



# Universidad Nacional Autónoma de México

## Facultad de Ingeniería

# Control Inteligente

Módulo de Control 2023-1

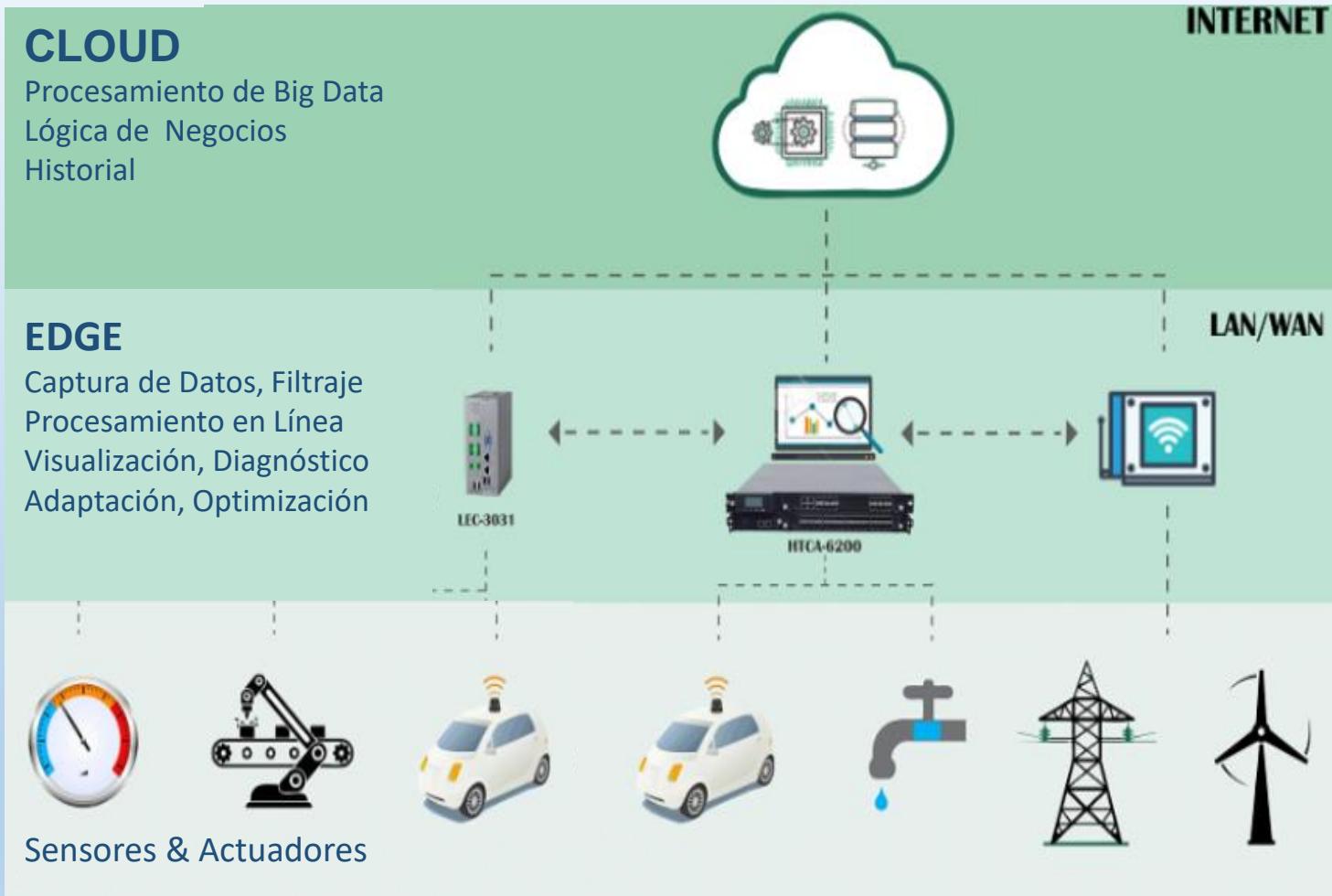
C. Verde  
Instituto de Ingeniería

[verde@unam.mx](mailto:verde@unam.mx)

[CVerdeR@iingen.unam.mx](mailto:CVerdeR@iingen.unam.mx)



# Potenciar la Tecnología





# Control Inteligente

- Octavo semestre,
- 3 créditos,
- Optativa,
- Horas/semanas 3,
- Horas/semestre 48,
- Modalidad: Curso teórico-práctico
- Horario: Marte y Jueves de 9:00 a 10:30

## Temario

1. Introducción al control inteligente y PID experto
2. Fundamentos de matemáticas borrosa
  - 2.1 Control con lógica borrosa
  - 2.2 Control y modelación Mamdani
  - 2.3 Control y modelación Takagi\_Sugeno
  - 2.4 Control borroso adaptable
3. Modelos de Redes Neuronales ANN
  - 3.1 Control con ANN adaptables
  - 3.2 Control con ANN y modos deslizantes RBF
4. Control con aprendizaje iterativo
  - 4.1 Algoritmo con una búsqueda inteligente
  - 4.2 Control neuronal discreto RBF



# Bibliografía y Referencias

- Jinkun Liu, 2018, Intelligent Control Design and MATLAB Simulation, <https://doi.org/10.1007/978-981-10-5263-7>, Springer Singapore.
- P. Alberto et al. Learning Control of Complex Systems. K. Astrom et al Chapter 6. Control of Complex Systems.
- Pedro Ponce-Cruz & Fernando D. Ramírez-Figueroa Intelligent Control Systems with LabVIEW, ISBN 978-1-84882-683-0 e-ISBN 978-1-84882-684-7 DOI 10.1007/978-1-84882-684-7.
- Dev, Prabhat et. al doi = {10.1016/j.matpr.2021.02.281}, Machine learning and its impact on control systems: A review, Materials Today: Proceedings. Es un buen panorama de control con el enfoque de aprendizaje.

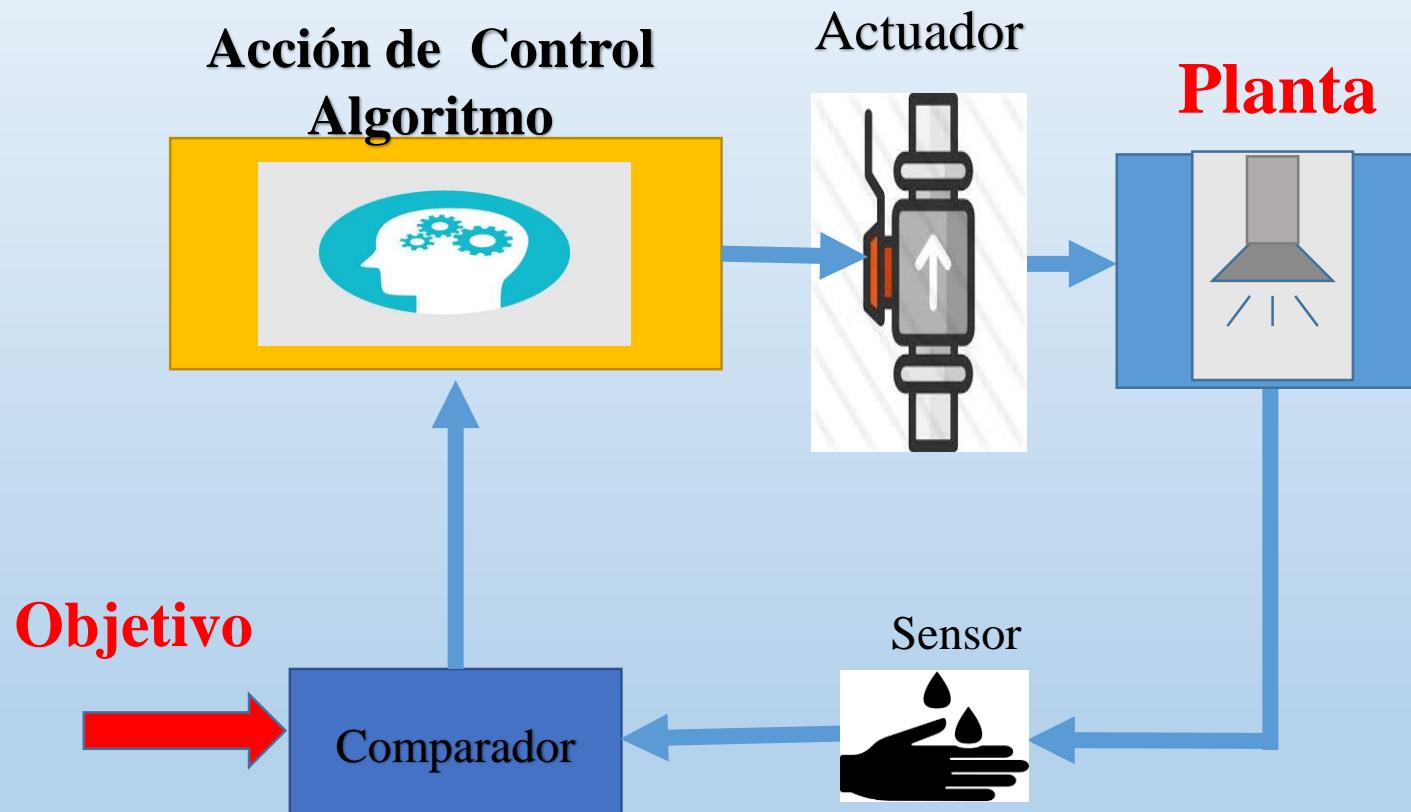


# Índice

- Introducción; S1, Lam 1-12.
- Control digital y PID; S2, Lam 13-31. TAREA 1 experto
- Fundamentos de Lógica borrosa; S3, Lam 32-51
- Inferencia y Ec. Borrosa If...Then; S4, Lam 52-58
- Control borrosa; S5, Lam 59-63 TAREA 2 Borroso,tanque
- Sintonización PI borroso; S6, Lam 64-67
- Modelo Takagi-Sugeno Retro de edo; S7, Lam 68-76 TAREA 3 control del tanque
- Control nolineal T-S; S8 77-80
- Redes neuronales artificiales; S9, Lam 81-90
- Esquema de aprendizaje e identificación; S10, Lam 91-95 TAREA 4 Generador de función con BPN.
- NN con RBF; S11, Lam 96-100
- Entrenamiento (W) gradiente RBF; S12, Lam 100-102
- Aproximación  $f(x)$  con RBF e implementación de control PI; S13, Lam 103-112, TAREA 5 CONTROL PI con gradiente
- Control con  $f(x)$  y perturbación desconocidas PI; S14, Lam 113-116,
- Sliding Mode Control+Versión Discreta, S15-16 , Lam 117-136 Tarea 6 SMC y RBF



