

Control Inteligente

Control con Redes Neuronales L10

Luis Moreno, Santiago Garrido, Dorin Copaci

Dpto. Ing. de Sistemas y Automática
Universidad Carlos III
Madrid

Oct 2019



Table of contents

- 1 Control con Redes Neuronales

Introducción

Cuando hablamos de control podemos distinguir dos grandes tipos básicos:

- *Regulación de sistemas.*
Se busca mantener la salida del sistema en un cierto valor independientemente de las perturbaciones a las que este sometido (ej: control de temperatura de una habitación).
- *Servosistemas.*
Se busca que la salida siga lo mejor posible una cierta trayectoria de referencia (ej: control de trayectoria en un robot). Nos centraremos en estos.

Control de sistemas mediante redes neuronales

En general la estructura del control está condicionada por la forma de identificar la dinámica inversa de la planta, así podemos tener:

- Modelo de **control directo**:
si la información acerca del sistema a controlar se extrae por un método explícito de identificación.
- Modelo de **control indirecto**:
si la identificación forma parte de las acciones de control.

Esquemas de control directo

Son esquemas de control basados en modelo en el sentido de que es necesario un modelo del sistema para diseñar el controlador, entre estos esquemas tenemos:

- Control directo inverso (incluido el entrenamiento del modelo inverso).
- Control con modelo interno
- Linealización por realimentación.
- Prealimentación con modelo inverso.
- Control óptimo (no lo veremos)

Esquemas de control indirecto

Son esquemas de control que están siempre basados en un modelo del sistema. La idea consiste en usar una red neuronal para modelar el sistema que va a ser controlado. Este modelo es usado en el diseño de un controlador más convencional. Entre estos esquemas tenemos:

- Posicionamiento de polos.
- Mínima varianza.
- Control predictivo.
- Control predictivo no lineal.

Control directo

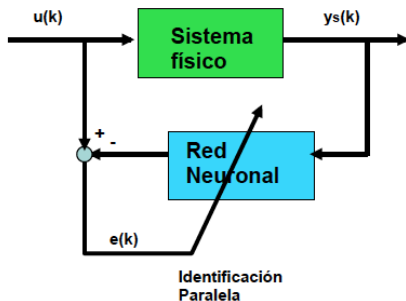
Control directo inverso

- Cuando hablamos de control directo nos referimos a que *el controlador es una red neuronal*.
- Idea básica:
 - Conectar la red en paralelo con la planta
 - Entrenarla con la dinámica inversa de esta, para luego utilizarla como controlador.



Control directo inverso

- Aprendizaje de la dinámica inversa del sistema



Control directo Inverso

- Esquema básico:
 - Se conecta la red en paralelo con la planta
 - Entrenarla con la dinámica inversa de esta, para luego utilizarla como controlador.
- Este modelo fue propuesto y usado por diferentes autores, entre ellos Albus(1975), Widrow(1986) y Miller(1987).
 - Durante el entrenamiento a la red se le muestra como entrada la salida de la planta y como salida deseada de la red la entrada a la planta, es decir la red aprende el comportamiento inverso.
 - Después del aprendizaje a la red se le introduce la salida que se desea para la planta y ella da como salida la acción de control adecuada.
 - En este modelo se supone que la planta es invariante en el tiempo, ya que una vez se realiza el aprendizaje el modelo queda fijado.

Control directo inverso

El problema principal de la identificación de la dinámica inversa ocurre cuando la inversa de dicha dinámica no es una función bien definida.

- La mayoría de los sistemas presentan problemas al no tener inversa.
- Otras veces el problema estriba en que para diferentes entradas la planta produce la misma salida.
- Estos problemas no son evitables, aunque hay soluciones prácticas que en ocasiones son suficientes.

Control directo inverso

Los problemas básicos de este modelo son en parte consecuencia de que es un control en bucle abierto, y por ello:

- El control depende de la exactitud con la que la red haya aprendido el modelo inverso de la planta.
- El modelo no contempla la presencia de perturbaciones externas, por lo que es muy sensible a perturbaciones.
- Inestabilidad.

