro ($pH = 7$), proporciona al sistema propiedades muy específicas que dificultan el control del pH cerca de su punto neutro.*

5.2 Control de pH

Las estrategias clásicas de control de pH incluyen al control PID y al control anticipativo de pH . En el primero, la medición del pH a la salida del tanque se retroalimenta a un controlador PID convencional que se encarga de manipular el flujo de corriente base Q_3 para neutralizar la mezcla. En el control anticipativo de pH se utiliza un controlador PID no lineal en combinación con un control de relación. En el controlador PID no lineal se introduce una función que intenta compensar las zonas de alta ganancia y de baja ganancia que experimenta el proceso en diferentes puntos de operación. En éste enfoque propuesto por Shinsky [66], se aplica una idea muy similar a la de un control multimodelo, debido a que la función no lineal que se introduce tiene la finalidad de detectar los puntos de operación de muy alta ganancia y de baja ganancia que caracterizan al proceso.

El estudio y simulaciones de las estrategias clásicas de control pH se hace con la finalidad de poder establecer una comparación con el desempeño de un control neuronal sencillo y

luego con un control multimodelo neuronal dinámico, el cuál es una de las propuestas de este trabajo para sistemas altamente no lineales, como es caso de los procesos de neutralización de pH .

5.2.1 Control PID de pH

El objetivo del sistema de control es neutralizar el pH de la corriente ácida Q_1 , manipulando la flujo de la corriente base Q_3 , que es la entrada de control a la planta.

En esta sección se estudia el desempeño que tiene un control PID clásico para regular el pH de un tanque de neutralización (figura 5-3), con y sin perturbaciones. Se incluye un elemento de saturación a la salida del controlador para limitar la señal de control, y así incluir, sino la dinámica de la válvula de control, si las limitaciones físicas de esta. Las perturbaciones que se aplican al proceso pueden cambiar el punto de operación del proceso, o en términos más generales, la región de operación. Esta situación puede producir un decremento en el desempeño del sistema de control.

Primero se prueba este control sin perturbaciones y luego se aplican las siguientes perturbaciones:

- a) Perturbación en el pH de la mezcla
- b) Perturbación en el punto de ajuste
- c) Perturbación en el flujo de alimentación
- d) Perturbación en la concentración del flujo de alimentación

La estrategia de control retroalimentado para el control del pH del tanque de neutralización se ilustra en la figura 5-4.

En este caso el pH de la corriente de salida, el cual se considera representativo del pH en el tanque, se retroalimenta al controlador PID que se encarga de manipular la corriente base Q_3 (entrada de control a la planta), con el objetivo de neutralizar la corriente ácida Q_1 .

Los parámetros de sintonización del controlador PID se obtuvieron por medio de prueba y error (debido a que no se obtuvieron buenos resultados con el metodo de Ziegler y Nichols

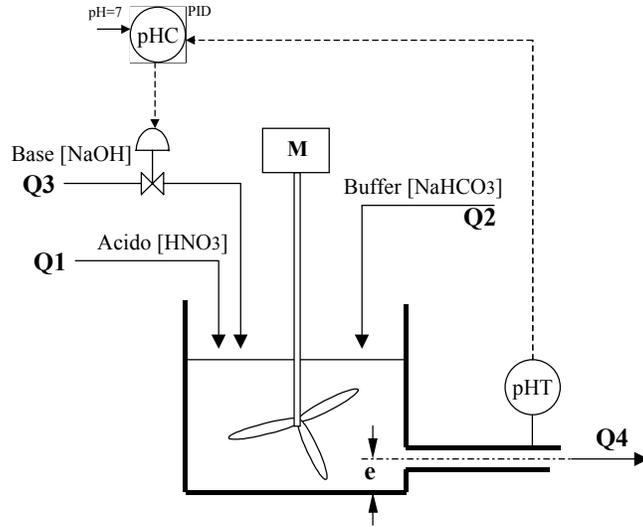


Figura 5-4: Control de pH con un PID retroalimentado.

[3]) , siendo los valores que entregarón un mejor desempeño del controlador los siguientes: $Kc = 1.5$, $Ti = 10$ y $Td = 0.01$. El bajo valor de la ganancia del controlador refleja la alta ganancia del proceso de neutralización. Con estos ajustes del controlador el pH de la mezcla en el tanque logra neutralizarse aproximadamente en 600s.

Las figuras 5-5 y 5-6 muestran la respuesta del proceso y la señal de control respectivamente. En esta simulación no hay perturbaciones en el proceso.

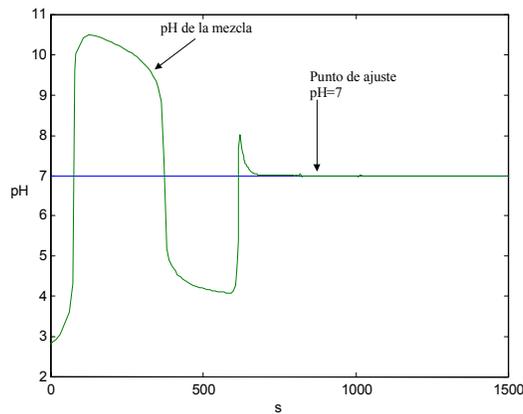


Figura 5-5: pH en la corriente de salida Q_4

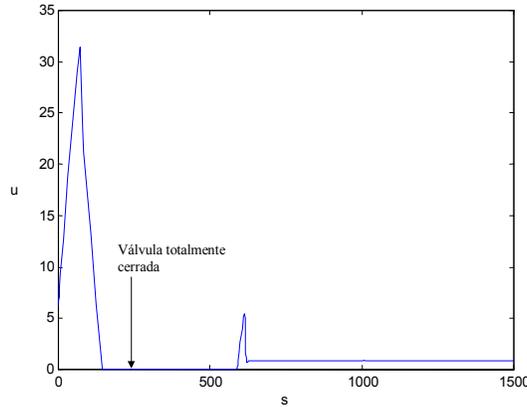


Figura 5-6: Señal de control Q_3

a) **Perturbación en el pH de la mezcla**

Una vez que el pH se estabilizó en su valor neutro, se introduce un cambio en el pH de la mezcla en $t = 800$. Este cambio se genera al modificar el invariante de reacción de la mezcla, lo que puede interpretarse como un cambio en la diferencia entre las concentraciones ácido-base en el tanque. La respuesta del proceso y la señal de control se muestran en las figuras 5-7 y 5-8.

En la figura 5-7, se observa como la alta ganancia del proceso produce grandes excursiones de la variable pH . Una vez que se presenta la perturbación (en $t = 800s$), la variable tarda más de $200s$ en estabilizarse. El pobre desempeño del controlador en este caso es evidente.

b) **Perturbación en el punto de ajuste**

Ahora se aplica un brusco cambio en el punto de ajuste como se muestra en la figura 5-9. Cuando el pH se ha estabilizado, se introduce primero un cambio positivo (de $pH = 7$ a 10) y luego uno negativo (de 7 a 4), cada uno por un período de 10 segundos. El desempeño del controlador PID y la señal de control se muestran en las figuras 5-9 y 5-10.

En este caso, se aplican dos perturbaciones en el punto de ajuste: la primera en un sentido positivo y la segunda en un sentido negativo. La figura 5-9, muestra como la variable controlada no tiene una respuesta simétrica ante estas perturbaciones, lo cuál es síntoma de

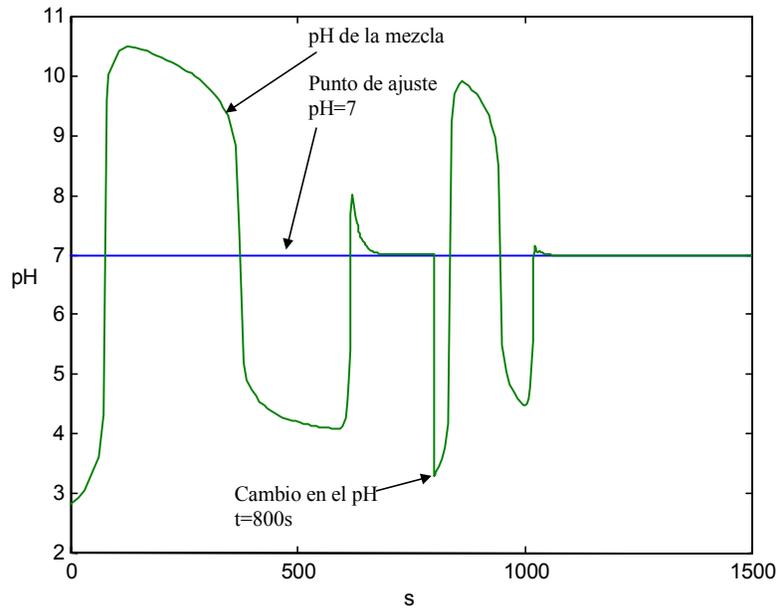


Figura 5-7: Respuesta ante una perturbación en el pH de la mezcla.

la no linealidad del proceso. Además, este controlador presenta un pobre desempeño ante la perturbación positiva.

c) **Perturbación en el flujo de alimentación Q_1**

El flujo nominal de la corriente de alimentación Q_1 es de 16.6mls^{-1} . La perturbación que se introduce también es en ambos sentidos, es decir, primero se aplica un escalón de corta duración y de amplitud $Q_1 + 10\text{mls}^{-1}$, y un poco después un escalón de $Q_1 - 10\text{mls}^{-1}$. La perturbación en el flujo de alimentación Q_1 , se muestra en la figura 5-11, mientras que la respuesta del proceso y la señal de control se muestran en las figuras 5-12 y 5-13.

En figura 5-12, se observa como el control PID , parece tener un buen desempeño ante este tipo de perturbación.

d) **Perturbación en la concentración del flujo de alimentación Q_1**

Esta perturbación se introduce a través del invariante de reacción W_1 , ya que en el modelo simplificado del tanque de neutralización de pH se tiene que $W_1 = [HNO_3]_1$, es

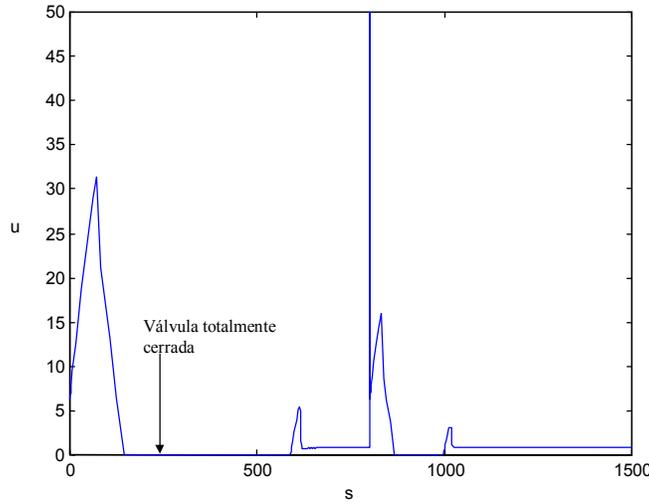


Figura 5-8: Señal de control

decir, la concentración de ácido en la corriente Q_1 es precisamente W_1 . El valor nominal de la concentración de Q_1 es 0.0001. El proceso experimenta un cambio en la concentración de la corriente de entrada como se muestra en la figura 5-14.

Las figuras 5-15 y 5-16 muestran la respuesta del proceso y la señal de control respectivamente. Como se aprecia en la figura 5-15, el sistema de control no tiene un buen desempeño ante esta perturbación.

5.2.2 Control PID no lineal

Una modificación natural del controlador PID para el control de pH , es un controlador no lineal que tenga una baja ganancia cerca del punto de control y una alta ganancia a desviaciones mayores. Esta característica del controlador no lineal de Shinskey [67] [68], se ilustra en la figura 5-17, en donde ΔpH , denota la desviación del punto de ajuste en el pH , y e la señal de entrada al controlador PID . Vagamente hablando esta ganancia no lineal es una especie de función inversa de la curva de pH .

Observación 5.4 *Cuando se desea compensar cambios en la composición de la alimentación utilizando un control anticipativo de pH , se presenta un problema: el estado de la pertur-*

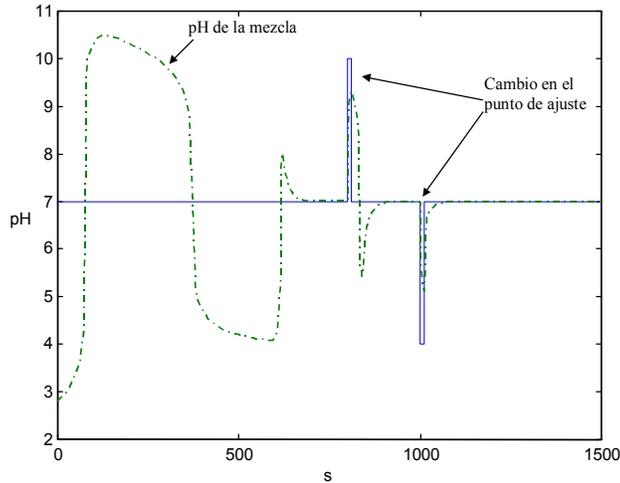


Figura 5-9: Respuesta ante perturbaciones en el punto de ajuste

bación en la alimentación no se determina sólo por la medición del pH. La perturbación se conoce sí, junto con el pH, se conoce la concentración total de cada sistema ácido-base presente [73]. Debido a que la capacidad de amortiguamiento comunmente varia con el tiempo, es deseable un enfoque adaptable. Otro punto importante en un control anticipativo de pH, es que el carácter no lineal del proceso tiene que considerarse en la acción anticipativa. Como regla, un control anticipativo lineal a partir del pH no debe utilizarse.

Control PID no lineal anticipativo de pH

Los sistemas de control de relación son sistemas anticipativos en donde una variable se controla en relación a otra para satisfacer un objetivo.

En un sistema de control de relación la variable que se controla es la relación R de dos flujos

$$R = \frac{f}{q}$$

Normalmente el control se realiza manipulando una válvula que ajusta el flujo de una de las dos corrientes, mientras que la otra corriente se deja libre o "loca".

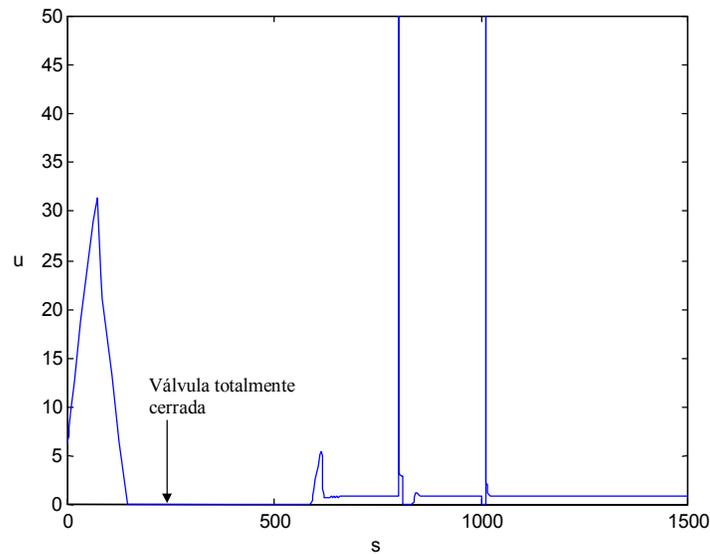


Figura 5-10: Señal de control

Una de las formas de implementar el control de relación, consiste en calcular la relación R , y retroalimentar éste valor al controlador

En la figura 5-18, se observa un divisor dentro del lazo cerrado, independientemente de que variable este manipulando el controlador. Si f es la variable manipulada entonces la ganancia del lazo variara con el flujo libre q ; pero si q es manipulada entonces el lazo se torna no lineal en el sentido que la ganancia del lazo cambia con la salida del controlador.

$$\frac{dR}{df} = \frac{1}{q}, \quad \frac{dR}{dq} = -\frac{R}{q}$$

Estos problemas se superan trasladando el cálculo de la relación R fuera del lazo cerrado. De esta manera la estación de relación se debe localizar en el circuito del punto de ajuste (ver figura 5-19), haciendo $r = Rq$, si f es la variable controlada, o $r = f/R$, si q es controlada. En esta configuración una variable se utiliza como variable controlada f , mientras que la variable libre q , sirve para generar el punto de ajuste. La variable libre se multiplica por el factor de relación R .