# Sistema de Control de la Regadera

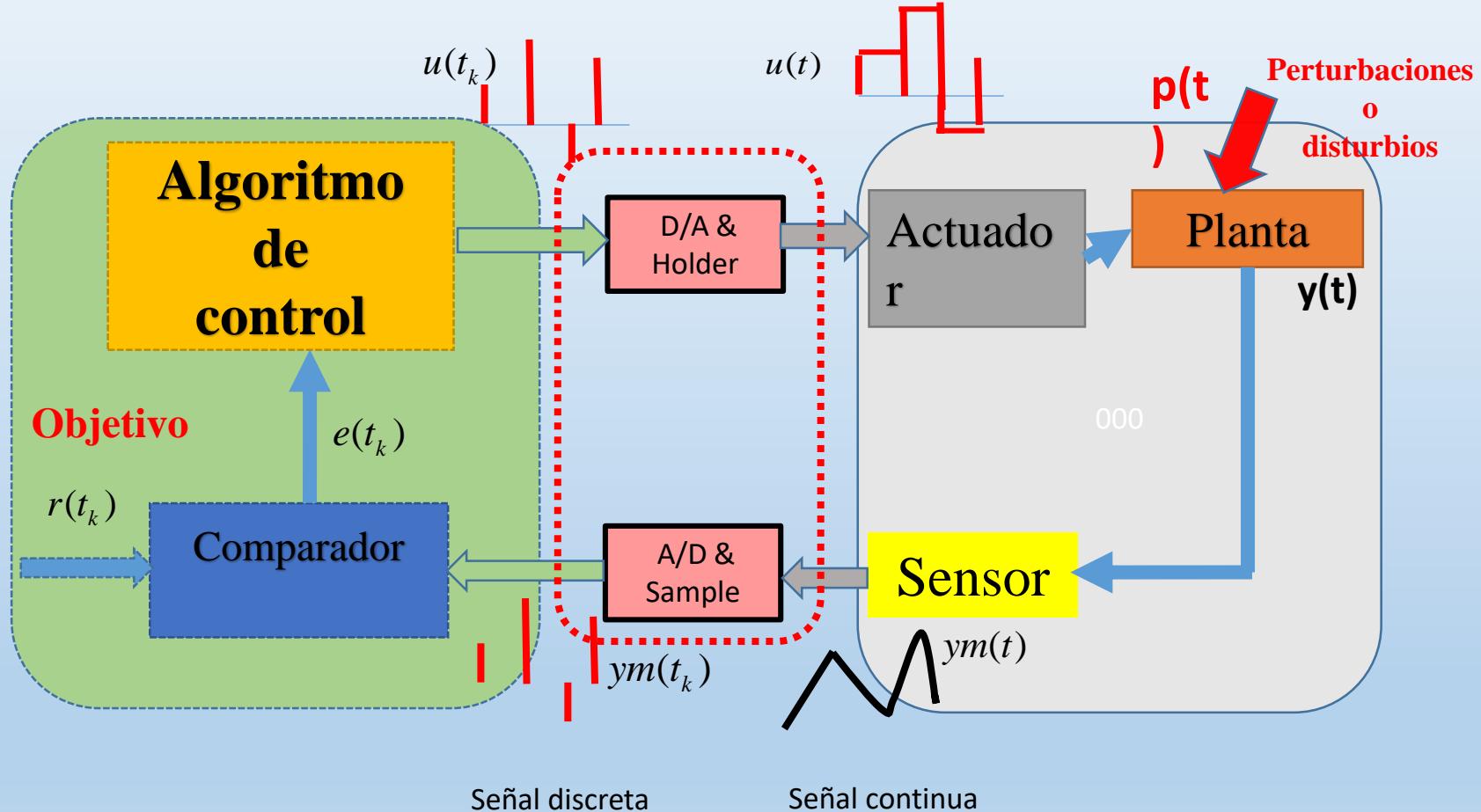




# Nomenclatura y Acrónimos

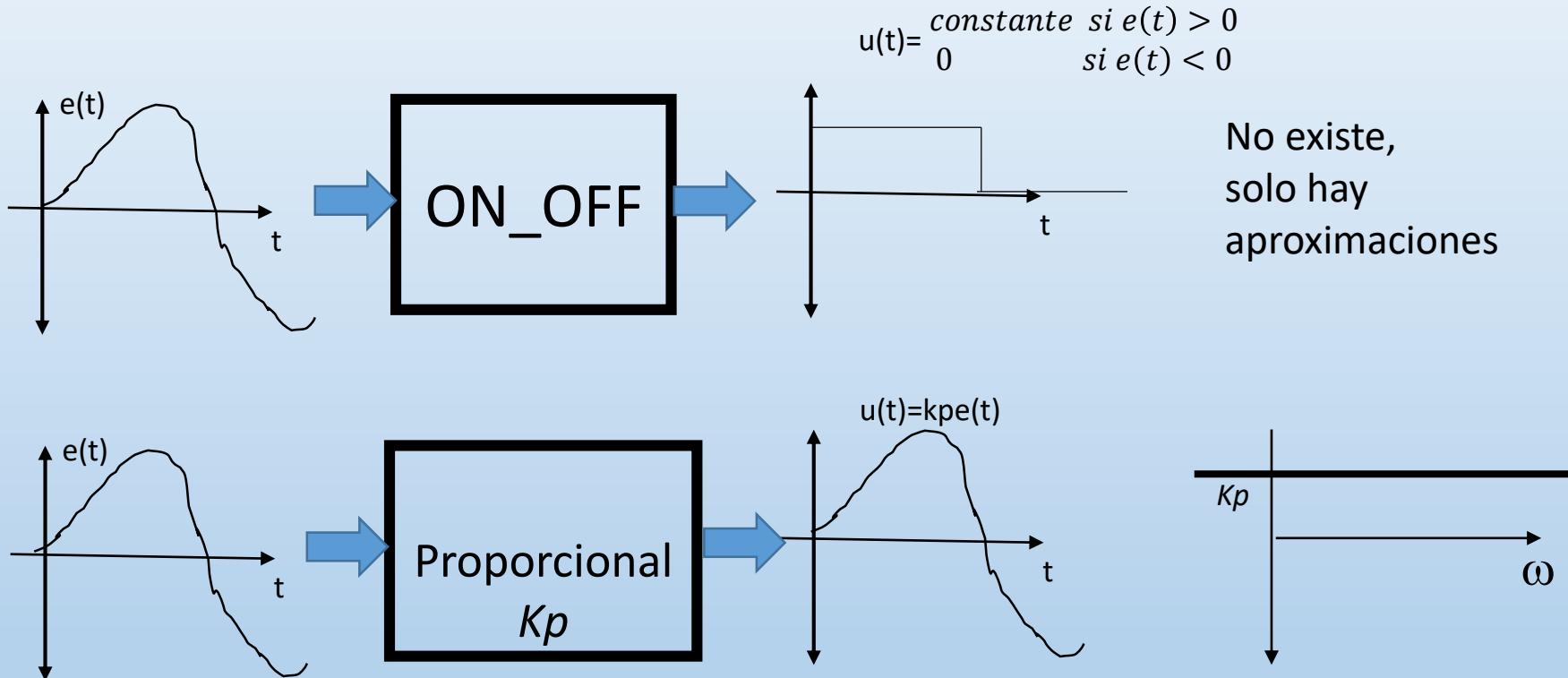
- ANN artificial Neural Networks
- PID Proportional, Integral & Derivative controller
- FL Fuzzy Logic
- TS Takashi-Sugeno model
- LMI Linear Matrix Inequality
- RBK Radial Back Propagation
- SISO single input single output System
- MIMO multiple inputs multiple outputs
- MM Mandami model
- ML Machine Learning
- KBS Knowledge based System
- Fs Fuzzy set
- PLC programmable logic controller
- DECS direct expert control System
- SECS Supervisor expert control System
- LTI linear time invariant System
- FM fuzzification model
- NLS nlineae static System
- SOC self-organizing controller
- SMC sliding mode control
- SRT self-tuning controller
- MRAC model reference adaptive controller
- MBC model based controller

# Esquema de Control Digital: Dos tipos de señales en el lazo



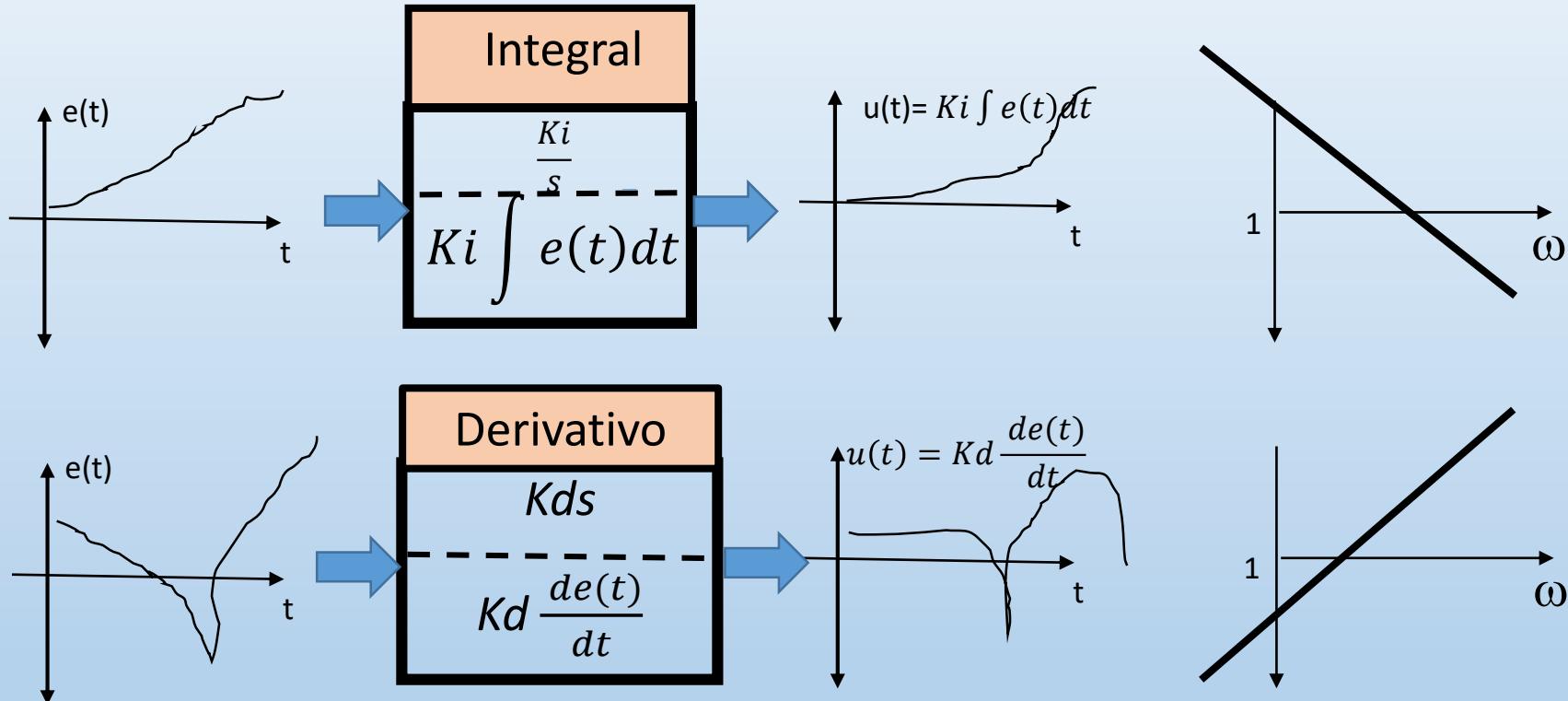


# Acciones Básicas de Control (1)





# Acciones Básicas de Control (2)



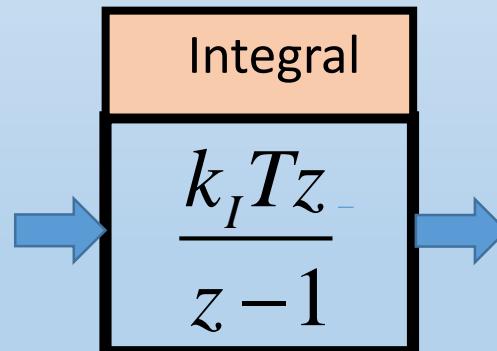


# Acciones Básicas del PID Discreto

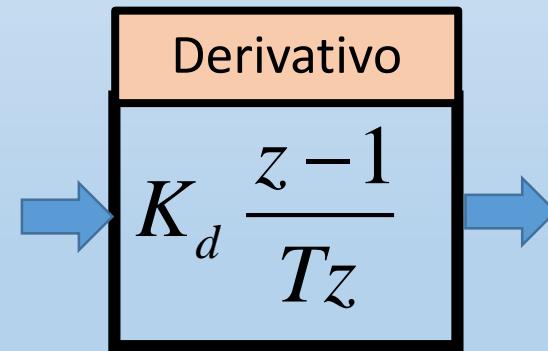


$$u(k) = \begin{cases} u_{\min} & \text{si } e(k) < 0 \\ u_{\max} & \text{si } e(k) > 0 \end{cases}$$

$$u(k) = k_p e(k)$$



$$u(k) = k_I \sum_0^k e(i)$$



$$u(k) = k_D (e(k) - 2e(k-1) + e(k-2))$$

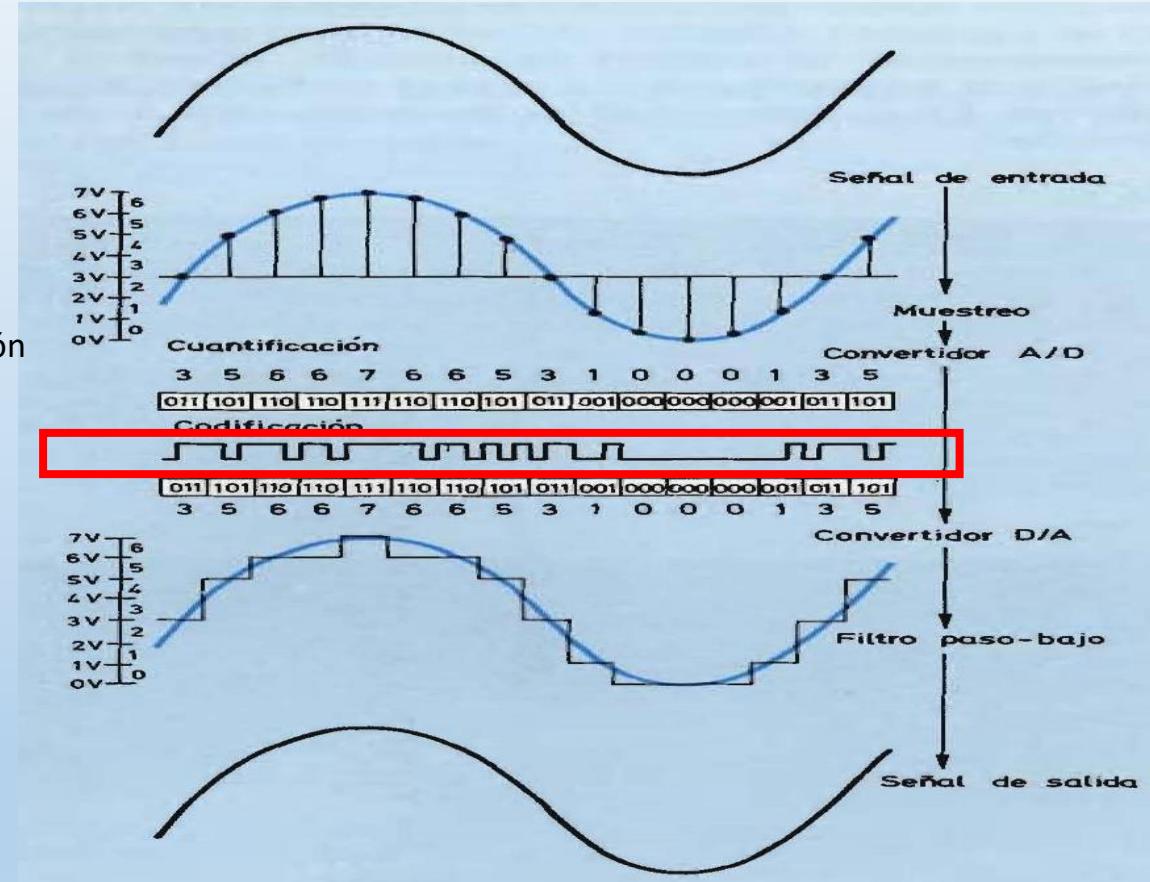
# Control Digital: Muestreo y Señales Digitales