Control directo inverso

- Este esquema de control con redes neuronales fué uno de los primeros, se entrenaba una red para que aprendiese la dinámica inversa del sistema y usarlo como controlador.
- Suponiendo que el sistema a controlar puede ser descrito por

$$y(k+1) = g[y(k), \dots, y(k-n+1), u(k), \dots, u(k-m)] \quad (1)$$

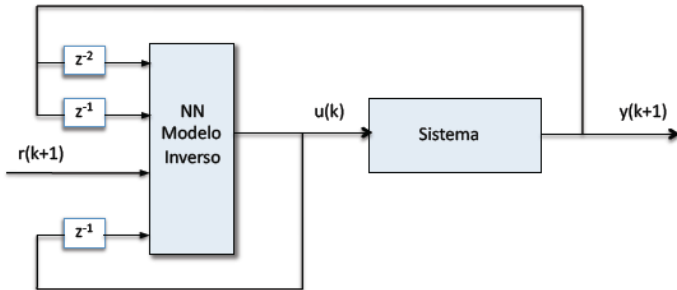
- La red que se busca debe de generar la entrada más reciente $u(k)$ que genere como salida $y(k+1)$ es decir

$$\hat{u}(k) = \hat{g}^{-1}[y(k+1), y(k), \dots, y(k-n+1), u(k), \dots, u(k-m)] \quad (2)$$

- Suponiendo que esta red pueda obtenerse de alguna forma, puede usarse para controlar el sistema si se sustituye $y(k+1)$ por la salida deseada en $k+1$, es decir la referencia $r(k+1)$

Control directo inverso

- Si la red ha aprendido de forma exacta la dinámica inversa, la señal $u(k)$ que genera conducirá al sistema a la salida deseada en $k + 1$, tal y como se muestra en el esquema.



Control directo inverso

Veremos dos métodos para obtener el modelo inverso:

- **Entrenamiento general.** Es un método off-line.
- **Entrenamiento especializado.** Es un método on-line.

Entrenamiento general.

- Se selecciona una red y se entrena **off line**.
- La diferencia con los casos vistos para identificación está en la elección de las entradas (los regresores) y salidas de la red.
- Las entradas y salidas son

$$\hat{u}(k) = NN_{inv}[y(k+1), y(k), \dots, y(k-n+1), u(k), \dots, u(k-m)] \quad (3)$$

- La red se entrena de forma que minimize el criterio

$$J(\theta, Z^N) = \frac{1}{2N} \sum_{i=1}^N [u(k) - \hat{u}(k|\theta)]^2 \quad (4)$$

Entrenamiento general

- Esta estrategia de entrenamiento general no es un método de diseño basado en modelo en el sentido de que *el controlador se infiere directamente a partir del conjunto de datos* sin que se necesite tener un modelo del sistema.
- Si se observa la función de transferencia global del controlador mas el sistema en lazo cerrado esta es $H(z) = z^{-1}$, es decir un retardo unidad.
- Se puede interpretar este tipo de control como una **linealización** del sistema dando lugar a un control de tipo **dead-beat** (tiempo mínimo) en el que la salida sigue a la referencia excepto por un retardo de un ciclo de muestreo.
- Si el sistema tiene un tiempo de retardo mayor que uno, el uso de modelos inversos para el control del sistema se hace algo más complicado.