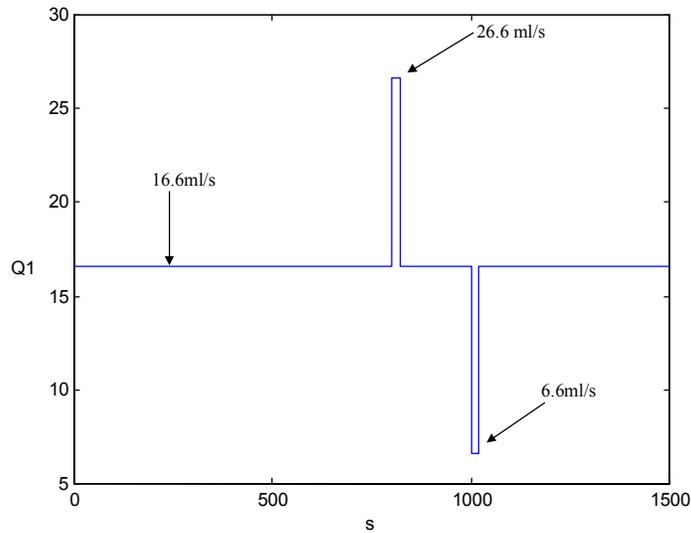


Figura 5-11: Perturbación en el flujo de alimentación Q_1 .

Debido a la gran sensibilidad de la curva en la región de neutralidad es necesario ajustar la relación R a través de un lazo retroalimentado. En nuestro ejemplo en particular, la variable manipulada es la corriente base Q_3 , la corriente de entrada Q_1 se deja libre para generar el punto de ajuste. Un diagrama del sistema recomendado en [66], se muestra a continuación.

Esta estrategia es un control de relación entre Q_1 y Q_2 , en donde el factor de relación R se ajusta a través de un lazo retroalimentado y un controlador PID de ganancia no lineal. El controlador PID no lineal tiene dos zonas: una de baja ganancia para contrarrestar la alta ganancia del proceso cerca del punto neutro; y una de alta ganancia para tratar de mantener una ganancia constante lejos del punto neutro. El controlador PID no lineal tiene un punto de ajuste igual al valor del pH deseado ($pH = 7$), y su salida es igual al factor de relación R que deben mantener los flujos Q_1 y Q_2 para mantener el pH en su punto neutro.

En la siguiente figura se observa la respuesta del proceso cuando se utiliza primero un controlador convencional PID retroalimentado, y en segundo lugar un control de relación más un control retroalimentado PID no lineal.

En la figura 5-21 se observa un mejor desempeño del controlador propuesto en [66]. Los ajustes del controlador PID no lineal fueron: $K_c = 1.5$, $T_i = 50$, $T_d = 0.01$, $z = 1$ y $kz = 0.2$;

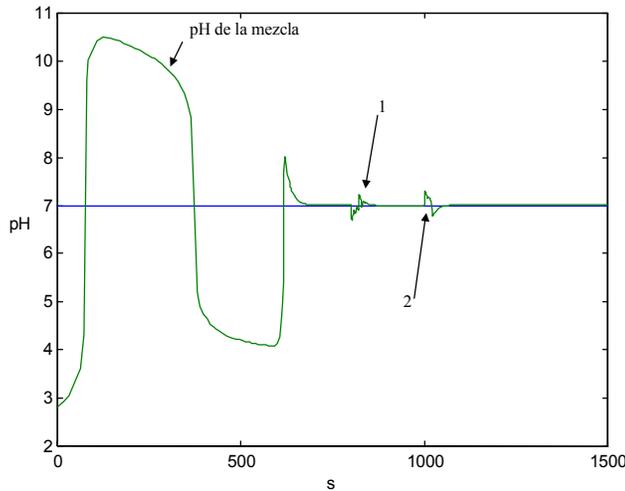


Figura 5-12: Respuesta ante perturbaciones en el flujo de alimentación Q_1

mientras que los ajustes del lazo interno que tiene un PI convencional con ajustes: $K_c = 1.5$, $T_i = 5$.

Debido a la gran no linealidad que presenta éste tipo de procesos los parámetros de sintonización se obtuvieron a prueba y error, considerando la parte intuitiva que proporciona el método de sintonización de Ziegles y Nichols.

A continuación se presentan las simulaciones utilizando esta estrategia de control en presencia de dos perturbaciones: primero en el flujo de alimentación y luego en la concentración del flujo de alimentación.

a) **Perturbación en el flujo de alimentación**

La perturbación que se introdujo en el flujo de alimentación Q_1 fué en ambos sentidos. Como el flujo nominal es $Q_1 = 16.6 \text{ mls}^{-1}$, la primer perturbación se aplicó en $t = 800s$ llevando el valor a 26.6 mls^{-1} por unos cuantos segundos. Luego en $t = 1000s$ se llevó el valor 6.6 mls^{-1} .

Como puede apreciarse en la figura 5-22, aunque las perturbaciones en el flujo de alimentación son considerables, esta estrategia de control tiene un buen desempeño, pues logra contrarrestar rápidamente las perturbaciones descritas.

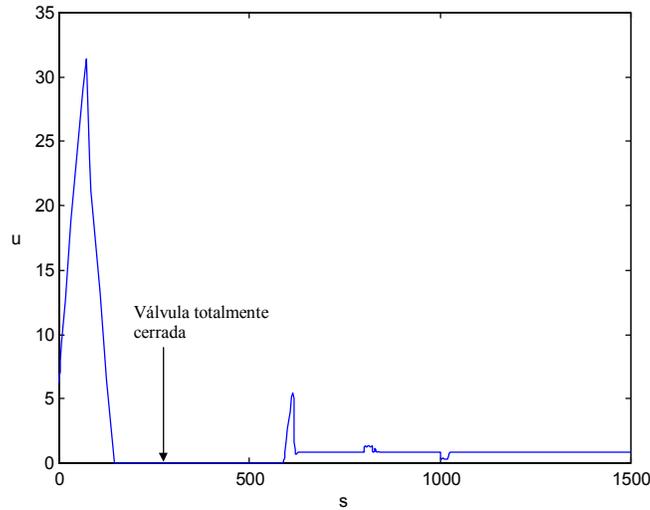


Figura 5-13: Señal de control

b) Perturbación en la concentración del flujo de alimentación.

En este caso la perturbación que se introdujo en la concentración del flujo de alimentación tiene las siguientes características. En primer lugar, el valor nominal de la concentración es de 0.0001grl^{-1} . En $t = 800\text{s}$ se aplicó un incremento en la concentración a 0.001grl^{-1} , y luego en $t = 1000\text{s}$ se aplicó un decremento a 0.00001grl^{-1} .

Como puede observarse en la figura 5-23 el desempeño de esta estrategia de control es muy pobre ante perturbaciones en la concentración del flujo de alimentación.

5.3 Identificación con un multimodelo neuronal dinámico

El proceso de neutralización de pH es un caso interesante para probar el desempeño del identificador multimodelo neuronal, ya que el comportamiento de la variable pH caracteriza al proceso con un comportamiento altamente no lineal. En los procesos de neutralización de ácidos y bases fuertes un pequeño cambio en la diferencia de concentraciones ácido-base producirá un cambio muy grande en el punto de operación del proceso [66]. El cambio en el valor del pH puede ser inclusive discontinuo como se muestra en las figuras 5-25 y 5-24. En esta prueba de simulación se utilizó un modelo simplificado del proceso (ver. 5.4.3) en donde

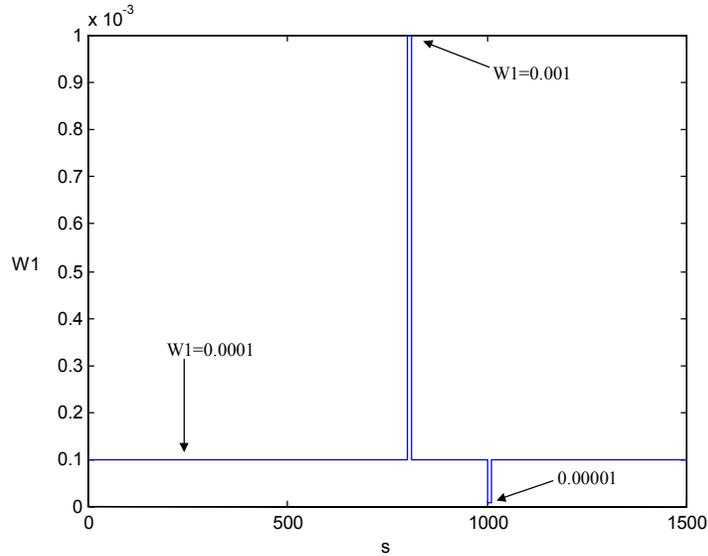


Figura 5-14: Perturbación en la concentración del flujo de alimentación Q_1 .

sólo se consideran ácidos y bases fuertes, es decir, $Q_2 = 0$. Los valores de los parámetros necesarios para la simulación se tomaron de [30].

Para evaluar la capacidad de aproximación de una red neuronal dinámica y compararla con la estimación se obtiene con el identificador multimodelo, se introduce un brusco cambio en el pH de la mezcla. En las figuras 5-24 y 5-26 se puede observar que en $t = 80$ s el pH de la mezcla experimenta un gran cambio en las condiciones de operación que lleva al pH de la mezcla de un $pH = 10$ a un $pH = 3$ aproximadamente.

5.3.1 Identificación con un modelo neuronal

Los parámetros de la red neuronal que se utilizó son los siguientes

$$A = -35, \dots P = 12, \quad W_{1,0} = 1, \quad W_{2,0} = 1$$

En la figura 5-24 se compara la estimación del identificador que utiliza sólo un modelo neuronal con el pH de la mezcla. En esta figura es posible apreciar como después de que se

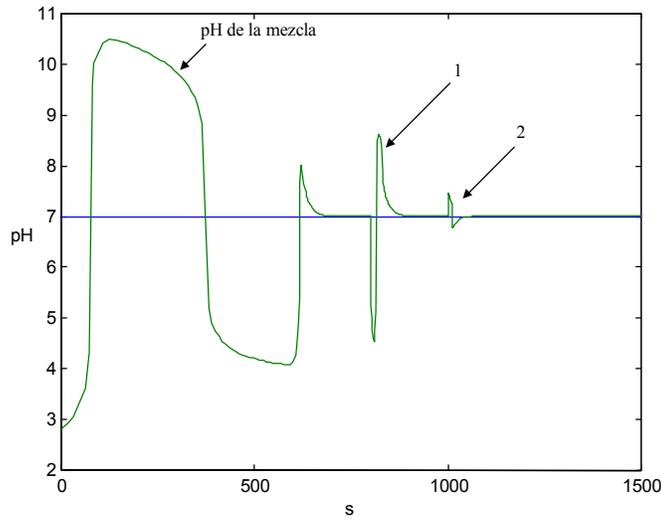


Figura 5-15: Respuesta ante perturbaciones en la concentración de Q_1

aplica el cambio en el pH de la mezcla, el identificador no es capaz de seguir perfectamente la salida de la planta (ver zona A), y se observa que la salida de la planta experimenta excursiones de hasta ± 3 unidades con respecto al valor del pH de la mezcla. Si se considera que se tiene una escala logarítmica, entonces cada cambio en una unidad en el pH representa un cambio en un orden de magnitud en la concentración ácido-base de la mezcla.

5.3.2 Identificación con un multimodelo neuronal

Ahora el identificador multimodelo se prueba bajo las mismas condiciones. En este experimento, el identificador multimodelo tiene dos redes neuronales dinámicas que se encargan de identificar continuamente la dinámica de la planta. Los parámetros de estas redes son los siguientes

$$\begin{aligned}
 A_1 &= -35, \dots, P_1 = 12, & W_{1,0}^1 &= 1, & W_{2,0}^1 &= 1 \\
 A_2 &= -2, \dots, P_2 = 3, & W_{1,0}^2 &= 0.1, & W_{2,0}^2 &= 0.1
 \end{aligned}$$

La figura 5-25 permite comparar el pH medido y la estimación del identificador multimod-

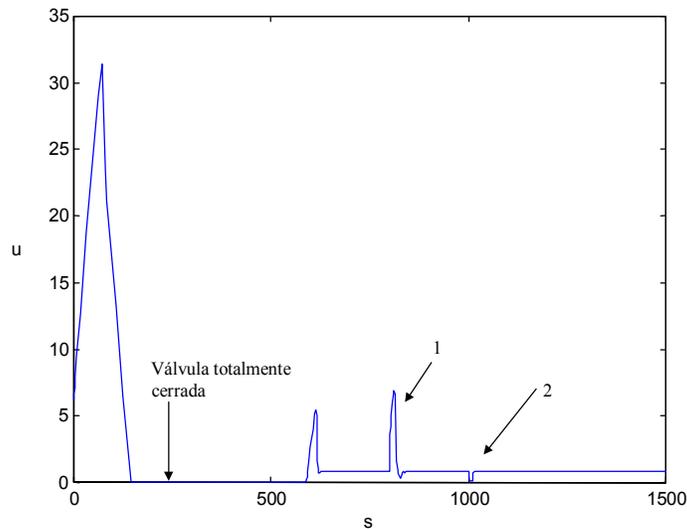


Figura 5-16: Señal de control

elo neuronal. En esta figura se puede apreciar como el identificador multimodelo neuronal presenta un buen desempeño.(ver zona A) después de que la planta experimento un gran cambio en el punto de operación. En el resto de la trayectoria el identificador multimodelo aproxima bien la dinámica de la planta.

En la figura 5-26 se tiene el estado del selector que nos informa cuál de los identificadores dinámicos es el que aproxima mejor la dinámica de la planta, y junto con la figura 5-27 puede observarse como en el intervalo $0 < t < 80 s$ el identificador 2, aproxima mejor la dinámica de la planta pero en el instante en que ocurre un cambio brusco en las condiciones de operación ($t = 80s$), el identificador 1 se selecciona como el mejor identificador.

En las figuras 5-24 y 5-25 se puede observar como una red neuronal dinámica puede tener un buen desempeño en un punto de operación, pero cuando las condiciones de operación cambian bruscamente, entonces es posible que otro identificador tenga un mejor desempeño en la nuevas condiciones.

En las figuras 5-27 y 5-28 se puede observar el error de identificación ($pH - \hat{pH}$), cuando se utiliza un identificador multimodelo y cuando se utiliza sólo un identificador.

A partir de los resultados de estas simulaciones se puede concluir que el identificador multimodelo neuronal dinámico propuesto puede identificar con mayor exactitud la dinámica

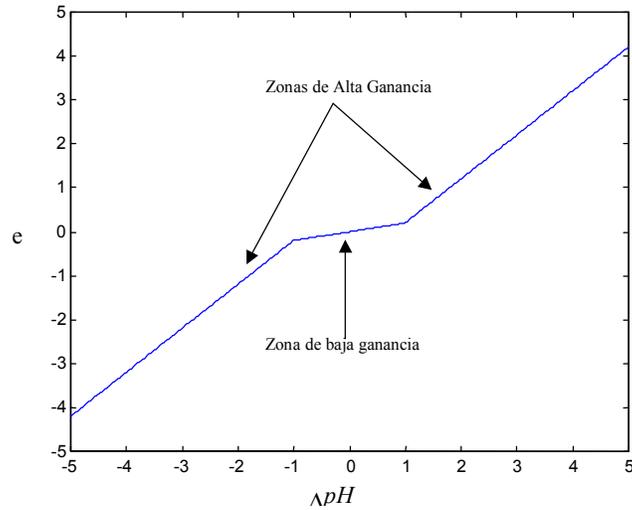


Figura 5-17: Relación del error entrada/salida de un PID no lineal

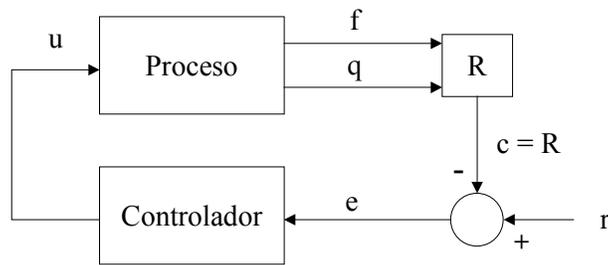


Figura 5-18: Estación de relación dentro del lazo de control

de procesos de neutralización no amortiguados ($Q_2 = 0$), en presencia de grandes variaciones en el pH de la mezcla.

5.4 Control multimodelo neuronal de pH

5.4.1 Control de pH con un modelo neuronal dinámico

La estructura de control neuronal que se utilizó para el control de pH se presenta en la figura 5-29. En esta figura puede apreciarse como la red neuronal dinámica juega el papel del identificador en línea de la planta.