Cuantificación  
& Discretización

Codificada

Retenida

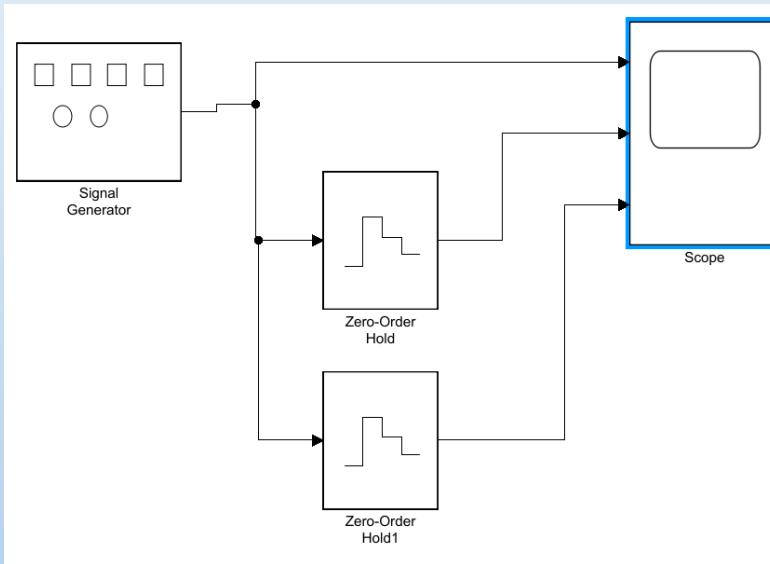
Filtrado



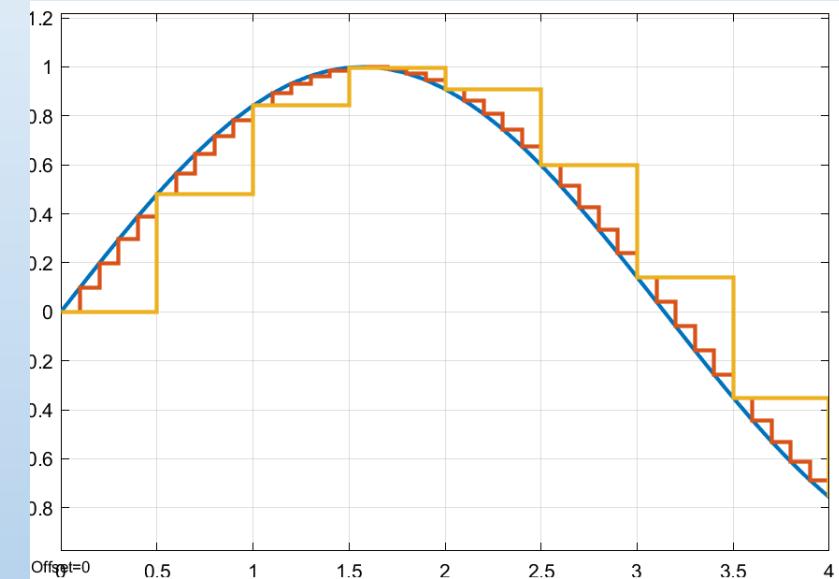


# Señal Analógica & Digital

## Proceso de A/D & D/A



- Señal Analógica y su Discretización



Muestreo de 0.5 unidades de tiempo  
Muestreo de 0.1 unidades de tiempo



# Sistema de Control con Adaptación (aprendizaje)

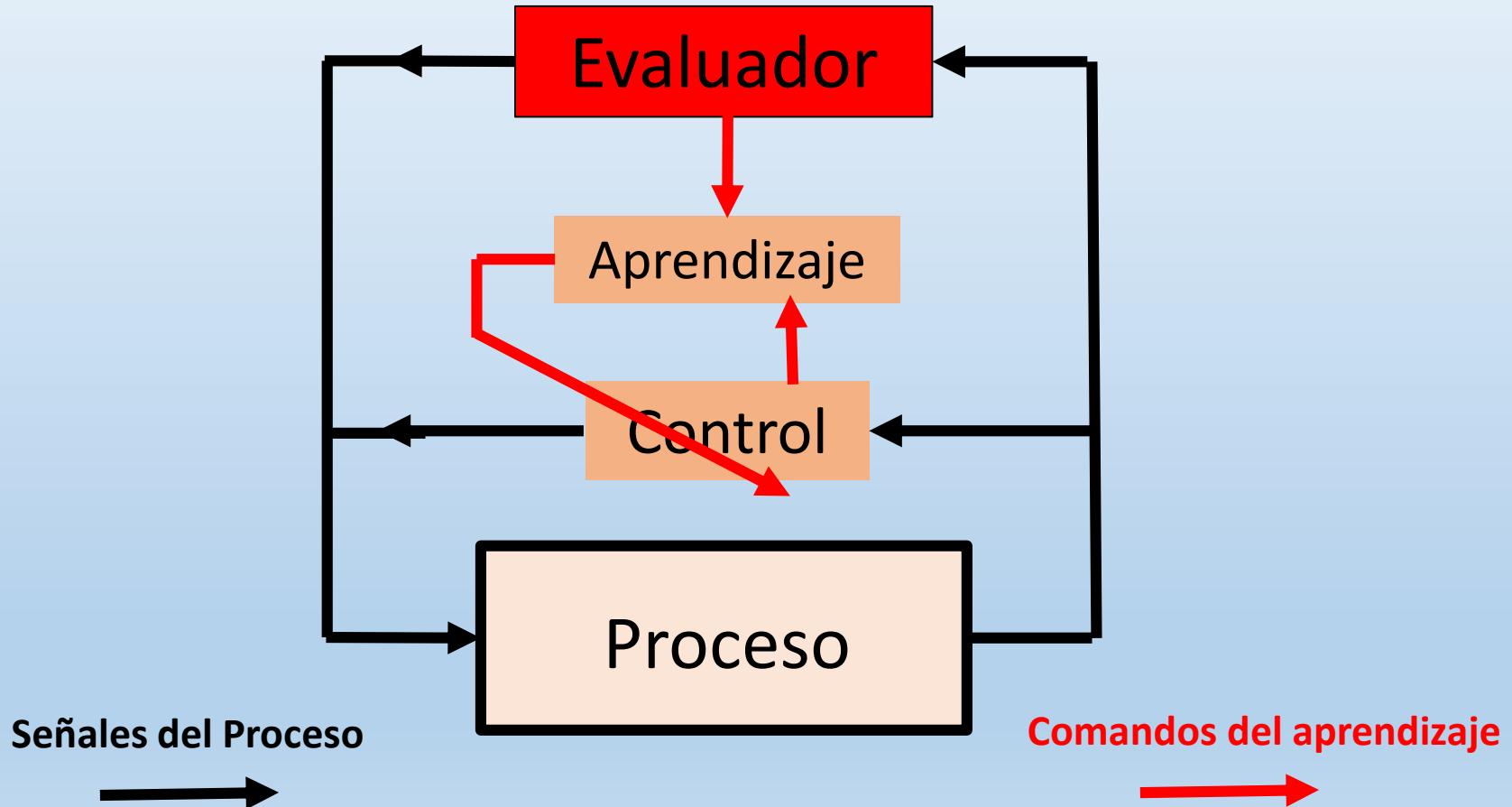


$$T_s = F_s(X_{sa}, U_{sa}, Y_{sa}, \lambda, X_{so}, p_s)$$
$$T_r = F_r(X_{ra}, U_{ra}, Y_{ra}, \lambda, X_{ro}, p_r)$$

$$X_a = [X, \Delta p]$$



# Sistema de Control con Aprendizaje Learning



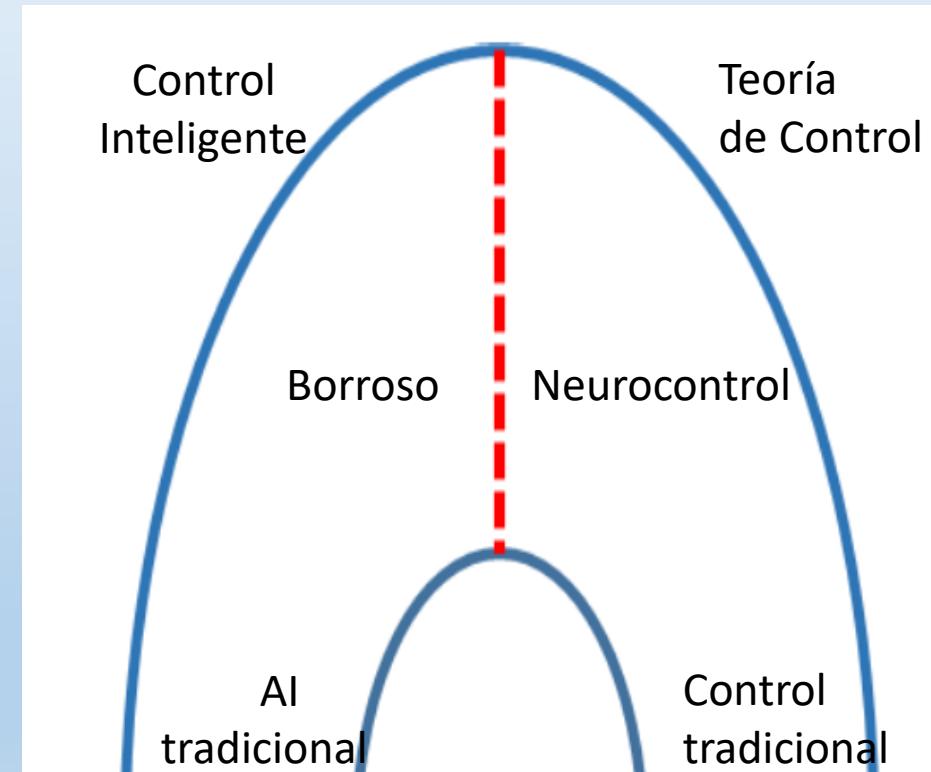
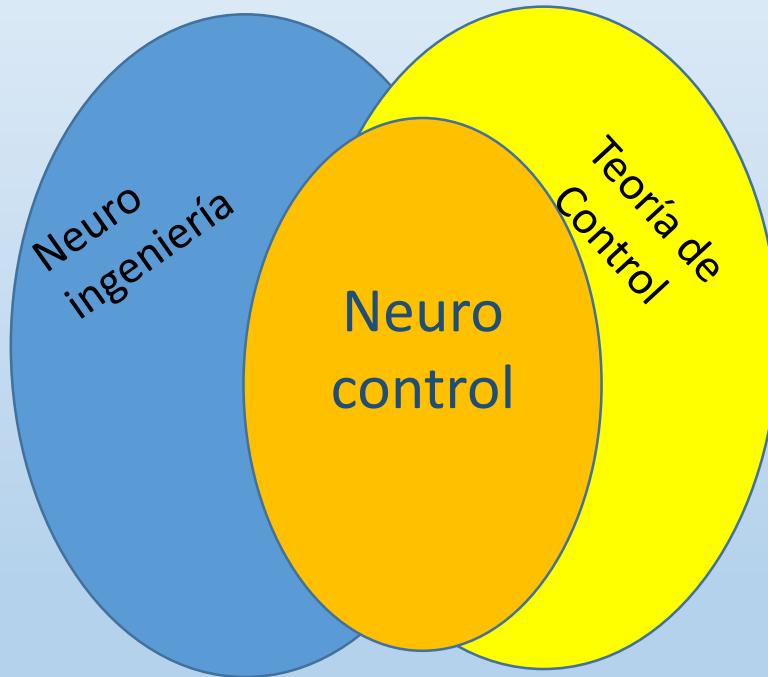


# Objetivo General

**Comprender el comportamiento de los sistemas de control avanzados diseñados con modelos de aprendizaje automático e inteligencia artificial.**

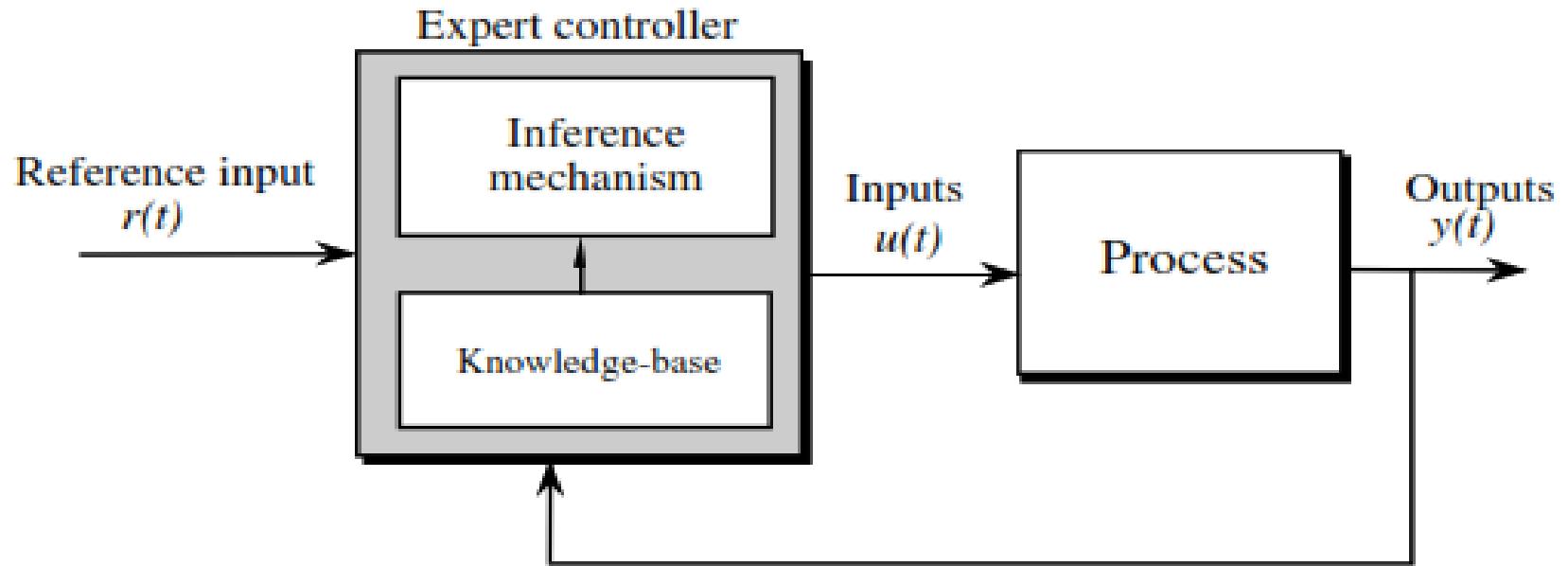


# Aspectos de Control Inteligente



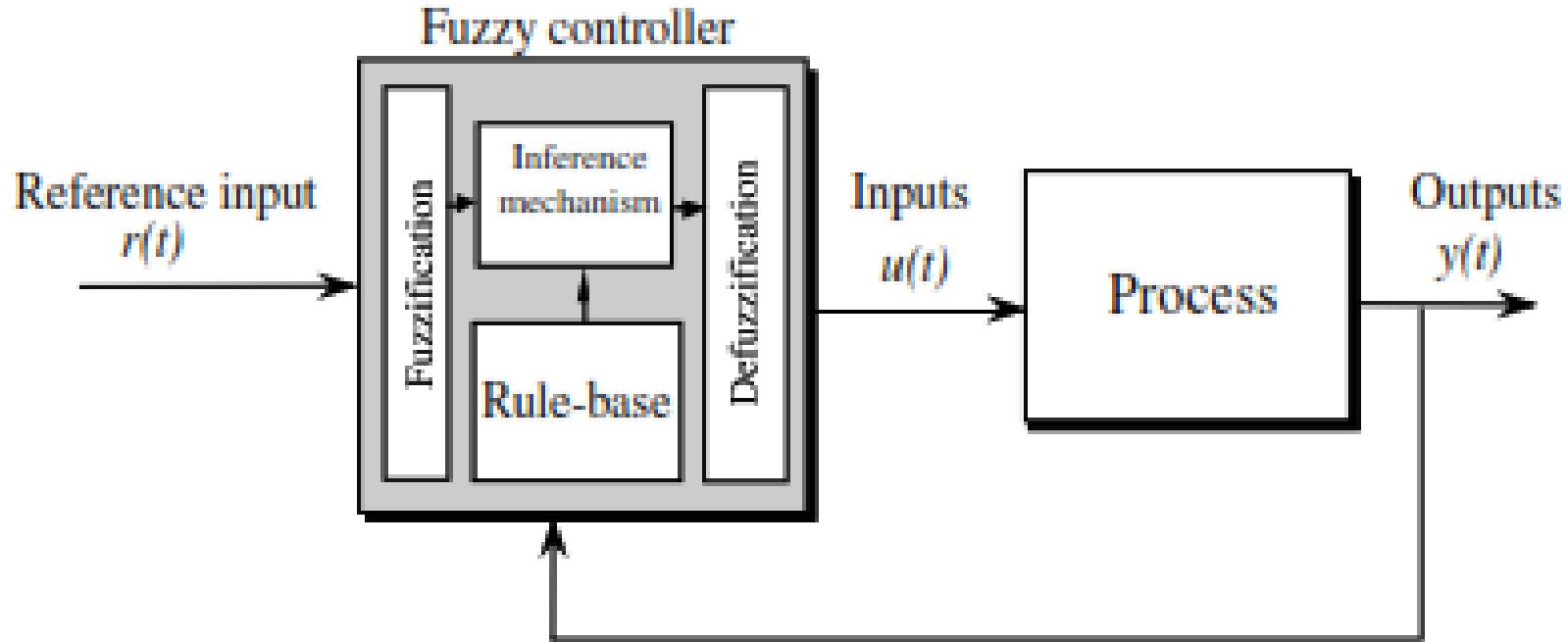


# Control Experto





# Control Borroso (Fuzzy)



If  $\sqrt{b^2 - 4ac} \leq 40$  and  $\beta > 50^\circ$  Then abre la válvula

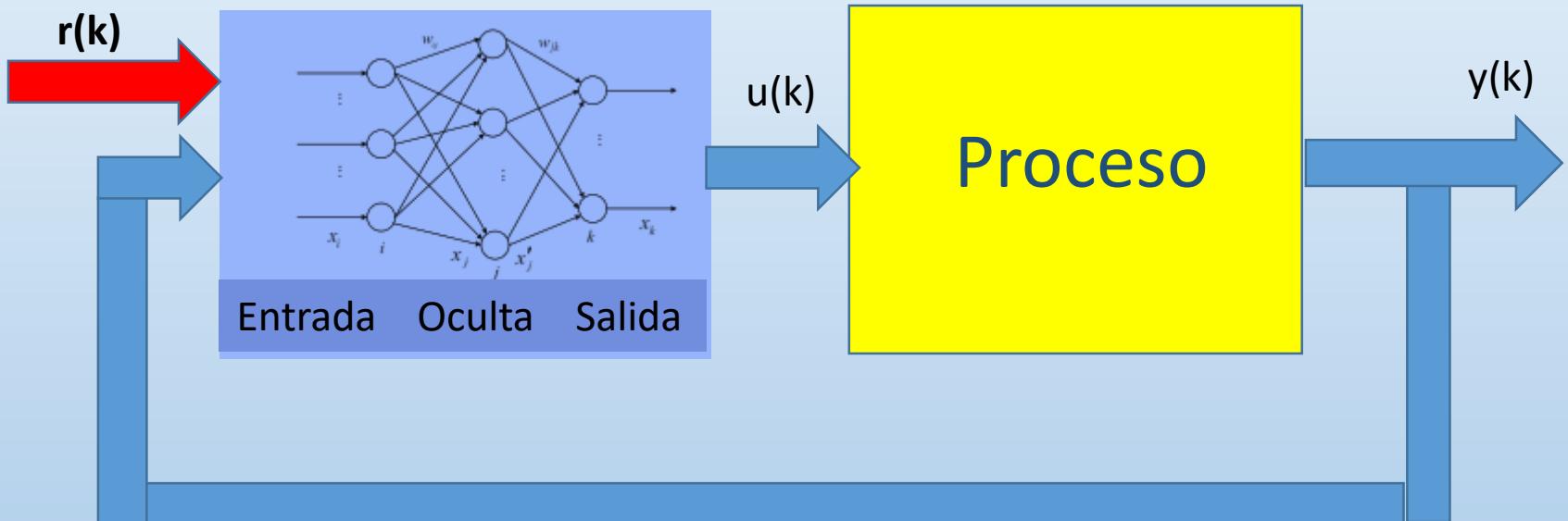
If  $\sqrt{b^2 - 4ac} \leq 40$  and  $\beta > 50^\circ$  Then abre un poco la válvula