Entrenamiento general

Inconvenientes

- Si el modelo inverso es inestable (zeros fuera del círculo unidad) el sistema en bucle cerrado suele inestabilizarse. Esto ocurre con frecuencia en la práctica.
- Otro problema aparece cuando existe más de una inverso, es decir cuando para dos entradas de control distintas $u_1(k) \neq u_2(k)$ se tiene que

$$\begin{aligned} g[y(k), \dots, y(k-n+1), u_1(k), \dots, u(k-m)] = \\ g[y(k), \dots, y(k-n+1), u_2(k), \dots, u(k-m)] \quad (5) \end{aligned}$$

Entrenamiento especializado

- El entrenamiento general usa como criterio a optimizar el tratar de minimizar la discrepancia entre la salida de la red y la secuencia de ordenes de control correctas, esto es muy difícil de obtener y puede dar lugar a grandes errores.
- Un criterio más deseable puede ser el tratar de minimizar el error entre la salida del sistema y la señal de referencia del modelo

$$J(\theta, Z^N) = \frac{1}{2N} \sum_{i=1}^N [r(k) - y(k)]^2 \quad (6)$$

- Desafortunadamente minimizar este criterio off-line es difícil ya que la salida del sistema $y(t)$ depende de la salida del modelo inverso $u(k - 1)$.

Entrenamiento especializado

- Una posibilidad es que la red pueda ser entrenada para minimizar en forma recursiva

$$J_k(\theta, Z^k) = J_{k-1}(\theta, Z^{k-1}) + [r(k) - y(k)]^2 \quad (7)$$

- Esto es un enfoque **on line** del problema y requiere un *controlador adaptativo*.
- Obtener un esquema de entrenamiento basado en este criterio no es inmediato, son necesarias algunas aproximaciones para hacerlo implementable.

Entrenamiento especializado

- Consideremos un método del gradiente recursivo, y supongamos que J_{k-1} ha sido minimizado, los pesos en el instante k son ajustados de acuerdo a

$$\hat{\theta}(k) = \hat{\theta}(k-1) - \mu \frac{de^2(k)}{d\theta} \quad (8)$$

donde $e(k) = r(k) - y(k)$, y

$$\frac{de^2(k)}{d\theta} = -\frac{dy(k)}{d\theta} e(k) \quad (9)$$

- y aplicando la regla de la cadena, el gradiente $\frac{dy(k)}{d\theta}$ puede calcularse como

$$\begin{aligned} \frac{dy(k)}{d\theta} &= \frac{\partial y(k)}{\partial u(k-1)} \frac{du(k-1)}{d\theta} \\ &= \frac{\partial y(k)}{\partial u(k-1)} \left\{ \frac{\partial u(k-1)}{\partial \theta} + \sum_{i=1}^n \frac{\partial u(k-1)}{\partial y(k-i)} \frac{dy(k-i)}{d\theta} \right. \\ &\quad \left. + \sum_{i=1}^m \frac{\partial u(k-1)}{\partial u(k-i)} \frac{du(k-i)}{d\theta} \right\} \end{aligned} \quad (10)$$

Entrenamiento especializado

- Se necesitan los Jacobianos del sistema $\frac{\partial y(k)}{\partial u(k-1)}$. Estos en general no son conocidos dado que el sistema no es conocido.
- Para resolver este problema, es necesario *identificar un modelo directo del sistema* (forward model) que nos permita obtener estimaciones de los Jacobianos

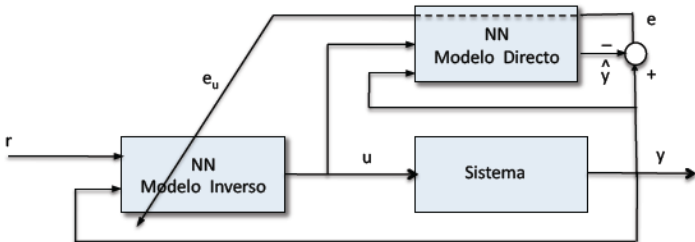
$$\frac{\partial y(k)}{\partial u(k-1)} \simeq \frac{\partial \hat{y}(k)}{\partial u(k-1)} \quad (11)$$

- El modelo inverso es una red recursiva que es ignorada durante el entrenamiento, en el sentido de que las dependencias de los regresores de los pesos de la red son ignoradas.