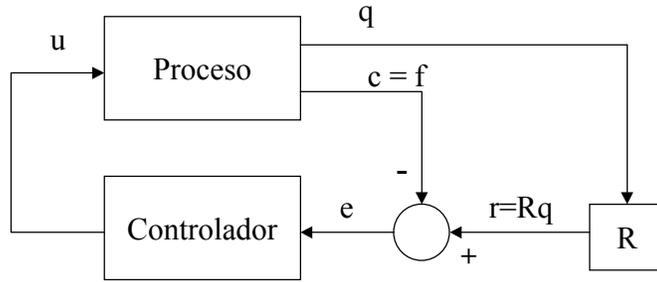


Figura 5-19: Estación de relación fuera del lazo de control

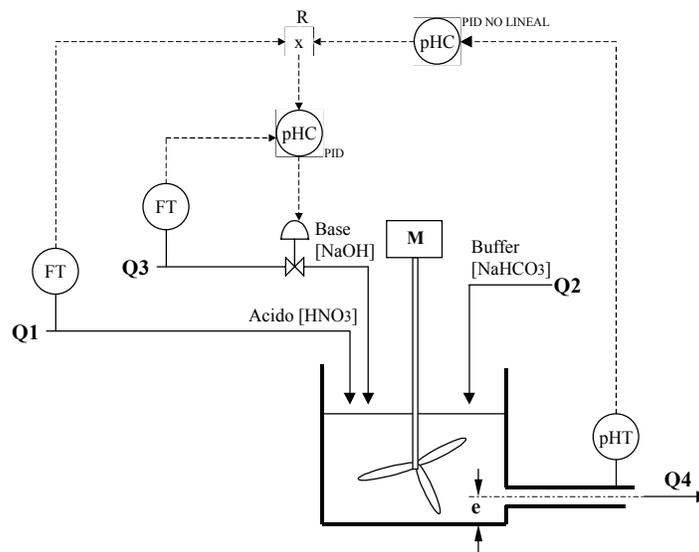


Figura 5-20: Estrategia de control anticipativo retroalimentado de pH

Las salidas de la red neuronal son la estimación de la dinámica de la planta \hat{y} , y el vector θ_t , es la información dinámica que se retroalimenta al controlador para el cálculo de la ley de control u . El vector θ_t , lleva la siguiente información

$$\theta_t = (W_{1,t}, sig(\hat{y}), \dot{\hat{y}})$$

en donde $W_{1,t}$ es la matriz de pesos de retroalimentación no lineal, y $sig(\hat{y})$ es una función sigmoïdal que se aplica a la salida de la red neuronal \hat{y} .

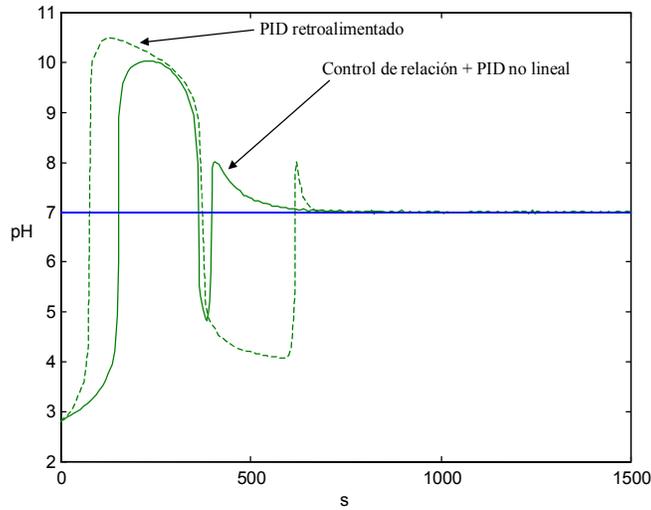


Figura 5-21: Comparación entre un control *PID* y un control anticipativo retroalimentado de *pH*.

La estructura de la red neuronal red neuronal dinámica que se utiliza para el identificador neuronal es una red sencilla de una capa, con la forma

$$\dot{\hat{y}}_t = A\hat{y}_t + W_{1,t}\sigma(\hat{y}_t) + u_t \quad (5.9)$$

en donde $\hat{y}_t \in R^1$ es el estado de la red neuronal, $W_{1,t} \in R^1$ es la matriz de retroalimentación de estado no lineal, $A \in R^1$ es una matriz estable, y $\sigma(\hat{y}_t)$ es una función sigmoideal de la forma:

$$\sigma(\hat{y}) = \frac{a}{1 + e^{-b\hat{y}}} - c$$

La ley de aprendizaje dinámico esta representada por la siguiente ecuación diferencial

$$\dot{W}_{1,t} = -K_1 P \Delta_t \sigma^T \quad (5.10)$$

en donde $K_1 \in R^1$ es una matriz definifa positiva, P es la solución de la ecuación matricial de Riccati (4.6), Δ_t es el error de identificación $\Delta_t = (y_t - \hat{y}_t)$, y $W_{1,0} = W_1^*$ es la condición

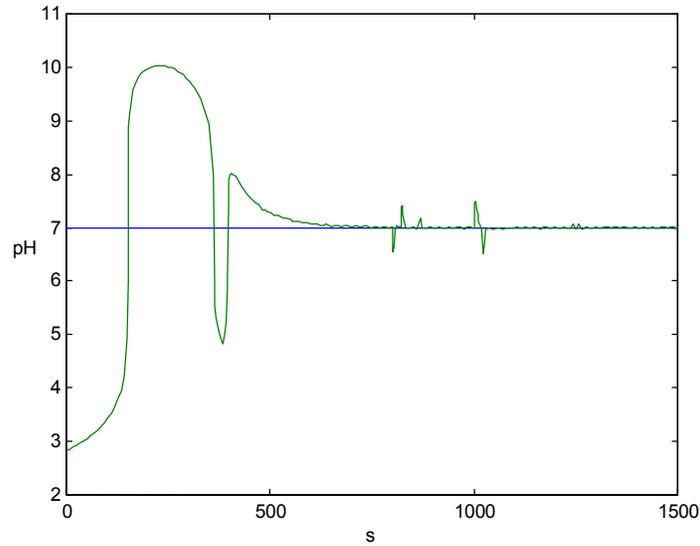


Figura 5-22: Control anticipativo retroalimentado ante perturbaciones en Q_1 .

inicial de la matriz de pasos.

Con la finalidad de comparar el desempeño de diferentes modelos neuronales y de utilizarlos posteriormente en el enfoque multimodelo, se implemento esta estrategia de control neuronal con dos modelos diferentes, es decir, con dos redes neuronales distintas:

Modelo de identificación 1.

La red neuronal dinámica 1, tiene los siguientes ajustes:

$$A_1 = -5, \quad P_1 = 8, \quad W_{1,0} = 1, \quad \sigma_1(\hat{y}) = \frac{0.01}{1 + e^{-0.1\hat{y}}} + 0.005 \quad (5.11)$$

Modelo de identificación 2.

La red neuronal dinámica 1, tiene los siguientes ajustes:

$$A_2 = -5.6, \quad P_2 = 8, \quad W_{2,0} = 1, \quad \sigma_2(\hat{y}) = \frac{-0.01}{1 + e^{-\hat{y}}} \quad (5.12)$$

El vector θ_t , lleva la siguiente información

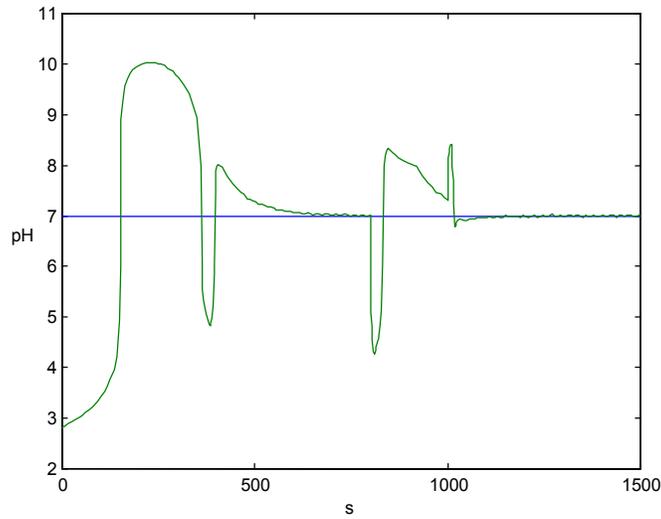


Figura 5-23: Control anticipativo retroalimentado ante perturbaciones en la concentración de Q_1 .

$$\theta_t = (W_{1,t}, sig(\hat{y}), \dot{\hat{y}})$$

en donde $W_{1,t}$ es la matriz de pesos de retroalimentación no lineal, y $sig(\hat{y})$ es una función sigmoideal que se aplica a la salida de la red neuronal \hat{y} .

$$\theta_t = (W_{1,t}, sig(\hat{y}), \dot{\hat{y}})$$

$W_{1,t}$ es la matriz de pesos de retroalimentación no lineal, $sig(\hat{y})$ es una función sigmoideal que se aplica a la salida de la red neuronal \hat{y} .

De acuerdo con lo expuesto en el capítulo 4 (Secc. 4.1.2), se aplicó la ley de control $u_t = u_{1,t} + u_{2,t}$, en donde $u_{1,t}$ es la parte de linealización directa

$$u_{1,t} = \varphi(x_t^*, t) - Ax_t^* - W_{1,t}\sigma(\hat{x}_t) \quad (5.13)$$

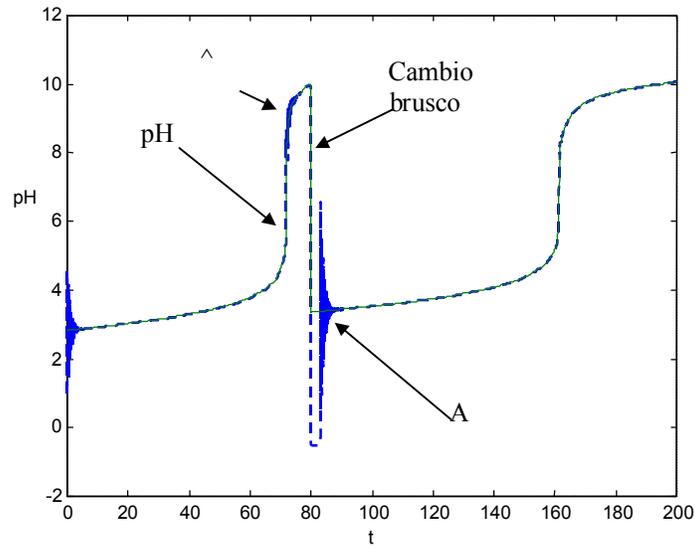


Figura 5-24: Identificación con una red neuronal

y $u_{2,t}$ corresponde al caso de compensación exacta de la dinámica no modelada d_t .

$$u_{2,t}^a = A(x_t - \hat{x}_t) - (\dot{x}_t - \dot{\hat{x}}_t) \quad (5.14)$$

La figura 5-30 muestra el desempeño del control neuronal utilizando las redes neuronales 1 y 2.

El índice de desempeño que se utilizó para evaluar la calidad de los modelos de identificación contiene sólo una medida del error cuadrático instantaneo:

$$J(e) = e^2(t)$$

5.4.2 Control de pH con un multimodelo neuronal dinámico

A pesar de que los dos controles con un modelo neuronal que se estudiarón en la sección anterior tienen un buen desempeño, se tiene la limitación de que un modelo neuronal puede tener un buen desempeño en una región de operación y un pobre desempeño si la región de

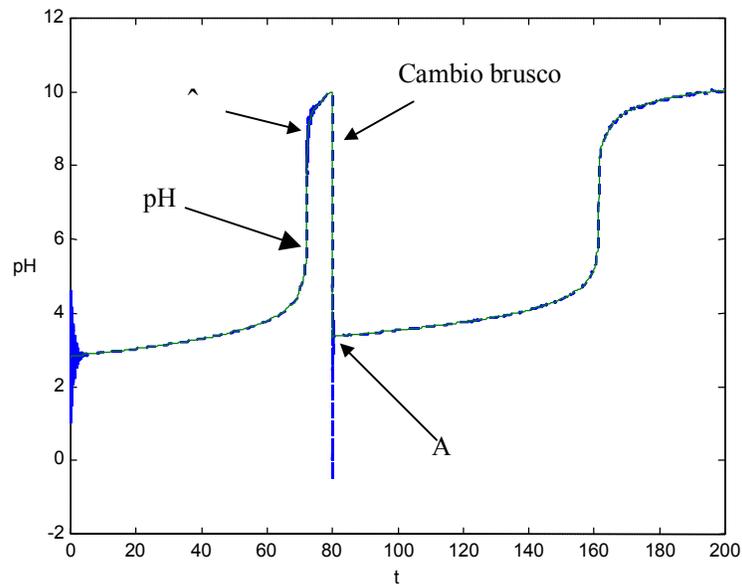


Figura 5-25: Identificación con múltiples redes neuronales

operación de la planta cambia.

Se propone contarrestar los posibles cambios en la región de operación de la planta, utilizando la estructura de un controlador multimodelo neuronal, como la de la figura 4-7.

Para el caso del control del proceso de neutralización, se utilizarón dos redes neuronales como identificadores de la planta. Los identificadores son los mismos que se utilizarón en la sección anterior (5.11) y (5.12), en forma independiente, pero ahora se integran en la construcción del controlador multimodelo neuronal propuesto y se ajusta un parámetro adicional, la constante de histéresis $h = 5$.

En la figura 5-32 se observa como la respuesta de este esquema de control mejora a las dos respuestas que se obtuvieron cuando se utilizó sólo una red neuronal en el proceso de identificación.

En la elección de la constante de histéresis se observó el desempeño del control con diferentes valores de h . Finalmente, en la simulación que se presenta en la figura 5-32, el mejor desempeño de controlador multimodelo neuronal se obtuvo con $h = 5$.

Observación 5.5 *Inicialmente la constante de histéresis se introdujo para reducir el número*

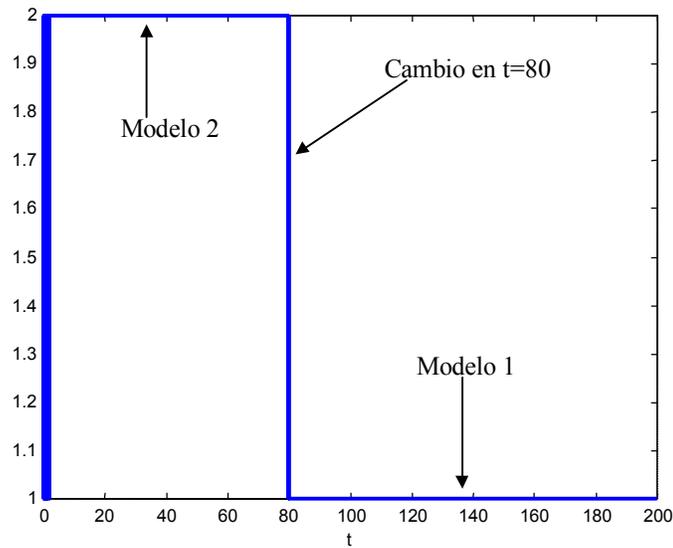


Figura 5-26: Selección de modelos

de conmutaciones del selector. Sin embargo, en esta simulación se observó que el ajuste de la constante de histéresis tiene un efecto interesante en el desempeño del controlador multimodelo. Este efecto puede deberse a que se tiene una retroalimentación dinámica para el cálculo de la señal de control. Como el valor de la constante de histéresis además de reducir el número de conmutaciones mueve el momento en que se hace la conmutación, entonces un cambio de modelo en el selector, también producirá a partir de ese momento, cambios en la dinámica de la planta, del modelo, y por consiguiente en los errores de identificación.

En los errores de identificación de los modelos 1 y 2 (figura 5-33), se puede observar que el identificador neuronal 2, tiene un menor error de identificación hasta $t < 38$ s, y después de que se presenta un brusco cambio en la dinámica del error en $t = 38$ s, el identificador multimodelo 1 aproxima mejor a la planta.

El estado del selector que se muestra en la figura 5-34, es consistente con las dinámicas del error de identificación de la figura 5-33, ya que en el intervalo $0 < t < 38$ s, el selector tiene al modelo 2 como el modelo que aproxima mejor a la dinámica de la planta, y en ese mismo intervalo, la dinámica del error de identificación del modelo 2 es menor.

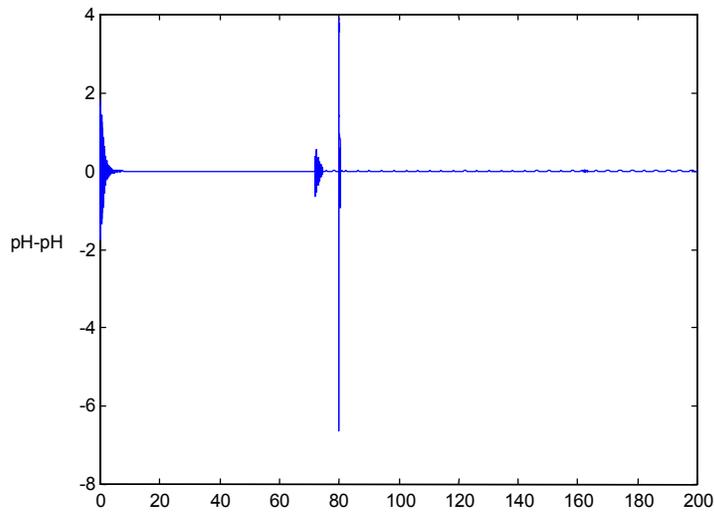


Figura 5-27: Error de identificación con múltiples redes neuronales

5.4.3 Datos de simulación

Para efectos de simulación a continuación se describe un modelo simplificado del tanque de neutralización, en donde se desprecia las corriente moderadora Q_2 (*buffer*). Bajo esta consideración las corrientes restantes son:

- Corriente ácida de entrada $Q_1(HNO_3)$
- Corriente base $Q_3(NaOH)$
- Corriente de salida Q_4 .