# Control Inteligente

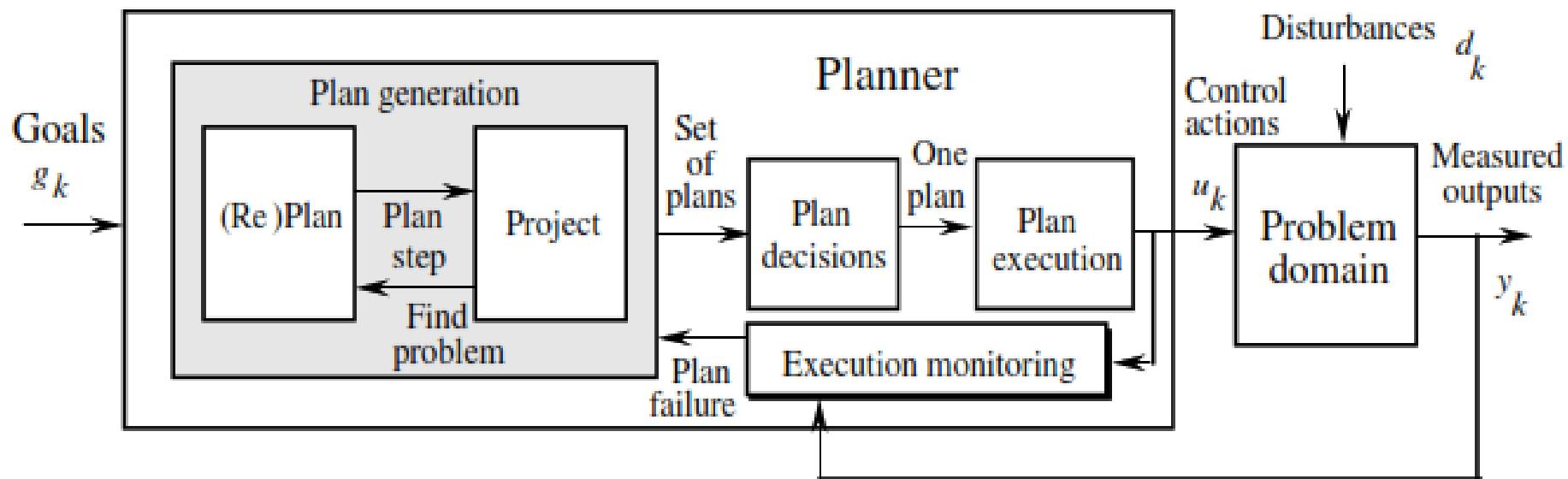
## Redes Neuronales Artificiales

Señal a seguir



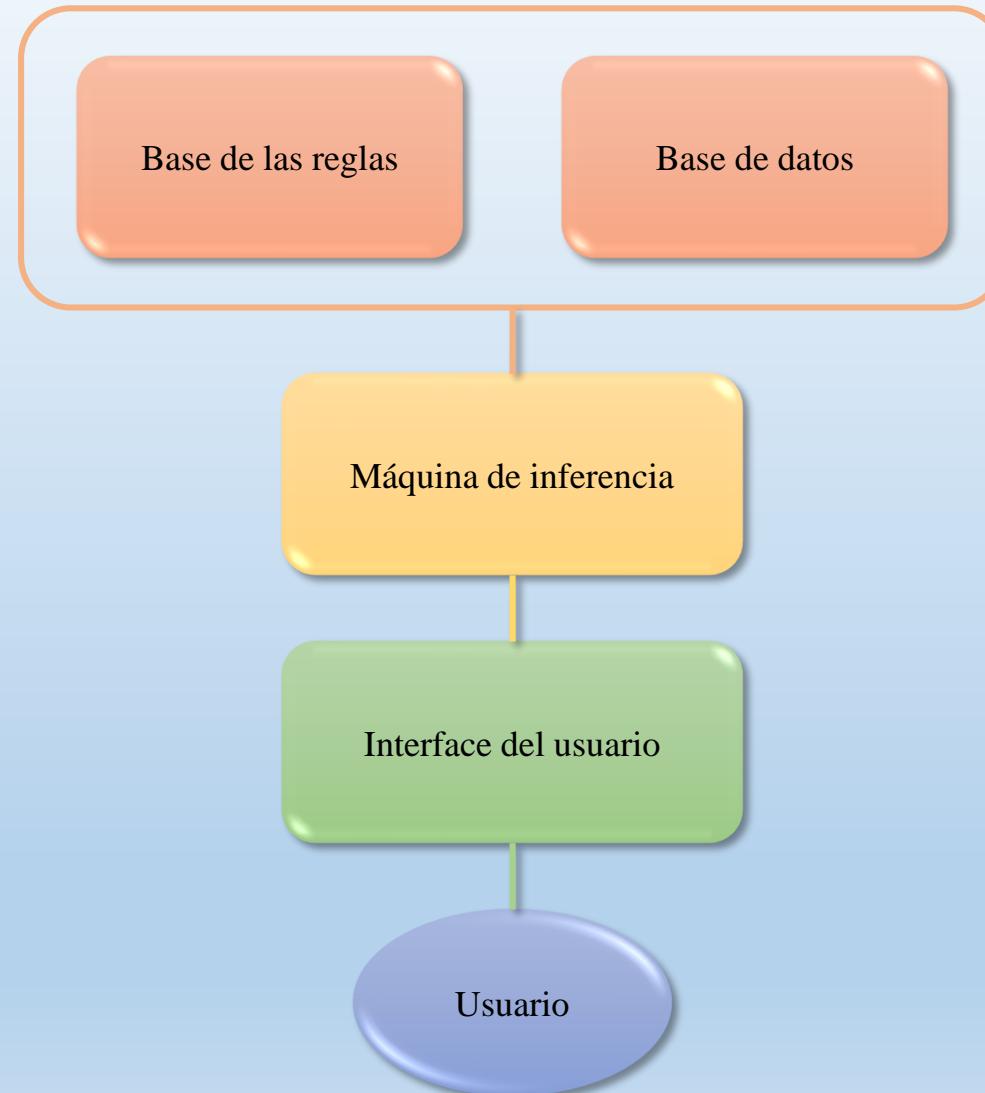


# Algoritmos de Búsqueda Inteligentes



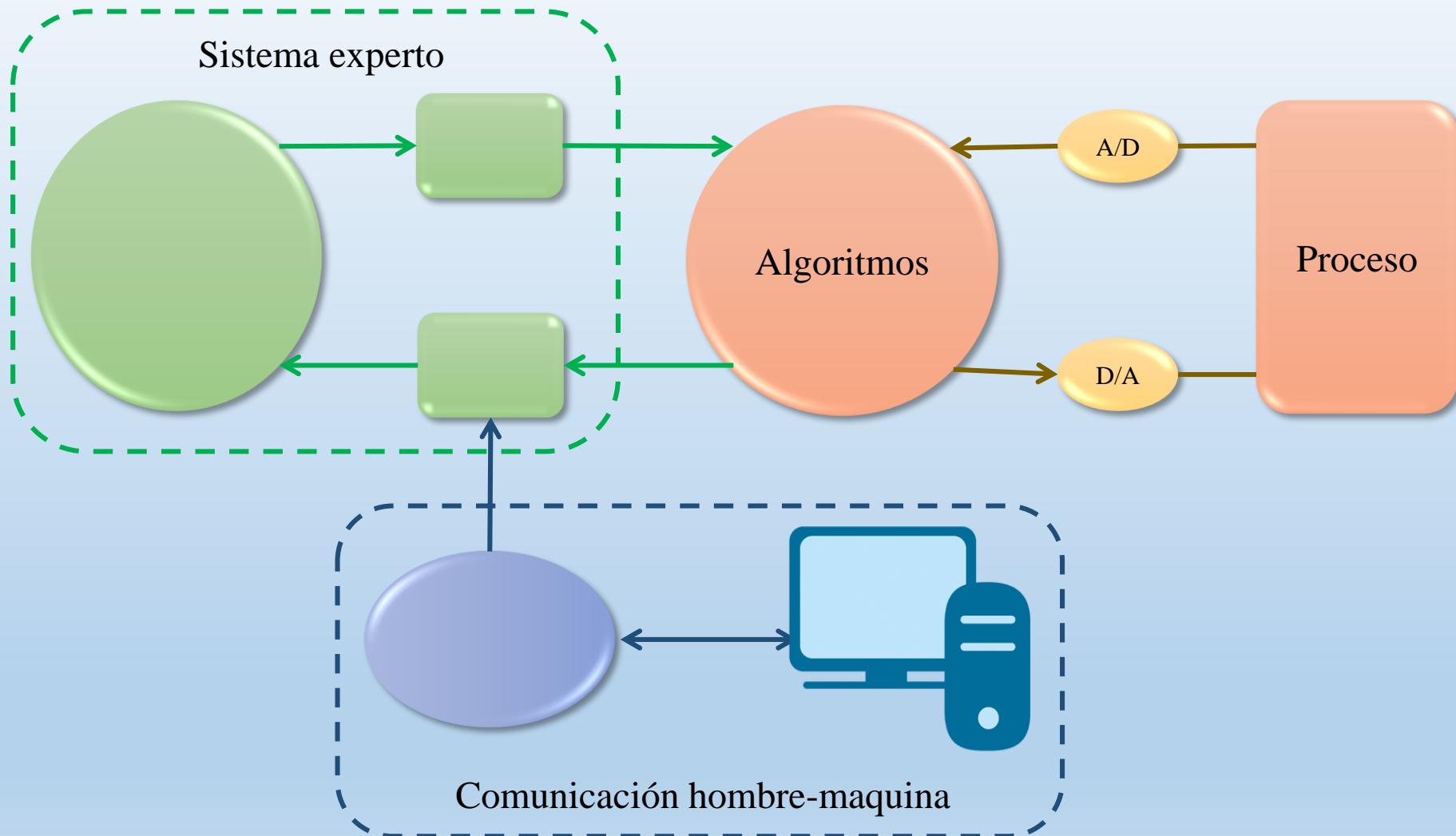


# Sistema Experto





# Control Experto





# PID con Lógica Heurística

Selección de modos de operación

Filtración de señales de entrada

Límite de validación y alarmas

Selección de programa de conmutación

**PID**

Limitación de la salida y de su velocidad

Lógica para condiciones de operación anormales

Condicionamiento de la señal

Lógica “anti-windup”

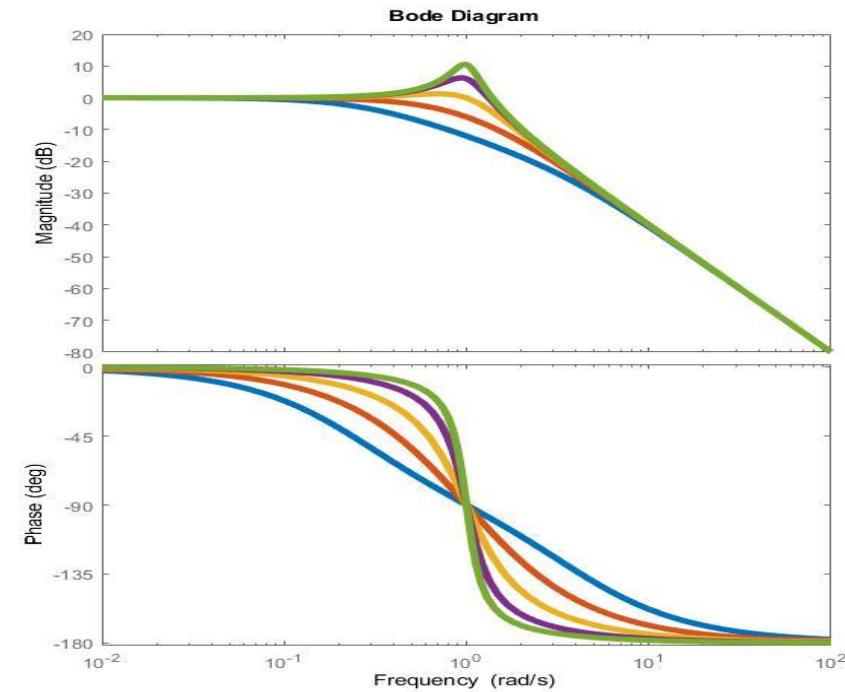
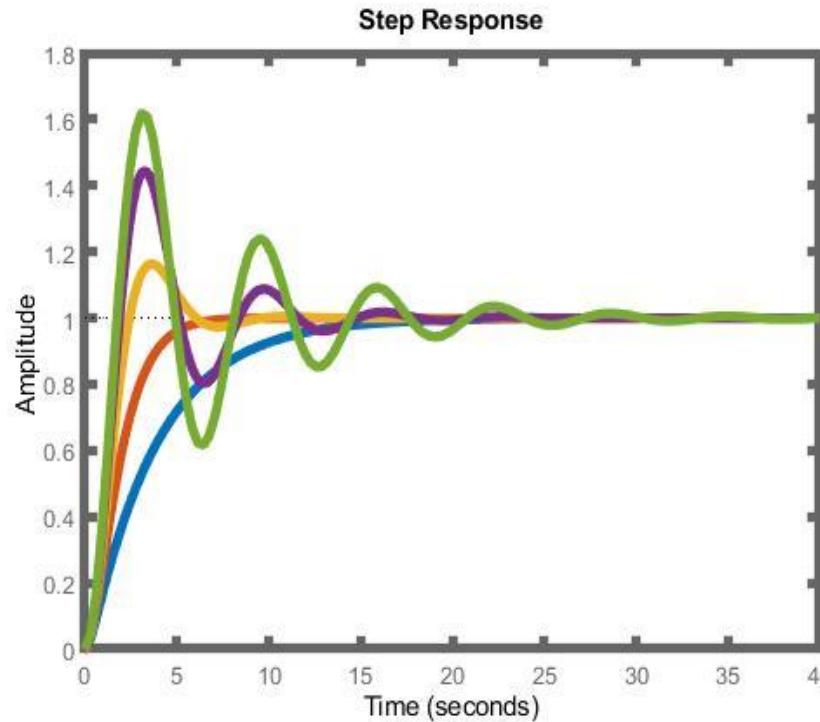