$$\frac{du(k-1)}{d\theta} = \frac{\partial u(k-1)}{\partial \theta} \quad (12)$$

Entrenamiento especializado

- Esquema



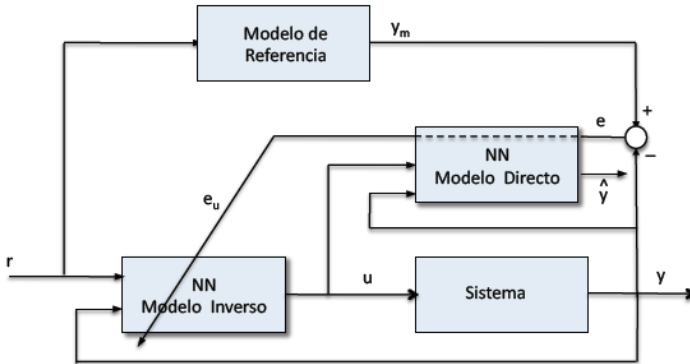
Entrenamiento especializado

- En este esquema las imprecisiones en el modelo no tienen un impacto grande en el entrenamiento.
- El Jacobiano es un factor escalar y en el algoritmo simplificado solo modifica el paso del aprendizaje, mientras que el Jacobiano tenga el signo correcto el algoritmo converge siempre que el paso sea suficientemente pequeño.
- Este esquema sigue siendo de tipo dead-beat por lo que genera una respuesta innecesariamente rápida a los cambios en las referencias, pudiendo dañar los actuadores.
- El esquema puede ser modificado para que la red neuronal de lugar en cadena cerrada a una respuesta de tipo filtro paso-bajo (primer orden o un segundo orden por ejemplo) es decir un MRAS (model reference adaptive system).

Entrenamiento especializado

MRAS

- Esquema



Entrenamiento especializado

MRAS

- Se busca un comportamiento de tipo

$$y_m(k) = \frac{B_m(z^{-1})}{A_m(z^{-1})} r(k) \quad (13)$$

donde el diseñador elige arbitrariamente los polinomios $B_m(z^{-1})$ y $A_m(z^{-1})$.

- En este caso se usa como señal de error $e(k) = y_m(k) - y(k)$ en vez de $e(k) = r(k) - y(k)$.
- El esquema de control adaptativo MRAS fué propuesto por Isermann en 1992 (sin redes neuronales).

Entrenamiento especializado

A veces se usa un sistema en dos etapas:

- Un entrenamiento general para hacer un primer ajuste del modelo inverso con lo que se obtiene una buena inicialización.
- Un entrenamiento especializado para hacer el ajuste fino del controlador.

Entre las ventajas podemos encontrar que

- No aparece el problema de sobreentrenamiento que se producía en el entrenamiento general ya que en muchos casos sólo se entrena al sistema con trayectorias de referencia con frecuencia muy parecidas.
- Al ser un esquema de entrenamiento on-line puede ser utilizado con *sistemas variantes en el tiempo* (variaciones lentas en el tiempo).

Control directo inverso

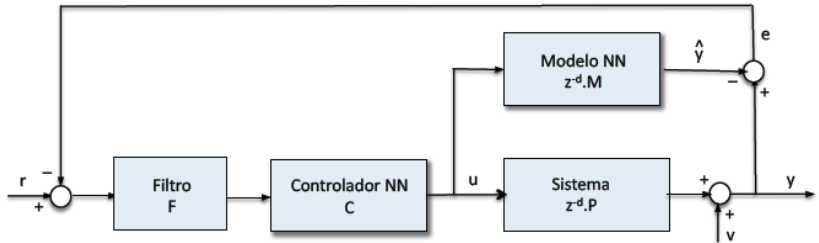
- Ventajas:
 - Es intuitivamente simple.
 - Es fácil de implementar.
 - Con el entrenamiento especializado el controlador puede ser optimizado para una trayectoria específica.
 - Es en principio posible aplicar un control especializado a sistemas variantes en el tiempo.
- Inconvenientes
 - No funciona en sistemas donde el inverso es inestable, lo que suele ocurrir a menudo cuando se usa muestreos a alta frecuencia.
 - Presenta problemas en sistemas que no son mapeos one-to-one (entrenamiento generalizado de los modelos inversos).
 - Presenta problemas con modelos inversos que no están bien amortiguados.
 - No hay opciones de ajuste.
 - En general son altamente sensibles a perturbaciones y ruido.