La única reacción que se considera en el modelo es



El invariante de reacción es

$$W = [H^+] - [OH^-]$$

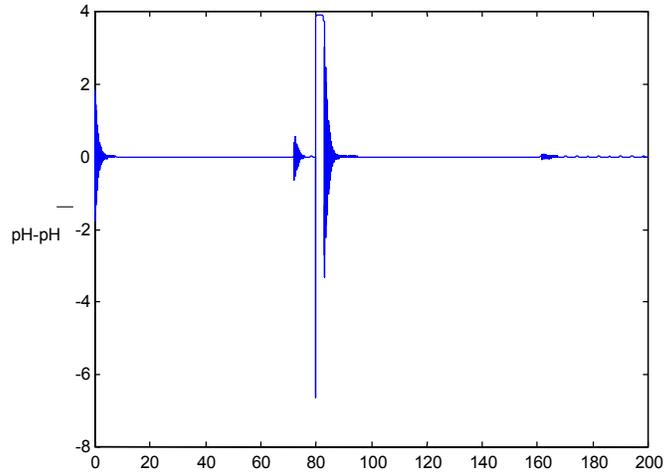


Figura 5-28: Error de identificación una red neuronal

y una ecuación implícita para la concentración de iones $[H^+]$ es:

$$W = [H^+] - K_w/[H^+]$$

a) **Balance de masa:**

$$A \frac{dh}{dt} = Q_1 + Q_3 - c(h + e)^d$$

b) **Balance de componentes:**

$$hA \frac{dW}{dt} = Q_1(W_1 - W) + Q_3(W_3 - W)$$

Los datos que se utilizarán en la simulación se tomarán de [30] y pueden consultarse en la tabla 5.2. Constante de la válvula $c = 8mls^{-1}\sqrt{cm}$, $W_{b1} = 0$, $W_2 = 0$, $A = 207cm^2$ es el área del tanque, $h = 14cm$ el nivel en el tanque, y $e = 5cm$ la altura a la salida

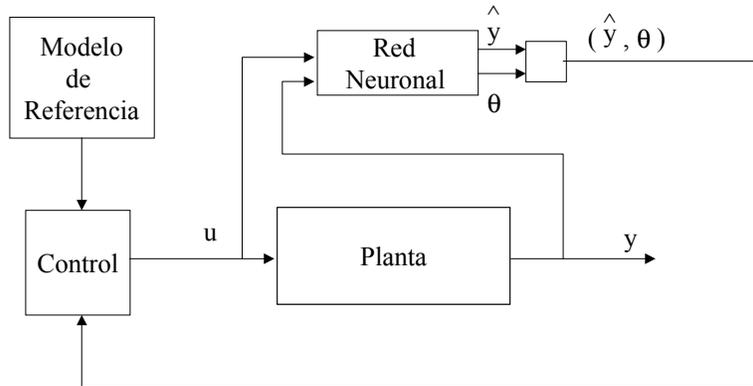


Figura 5-29: Estructura de control con un modelo neuronal

del tanque.

Símbolo	Variable	Valor nominal
Q_1	Flujo de la corriente ácida	$16.6mls^{-1}$
Q_2	Flujo de la corriente moderadora (Buffer)	$0.55mls^{-1}$
Q_3	Flujo de la corriente base	$15.6mls^{-1}$
$[HNO_3]_1$	Concentración de ácido en Q_1	$0.003molar$
$[NaHCO_3]_3$	Concentración del regulador en Q_2 (Trazas)	$0.00005molar$
$[NaOH]_3$	Concentración de la base en Q_3	$0.003molar$
$[NaHCO_3]_2$	Concentración del regulador en Q_2	$0.03molar$
pK_{a1}	$-\log_{10} K_{a1}$	6.35
pK_{a2}	$-\log_{10} K_{a2}$	10.33
pK_w	$-\log_{10} K_w$	14.00
W_{a1}	$[HNO_3]_1$	$0.003molar$
W_{a2}	$-[NaHCO_3]_2$	$-0.03molar$
W_{a3}	$-[NaHCO_3]_3 - [NaOH]_3$	$-0.00305molar$
W_{b2}	$[NaHCO_3]_2$	$0.03molar$
W_{b3}	$[NaHCO_3]_3$	$0.00005molar$
W_1	$[HNO_3]_1$	$0.003molar$
W_3	$-[NaOH]_3$	$-0.03molar$

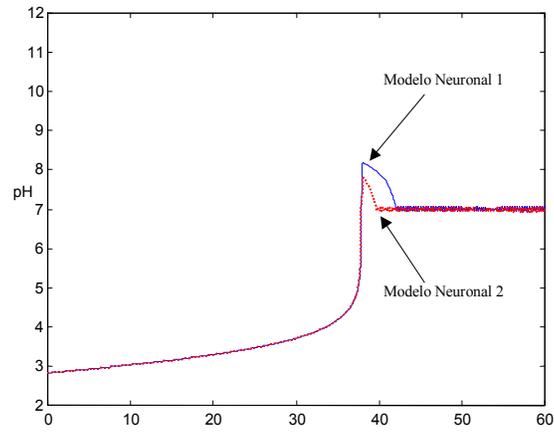


Figura 5-30: Respuestas con la red neuronal 1 y con la red neuronal 2

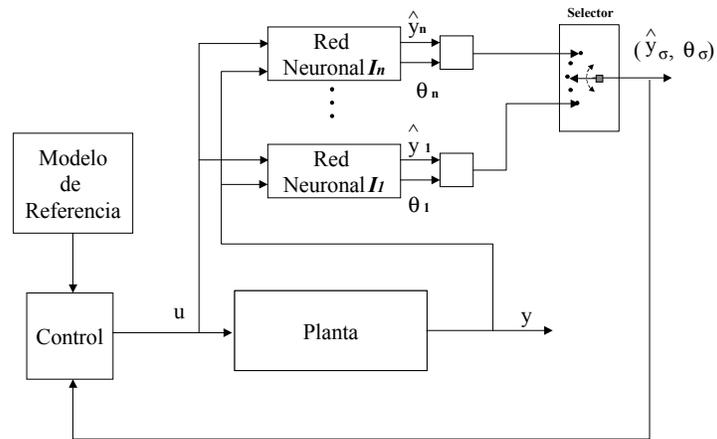


Figura 5-31: Estructura de control con múltiples redes neuronales

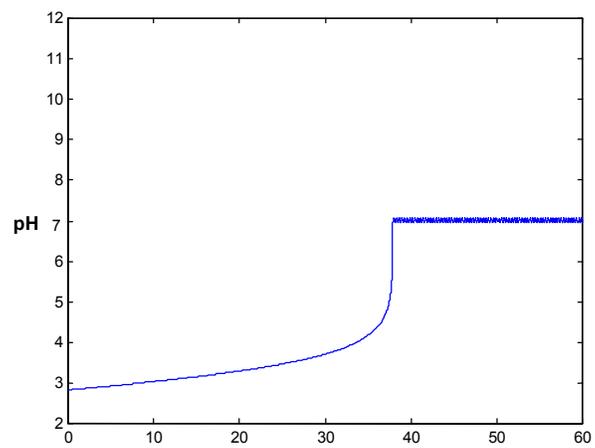


Figura 5-32: Respuesta con un control multimodelo neuronal

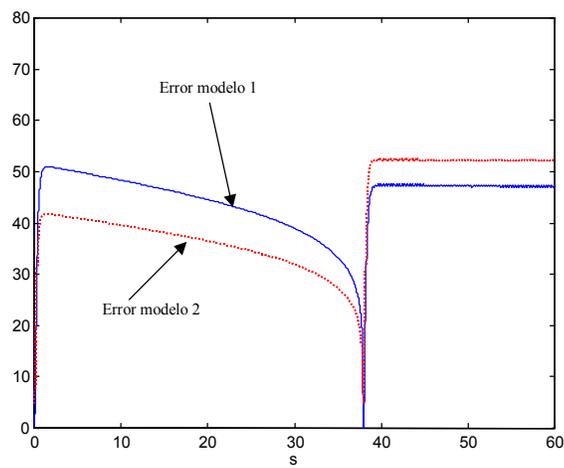


Figura 5-33: Error de los modelos 1 y 2

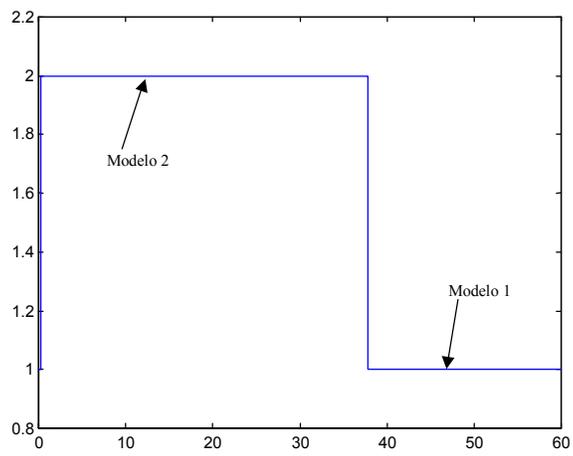


Figura 5-34: Selector

Capítulo 6

Identificación y control de un proceso de fermentación con múltiples RNA's dinámicas

Algunos procesos químicos presentan dinámicas complejas cuando las condiciones de operación cambian frecuentemente. Por ejemplo, los procesos de fermentación pueden experimentar diferentes procesos metabólicos dependiendo de las condiciones de operación [17]. De aquí que el modelo dinámico de estos procesos cambie cuando el punto de operación o las condiciones ambientales se modifican. Una red neuronal dinámica no puede seguir estos cambios en forma instantánea [12].

La microbiología, la bioquímica y la genética son disciplinas que nacen de la necesidad de entender los procesos de fermentación [18]. Por otro lado, los avances en teoría de control y en ingeniería de procesos, en relación con las aportaciones de estas tres disciplinas, pueden ayudar a desarrollar aplicaciones a procesos industriales, contribuyendo de esta manera al desarrollo de la biotecnología.

Los primeros procesos de fermentación tradicional fueron para la producción de cerveza y vinos, y para la fermentación de ácido láctico de productos derivados de la leche. La fermentación de alcoholes se atribuye a los Sumarios (10,000 años A.C), mientras que los Egipcios (5000 A.C.) aprendieron a utilizar hornos de levadura.

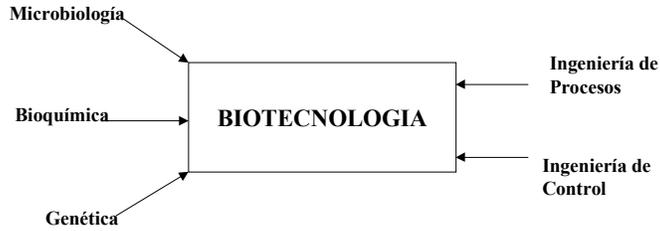


Figura 6-1: Relación de la biotecnología con otras disciplinas

El desarrollo de la microbiología se inició con Luis Pasteur en 1867, quien en su libro "Etude sur la biere" mostró que la levadura es la responsable de la fermentación; también descubrió que la bacteria de ácido láctico produce la descomposición del vino y de la cerveza.

En 1882, Robert Koch desarrolló un medio solido en el cuál es posible aislar medios de cultivo con sólo una célula. Un año después, Emil Christian aplicó por primera vez medios de cultivo puros a procesos de fermentación.

Un gran paso fué el desarrollo de la tecnología de fermentación. Este periodo inició en 1940 con la producción de penicilina, dando lugar al desarrollo de grandes fermentadores sumergidos. Poco después, en 1950 se introdujeron procesos continuos de fermentación y en la década de los 70s las células y enzimas inmovilizadas. En esta década la tecnología de computo entra en escena permitiendo por primera vez la implementación de algoritmos de control en computadoras.

La microbiología permitio conocer que en el proceso de fermentación de etanol las células estan creciendo y a la vez germinando nuevas células. La figura 6-2 ilustra como la célula de levadura esta asimilando componentes del azucar y del nitrógeno, y también esta secretando etanol y dióxido de carbono. Sin embargo, esto es sólo una fotografía superficial de lo que sucede en la célula de levadura.

La bioquímica llega más lejos y estudia los fenómenos que existen dentro de la célula, en donde la sucrosa es hidrolisada fuera de la pared de la célula, es decir, se transforma en glucosa y fructuosa. Luego los azúcares son fosforizados y pasan a traves de una serie de reacciones bioquímicas, cada una de las cuales se controla por una enzima para producir la energía necesaria para la célula. Los productos finales, dioxido de carbono y etanol se liberan.

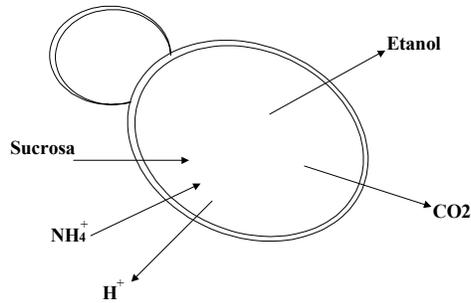


Figura 6-2: Fermentación y germinación de una célula de levadura

Todo el sistema se controla por algo así como 20 diferentes enzimas. Las enzimas se controlan mediante un sistema muy complejo dentro de la célula que tiene que ver con la síntesis y con la actividad de la enzima. Esto proporciona una idea de la complejidad del proceso de biosíntesis de una enzima. Finalmente cuando se ha formado la enzima su actividad se controla por medio de la activación e inhibición del producto final, en donde nuevamente el sistema de control es muy complejo. La Ingeniería de control debe buscar desarrollar sistemas de control que permitan mejorar los procesos de fermentación que experimentan una gran cantidad de reacciones internas que dan lugar a dinámicas muy complejas.

Los procesos de fermentación son un problema adecuado para evaluar el desempeño de un control multimodelo, debido a que en estos procesos los microorganismos pueden experimentar diferentes cambios metabólicos cuando las condiciones de operación cambian [17].

En este trabajo se propone un control adaptable multimodelo neuronal que se ilustra en la figura 4-7, con la finalidad de contrarrestar las variaciones en la dinámica de la planta, producto de la complejidad de la reacciones bioquímicas internas.

6.1 Dinámica de reactores biológicos

Un reactor biológico es un tanque en el que se presentan, en forma simultánea, diversas reacciones en un medio líquido. Estas reacciones pueden clasificarse en: reacciones de crecimiento microbiológico (comunmente llamadas reacciones microbiológicas) y reacciones de catalizadas

por enzimas (comunmente llamadas reacciones bioquímicas o biotransformaciones).

Reacciones de crecimiento microbiológico

El crecimiento de microorganismos, como por ejemplo bacterias, levaduras, etc., se produce por medio del consumo de nutrientes o substratos, los cuales contienen carbono, nitrógeno, oxígeno, etc. La masa de los microorganismos vivientes en el reactor se le llama biomasa.

Una reacción microbiológica típica se representa por:



en donde S representa la fuente de carbono (por ejemplo glucosa), O el oxígeno disuelto, X la biomasa y P expresa la existencia de un producto (o metabolito) asociado con el crecimiento de microorganismos. Esta es una reacción autocatalizada debido a que la producción de biomasa requiere de un cantidad inicial de biomasa

Reacciones catalizadas por enzimas

Estas son reacciones en donde un substrato S se transforma en un producto P a través de la acción catalítica de una enzima E . Una reacción bioquímica se representa por



Las enzimas son proteínas producidas por microorganismos vivos y estas pueden ser de dos tipos: enzimas extracelulares y enzimas intracelulares.

Las enzimas extracelulares son secretadas por la células en el dentro del medio líquido mientras que las enzimas intracelulares están confinadas dentro de las células de tal manera que la misma biomasa puede considerarse como catalizador:



Observación 6.1 *Un proceso biológico se define como un conjunto de m reacciones biológicas*

icas (como 6.1, 6.2 y 6.3), que involucran a n componentes. Al conjunto de estas reacciones se les conoce como red de reacción del proceso. La red de reacción representa una descripción cualitativa de las reacciones involucradas en el proceso, en donde pueden omitirse tanto los substratos que no son limitadores como los productos de una reacción que no son substratos en otras reacciones.