# Respuesta al Escalón y Frecuencia del Sistema

$$\frac{1}{s^2 + 2\xi s + 1}$$

Azul  $2\xi = 4$ ; Naranja  $2\xi = 2$ ; Amarillo  $2\xi = 1$ ; Morado  $2\xi = 0.5$ ; Verde  $2\xi = 0.3$

Azul:  $-3.7321, -0.2679$ ; Naranja:  $-1, -1$ ; Amarillo:  $= 0.5 + 0.866i, -0.5 - 0.866i$ ; Morado:  $-0.25 + 0.9682i, -0.25 - 0.9682i$   
Verde:  $-0.1500 + 0.9887i, -0.1500 - 0.9887i$





# Control PID Experto

Cap 2, Liu

$$e(k) = y_d(k) - y(k), \quad \Delta e(k-1) = e(k) - e(k-1)$$

Reglas:

- 1) Si  $|e(k)| > M_1$  control para minimizar rapido
- 2) Si  $e(k)\Delta e(k) > 0 \circ \Delta e(k) = 0$ 
  - a) Si  $|e(k)| > M_2$  Usar un PID con altas ganancias
  - b) Si  $|e(k)| < M_2$  Usar un PID con ganancias moderadas
- 3) Si  $e(k)\Delta e(k) < 0, \Delta e(k)\Delta e(k-1) > 0$   
 $u(k) = \text{valor constante}$
- 4) Si  $e(k)\Delta e(k) < 0, \Delta e(k)\Delta e(k-1) < 0$ 
  - a) Si  $|e(k)| > M_2$  Usar un P con alta ganancia
  - b) Si  $|e(k)| < M_2$  Usar un P con ganancia moderada
- 5) Si  $|e(k)| < \varepsilon$  Usar un PI moderado



# Ejemplo Pag 9. de Liu

Consider a plant as

$$G_p(s) = \frac{523500}{s^3 + 87.35s^2 + 10470s}$$

The sampling time is 1 ms, using MATLAB command “c2d”, the plant can be discrete as

$$\begin{aligned}y(k) = & -\text{den}(2)y(k-1) - \text{den}(3)y(k-2) - \text{den}(4)y(k-3) \\& + \text{num}(2)u(k-1) + \text{num}(3)u(k-2) + \text{num}(4)u(k-3)\end{aligned}$$

where num() and den() can be gotten by the command tfdata.

tradicional

```
% PID Controller
clear all;
close all;
ts=0.001;
sys=tf(5.235e005,[1,87.35,1.047e004,0]); %Plant establecer pl
dsys=c2d(sys,ts,'z'); convertir a G(z) con retenedor
[num,den]=tfdata(dsys,'v'); ecuación recursiva

u_1=0;u_2=0;u_3=0;u_4=0;
y_1=0;y_2=0;y_3=0;
ei=0;

kp=0.96;ki=0.03;kd=0.01;

error_1=0;
```

Ref y=1

```
for k=1:1:500
    time(k)=k*ts;

    yd(k)=1.0;
    y(k)=-den(2)*y_1-den(3)*y_2-den(4)*y_3+num(1)*u_1+num(2)*u_2+num(3)
    *u_3+num(4)*u_4;

    error(k)=yd(k)-y(k); % Calculating P
    derror(k)=error(k)-error_1; % calculating D
    ei=ei+error(k)*ts;

    u(k)=kp*error(k)+kd*derror(k)/ts+ki*ei; %PID Controller

    u_4=u_3;u_3=u_2;u_2=u_1;u_1=u(k);
    y_3=y_2;y_2=y_1;y_1=y(k);
    error_1=error(k);
end
```



# Script del PID experto

```
kp=0.6;ki=0.03;kd=0.01;
for k=1:1:500
time(k)=k*ts;
yd(k)=1.0; %Tracing Step Signal
%Linear model
y(k)=-den(2)*y_1-den(3)*y_2-den(4)*y_3+num(1)*u_1+num(2)*u_2+num(3)
*u_3+num(4)*u_4;

error(k)=yd(k)-y(k); % Calculating P
derror(k)=error(k)-error_1; % Calculating D
ei=ei+error(k)*ts; % Calculating I
u(k)=kp*error(k)+kd*derror(k)/ts+ki*ei; %PID Controller

%Expert control rule
if abs(error(k))>0.8 %Rule1:Unclosed control rule
    u(k)=0.45;
elseif abs(error(k))>0.40
    u(k)=0.40;
elseif abs(error(k))>0.20
    u(k)=0.12;
elseif abs(error(k))>0.01
    u(k)=0.10;
end
```

```
if error(k)*derror(k)>0 | (derror(k)==0) %Rule2
    if abs(error(k))>=0.05
        u(k)=u_1+2*kp*error(k);
    else
        u(k)=u_1+0.4*kp*error(k);
    end
end

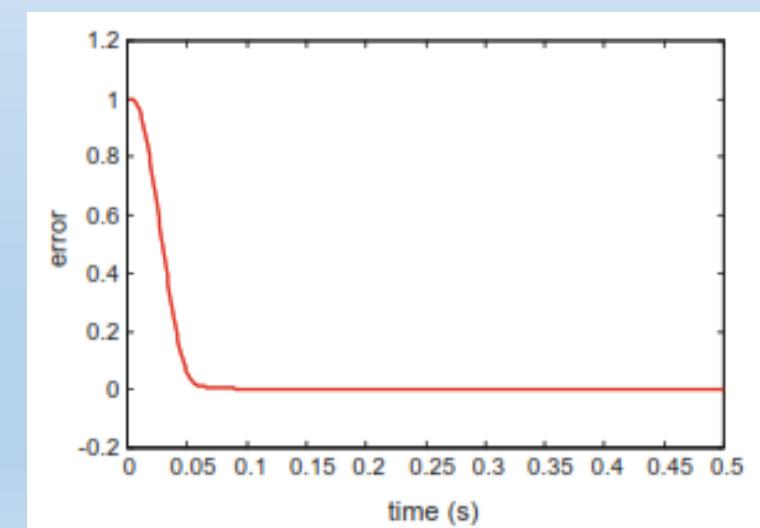
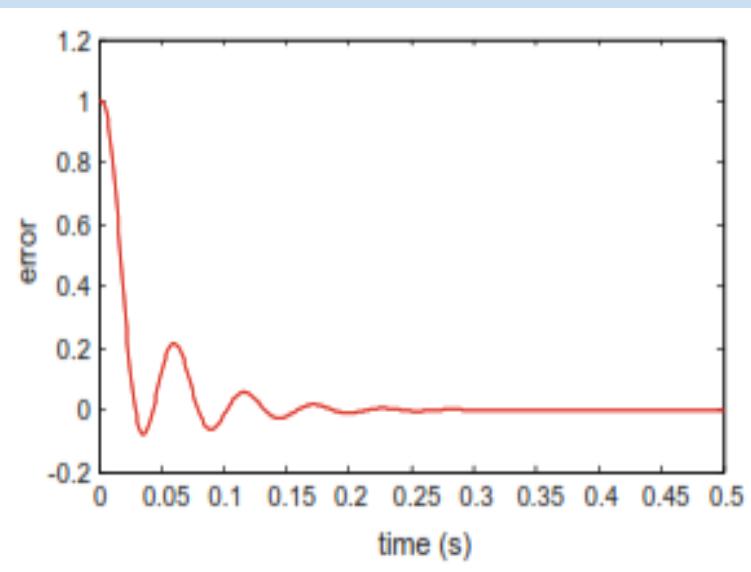
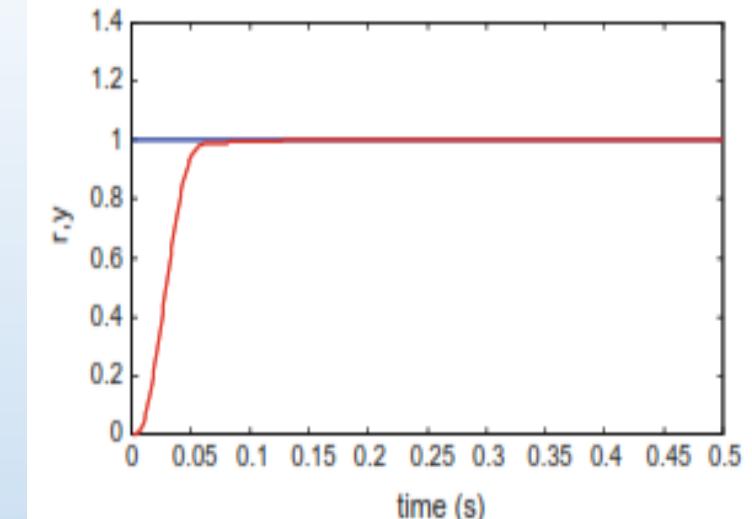
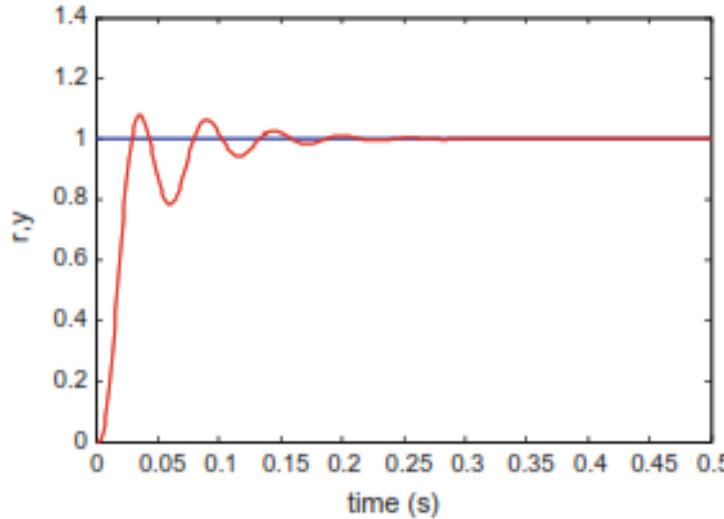
if (error(k)*derror(k)<0&derror(k)*derror_1>0) | (error(k)==0) %Rule3
    u(k)=u(k);
end

if error(k)*derror(k)<0&derror(k)*derror_1<0 %Rule4
    if abs(error(k))>=0.05
        u(k)=u_1+2*kp*error(k);
    else
        u(k)=u_1+0.6*kp*error(k);
    end
end

if abs(error(k))<=0.001 %Rule5:Integration separation PI control
    u(k)=0.5*error(k)+0.010*ei;
end
u_4=u_3;u_3=u_2;u_2=u_1;u_1=u(k);
y_3=y_2;y_2=y_1;y_1=y(k);
error_1=error(k);
derror_1=derror(k);
end
```



# Respuesta con los 2 PIDs



PID tradicional

PID experto



## Tarea por Triada

- Tarea entregar 8 de septiembre:

Considere el sistema con función de transferencia:

$$G(s) = \frac{3000}{s^3 + 150s^2 + 1000s + 500}$$

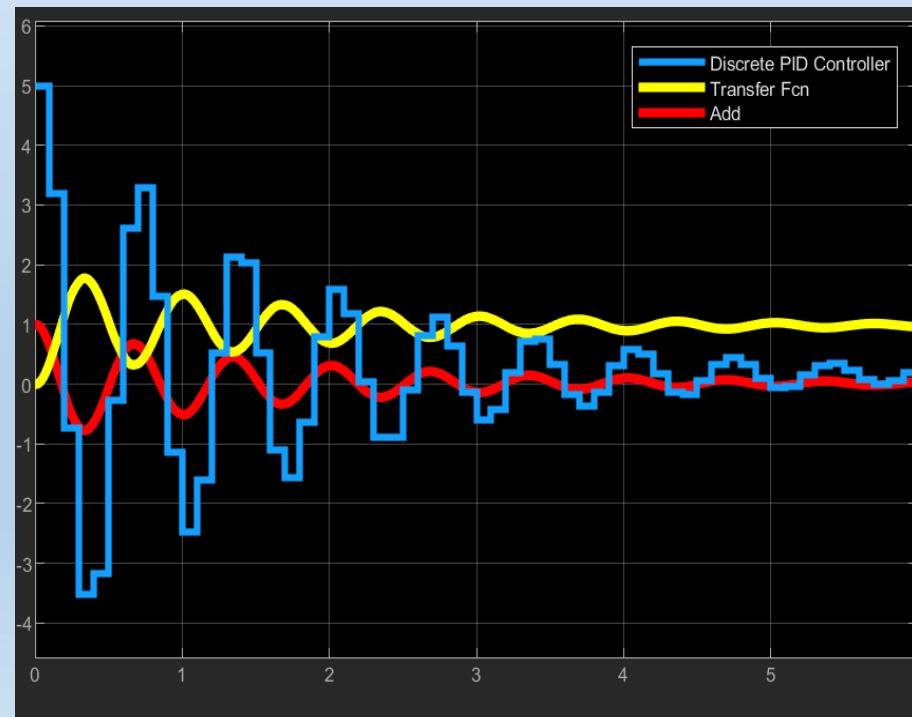
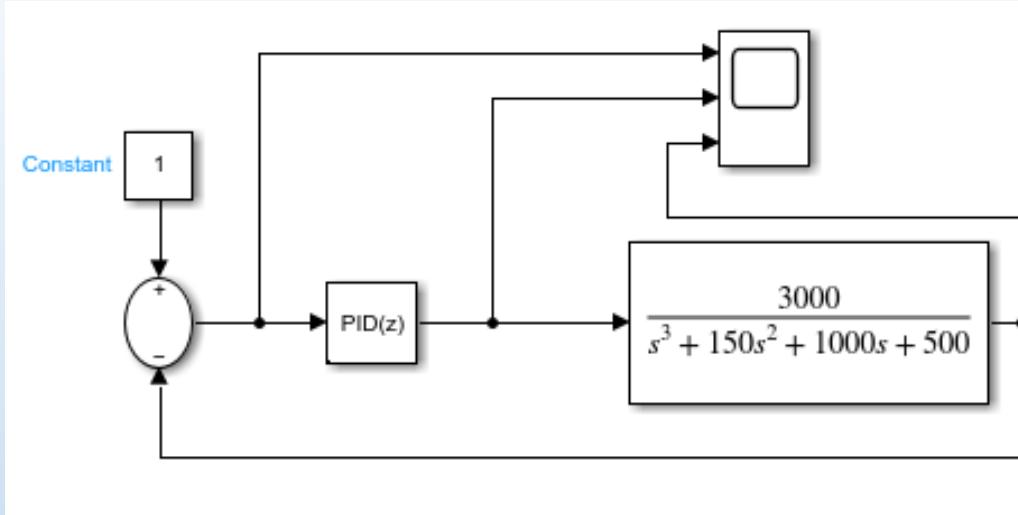
Diseñe un controlador PID discreto tradicional ( $K_p=5$ ,  $K_i=0.5$ ,  $T_s=0.1$ )

Diseñe un controlador PID experto.

Compare el desempeño de ambos PID. La triada que diseña el mejor PID experto tiene la nota máxima.