Modelo Interno de Control (IMC)

- Esta estrategia de control es muy próxima al control directo inverso, se ha usado en control de procesos químicos pero es utilizable en otros tipos de sistemas.
- El concepto de IMC se debe a Morari y Zafiriou (1989) y la idea de usar redes neuronales para IMC fué sugerida por Hunt en 1991 [Hunt(1992)].
- Este esquema de control requiere un modelo directo así como un modelo de la dinámica inversa del sistema a controlar, ver el esquema de la transparencia siguiente.

Modelo Interno de Control (IMC)

- Esquema



Modelo Interno de Control (IMC)

- A diferencia del control directo inverso, la realimentación no se compone directamente a partir de la salida del sistema sino que el error entre la salida del sistema y la salida del modelo es realimentado.
- Si suponemos que el modelo es perfecto y que no hay perturbaciones la señal de realimentación será cero.
- El controlador en este caso se comporta como una prealimentación pura a partir de la señal de referencia.

Modelo Interno de Control (IMC)

- Veamos cual es la función de transferencia del sistema en bucle cerrado en el caso de comportamiento lineal de los diferentes bloques F,C,P y M:

$$\begin{aligned}y(k) &= \frac{z^{-d}FCP}{1 + z^{-d}FC(P - M)}r(k) + \frac{1 - z^{-d}FCM}{1 + z^{-d}FC(P - M)}v(k) \\ &= v(k) + \frac{z^{-d}FCP}{1 + z^{-d}FC(P - M)}[r(k) - v(k)]\end{aligned}\quad (14)$$

y

$$u(k) = \frac{FC}{1 + z^{-d}FC(P - M)}[r(k) - v(k)]\quad (15)$$

Modelo Interno de Control (IMC)

- La condición para que el IMC sea estable es que el sistema a ser controlado y el modelo inverso sean ambos estables.
- Esto es un tanto restrictivo ya que el sistema debe de ser estable en bucle abierto.
- Bajo condiciones ideales $M = P$ y $C = P^{-1}$ las expresiones anteriores se reducen a

$$\begin{aligned}u(k) &= \frac{F}{P}[r(k) - v(k)] \\y(k) &= z^{-d}Fr(k) + [1 - z^{-d}F]v(k)\end{aligned}\tag{16}$$

- El filtro paso bajo F tiene impacto sobre la conducta del sistema en bucle cerrado, en la conducta de seguimiento del modelo y en la capacidad de rechazo de perturbaciones.

Modelo Interno de Control (IMC)

- Este filtro debe de cumplir dos condiciones:
 - Debe de ser estable y tener ganancia unitaria para asegurar el seguimiento de la referencia.
 - El numerador de $1 - z^{-d}F$ debe contener como factor el polinomio $\Phi(q^{-1})$ con la propiedad de $\Phi(q^{-1})d(t) = 0$
 - Para $F = 1$ se tiene un controlador dead-beat.
- Para el caso no lineal, el sistema y su inverso se modelan con redes neuronales ([Hunt(1992)]).

Modelo Interno de Control (IMC)

- Las ventajas e inconvenientes de este esquema de control son similares a las del control directo inverso.
- Características adicionales:
 - Requiere que el sistema sea estable en bucle abierto.
 - Debido a la estructura del controlador IMC es difícil asegurar que el modelo inverso este entrenado con un conjunto realista de datos.