Los componentes que aparecen comúnmente en una red de reacción son:

- Población de microorganismos
- Enzimas
- Substratos externos
- Productos / substratos internos.

Los substratos externos son substratos que se introducen al reactor desde el exterior, mientras que los productos/substratos internos son los componentes que se producen por una reacción y que posiblemente pueden actuar como substratos de otras reacciones.

Modos de operación en reactores biológicos

En las aplicaciones industriales normalmente existen dos maneras de operar los reactores biológicos: operación en modo continuo y en modo por lotes con alimentación.

En un reactor biológico continuo se tiene un tanque con agitación en donde los substratos se alimentan en forma continua al reactor y una corriente de salida (efluente) se remueve del sistema de tal forma que el volumen del cultivo se mantiene constante (figura 6-3).

Por otro lado, un reactor biológico en modo de operación por lotes con alimentación opera sin efluente [27], es decir, el tanque inicialmente contiene una pequeña cantidad de substrato y de microorganismos, y se alimenta en forma continua con una corriente de substratos. Cuando el tanque se llena, y posiblemente después de un tiempo de reacción adicional, el contenido del tanque se remueve como un lote (figura 6-4).

Modelo dinámico general

A continuación se presenta un modelo general en variables de estado que describe la dinámica de los reactores biológicos.

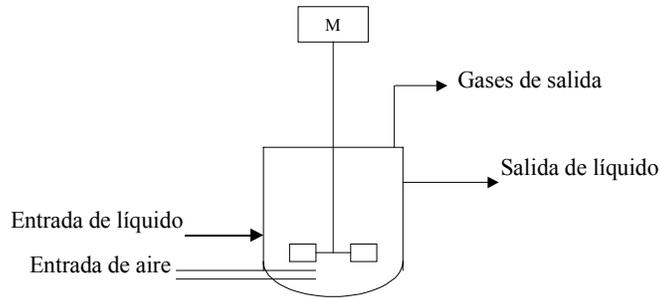


Figura 6-3: Reactor biológico en operación continua

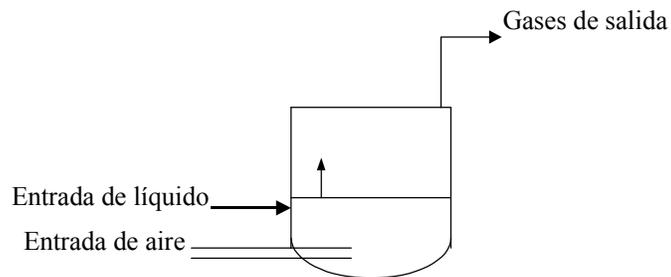


Figura 6-4: Reactor biológico en operación por lotes con alimentación

Las reacciones en el reactor se expresan en la red de reacción la cual esta conformada por m reacciones y n componentes (con $n > m$). El modelo del reactor biológico expresa el balance de masa de varios componentes dentro del reactor. Las concentraciones instantaneas de los n componentes se denotan por $\xi_i(t), i = 1, \dots, n$. El vector de estas concentraciones se llama composición del reactor y es el vector de estado del modelo.

$$\xi = (\xi_1(t), \xi_2(t), \dots, \xi_n(t))$$

La dinámica del balance de masa de cada componente en el reactor es resultado de dos mecanismos: de la cinética de reacción y del intercambio de materia con el ambiente. Esta dinámica se expresa como:

$$\dot{\xi}_i = \sum_{j \neq i} c_{ij} r_j(\xi) - D\xi_i + F_i - Q_i \quad (6.4)$$

en donde la sumatoria representa la cinética de reacción y los términos restantes el intercambio de materia con el ambiente. En el intercambio de materia con el ambiente F_i es la velocidad de suministro del componente ξ_i al reactor por unidad de volumen $[(gr\ h^{-1})\ L^{-1}]$, y Q_i representa la velocidad de remoción del componente ξ_i (en forma gaseosa) por unidad de volumen $[(gr\ h^{-1})\ L^{-1}]$.

En reactores continuos $D\xi_i$ representa la velocidad de remoción del componente ξ_i en forma líquida a través de la corriente de salida (efluente), D es la velocidad de dilución y D^{-1} es el tiempo de residencia. En reactores en operación por lotes con alimentación $D\xi_i$ representa la dilución del componente ξ_i en el tanque ocasionada por un incremento del volumen. La velocidad de dilución D es la velocidad específica de incremento de volumen $D \triangleq \frac{\dot{V}}{V}$, en donde V denota el volumen del líquido en el reactor. El término $\sum c_{ij} r_j(\xi)$, representa la velocidad de consumo y/o producción del componente ξ_i en el reactor de acuerdo con la red de reacción. La notación $j \neq i$, significa que la sumatoria se toma sobre las reacciones con índice j que involucran al componente ξ_i . Una velocidad de reacción $r_j(\xi)$ (unidades de masa por unidad de volumen), se asocia con cada reacción de la red y normalmente se representa con una función racional no negativa del estado ξ .

El vector de velocidades de reacción se conoce como cinética de reacción y se expresa como:

$$r(\xi) = (r_1(\xi), r_2(\xi), \dots, r_m(\xi))$$

Los coeficientes c_{ij} son coeficientes de producción adimensionales (unidades de masa por unidades de masa), los cuales son negativos cuando el componente ξ_i es un sustrato (es decir, cuando aparezcan en el lado izquierdo de la red de reacción), y positivos cuando el componente ξ_i sea biomasa o producto (es decir, cuando aparezcan en el lado derecho del

la red de reacción).

Introduciendo la siguiente notación matricial en la ecuación (6.4):

$$\begin{aligned} F &= (F_1, F_2, \dots, F_n)^T, \quad Q = (Q_1, Q_2, \dots, Q_n)^T \\ C &= \text{matriz } (n \times m) \text{ de coeficientes de producción } c_{ij} \end{aligned}$$

la dinámica de un reactor biológico se puede representar por el siguiente modelo en variables de estado:

$$\dot{\xi} = Cr(\xi) - D\xi + F - Q \quad (6.5)$$

Este modelo se conoce como *Modelo Dinámico General de Reactores Biológicos* [9].

Proceso de fermentación de penicilina

Existen dos ventajas fundamentales para utilizar procesos de fermentación por lotes con alimentación en lugar de los procesos por lotes convencionales: grandes posibilidades de incrementar la productividad del reactor y de mejorar el sistema de control. Estas ventajas adquieren una mayor importancia cuando los productos son metabolitos secundarios (por ejemplo penicilina), debido a que su producción en general está determinada por el estado del cultivo, el cual a su vez está relacionado con las condiciones ambientales del reactor.

La determinación de una estrategia de control óptima no es fácil y a menudo se complica por la presencia de restricciones que se imponen en las variables de estado. La optimización de procesos de fermentación por lotes con alimentación se ha intentado manipulando el flujo de alimentación del substrato, que es la variable de control natural en este tipo de procesos. Sin embargo, formular el problema considerando al flujo de alimentación como la variable de control produce un problema de control óptimo singular [63].

Esta situación a menudo se complica por la presencia de restricciones de estado, por ejemplo: una máxima concentración de biomasa permitida más allá de la cual la transferencia de oxígeno se ve limitada, o una máxima concentración de substrato para minimizar

reacciones secundarias indeseables.

La combinación de un problema de optimización singular con restricciones de estado representa un problema numérico importante. En la práctica normalmente se utilizan perfiles constantes por pedazos o rampas en el flujo de alimentación, bajo el conocimiento de que es posible mejorar la operación si se divide en varias etapas las diferentes fases del proceso de fermentación, y utilizando el perfil de flujo de alimentación para dividir dichas etapas.

En muchos casos se presenta un problema cuando el desempeño del reactor biológico depende en gran medida tanto de la concentración de biomasa como de substrato en el reactor, y se dispone sólo de una variable de control, el flujo de alimentación. En [63] se prueba que en presencia de restricciones de estado la elección de la variable de control es muy importante, y se propone que el desempeño del reactor biológico se puede optimizar mejor en términos de un perfil de substrato en el reactor que se implementa manipulando la concentración de substrato en el flujo de alimentación.

El proceso de fermentación de penicilina que se propone en [63] se toma como ejemplo para estudiar el desempeño del esquema de control multimodelo neuronal. Se asume que se conocen las restricciones en la concentración de substrato y de biomasa que permiten incrementar la productividad del reactor biológico, y que el proceso puede operarse en dos fases: fase de crecimiento y fase de producción de penicilina.

Modelo matemático

Se utiliza un modelo propuesto por Bajpai y Reuss en donde los valores de los parámetros del modelo se estimaron con datos experimentales, y que es consistente con los valores observados de la velocidad específica de crecimiento $\mu(S, X)$, y de formación de penicilina $\theta(S)$ [7]. El modelo utiliza el modelo de crecimiento de Contois [16] para el cálculo de la velocidad de crecimiento específica:

$$\mu(S, X) = \frac{(\mu_m S)}{S + K_1 X}$$

en donde S denota la concentración de substrato en el reactor y X la concentración de microorganismos.

El consumo de substrato requerido para el crecimiento de microorganismos y para la

formación de penicilina se considera proporcional a las velocidades correspondientes con factores de producción Y_X y Y_P respectivamente. Los requerimientos de mantenimiento de células se satisfacen por el sustrato a una velocidad determinada por la demanda de mantenimiento específico constante M_X .

La penicilina se forma a una velocidad específica θ , que se determina por el siguiente modelo de inhibición de sustrato:

$$\theta(S) = \frac{\theta_m}{1 + K_P/S + S/K_i}$$

También se considera una velocidad de decaimiento constante K , para la penicilina.

Las consideraciones anteriores producen el siguiente modelo dinámico para la fermentación de penicilina en un proceso por lotes con alimentación:

$$\begin{aligned} \dot{X} &= \mu(S, X)X - \frac{FX}{V} \\ \dot{S} &= \frac{F(S_I - S)}{V} - \frac{\theta(S)X}{Y_P} - \frac{\mu(S, X)X}{Y_X} - M_X X \\ \dot{P} &= \theta(S)X - KP - \frac{FP}{V} \\ \dot{V} &= F \end{aligned} \tag{6.6}$$

en donde F es el flujo de alimentación, V es el volumen en el reactor, S_I es la concentración del sustrato en la alimentación, y K es la velocidad constante de decaimiento de penicilina.

El modelo anterior (6.6) no toma en cuenta el hecho de que una alta concentración de biomasa afecta la capacidad de transferencia de oxígeno en el reactor, lo que puede dar lugar a una limitación de oxígeno [7]. En lugar de incluir la dinámica de transferencia de oxígeno en el modelo (6.6), parece más realista evaluar el desempeño del reactor imponiendo un límite en la concentración de biomasa en el reactor [63]. De la misma manera es deseable limitar la concentración de sustrato en el reactor con la finalidad de evitar reacciones no deseadas. Por lo anterior, las siguientes restricciones deben observarse en la operación del

proceso de fermentación.

$$S \leq S_{\max}, \quad X \leq X_{\max} \quad (6.7)$$

en donde S_{\max} , y X_{\max} representan los límites superiores en la concentración de substrato y biomasa respectivamente. En este ejemplo se considera: $S_{\max} = 100 \text{ gr } L^{-1}$, y $X_{\max} = 40 \text{ gr } L^{-1}$ [63].

Estrategia de control

La estrategia de control se divide en dos fases: la primera es la fase de crecimiento de microorganismos, y la segunda fase es de producción de penicilina. La figura 6-10 ilustra la estrategia de control.

En la fase de crecimiento se debe acumular biomasa lo más rápido posible sin que se rebasen las restricciones (6.7). La variable de control (en éste caso la concentración del flujo de alimentación S_I), debe ser tal que, la concentración de substrato en el reactor S se mantenga en el valor más alto (si es posible en S_{\max}), lo cual acelerará el crecimiento de biomasa hasta que $X = X_{\max}$.

En la fase de producción de penicilina se debe mantener la concentración de biomasa en su valor límite X_{\max} , ya que a mayor cantidad de microorganismos mayor será la producción de penicilina, y al mismo tiempo se debe mantener la concentración de substrato en el reactor S en un nivel bajo para permitir una máxima velocidad de producción de penicilina.

Observación 6.2 *En la fase crecimiento se aplica un escalón $S_I = S_{\max}$. Este valor constante de S_I eleva la concentración en el reactor S , pero esta concentración no se puede mantener constante en el reactor debido a que el crecimiento exponencial de microorganismos tiene como consecuencia un consumo exponencial de substrato, del tal manera que la concentración de substrato en el reactor decae exponencialmente, y una vez que el proceso entra en la fase de producción, la concentración de substrato es suficientemente baja para propiciar la producción de penicilina.*

Observación 6.3 *En muchos casos la variable de control que se utiliza es el flujo de alimentación F . Sin embargo, la limitante de utilizar el flujo de alimentación F como señal de*

control puede observarse en las siguientes ecuaciones:

$$\dot{X} = \mu(S, X)X - \frac{FX}{V} \quad (6.8)$$

$$\dot{S} = \frac{F(S_I - S)}{V} - \frac{\theta(S)X}{Y_P} - \frac{\mu(S, X)X}{Y_X} - M_X X \quad (6.9)$$

Cuando la concentración de biomasa alcanzó su valor máximo el crecimiento debe cesar ($\dot{X} = 0$), esto determina a la variable de control:

$$F = V\mu(S, X) \quad (6.10)$$

Como resultado, se ha perdido la libertad para controlar la concentración de substrato en el reactor S , ya que la dinámica del substrato esta completamente definida por la ecuación (6.9), con F dada por la ecuación (6.10).

Los valores de los parámetros del modelo se tomarón de [63] y se presentan en la tabla 6.11.

Parámetro	Descripción	Valores
μ_m	Parámetro de crecimiento de microorganismos.	0.11 h^{-1}
K_1	Parámetro de crecimiento de microorganismos.	0.006
θ_m	Parámetro de formación de penicilina.	0.004 h^{-1}
K_P	Parámetro de formación de penicilina.	0.0001 gL^{-1}
K_i	Parámetro de formación de penicilina.	0.1 gL^{-1}
K	Factor de velocidad de decaimiento de penicilina.	0.01 h^{-1}
Y_X	Factor de producción de microorganismos.	0.47
Y_P	Factor de producción de penicilina..	1.2
M_X	Constante de mantenimiento de células.	0.029 h^{-1}

Tabla 6.1 Parámetros del modelo de fermentación de penicilina

6.2 Identificación con un multimodelo neuronal dinámico

En los procesos de fermentación los microorganismos pueden experimentar diferentes transformaciones metabólicas cuando las condiciones de operación cambian. Esta situación hace que la dinámica de este tipo de procesos sea muy sensible a cambios en las condiciones de operación. Por esta razón este tipo de procesos son interesantes para poner a prueba el desempeño del identificador multimodelo neuronal que se propone en el presente trabajo.

El proceso de fermentación se lleva a cabo en un reactor biológico con microorganismos, los cuales crecen al comer un substrato que es la alimentación al reactor. Los dos componentes principales del proceso son la concentración de microorganismos x_1 , y la concentración de substrato en el reactor x_2 . El modelo del bioreactor es un modelo de fermentación continuo y su dinámica se representa por el siguiente conjunto de ecuaciones diferenciales.

$$\begin{aligned}\dot{x}_1 &= \frac{a_1 x_1 x_2}{a_2 x_1 + x_2} - u x_1 \\ \dot{x}_2 &= -\frac{a_3 a_1 x_1 x_2}{a_2 x_1 + x_2} - u x_2 + u a_4 \\ y &= x_1\end{aligned}\tag{6.12}$$

en donde u es el substrato que se alimenta al reactor y las constantes a_i son positivas. El modelo considera que la velocidad de crecimiento de microorganismos esta dada por el modelo de Contois [16]. Para efectos de simulación los valores de los parámetros fueron $a_1 = a_2 = a_3 = 1$ y $a_4 = 0.1$. Estos valores garantizan que el volumen es constante cuando el substrato se transforma en microorganismos. El modelo y los datos de simulación se tomarán de [19].

El experimento de simulación tiene la finalidad de evaluar el desempeño del identificador multimodelo neuronal cuando se introducen variaciones en la entrada u (substrato de alimentación), que hagan que la planta experimente una dinámica compleja que nunca alcance un valor en estado estacionario.

6.2.1 Identificación con un modelo neuronal

La red neuronal dinámica que se utilizó en el proceso de identificación tiene la forma:

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)\gamma(u_t)$$

con la ley de adaptación de pesos dada por:

$$\begin{aligned}\dot{W}_{1,t} &= -P\Delta_t\sigma^T(\hat{x}_t) \\ \dot{W}_{2,t} &= -P\phi(\hat{x}_t)\gamma(u_t)\Delta_t^T\end{aligned}$$

los parámetros de la red neuronal se ajustaron de la siguiente manera:

$$A_1 = \text{diag}[-3, -3], \quad P_1 = \text{diag}[2, 2], \quad W_{1,0}^1 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad W_{2,0}^1 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

El resultado de utilizar sólo un identificador neuronal se muestra en la figura 6-5.