# PI Discreto Tradicional





## 2. Fundamentos de lógica y control borroso

2.1 Control con lógica borrosa

2.2 Control y modelación Mamdani

2.3 Control y modelación Takagi\_Sugeno

2.4 Control borroso adaptable



# Control Borroso Perspectiva Industrial

- Basado en la FL, (Lofti Zadeh 1965) la cual maneja cantidades vagas o lingüísticas, junto con reglas borrosas para diseñar la ley de control.
- Japón fue el mayor promotor de la tecnología (cámaras, lavadoras, molinos)
  - If (temp. es > al promedio)
  - AND (gradiente es + y grande)
  - OR (temp demasiado alta)
  - THEN (enfriamiento debe ser fuerte)
- Condición de bloqueo que pueden ser una concatenación de lógica y sus conclusiones.
- Categorías: sistemas independientes, procesadores borrosos, micro con funciones borrosas, bloques de funciones.
- Ventajas: Desarrollo rápido en comparación con métodos analíticos.
- Diferencia con los controles expertos, los FC si tienen reglas específicas, los EC solo tienen experiencia del operador.



# Control Borroso Perspectiva Industrial

- Un FKBS es un sistema de control que satisface desempeño adecuado, seguro, y robusto, ayudado por la LF incorporando un conocimiento que no puede ser incorporado fácilmente en un modelo analítico.
- ¿Desempeño adecuado?

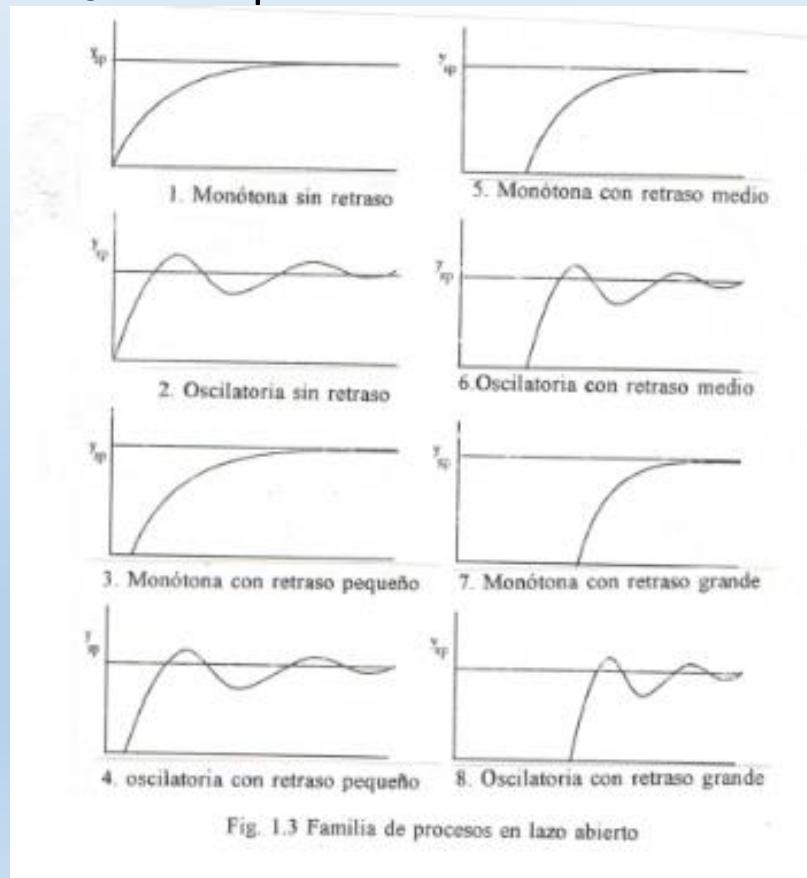


Fig. 1.3 Familia de procesos en lazo abierto

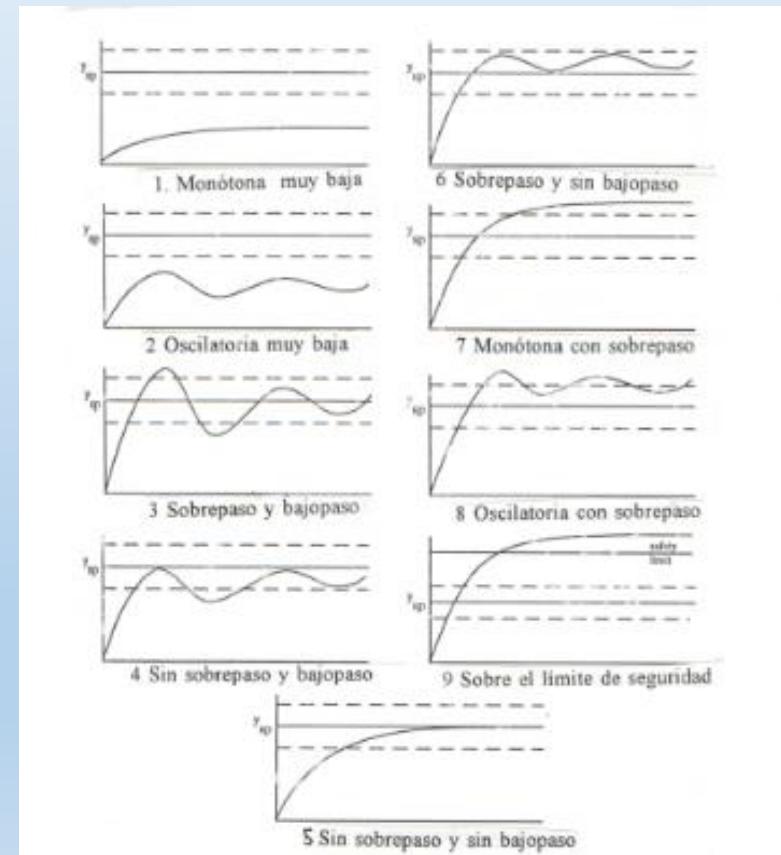


Fig. 1.4 Características típicas en lazo cerrado



# Problemas de la Caracterización de un Proceso

- No es trivial el describirlo aún cuando se desea modelarlo con poco detalle.
- No puede llevarse a un trabajo de rutina.
- Reglas activas  $\forall k$  y de restricciones  $\approx$  límites
- Demanda una definición concreta del problema que se requiere resolver.

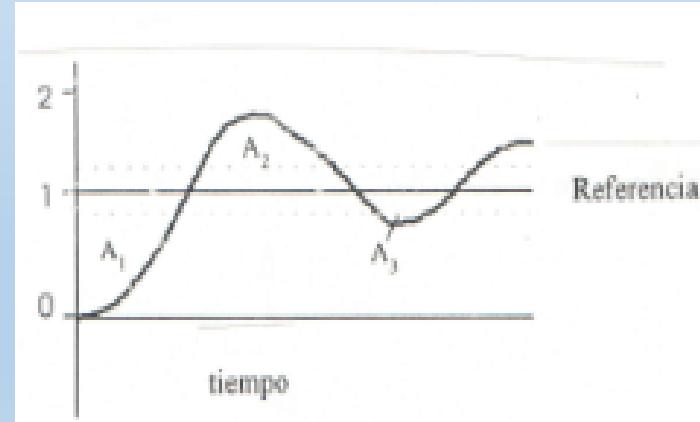


Fig 1.5 Zonas prohibidas de la respuesta



# Ejemplo

LN Large Negative  
SN Small Negative  
Z Zero  
SP Small Positive  
LP Large Positive  
C Constraint  
DC Drastic C

Para las regla activa:

IF  $e(k)$  is SN &  $\Delta e(k)$  is SN  
THEN  $\Delta u(k)$  is Z

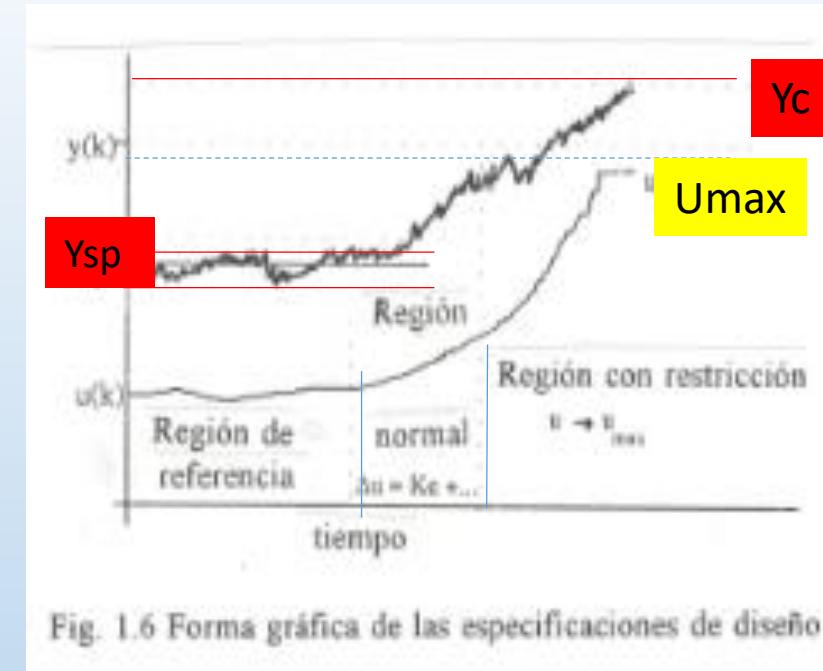


Fig. 1.6 Forma gráfica de las especificaciones de diseño

		$\Delta error$		
		<u>SN</u>	<u>S</u>	<u>LP</u>
error	<u>LN</u>	SN	LN	LN
	<u>SN</u>	Z	SN	SN
	<u>SP</u>	SP	SP	Z
<u>LP</u>	LP	LP	SP	



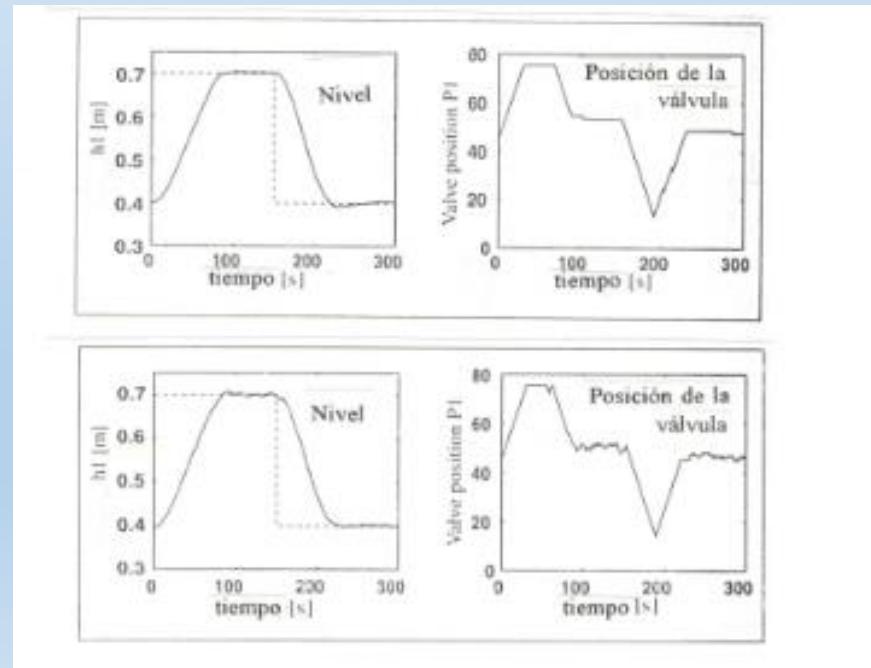
# Para las regla de restricciones

IF  $e(k)$  es C THEN  $\Delta u(k)$  es DC

IF  $e(k)$  entra en C THEN sumarle a  $\Delta u(k)$  los valores obtenidos con la primera regla

IF  $e(k)$  deja C THEN restarle al valor total de  $\Delta u(k)$  los valores obtenidos con la segunda regla

Aplicado al control de nivel de un tanque: Simulink (arriba) y Experimento real (abajo)





# Fundamentos de Lógica Borrosa

Definición: Función de pertenencia de un conjunto A con su función de pertenencia o grado de pertenencia dada por

$$\mu_A(x) = \begin{cases} 1 & x \in A \\ (0,1) & x \in A \text{ par} \\ 0 & x \notin A \end{cases}$$

donde cada  $x$  tiene su propia  $\mu_A(x)$

Definición: El conjunto borroso A está completamente caracterizado por el conjunto parejas o tuples

Universo contable con  $|A| = n$

$$A = \{(x, \mu_A(x)) \mid x \in A\}, = \{(x_1, \mu_A(x_1)) + (x_2, \mu_A(x_2)) + \dots (x_n, \mu_A(x_n))\}$$

$$= \sum_{i=1}^n \mu_A(x_i) / x_i = \mu_A(x_1) / x_1 + \mu_A(x_2) / x_2 + \dots \mu_A(x_n) / x_n$$

Universo continuo

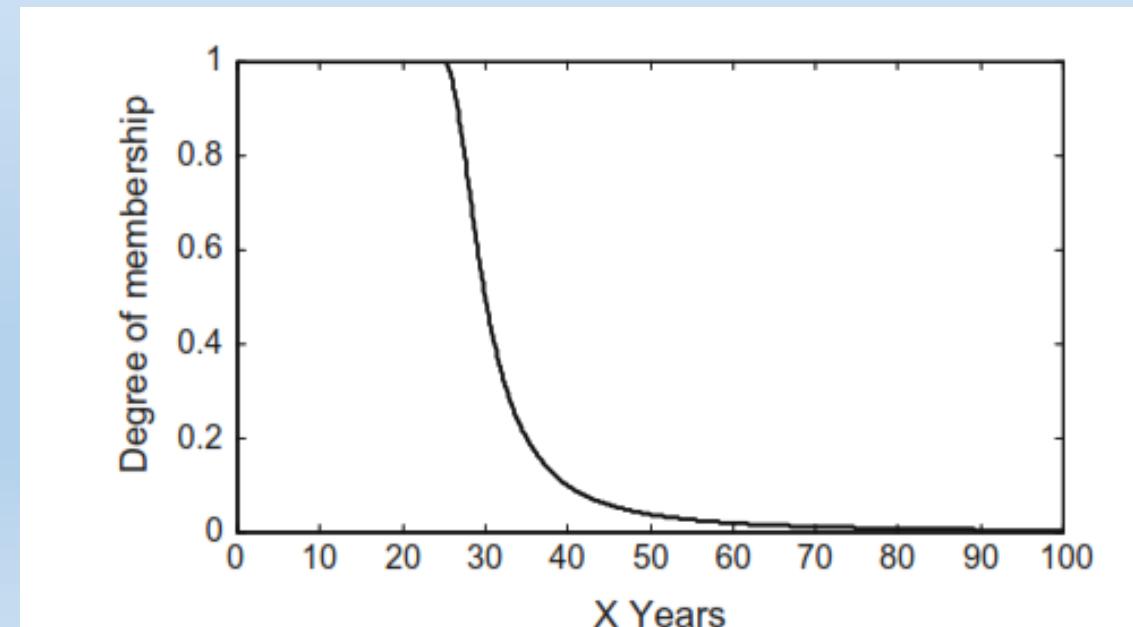
$$A = \int_A \mu_A(x_i) / x_i$$



# Ejemplo

- El conjunto de jóvenes borrosos

$$\mu_A(x) = \begin{cases} 1 & 0 \leq x \leq 25 \\ \left(1 + \left(\frac{x-25}{5}\right)^2\right)^{-1} & 25 < x \leq 100 \end{cases}$$