Linealización por realimentación

- Si asumimos que el sistema puede ser escrito en la denominada forma canónica controlable no lineal

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_3(t) \\ \vdots \\ \tilde{f}[x(t)] + \tilde{g}[x(t)]u(t) \end{bmatrix} \quad (17)$$

donde:

- \tilde{f} y \tilde{g} son funciones no lineales del estado.
- t es el tiempo actual (no un múltiplo del periodo de muestreo)

Linealización por realimentación

- El sistema anterior puede ser linealizado mediante la introducción de una nueva variable control en la entrada del sistema $v(t)$

$$w(t) = \tilde{f}[x(t)] + \tilde{g}[x(t)]u(t) \quad (18)$$

tal que se genera la siguiente redefinición de la señal de control (asumiendo que $\tilde{g}[x(t)] \neq 0$)

$$u(t) = \frac{w(t) - \tilde{f}[x(t)]}{\tilde{g}[x(t)]} \quad (19)$$

tal que el sistema queda en una forma lineal

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_3(t) \\ \vdots \\ w(t) \end{bmatrix} \quad (20)$$

Linealización por realimentación

- Si asumimos que el sistema no lineal puede ser modelado en entrada-salida de la forma

$$y(t) = f[y(t-1), \dots, y(t-n), u(t-2), \dots, u(t-n)] \\ + g[y(t-1), \dots, y(t-n), u(t-2), \dots, u(t-n)]u(t-1)$$

o de forma equivalente

$$\dot{x}(t+1) = \begin{bmatrix} \dot{x}_1(t+1) \\ \dot{x}_2(t+1) \\ \vdots \\ \dot{x}_n(t+1) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ x_3(t) \\ \vdots \\ \tilde{f}[x(t)] + \tilde{g}[x(t)]u(t) \end{bmatrix}$$
$$y(t) = x_n(t)$$

con un vector de estado definido en la forma

$$x(t) = [y(t-n+1), \dots, y(t-1), y(t)]^T \quad (21)$$

Linealización por realimentación

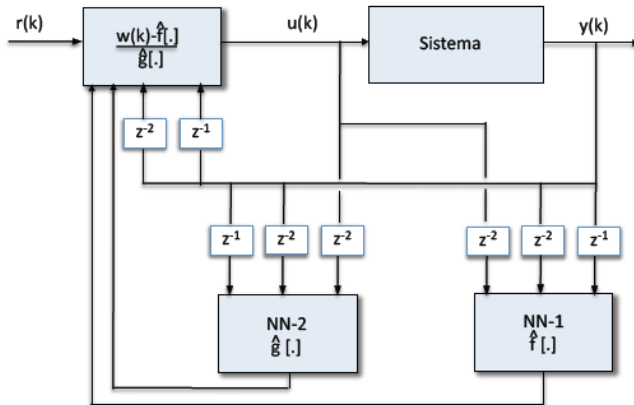
- Si asumimos que f y g son funciones conocidas, la introducción de la siguiente redefinición de la señal de control linealiza el sistema en los instantes de muestreo:

$$u(t) = \frac{w(t) - f[y(t-1), \dots, y(t-n+1), u(t-1), \dots, u(t-m+1)]}{g[y(t-1), \dots, y(t-n+1), u(t-1), \dots, u(t-m+1)]}$$

- En el caso de que el sistema no sea conocido es posible modelarlo mediante la introducción de dos redes neuronales que aproximen las funciones f y g tal y como se muestra en el esquema siguiente.

Linealización por realimentación

- Esquema



Linealización por realimentación

- Ventajas:
 - Es fácil de implementar.
 - Sólo se necesita un modelo del sistema a ser controlado.
 - El ajuste de la respuesta del sistema en bucle cerrado puede hacerse sin reentrenar el modelo.
 - Una vez linealizado puede incluirse un control por realimentación de estado que sitúe los polos en cadena cerrada en las posiciones deseadas.
- Inconvenientes
 - Está limitado a una cierta clase de sistemas. Es difícil de resolver si un sistema desconocido no pertenece a esta clase de sistemas.
 - La selección de la estructura del modelo es complicada debido a que deben elegirse dos arquitecturas de red.
 - No hay parámetros de diseño para ajustar el controlador.
 - Presenta problemas cuando la inversa (localmente) es inestable o esta al borde de la estabilidad.