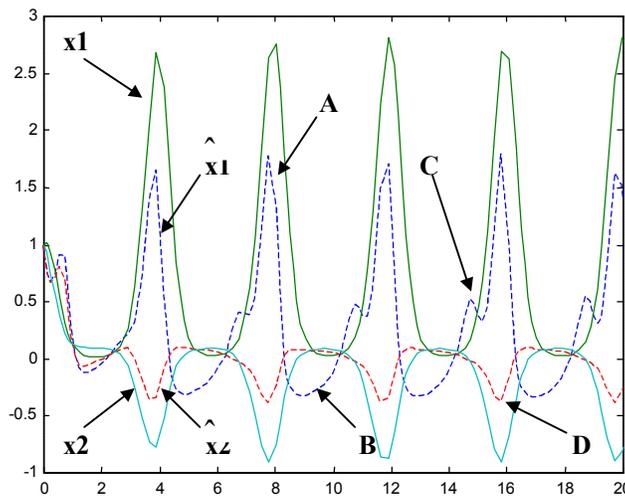


Figura 6-5: Identificación con una red neuronal

En las zonas A,B,C, y D de la figura 6-5 se observa que una red neuronal dinámica

tiene algunas dificultades para aproximar el comportamiento del proceso de fermentación. A continuación se presentan los resultados de simulación cuando se utiliza el identificador multimodelo neuronal y se comparan con los errores de identificación de ambos enfoques.

6.2.2 Identificación con un multimodelo neuronal

En identificador multimodelo neuronal esta formado por tres diferentes RNAs dinámicas. Los parámetros de ajuste de estas redes neuronales son los siguientes:

$$\begin{aligned}
 A_1 &= \text{diag}[-3, -3], & P_1 &= \text{diag}[2, 2], & W_{1,0}^1 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, & W_{2,0}^1 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \\
 A_2 &= \text{diag}[-2, -2], & P_2 &= \text{diag}[2, 2], & W_{1,0}^2 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, & W_{2,0}^2 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \\
 A_3 &= \text{diag}[-11, -11], & P_3 &= \text{diag}[7, 7], & W_{1,0}^3 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, & W_{2,0}^3 &= \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}
 \end{aligned}$$

En la figura 6-6 se muestra el resultado de utilizar el identificador multimodelo neuronal para identificar la dinámica de este proceso de fermentación.

Si se observan las zonas A, B, C, y D que se señalan en figuras 6-6 y 6-5, se puede concluir que el identificador multimodelo aproxima mejor la dinámica de la planta que cuando se utiliza sólo un identificador. Por ejemplo, en las zonas A y C, asociadas al crecimiento de microorganismos x_1 , se puede apreciar que el multimodelo logra seguir más tiempo a x_1 en su ascenso, y no experimenta un cambio de dirección (ver zona C) como en el caso de la figura 6-5.

La figura 6-7 muestra el estado del selector en su tarea de buscar al identificador dentro de la estructura multimodelo, que mejor represente la dinámica de la planta.

Es interesante observar (figuras 6-6 y 6-7), como la selección de los modelos no es repetitiva como sucede con la dinámica de la planta. Esto se debe a que se utilizan identificadores dinámicos que continuamente están compitiendo por una mejor aproximación de la dinámica de la planta.

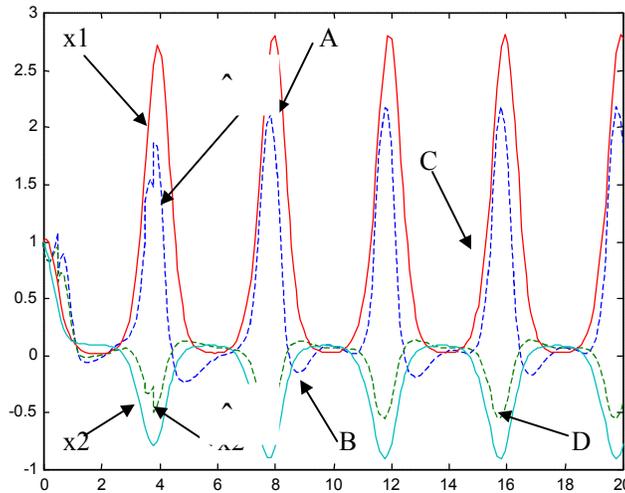


Figura 6-6: Identificación con múltiples redes neuronales

En la figura 6-8 se observa el error de identificación en la concentración de microorganismos ($x_1 - \hat{x}_1$). En este caso se verifica que el identificador multimodelo aproxima mejor la dinámica de la concentración de microorganismos. Para la concentración de sustrato x_2 , la figura 6-9 también refleja un mejor desempeño del identificador multimodelo.

De estos resultados de simulación se puede concluir que el desempeño del identificador multimodelo con redes neuronales dinámicas puede tener mayor capacidad de aproximación de dinámicas complejas, como es el caso de este proceso de fermentación continua.

6.3 Control con un multimodelo neuronal dinámico

6.3.1 Control con un modelo neuronal

En la figura 6-10 se observa como se implementa la estrategia de control que se describe en el proceso de fermentación de penicilina. En la fase de crecimiento (fase 1) se utilizó un escalón en la concentración del flujo de alimentación, con el cuál es posible producir un crecimiento exponencial de los microorganismos. El instante en el que se debe hacer la transferencia a la fase de producción es cuando se alcanza el límite máximo de concentración

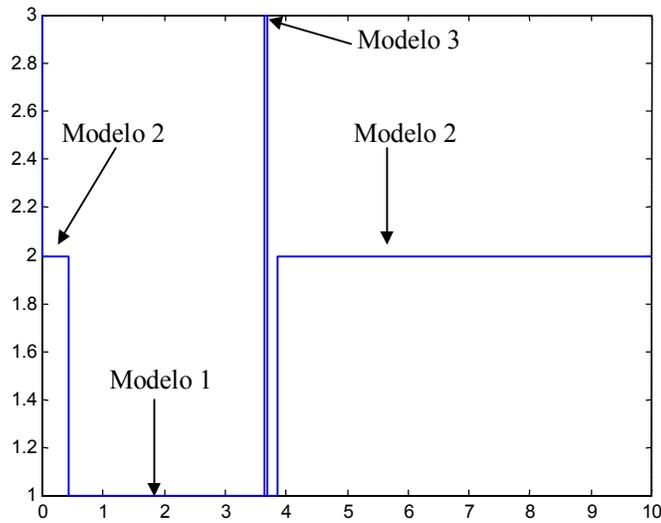


Figura 6-7: Selección de modelos

de microorganismos X_{\max} , por esta razón se introduce un selector que transfiere a la fase de producción en el instante en que se alcanza dicho valor máximo.

Observación 6.4 *En teoría si en la fase de producción se mantiene la concentración de microorganismos en su nivel máximo X_{\max} , se lograría maximizar la producción de penicilina. Pero al mismo tiempo también es necesario mantener la concentración de substrato en el reactor S en un nivel bajo (porque es una condición para tener un velocidad de formación de producto máxima), entonces los esfuerzos del controlador para mantener X constante en su valor máximo pueden producir incrementos en S , que ocasionarían una deficiencia en la producción de penicilina. Debido a que en la fase de producción la concentración de microorganismos no se pudo mantener constante, se observa un deacimamiento en la concentración de microorganismos.*

En la fase de producción de penicilina (fase 2), el objetivo de control es mantener la concentración de microorganismos en su valor más alto. Para realizar esta tarea se utiliza una estrategia de control adaptable indirecto en donde el modelo de identificación se obtiene con una red neuronal de la forma:

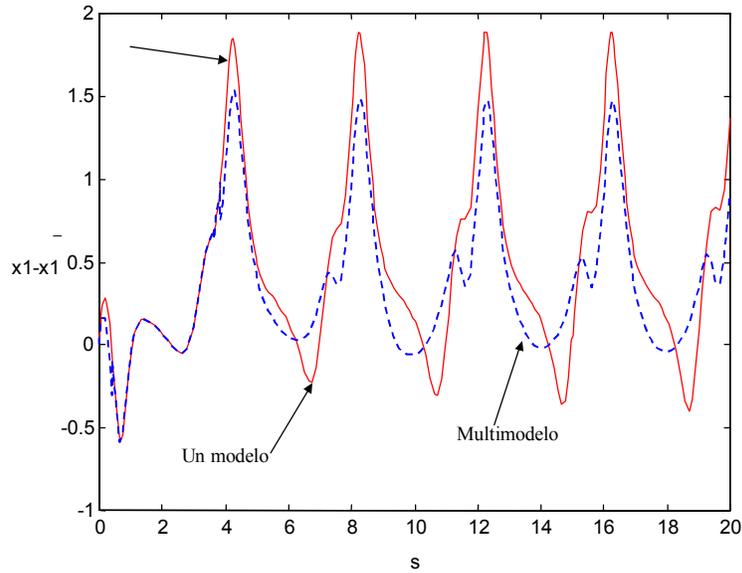


Figura 6-8: Error de identificación en la concentración de microorganismos x_1

$$\dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + u_t \quad (6.13)$$

y la ley de aprendizaje para el ajuste de pesos es

$$\dot{W}_{1,t} = -P\Delta_t\sigma^T \quad (6.14)$$

El modelo de identificación neuronal se ajustó con los siguientes parámetros:

$$A = -0.2, \quad P = 7.3, \quad W_{1,0} = 1, \quad \sigma(\hat{X}) = \frac{-0.12}{1 + e^{-2\hat{X}}} \quad (6.15)$$

y la ley de control que se utilizó corresponde al caso de aproximación exacta de la dinámica no modelada que se establecieron en el capítulo 4 (ecuaciones (4.16) y (5.14)).

En este tipo de procesos de fermentación en donde se tiene producción de penicilina, es importante que la concentración de microorganismos sea alta en la fase de producción, ya que la producción de penicilina es directamente proporcional a la cantidad de microorganismos.

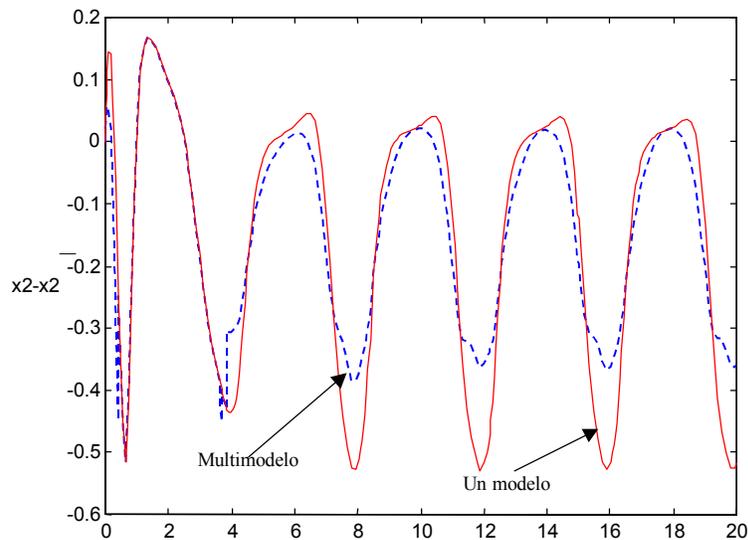


Figura 6-9: Error de identificación en la concentración de sustrato x_2 .

Sin embargo, en este proceso el objetivo principal es maximizar la concentración del producto, en este caso la concentración de penicilina.

Con este control neuronal se alcanzó una concentración de microorganismos de $30.7 \text{ gr } L^{-1}$, y la concentración final de penicilina fue de $6.2 \text{ gr } L^{-1}$, y la concentración final de penicilina fue de $6.2 \text{ gr } L^{-1}$.

A pesar de que en la fase de crecimiento se aplicó un escalón de $100 \text{ gr } L^{-1}$, la concentración en el reactor experimenta un decaimiento exponencial. Esto seguramente se debe a que el aumento exponencial en la concentración de microorganismos produce una demanda también exponencial de alimento. En la figura 6-13, se observa el decaimiento exponencial de la concentración de sustrato en el reactor. El error de identificación del modelo neuronal se muestra en en la figura 6-14.

6.3.2 Control con un multimodelo neuronal

En la sección anterior se utilizó sólo un modelo neuronal para identificar la fase de producción de penicilina. Ahora se utilizarán dos modelos neuronales para identificar la dinámica de la fase de producción. Con estos dos identificadores en línea se construye un controlador

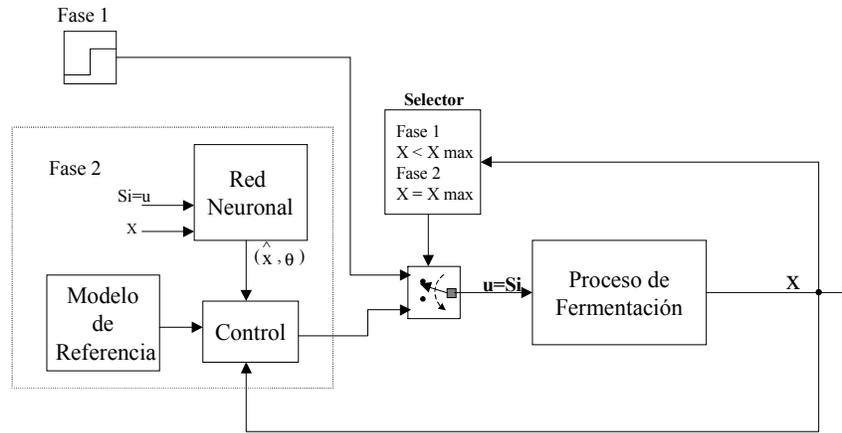


Figura 6-10: Control en 2 fases de un reactor biológico

multimodelo neuronal como el de la figura 4-7. Las dos redes neuronales dinámicas que se utilizan para la identificación de la planta son las siguientes:

$$A_1 = -1. \quad P_1 = 4, \quad W_{1,0}^1 = 1, \quad \sigma_1(\hat{X}) = \frac{-0.15}{1 + e^{-2\hat{X}}} - 0.02 \quad (6.16)$$

$$A_2 = -0.2. \quad P_2 = 7.3, \quad W_{1,0}^2 = 1, \quad \sigma_2(\hat{X}) = \frac{-0.12}{1 + e^{-2\hat{X}}} \quad (6.17)$$

La segunda red neuronal es idéntica a la que se utilizó en la sección anterior. La respuesta del controlador multimodelo se compara con la respuesta del controlador con un modelo neuronal de la sección anterior.

En la figura 6-15 se observa como el controlador multimodelo logra contrarrestar el decaimiento en la concentración de microorganismos, en comparación con el controlador que utiliza sólo un modelo neuronal. El valor final de la concentración de biomasa cuando se utiliza el controlador multimodelo es de 32.2 grL^{-1} , lo cuál representa un incremento en la concentración de 1.5 grL^{-1} .

De interés especial es la concentración de penicilina ya que las variaciones en la concentración de penicilina se reflejarán directamente en los ingresos económicos que se obtengan en cada lote. Como una mayor cantidad de microorganismos producirán una mayor cantidad de metabolitos secundarios (penicilina), es importante mantener la concentración de

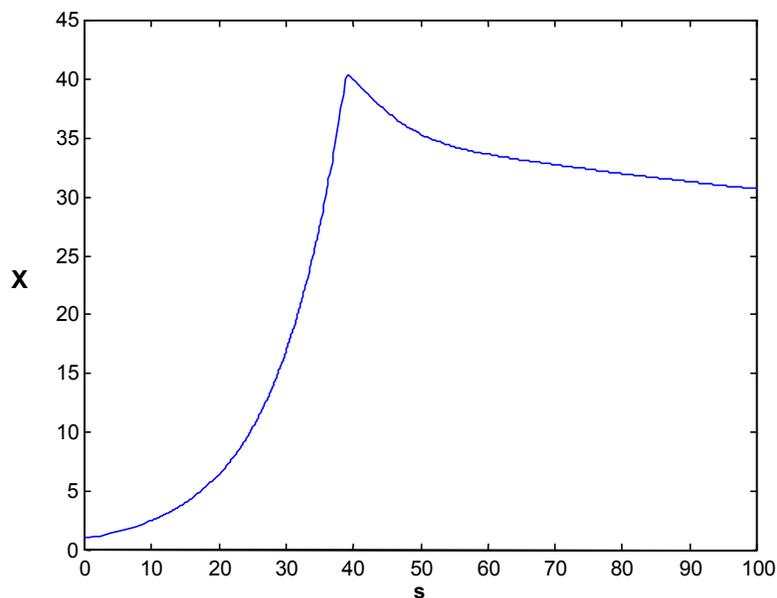


Figura 6-11: Control con una red neuronal

microorganismos en su nivel máximo aceptable X_{\max} durante la fase de producción.