# Cálculo de Conjuntos Borrosos

1. Conjunto nulo:  $A = \emptyset \Leftrightarrow \mu_A(x) = 0$
2. Conjunto universal:  $A = U \Leftrightarrow \mu_A(x) = 1$
3. Conjuntos iguales:  $A = B \Leftrightarrow \mu_A(x) = \mu_B(x)$
4. Conjunto complemento:  $\overline{A} \Leftrightarrow \mu_{\overline{A}}(x) = 1 - \mu_A(x)$
5.  $B$  Subconjunto de  $A$ :  $B \subseteq A \Leftrightarrow \mu_B(x) \leq \mu_A(x)$
6. Conjunto unión C:  
$$C = A \cup B = \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x)$$

## 7. Conjunto intersección

$$C = A \cap B = \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x)$$



# Ejemplo

- Sean los conjuntos borrosos

$$A = \frac{0.9}{x_1} + \frac{0.2}{x_2} + \frac{0.8}{x_3} + \frac{0.4}{x_4}, \quad B = \frac{0.3}{x_1} + \frac{0.1}{x_2} + \frac{0.4}{x_3} + \frac{0.6}{x_4},$$

entonces

$$A \cup B = \frac{—}{x_1} + \frac{—}{x_2} + \frac{—}{x_3} + \frac{—}{x_4}, \quad A \cap B = \frac{—}{x_1} + \frac{—}{x_2} + \frac{—}{x_3} + \frac{—}{x_4},$$

- Sea

$$\mu_A(x_2) = 0.4, \text{ entonces } \mu_{\bar{A}}(x_2) = 0.6,$$

**NOTA**

$$\mu_A(x_2) \vee \mu_{\bar{A}}(x_2) = 0.6 \neq 1, \quad \mu_A(x_2) \wedge \mu_{\bar{A}}(x_2) = 0.4 \neq 0$$



# Operadores Convencionales

Sean tres conjuntos borrosos A, B y C

Operador intersección

$$C = A \cap B$$

Operador intersección básico

$$\mu_C(x) = \text{Min}\{\mu_A(x), \mu_B(x)\}$$

Operador producto algebráico

$$\mu_C(x) = \mu_A(x) \cdot \mu_B(x)$$

Operador producto limitado

$$\mu_C(x) = \text{Max}\{0, \mu_A(x) + \mu_B(x) - 1\}$$

Operador unión

$$C = A \cup B$$

Operador unión básico

$$\mu_C(x) = \text{Max}\{\mu_A(x), \mu_B(x)\}$$

Operador OR probabilístico

$$\mu_C(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x)$$

Operador suma limitada

$$\mu_C(x) = \text{Min}\{1, \mu_A(x) + \mu_B(x)\}$$

Operador balanceado:  $C = A \circ B$

$$\mu_C(x) = [\mu_A(x) \cdot \mu_B(x)]^{1-\gamma} [1 - (1 - \mu_A(x)) \cdot ((1 - \mu_B(x))]^\gamma \quad \gamma = (0, 1)$$

If  $\gamma=0$   $\mu_C(x) = \mu_A(x) \cdot \mu_B(x) \Rightarrow C = A \cap B$

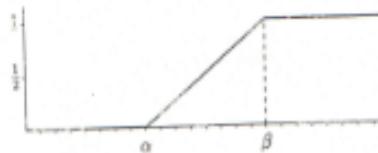
If  $\gamma=1$   $\mu_C(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x) \Rightarrow C = A \cup B$



# Funciones de Pertenencia Típicas

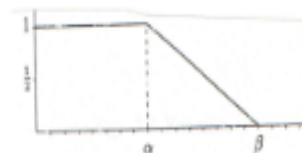
Función Γ:

$$\Gamma(u; \alpha, \beta) = \begin{cases} 0 & u < \alpha \\ (u - \alpha)/(\beta - \alpha) & \alpha \leq u \leq \beta \\ 1 & u > \beta \end{cases}$$



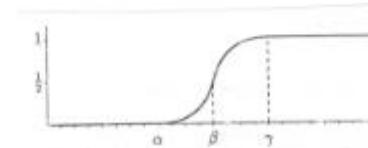
Función L:

$$L(u; \alpha, \beta) = \begin{cases} 1 & \alpha \leq u \leq \beta \\ 0 & u < \alpha \text{ or } u > \beta \end{cases}$$



Función S:

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 & \text{for } x \leq \alpha \\ 2\left(\frac{x - \alpha}{\gamma - \alpha}\right)^2 & \text{for } \alpha < x \leq \beta \\ 1 - 2\left(\frac{x - \gamma}{\gamma - \alpha}\right)^2 & \text{for } \beta < x \leq \gamma \\ 1 & \text{for } x > \gamma \end{cases}$$



Función Λ:

$$\Lambda(u; \alpha, \beta, \gamma) = \begin{cases} 0 & u \leq \alpha \\ (\alpha - u)/(\beta - \alpha) & \alpha \leq u \leq \beta \\ (\gamma - u)/(\beta - \alpha) & \beta \leq u \leq \gamma \\ 0 & u \geq \gamma \end{cases}$$

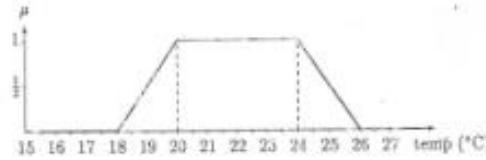




# Funciones de Pertenencia Típicas

Función Π:

$$\Delta(u; \alpha, \beta, \gamma) = \begin{cases} 0 & u < \alpha, \\ (\mu - \alpha)/(\beta - \alpha) & \alpha \leq u < \beta \\ 1 & \beta \leq u < \gamma \\ (\gamma - u)/(\delta - \gamma) & \gamma \leq u \leq \delta \\ 0 & u > \delta \end{cases}$$





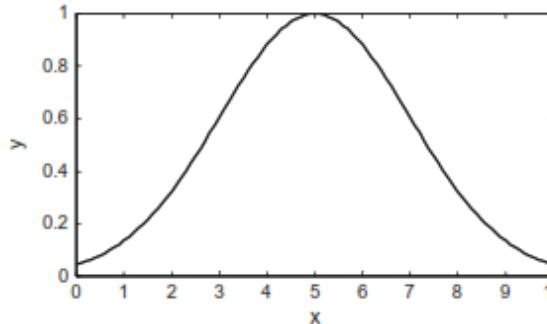
# Función de Pertenencia & Script

Gaussiana:

$$f(x, \sigma, c) = \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \quad \sigma \leq 0$$

```
if M==1      %Guassian membership function
```

```
x=0:0.1:10;  
y=gaussmf(x,[2 5]);  
plot(x,y,'k');  
xlabel('x');ylabel('y');
```

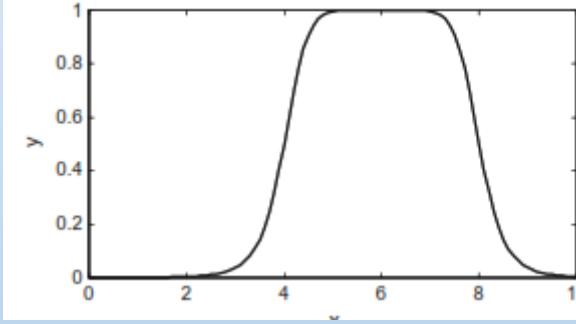


Campana:

$$f(x, a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}}$$

```
elseif M==2      %General Bell membership
```

```
x=0:0.1:10;  
y=gbellmf(x,[2 4 6]);  
plot(x,y,'k');  
xlabel('x');ylabel('y');
```

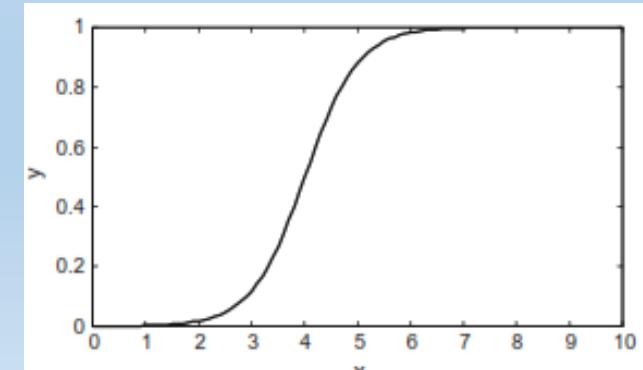


Función Sigma:

$$f(x, a, b, c) = \frac{1}{1 + \exp(-a(x-c))}$$

```
elseif M==3      %S membership function
```

```
x=0:0.1:10;  
y=sigmf(x,[2 4]);  
plot(x,y,'k');  
xlabel('x');ylabel('y');
```



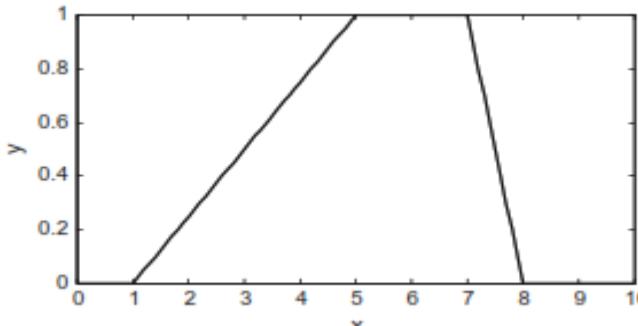


# Función de Pertenencia & Script

Trapecio:

$$f(x,a,b,c,d) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c \leq x \leq d \\ 0 & x \geq d \end{cases}$$

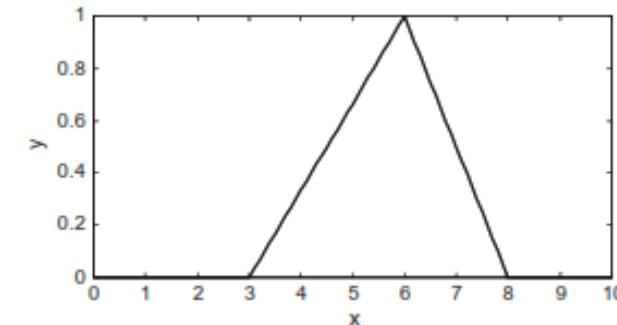
```
elseif M==4 %Trapezoid membership function
x=0:0.1:10;
y=trapmf(x, [1 5 7 8]);
plot(x,y, 'k');
xlabel('x');ylabel('y');
```



Triangular:

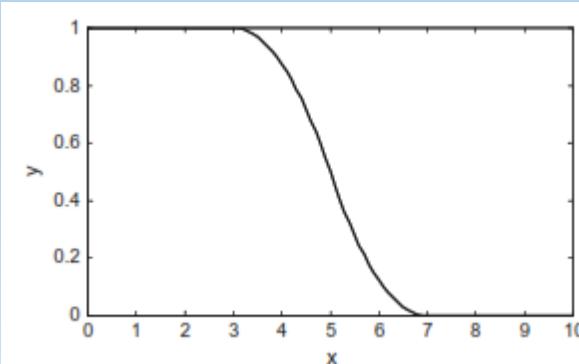
$$f(x,a,b,c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & x \geq c \end{cases}$$

```
elseif M==5 %Triangle membership function
x=0:0.1:10;
y=trimf(x, [3 6 8]);
plot(x,y, 'k');
xlabel('x');ylabel('y');
```



Función Z:

```
elseif M==6 %Z membership function
x=0:0.1:10;
y=zmf(x, [3 7]);
plot(x,y, 'k');
xlabel('x');ylabel('y');
```





# Generación de Variables Borrosas

Paso 1 Definir los N=7 conjuntos borrosos, ejemplo

$$e = \{ \underline{NB}, \underline{NM}, \underline{NS}, \underline{Z} \quad \underline{PS} \quad \underline{PM} \quad \underline{PB} \}$$

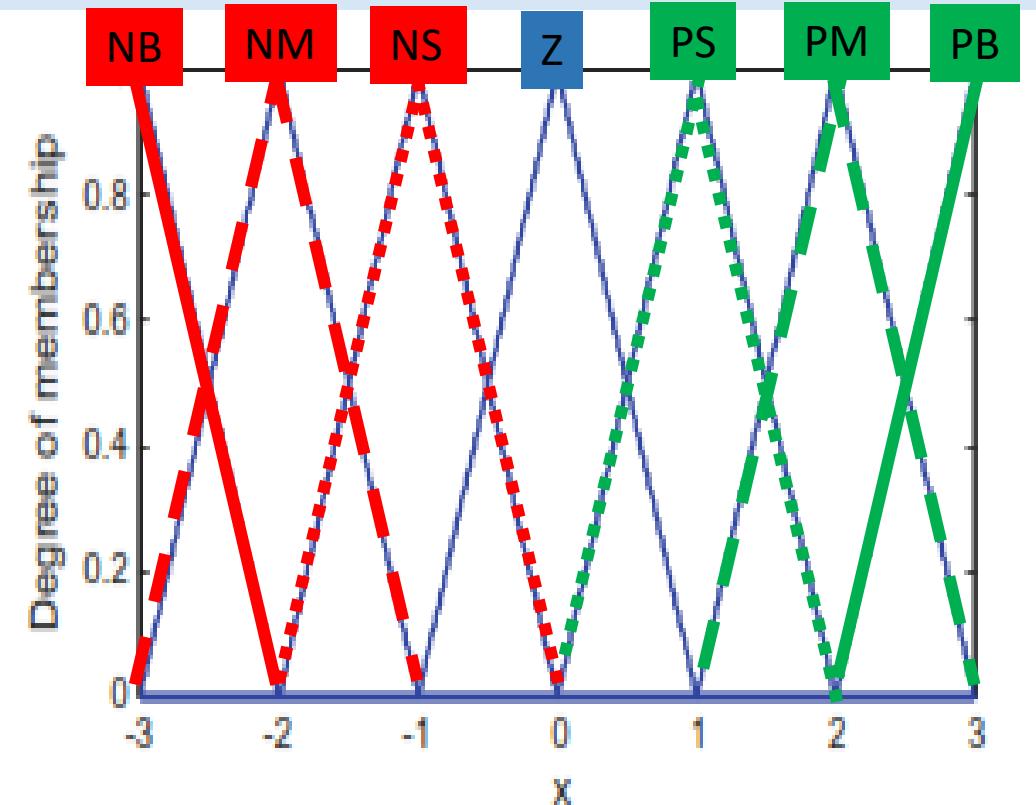
Paso 2 Definir el tipo de MF de cada conjunto, triángulo

Paso 3 Considerar el dominio de la variable, [-3,3]

```
%Define N+1 triangle membership function
clear all;
close all;
N=6;

x=-3:0.01:3;
for i=1:N+1
    f(i)=-3+6/N*(i-1);
end
u=trimf(x, [f(1), f(1), f(2)]);

figure(1);
plot(x,u);
for j=2:N
    u=trimf(x, [f(j-1), f(j), f(j+1)]);
    hold on;
    plot(x,u);
end
u=trimf(x, [f(N), f(N+1), f(N+1)]);
hold on;
plot(x,u);
xlabel('x');
ylabel('Degree of membership');
```





# Relación o Matriz Borrosa

- Alumnos y su Desempeño en 4 Asignaturas

$N_{om} / A_{sig}$	<i>Mat</i>	<i>Fís</i>	<i>Eléc</i>	<i>Mec</i>
<i>Juan</i>	0.7	0.9	0.8	0.65
<i>Pedro</i>	0.9	0.85	0.76	0.7
<i>Mario</i>	0.5	0.95	0.85	0.8
<i>José</i>	0.5	0.6	0.7	0.6

$$R = \begin{bmatrix} 0.7 & 0.9 & 0.8 & 0.65 \\ 0.9 & 0.85 & 0.76 & 0.7 \\ 0.5 & 0.95 & 0.85 & 0.8 \\ 0.5 & 0.6 & 0.7 & 0.6 \end{bmatrix}$$



# Cálculo de Matriz “Relación” Borrosa

Para dos matrices borrosas A y B de igual dimensión se pueden definir varias matrices asociadas a dichos conjuntos.