Feedforward Control

- Los controles con realimentación se utilizan principalmente para estabilizar sistemas inestables y/o reducir el efecto de perturbaciones o inexactitudes en el modelo.
- Los controles de tipo feedforward o pre-alimentaciones suelen usarse para mejorar el seguimiento de las señales de referencia.
- Pueden usarse conjuntamente ambos esquemas, esto es muy usual en robótica por ejemplo (la técnica denominada computed-torque o par-calculado es una prealimentación).

Feedforward Control

- La mayor parte de los sistemas se regulan con controladores de tipo PID (P, PI, PD y PID), esto se hace incluso con sistemas no lineales y a pesar de que la simplicidad de su concepto limita las prestaciones.
- La razón principal para esto es la facilidad con la que se ajusta el regulador y la existencia de sistemas de auto-ajuste que lo hacen de forma automática sin más que pulsar un botón.
- Su uso continuado durante años y la confianza que inspiran a los diseñadores hace que sea difícil convencerlos de usar otras técnicas como las redes neuronales.
- Los controladores Feedforward ofrecen una solución de compromiso ya que permiten mejorar un control por realimentación preexistente que puede ser un PID clásico.

Feedforward Control

Dinámico

- Hay diversas formas de implementar un control feedforward, el más intuitivo es usar el denominado *control feedforward dinámico*.
- Esta estrategia de control feedforward usa un modelo inverso de NN entrenado mediante un entrenamiento general o especializado tal y como se vió.

$$u(t) = \hat{g}[y(t+d), \dots, y(t+d-n), u(t-1), \dots, u(t-m+1)] \quad (22)$$

- El componente feedforward que se le añade a la entrada de control al sistema es obtenido sustituyendo en la expresión las salidas del sistema por los valores de referencia, es decir

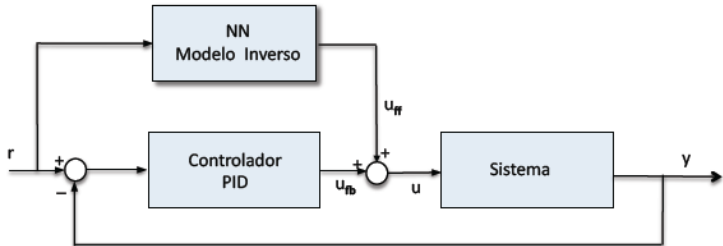
$$u_{ff}(t) = \hat{g}[r(t+d), \dots, r(t+d-n), u_{ff}(t-1), \dots, u_{ff}(t-m+1)] \quad (23)$$

este esquema de prealimentación aditiva se muestra en la siguiente figura:

Feedforward Control

Dinámico

- Esquema



Feedforward Control

Dinámico

- Si el modelo inverso es estable, la introducción del control feedforward no cambia las propiedades de estabilidad del sistema en bucle cerrado.
- En el caso de que se conozca la trayectoria de referencia completa la implementación de esta esquema de control se hace particularmente simple y es posible calcular las contribuciones del controlador feedforward de antemano y almacenar toda la secuencia de entradas de control $\{u_{ff}\}$.

Feedforward Control

Estático

- Una alternativa útil al control feedforward dinámico consiste en usar un feedforward estático.
- Este concepto elimina completamente los problemas potenciales de estabilidad que existen con los modelos dinámicos.
- La idea consiste en usar un *modelo NNFIR para describir la dinámica inversa del sistema*.

$$u_{ff}(t) = \hat{g}[r(t+d), \dots, r(t+d-n_{ff})] \quad (24)$$

- Aunque los modelos NNFIR no permiten modelar con precisión un sistema dinámico resultan muy atractivos ya que no presentan realimentación.
- Los errores que pueda presentar el modelo usado en el control feedforward resultan compensados por el control realimentado.