En la práctica los procesos de fermentación presentan dinámicas variantes en el tiempo, en particular las velocidades de crecimiento y de producción son muy sensibles a variaciones en las condiciones de operación. El controlador multimodelo neuronal se implementa con la finalidad de contrarrestar estas variaciones. En las figuras 6-15 y 6-16, se puede apreciar como una mayor concentración de microorganismos durante la fase de producción trae como consecuencia un incremento en la concentración final de penicilina.

El cálculo de la ley de control se basa en información que se generará en el proceso de identificación en línea de la planta. Es razonable suponer que si un identificador aproxima mejor la dinámica de la planta entonces este producirá un controlador con mejor desempeño. En la figura 6-17 se puede apreciar que el error de identificación del multimodelo neuronal es menor al error de identificación cuando se utiliza sólo un modelo neuronal, de aquí que las concentraciones finales de microorganismos y de penicilina sean mayores cuando se utiliza el controlador multimodelo.

Este resultado es consistente con las concentraciones de microorganismos y de penicilina

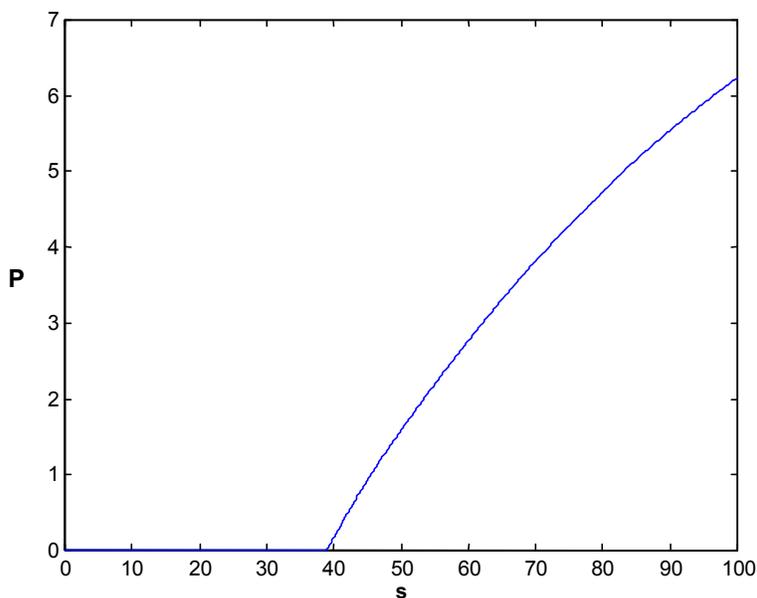


Figura 6-12: Concentración de penicilina con una red neuronal

que se muestran en las figuras 6-15 y 6-16. Debido a que el identificador multimodelo aproxima mejor la dinámica de la planta, las concentraciones de biomasa y de penicilina cuando se utiliza el enfoque multimodelo se incrementan.

Los modelos de identificación realizan un aprendizaje en línea de la dinámica de la planta, al mismo tiempo que, para cada uno de los modelos, se calcula un índice de desempeño que mide la calidad de cada modelo en cada instante de tiempo. A través de un algoritmo de selección con histéresis, se selecciona el modelo de identificación que mejor representa a la planta en sus diferentes regiones de operación. De esta manera, el modelo seleccionado o activo, puede observarse monitoreando el estado que del selector.

En la figura 6-18, se observa como los modelos de identificación se están seleccionando conforme evoluciona el proceso. Se debe aclarar que la fase de producción inicia aproximadamente en $t > 40 h$. Cuando arranca esta fase, el modelo 1 está activo y permanece así aproximadamente hasta $t = 62 h$. En este momento el algoritmo de selección detecta que el modelo 2 es más exacto que el del modelo 1, y transfiere la salida al modelo 2.

En la figura 6-17, se observan las dinámicas de los errores de identificación del multi-

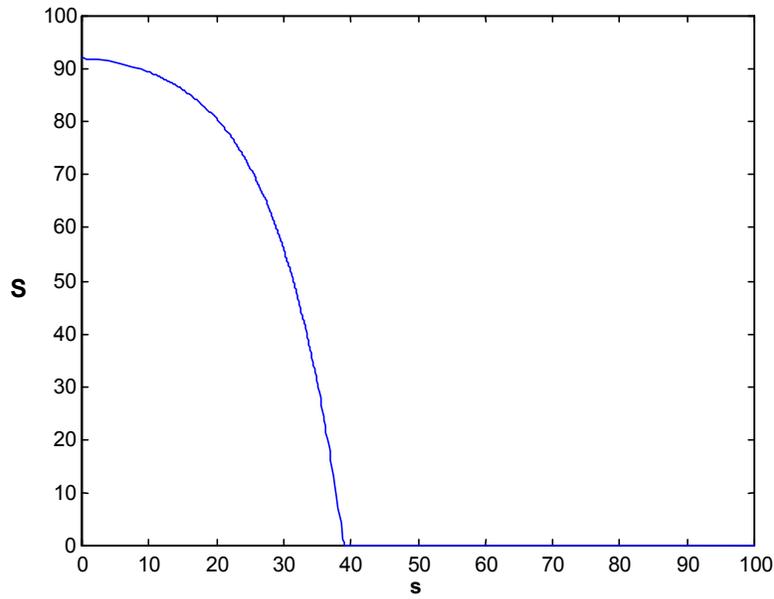


Figura 6-13: Concentración de substrato en el reactor

modelo neuronal y de cada uno de los modelos que conforman al controlador multimodelo. Los cruces de los errores de identificación de los modelos 1 y 2, coinciden con los instantes en que el selector transfiere de un modelo a otro. Aproximadamente en $t = 62$, se puede apreciar claramente un cruce que corresponde a una transferencia del selector en la figura 6-18.

En la figura 6-19 también se puede observar que el error de identificación del multimodelo neuronal es menor que cada uno de los errores de los modelos neuronales 1 y 2. El hecho de que el multimodelo aproxime mejor la dinámica de la planta contribuye a que se obtenga una mayor concentración tanto de biomasa como de penicilina cuando se utiliza el controlador multimodelo neuronal.

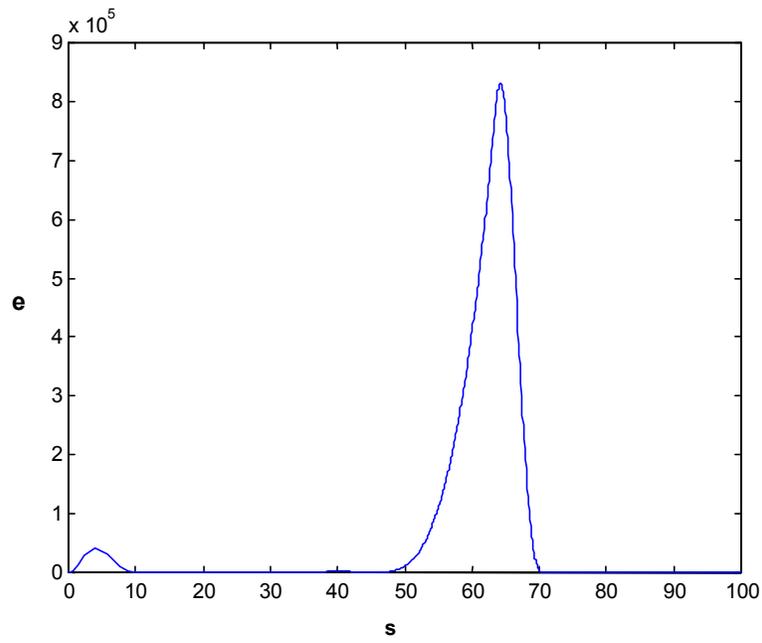


Figura 6-14: Error de identificación con una red neuronal.

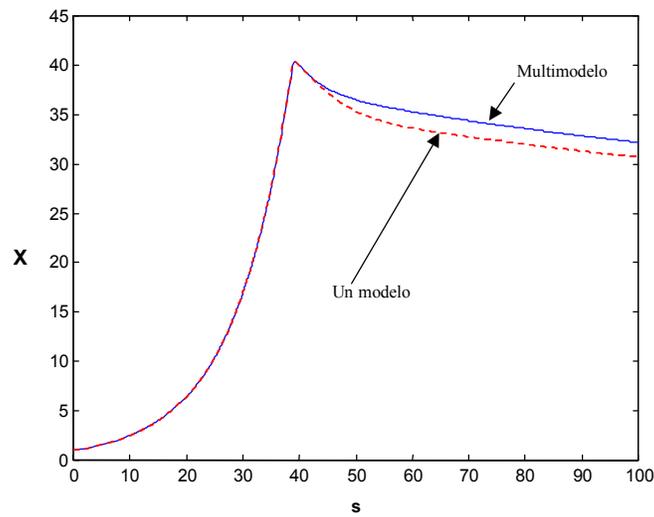


Figura 6-15: Comparación en la concentración de microorganismos

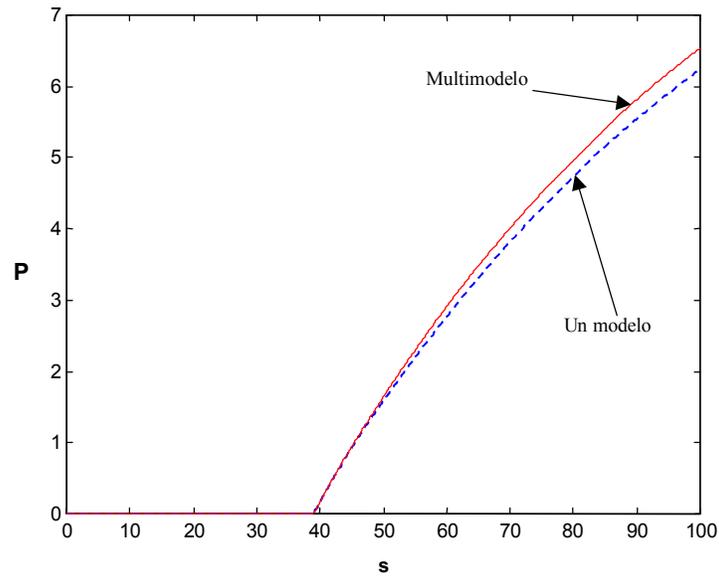


Figura 6-16: Comparación de la concentración de penicilina

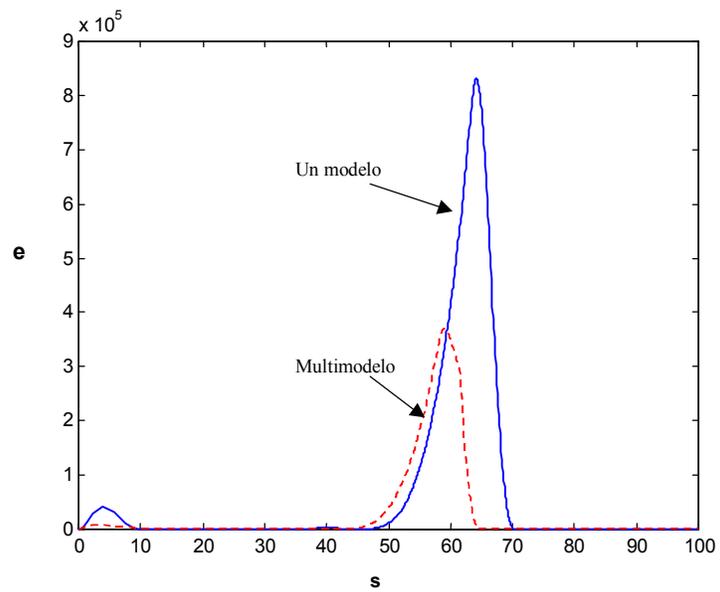


Figura 6-17: Errores de identificación con un modelo y con un multimodelo neuronal

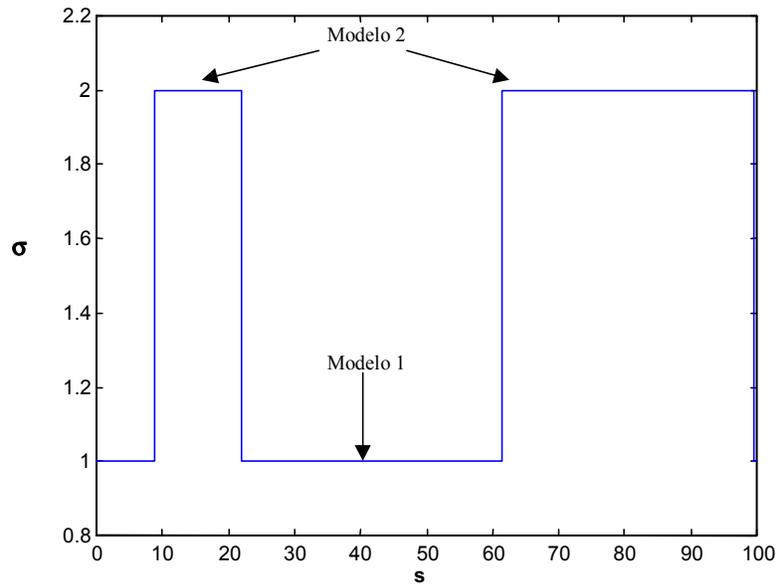


Figura 6-18: Selección de modelos

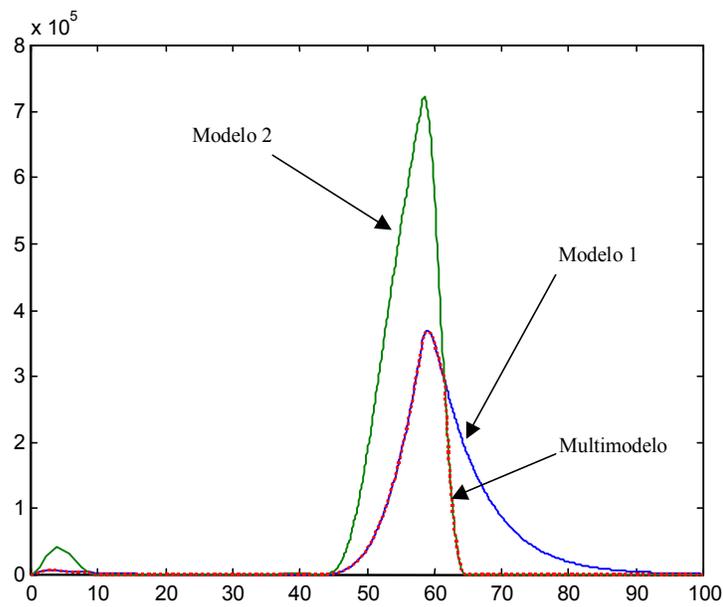


Figura 6-19: Errores de identificación multimodelo y de las redes neuronales 1 y 2.

Capítulo 7

Conclusiones

La necesidad de poder representar el comportamiento de procesos químicos que experimentan comportamientos diferentes cuando las condiciones de operación cambian ha dado lugar al surgimiento del enfoque multimodelo.

La técnica multimodelo no es una idea no es nueva y a la fecha se han desarrollado diferentes enfoques. La mayoría de los trabajos utilizan modelos fijos para modelar el comportamiento de la planta en diferentes condiciones de operación, lo que asume que se conoce *a priori* el comportamiento de la planta. Sin embargo, se puede presentar un problema cuando las nuevas condiciones de operación generen un comportamiento de la planta que no se conocía *a priori*.

De la combinación del enfoque multimodelo con las técnicas de identificación con redes neuronales dinámicas surge el identificador multimodelo neuronal que se propone en el presente trabajo. Este identificador se utilizó para identificar dos procesos químicos altamente no lineales: un proceso de neutralización y un proceso de fermentación. En ambos casos, los resultados de simulación muestran que el identificador multimodelo neuronal puede representar con mayor exactitud el comportamiento de la planta en presencia de cambios bruscos en el punto de operación, como en el caso del proceso de neutralización; o en presencia de dinámicas complejas como es el caso del proceso de fermentación. Se debe notar que el identificador multimodelo neuronal no tiene conocimiento previo de los cambios en la dinámica de la planta, lo cual es una muestra de su capacidad de modelar situaciones inesperadas.

Es razonable suponer que en un control adaptable indirecto, un buen modelo de identifi-

cación corresponderá a un buen desempeño del sistema de control. El controlador multimodelo se basa en la idea anterior, y por esta razón se construye un identificador multimodelo y un algoritmo de selección que se encargan de proporcionar al controlador la información dinámica del mejor identificador.

El controlador multimodelo neuronal se probó en el control de pH en el proceso de neutralización y en un proceso de fermentación, para la producción de penicilina. Los resultados que se obtuvieron muestran que el control multimodelo tiene un mejor desempeño que el control que utiliza sólo un modelo de identificación.

La aplicación del controlador multimodelo neuronal a procesos químicos arroja resultados alentadores que muestran la viabilidad de combinar el enfoque multimodelo con redes neuronales dinámicas en otras áreas de aplicación.

Los esquemas de identificación y de control multimodelo neuronal propuestos se implementarán a nivel simulación con buenos resultados y pueden servir como punto de partida para el desarrollo de experimentos en laboratorio, en donde se presentarán otras dificultades interesantes, ante las cuales habrá que evaluar el desempeño del controlador multimodelo en presencia de errores de medición, ruido u otros efectos dinámicos debidos a los sensores, actuadores y válvulas de control.

Bibliografía

- [1] AI-Akhras, M.A., Aly G.M., Green R.J., "Neural network learning approach of intelligent multimodel controller," *IEE Proceedings .-Control Theory Appl.* **43**(4), pp. 395-400, 1996.
- [2] Aly, G., Badr, A. and Binder, Z., "Multimodel control of MIMO system: Location and control algorithms," *Int. J. Sys. Sci.*, **19**(9), pp.1687-1698, 1988.
- [3] Astrom, K.J. and Hagglund, T., "Automatic Tuning of PID Controllers," *Instrument Society of America*, 1988.
- [4] Athans, M., et al., "The Stochastic Control of the F-8C Aircraft Using a Multimodel Adaptive Control (MMAC) Method - Part.I Equilibrium Flight," *IEEE Trans. on Automatic Control*, AC-22: pp.768-780, 1977.
- [5] Badr, A., Binder, Z. and Rey, D., "Weighted multimodel control," *Int. J. Sys. Sci.*, **23**(1), pp.145-149, 1992.
- [6] Badr, A., Binder, Z. and Rey, D., "Application of tracking multimodel control to a nonlinear thermal process," *Int. J. Sys. Sci.*, **21**(9), pp.1795-1803, 1990.
- [7] Bajpai, R.K. and Reuss, R., *Biotechnology and Bioeng.*, **23**, pp. 717, 1981.
- [8] Balchen, J.B., "How have we arrived at the present state of knowledge in process control? Is there a lesson to be learned," *Journal of Process Control*, **9**, pp. 101-108, 1999.
- [9] Bastin G., "Nonlinear and adaptive control in biotechnology: A tutorial" *European Control Conference*, Grenoble, France, July 2-5, pp. 2001-2012, 1991.

- [10] Bhat, N., Minderman, P., McAvoy, T. and Wang, N., "Modeling Chemical Process Systems via Neural Computation," *Proc. 3rd. Int. Symp. 'Control for profit'*, Newcastle-upon-Tyne, 1989.
- [11] Bhat, N., Minderman, P. and McAvoy, T., "Use of Neural Nets for modelling of Chemical Process Systems," *Preprints IFAC Symp. Dycord +89, Maastricht*, The Netherlands, Aug. 21-23, pp.147-153, 1989.
- [12] Boukhris, A., Mourot, G., and Ragot, J., "Nonlinear Dynamic System Identification: A Multimodel Approach," *International Journal of Control*, **72**, pp. 591-604, 1999.
- [13] Chen. L., G. Bastin and D. Dochain, "On the model identifiability of stirred tank bioreactors," *ECC*, Grenoble, July, 1991.
- [14] Chinrungrueng, C. and Séquin, C.H., "Optimal adaptive k-means algorithm with dynamic adjustment of learning," *IEEE Trans. on Neural Networks*, **6**, pp. 157-169, 1995.
- [15] Commuri, S. and Lewis, F.L., "CMAC Neural Networks for Control of Nonlinear Dynamical Systems: Structure, Stability and Passivity," *Automatica*, **33**(4), pp. 635-641, 1996.
- [16] D. Contois, "Kinetics of bacterial growth relationship between population density and specific growth rate for continuous cultures", *J. Genetic Microbial*, **21**, pp. 40-50, 1959.
- [17] Eikens, B., and Karim.N., "Process identification with multiple neural network models," *International Journal of Control*, **72**(7/8), 576-590, 1999.
- [18] Enari, T.-M., "Biotechnology and control engineering," *IFAC Conf. 'Modeling and control of biotechnical processes'*, eds. Halme, Helsinki, Finland, pp.1-3, 1982.
- [19] Gauthier, J.P., Hammouri, H., and Othman S., "A simple observer for Nonlinear systems," *IEEE*, 1992
- [20] Gustafsson, T.K. y Waller, K.V., "Dynamic Modeling and reaction invariant control of pH," *Chemical Engineering Science*, **38**, pp. 389-398, 1983.

- [21] Ioannou, P.A. and Sun, J., *Robust Adaptive Control*, Prentice Hall Inc. Upper Saddle River: NJ, 1996.
- [22] Hall, R.C. and Seborg D.E., "Modeling and selftuning control of a multivariable pH neutralization process. Part I: Modeling and multiloop control," *Proc. of the ACC*, Pittsburgh, **2**, pp.1822-1827, 1989.
- [23] Hathaway, R.J., and Bezdek, J.C., "Switching regression models and fuzzy clustering," *IEEE Trans. on Fuzzy Systems*, **1**, pp.195-203, 1993.
- [24] Haykin, S., *Neural networks a comprehensive foundation*, Macmillan College Publ. Co., New York, 1994.
- [25] Hunt, K.J., Sbarbaro, D., Zbikowski, R. and Gawthrop, P.J., "Neural Networks for Control Systems - A Survey," *Automatica*, **28**(6), pp.1083-1112, 1992
- [26] Jagannathan, S. and Lewis, F.L., "Identification of Nonlinear Dynamical Systems Using Multilayered Neural Networks," *Automática*, **32**(12), pp. 1707-1712, 1996.
- [27] Johnson, A., "The control of feed-batch fermentation processes - A survey," *Automatica*, **23**(6), pp. 691-705, 1987.
- [28] Johansen, T.A., and Foss, B.A., "Multiple model approaches to modelling and control," *International Journal of Control*, **72**(7/8), pp. 575, 1999.
- [29] Johansen, T.A., Operating regime based process modeling and identification. *PhD thesis*, Norwegian Institute of Thecnology, University of Trondheim, 1994.
- [30] Johansen, T.A., "Identification of Non-Linear Systems using Empirical Data and Prior Knowledge- An Optimization Approach," *Automatica*, **32**(3), pp. 337-356, 1996.
- [31] Kohonen, *Self Organizing Maps*, Heidelberg: Springer, 1995.
- [32] Kosmatopoulous, E.B., Polycarpou, M.M. and Ioannpu, P.A., "High order neural networks structures for identification of dynamical systems," *IEEE Trans. on Neural Networks*, **6**(2), pp.442-451, 1995.

- [33] Kosmatopoulous, E.B., Christodoulou, M.A., and Ioannpu, P.A., "Dynamic Neural Networks that ensure exponential identification error convergence," *Neural Networks*, 10(2), pp.299-314, 1997.
- [34] Lainiotis, D.G., "Partitioning: A unifying framework for Adaptive Systems, I. Estimation, II. Control," *Proceedings of the IEEE*, **64**, pp.1126-1143, 1182-1197, August 1976.
- [35] Levin, A.U., and Narendra, K.S., "Control of Nonlinear Dynamical Systems Using Neural Networks: Controllability and Stabilization," *IEEE Tran. on Neural Networks*, 4(2), pp. 192-206, March 1996.
- [36] Levin, A.U., and Narendra, K.S., "Control of Nonlinear Dynamical Systems Using Neural Networks, Part.II: Observability, identification and control," *IEEE Tran. on Neural Networks*, 1995.
- [37] Lewis, F.L., Yesildirek A. and Liu K., "Multilayer Neural-Net Robot Controller with Guaranteed Tracking Performance," *IEEE Trans. on Neural Networks*, **7**(2), 388-399, 1996.
- [38] Lisboa, P.J., "Newral Netwoks for control: evolution, revolution or renaissance," Page,G.F., Gomm, J.B., and Williams D. (Eds), *Application of Neural Networks to Modeling and Control*, Champan and Hall, 1993.
- [39] Maartensson, B., Adaptive Stabilization, *Ph. D. thesis*, Lund Institute of Technology, Sweden, 1986.
- [40] Middleton, R.H., Goodwing, G.C., Hill, D.J., and Mayne D.Q., "Desing Issues in Adaptive Control," *IEEE Trans. on Automatic Control*, AC-33(1): pp.50-58, January, 1988.
- [41] Miller III, W.T., Sutton, R.S. and Werbos, P.J.,"Neural Networks for Control," *MIT Press, Cambridge Mass.*, 1990.
- [42] Montague, G.A., Hofland, A.G., Lant, P.A., Di Massimo, C., Sauders, A., Tham, M.T. and Morris, A.J., "Model Based Estimation and Control: Adaptive Filtering, Nonlinear

- Observers and Neural Networks," *Proc. 3rd. Symp. 'Control for Profit'*, Newcastle-upon-Tyne, 1989.
- [43] Morari, M. and Zafiriou, E., *Robust Process Control*, Prentice Hall, Englewood Cliffs, N.Y., 1989.
- [44] Moore, R.L., *InTech* **27**(1):59, 1980.
- [45] Morse, A.S., Mayne D.Q. and Goodwing, G.C., "Applications of Hysteresis Switching in Parameter Adaptive Control," *IEEE Trans. on Automatic Control*, AC-37(9): pp.1343-1354, September, 1992.
- [46] Najim K. and Poznyak A.S., *Learning Automata Theory and Applications*, Pergamon, 1994.
- [47] Narendra, K.S. and Annaswamy, A.M., "*Stable Adaptive Systems*," Englewood Cliffs, N.J., Prentice Hall, 1989.
- [48] Narendra, K.S., Balakrishnan, J., Ciliz, M.K., "Adaptation and Learning Using Multiple Models, Switching, and Tuning," *IEEE Control Systems*, pp. 37-51, June 1995.
- [49] Narendra, K. S. and Balkrishnan, "Performance Improvement in Adaptive Control Systems Using Multiple Models and Switching," *Proc. Seventh Yale Workshop on Adaptive and Learning Systems*, Center for Systems Science, Yale University, New Haven, CT, pp. 27-33, May 1992.
- [50] Narendra, K. S. and Balkrishnan, "Improving Transient Response of Adaptive Control Systems Using Multiple Models and Switching," *IEEE Trans. on Automatic Control*, AC-39(9): pp.1861-1866, September, 1994.
- [51] Narendra, K. S. and Balkrishnan, "Intelligent Control Using Fixed and Adaptive Models," *Technical Report 9410*, Center for Systems Science, Yale University, New Haven, CT, October, 1994.
- [52] Narendra, K.S. and Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Tran. on Neural Networks*, **1**(1), pp. 4-26, March, 1990.

- [53] Narendra, K.S. and Mukhopadhyay, "Intelligent Control Using Neural Networks," *IEEE Control Systems Magazine*, **12**(2), 11-19, April, 1992.
- [54] Pineda, F.J. and W.Yu, "Chemical Process Identification with Multiple Dynamic Neural Networks," *accepted in the International NAISO Congress on Information Science Innovations*, Dubai, U.A.E., 2001.
- [55] Pineda, F.J. and W.Yu, "Chemical Process Modeling with Multiple Dynamic Neural Networks," *2001 Europe Control Conference*, Porto, Portugal, 2001, accepted in the Invited Session on Nonlinear Process Control.
- [56] Pineda, F.J. and W.Yu, "Identificación de Procesos Químicos con un enfoque multi-modelo y redes neuronales dinámicas," *Conferencia de Ingeniería Eléctrica "cie2000"*, CINVESTAV, México, 2000.
- [57] Pineda, F.J. and W.Yu, "Chemical Process Identification and Control with Multiple Dynamic Neural Networks," submitted to *Journal of Process Control*.
- [58] Pedrycz, W., "Fuzzy multimodels," *IEEE Trans. on Fuzzy Systems*, **4**, pp.139-148. 1996.
- [59] Poznyak, A.S., Yu W., Edgar N. Sanchez and Jose P. Perez, Stability Analysis of Dynamic Neural Control, *Expert System with Applications*, **14**(1), pp. 227-236, 1998.
- [60] Poznyak, A.S., Yu, W., Sanchez, E.N., "Nonlinear Adaptive Trajectory Tracking Using Dynamic Neural Networks," *IEEE Trans. on Neural Networks*, **10**(5), pp. 1402-1411, 1999.
- [61] Poznyak, A.S. and K. Najim, "Learning Automata and Stochastic Optimization," Springer Verlag, 1997.
- [62] Rovithakis, G.A. and Christodoulou, M.A., "Adaptive Control of Unknown Plants Using Dynamical Neural Networks," *IEEE Trans. on Syst., Man and Cybern.*, **24**, pp. 400-412, 1994.

- [63] San K. Y. and G. Stephanopoulos, "Optimization of fed-batch penicillin fermentation: a case of singular optimal control with state constraints," *Biotechnology and Bioengineering*, **34**, pp. 72-78, 1989.
- [64] Sanchez, E.N. and Perez, J.P., "Input to State Stability Analysis for Dynamic Neural Networks," to appear in *IEEE Trans. on Circuits and Systems-I*.
- [65] Selmic, R.R. and Lewis, F.L., "Neurocontrol for compensation of actuator nonlinearities," *10 Yale Workshop on Adaptive and Learning Systems*, pp.83-92, 1998.
- [66] Shinskey, F.G., *Process control systems, Application, Design, and Tunning*, 3rd Edition, McGraw-Hill Publishing Company, 1988.
- [67] Shinskey, F.G., *pH and pION Control in Process and Waste and Waste Streams*, John Wiley and Sons, 1973.
- [68] Shinskey, F.G., "Adaptive pH Controller Monitors Nonlinear Process," *Control Engineering*, **57**, February, 1974.
- [69] Sontag, E.D. and Wang, Y., "On Characterization of the Input to State Stability Property," *System and Control Letters*, **24**, pp. 351-359, 1995.
- [70] Sorheim, E., "A combined network architecture using ART2 and back propagation for adaptive estimation of dynamical processes," *Modeling, Identification and Control*, **11**, pp. 191-199, 1990.
- [71] Stokbro, K., Hertz, J.A., and Umberger, D.K., "Exploiting neurons with localized receptive fields to learn chaos," *Complex Systems*, **4**, pp. 603-611, 1990.
- [72] Sugeno, M., and Yasukawa, T., "A fuzzy logic based approach to qualitative modeling," *IEEE Trans. on Fuzzy Systems*, **1**, pp. 7-31, 1993.
- [73] Waller, K.V., Gustafsson, T.K., "Fundamental Properties of Continuous pH Control," *ISA Trans.* **22**(1), pp.25-34, 1983.

- [74] Willis, M.J., Massimo, C.D., Montague, G.A., Tham, M.T. and Morris, A.J., "Artificial Neural Networks in Process Engineering," *IEE Proc.-D*, **138**(3), May 1991, pp.256-266.
- [75] J.C.Willems, "Least squares optimal control and algebraic Riccati equations", *IEEE Trans. on Automatic Control*, **16**(6), pp 621-634, 1971.
- [76] Yamashita S., H. Hoshi and T. Inagaki, "Automatic control and optimization of fermentation processes," *Perlam D. (Ed.), Fermentation Advances*, Academic Press, New York, pp.441-463, 1969.
- [77] Weigend, A.S., Mangeas, M. and Srivastava, A.N., "Nonlinear gated expertes for time series: discovering regime and avoiding overfittings," *International Journal of Neural Systems*," **6**, pp. 373-399, 1995.
- [78] Poznyak, A.S., Yu, W., Sánchez, E.N., and Perez, J. P., "Nonlinear Adaptive Trajectory Tracking Using Dynamic Neural Networks," *IEEE Trans. Neural Networks*, **6**(10), 1402-1411, 1999
- [79] Wen Yu. and Poznyak, A S., Indirect Adaptive Control via Parallel Dynamic Neural Networks, *IEE Proceedings - Control Theory and Applications*, **146**(1), 25-30, 1999.
- [80] Wen Yu and Xiaoou Li, "Some Stability Properties of Dynamic Neural Networks," *IEEE Trans. Circuits and Systems*, Part I, accepted for publication.
- [81] Wen Yu and Xiaoou Li, Some New Results on System Identification with Dynamic Neural Networks, *IEEE Trans. Neural Networks*, accepted for publication
- [82] Werbos, P.J., "Neurocontrol and Supervised Learning: an overview and evaluation," In White, D.A., and Sofge, D.A., (Eds.), *Handbook of Intelligent Control*, Van Nostrand Reihhold, New Yor,. 1992.