Feedforward Control

- Ventajas:
 - Es fácil de implementar.
 - Puede introducirse gradualmente.
 - Mejora el seguimiento de la referencia sin incrementar la sensibilidad al ruido.
 - Si existe un control realimentado que estabilice el sistema los experimentos se realizan fácilmente
- Inconvenientes
 - Requiere la existencia de un controlador realimentado.
 - No reduce el efecto de las perturbaciones que actúan sobre el sistema.
 - Presenta problemas si el inverso es inestable o poco amortiguado.
 - Un mal control feedforward puede dañar las prestaciones globales en vez de mejorarlas.

Control indirecto

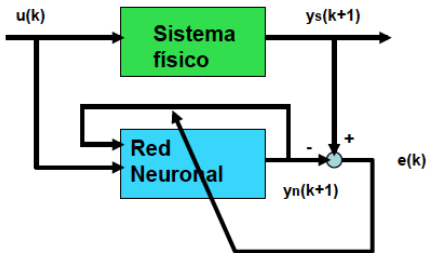
Control Indirecto

En este modelo, la red se entrena para realizar la función de controlador, para ello se usan dos redes:

- A una de las redes se la entrena para que aprenda la dinámica directa de la planta, para lo que a la entrada de la red se le dan las entradas a la planta y a la salida de la red se sitúa la salida de la planta.
- La segunda red se entrena con el algoritmo de control correspondiente para que controle el proceso.

Control Indirecto

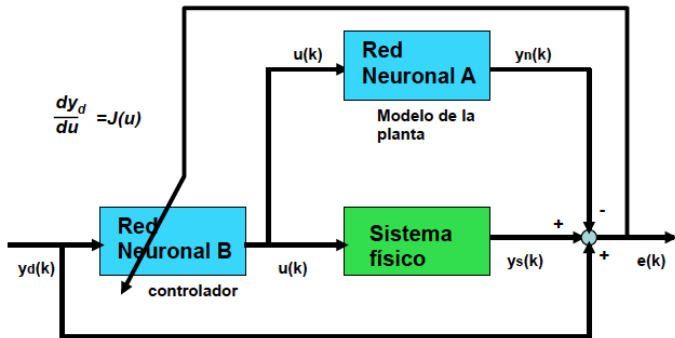
- Combina dos redes (fué propuesto por Jordan 1988)
 - Una se entrena con la dinámica directa de la planta
 - La segunda red se entrena para aprender la dinámica inversa



Aprendizaje de la dinámica directa

Control Indirecto

- Con el modelo directo de planta se puede calcular la derivada del error con respecto a las entradas de forma eficiente, mediante backpropagation lo que nos permite evaluar el Jacobiano de la planta.



Control Indirecto



- La utilidad de este Jacobiano estriba en que a pesar de que la inversa de la planta no este bien definida, podemos back-propagar el error a la segunda red neuronal para que esta simule la dinámica inversa de la planta.
- Con las salidas deseadas y este modelo inverso (controlador) calculamos la señal de control $u(k)$.
- El error resultante entre planta y salida deseada se retro-propaga hacia atrás en las capas del modelo directo hasta llegar a la entrada de la planta.
- Este error en la entrada se usa para el entrenamiento o ajuste del modelo inverso.

Esquemas de control indirecto

Son esquemas de control que están siempre basados en un modelo del sistema. La idea consiste en usar una red neuronal para modelar el sistema que va a ser controlado. Este modelo es usado en el diseño de un controlador más convencional. Entre estos esquemas tenemos:

- Posicionamiento de polos.
- Mínima varianza.
- Control predictivo.
- Control predictivo no lineal.

Bibliografía

-  Hunt, K. J., Sbarbaro, D., Zbikowski, R., & Gawthrop, P. J. (1992). Neural networks for control systems a survey. *Automatica*, 28(6), 1083-1112.
-  Sánchez Camperos, E. N., Alanís García, A. Y. (2006). *Redes Neuronales: Conceptos fundamentales y aplicaciones a control automático*. Edgar Nelson Sánchez Camperos y Alma Yolanda Alanís García.

- Fin L10