1. Igualdad: For  $a_{i,j} = b_{i,j} \Rightarrow A = B$
2. Subconjunto: For  $a_{i,j} \leq b_{i,j} \Rightarrow A \subseteq B$
3. Unión: For  $c_{i,j} = a_{i,j} \vee b_{i,j} \Rightarrow C = A \cup B$
4. Intersección: For  $c_{i,j} = a_{i,j} \wedge b_{i,j} \Rightarrow C = A \cap B$
5. Complemento: For  $c_{i,j} = 1 - a_{i,j} \Rightarrow C = \bar{A}$

$$\text{Ex. 3.7 } A = \begin{bmatrix} 0.7 & 0.1 \\ 0.3 & 0.9 \end{bmatrix}, B = \begin{bmatrix} 0.4 & 0.9 \\ 0.2 & 0.1 \end{bmatrix}$$

$$A \cap B = \begin{bmatrix} 0.7 \wedge 0.4 & 0.1 \wedge 0.9 \\ 0.3 \wedge 0.2 & 0.9 \wedge 0.1 \end{bmatrix} = \begin{bmatrix} 0.4 & 0.1 \\ 0.2 & 0.1 \end{bmatrix}$$

$$A \cup B = \begin{bmatrix} 0.7 \vee 0.4 & 0.1 \vee 0.9 \\ 0.3 \vee 0.2 & 0.9 \vee 0.1 \end{bmatrix} = \begin{bmatrix} 0.7 & 0.9 \\ 0.3 & 0.9 \end{bmatrix}$$

$$\bar{A} = \begin{bmatrix} 1 - 0.7 & 1 - 0.1 \\ 1 - 0.3 & 1 - 0.9 \end{bmatrix} = \begin{bmatrix} 0.3 & 0.9 \\ 0.7 & 0.1 \end{bmatrix}$$



# Composición de Relaciones Borrosas

- Si  $A$  y  $B$  son relaciones en  $x \times y$  y  $y \times z$  entonces  $C = A \circ B$  tal que  $c_{i,j} = \vee_k \{a_{ik} \wedge b_{kj}\}$

```
clear all;
close all;
A=[0.8,0.7;
 0.5,0.3];
B=[0.2,0.4;
 0.6,0.9];
%Compound of A and B
for i=1:2
  for j=1:2
    AB(i,j)=max(min(A(:,i),B(:,j)'));
  end
end

%Compound of B and A
for i=1:2
  for j=1:2
    BA(i,j)=max(min(B(:,j),A(:,i)'));
  end
end
```

Ejemplo

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}, C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

$$c_{11} = (a_{11} \wedge b_{11}) \vee (a_{12} \wedge b_{21})$$

$$c_{12} = (a_{11} \wedge b_{12}) \vee (a_{12} \wedge b_{22})$$

$$c_{21} = (a_{21} \wedge b_{11}) \vee (a_{22} \wedge b_{21})$$

$$c_{22} = (a_{21} \wedge b_{12}) \vee (a_{22} \wedge b_{22})$$

Nota: La operación no permuta



# Ejemplo

- Similitud entre nietas & abuelos y nietos y abuelos

$$R \circ S = \begin{bmatrix} 0.2 & 0.2 \\ 0.5 & 0.6 \end{bmatrix} \Rightarrow \begin{array}{ll} \begin{matrix} Pa & Ma \\ Ho & 0.2 & 0.8 \end{matrix} & \begin{matrix} Ao & Aa \\ Pa & 0.5 & 0.7 \end{matrix} \\ \begin{matrix} Ha & 0.6 & 0.1 \end{matrix} & \begin{matrix} Ma & 0.1 & 0 \end{matrix} \end{array}$$

Matrix R                                  Matrix S

$$R \circ S = \begin{bmatrix} 0.2 & 0.2 \\ 0.5 & 0.6 \end{bmatrix} \Rightarrow \begin{array}{ll} \begin{matrix} Ao & Aa \\ No & 0.2 & 0.2 \end{matrix} & \begin{matrix} Na & 0.5 & 0.6 \end{matrix} \end{array}$$



# Inferencia & Ecuación Borrosa

Semana 4

- De acuerdo con el método de inferencia de **Mandani**.  
Inferencia: If A & B THEN C

$$A \wedge B \rightarrow C$$

$$R = (A \times B)^{\textcolor{red}{Tc}} \times C$$

con **Tc** es la transformación en forma de vector **columna**

Para otros conjuntos  $A_1$ ,  $B_1$  y  $C_1$ , usando la matriz de relación  $R$

$$C_1 = (A_1 \times B_1)^{\textcolor{red}{Tf}} \times R$$

con **Tf** es la transformación en forma de vector **fila**



# Ejemplo y Programa

Sea  $A = \frac{0.5}{a_1} + \frac{1}{a_2} + \frac{0.1}{a_3}$ ;  $B = \frac{0.1}{b_1} + \frac{1}{b_2} + \frac{0.6}{b_3}$ ;  $C = \frac{0.4}{c_1} + \frac{1}{c_2} + \frac{0}{c_3}$  entonces la relación de

"IF A AND B THEN C" es una relación R

$$A \times B = \begin{bmatrix} 0.5 \\ 1 \\ 0.1 \end{bmatrix} \circ [0.1 \ 1 \ 0.6] = \begin{bmatrix} 0.1 & 0.5 & 0.5 \\ 0.1 & 1 & 0.6 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}$$

$$A \times B^{T_c} = \begin{bmatrix} 0.1 \\ 0.5 \\ 0.5 \\ 0.1 \\ 1 \\ 0.6 \\ 0.1 \\ 0.1 \\ 0.1 \end{bmatrix}; R = (A \times B)^{T_c} \times C = (A \times B^{T_c}) \circ [0.4 \ 1 \ 0] = \begin{bmatrix} 0.1 & 0.1 \\ 0.4 & 0.5 \\ 0.4 & 0.5 \\ 0.1 & 0.1 \\ 0.4 & 1 \\ 0.4 & 0.6 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \end{bmatrix}$$

Para otro  $A_l = [1 \ 0.5 \ 0.1]$  y  $B_l = [0.1 \ 0.5 \ 1]$  usando R, se tiene

$$A_l \times B_l = \begin{bmatrix} 0.1 & 0.5 & 1 \\ 0.1 & 0.5 & 0.5 \\ 0.1 & 0.1 & 0.1 \end{bmatrix};$$

$$(A_l \times B_l)^{T_f} = [0.1 \ 0.5 \ 1 \ 0.1 \ 0.5 \ 0.5 \ 0.1 \ 0.1 \ 0.1]$$

$$C = [0.1 \ 0.5 \ 1 \ 0.1 \ 0.5 \ 0.5 \ 0.1 \ 0.1 \ 0.1] \circ [R] = [0.4 \ 0.5]$$

```

clear all;
close all;

A=[0.5;1;0.1];
B=[0.1,1,0.6];
C=[0.4,1];

%Compound of A and B
for i=1:3
    for j=1:3
        AB(i,j)=min(A(i),B(j));
    end
end
%Transfer to Column
T1=[];
for i=1:3
    T1=[T1;AB(i,:)];
end
%Get fuzzy R
for i=1:9
    for j=1:2
        R(i,j)=min(T1(i),C(j));
    end
end
A1=[1,0.5,0.1];
B1=[0.1,0.5,1];

for i=1:3
    for j=1:3
        AB1(i,j)=min(A1(i),B1(j));
    end
end
%Transfer to Row
T2=[];
for i=1:3
    T2=[T2,AB1(i,:)];
end

```



# Ecuación y Variables

Sean  $A$  y  $B$  dos conjuntos borrosos y  $R$  una relación, la ecuación de relación es de la forma

$$A \circ R = B$$

que representa la composición de  $A$  con  $R$  donde la relación pueden involucrar MF desconocidas.

Ejemplo: Resolver la ecuación borrosa

$$[0.6 \quad 0.2 \quad 0.4] \circ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0.4$$

$$(0.6 \wedge x_1) \vee (0.2 \wedge x_2) \vee (0.4 \wedge x_3) = 0.4$$

$$\text{Dado que } (0.2 \wedge x_2) < 0.4 \Rightarrow x_2 = [0 \quad 1]$$

La ecuación se reduce a

$$(0.6 \wedge x_1) \vee (0.4 \wedge x_3) = 0.4$$

y entonces se tienen dos soluciones (condiciones) posibles

$$\text{Condición 1: } (0.6 \wedge x_1) = 0.4 \quad (0.4 \wedge x_3) \leq 0.4 \Rightarrow x_1 = 0.4, \quad x_3 = [0 \quad 1]$$

$$\text{Condición 2: } (0.6 \wedge x_1) \leq 0.4 \quad (0.4 \wedge x_3) = 0.4 \Rightarrow x_1 = [0 \quad 0.4], \quad x_3 = [0.4 \quad 1]$$



# Control con Lógica Borrosa

- Control parcialmente cierto, que puede llegar a ser falso o verdadero.
- Intervalos de las variables más comunes en control

$$v = \{NB \quad NS \quad ZO \quad PS \quad PB\}$$

$$v = \{NB \quad NM \quad NS \quad ZO \quad PS \quad PM \quad PB\}$$

$$v = \{NB \quad NM \quad NS \quad NZ \quad PZ \quad PS \quad PM \quad PB\}$$

- Reglas más comunes

R1: ***IF*  $e \in NB$  *AND*  $\Delta e \in NB$  *THEN*  $u \in PB$**

R2: ***IF*  $e \in NB$  *AND*  $\Delta e \in NS$  *THEN*  $u \in PM$**

de las cuales se obtiene la matriz de relación R.

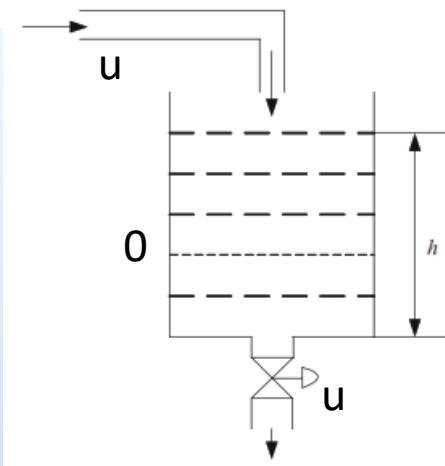
- Inferencia y Cuantificación. Para cualquier otro premisa de A y B se tiene la matriz conclusión

$$C = (A \times B) \circ R$$

que debe ser cuantificada en casos reales.



# Control de Nivel en un Tanque



Paso 1. Sea  $h_0$  el nivel ideal,  $e = \Delta h = h_0 - h$  el error

Variables :  $e = \{-3, -2, -1, 0, +1, +2, +3\}$ ,  $u = \{-4, -3, -2, -1, 0, +1, +2, +3, +4\}$

Paso 2  $e = \{NB, NS, 0, PS, PB\}$ ,

$u = \{NB, NS, 0, PS, PB\}$

IF  $h$  es mayor que 0, THEN drena (negativa);

IF  $h$  es menor que 0, THEN llena (positiva);

error	-3	-2	-1	0	+1	+2	+3	$u$	-4	-3	-2	-1	0	+1	+2	+3	+4
$PB$	0	0	0	0	0	0.5	1	$PB$							0.5	1	1
$PS$	0	0	0	0	1	0.5	0	$PS$						0.5	1	0.5	
0	0	0	0.5	1	0.5	0	0	0					0.5	1	0.5		
$NS$	0	0.5	1	0	0	0	0	$NS$		0.5	1	0.5					
$NB$	1	0.5	0	0	0	0	0	$NB$	1	0.5							

Paso 3 Reglas

- 1) IF  $e = NBe$  THEN  $u = NBu$
- 2) IF  $e = NSe$  THEN  $u = NSu$
- 3) IF  $e = 0e$  THEN  $u = 0u$
- 4) IF  $e = PSe$  THEN  $u = PSu$
- 5) IF  $e = PBe$  THEN  $u = PBu$

Paso 4

La matriz de relación

$$R = (NBe \times NBu) \cup (NSe \times NSu) \cup (0e \times 0u) \cup (PSe \times PSu) \cup (PBe \times PBu)$$



# Matrices de Inferencia

$$Oe \times Out = \begin{bmatrix} 0 \\ 0 \\ 0.5 \\ 1.0 \\ 0.5 \\ 0 \\ 0 \end{bmatrix} \times [0 \ 0 \ 0 \ 0.5 \ 1 \ 0.5 \ 0 \ 0 \ 0] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 1.0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



## Paso 5 Decisión $u = e \circ R$

Cuando  $e = NB$ ,  $e = [ 1. \quad 0.5 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 ]$ ,

$$u = e \circ R = [ 1 \quad 0.5 \quad 0.5 \quad 0.5 \quad 0 \quad 0 \quad 0 \quad 0 ]$$

## Paso 6 Cuantificar

$$u = \frac{1}{-4} + \frac{0.5}{-3} + \frac{0.5}{-2} + \frac{0.5}{-1} \Rightarrow u = -4$$

El programa se encuentra en el libro.



# Procedimiento de un control borroso

1. Definir las variables medidas, deseadas y las acciones:

$$e(k) = r(k) - y(k); \Delta e(k) = e(k - 1) - e(k); u_i(k)$$

2. Cuantificar las variables en números borrosos con su MF,  $|e|=5$ ,

$$e = \{NB \quad NS \quad Z \quad PS \quad PB\}$$

3. Diseñar las reglas borrosas y describirlas en una tabla

#reglas=n=5)

IF e	NBe	NSe	Ze	PSe	PBe
THEN u	NBu	NSu	Zu	PSu	PBu

4. Obtener la matriz de relación borrosa R considerando las reglas y las variables de entrada y salida  $|e|x|s|$

$$R = r_1 \cup r_2 \dots r_n$$

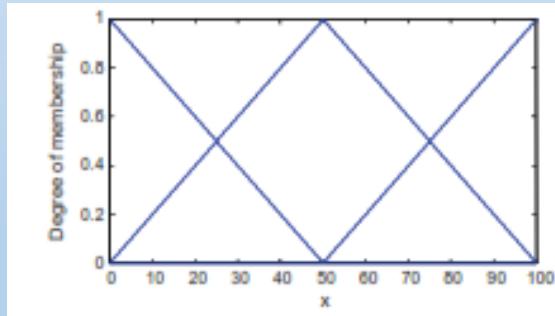
5. Tomar la decisión para cada entrada obteniendo las acciones borrosas  $u_i$  con su respectiva MF

6. Convertir las acciones borrosas en valores reales

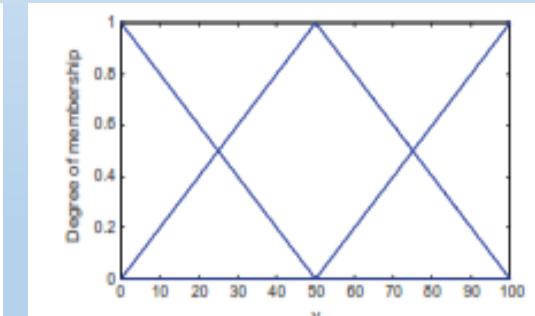


# Control del Tiempo de Lavado

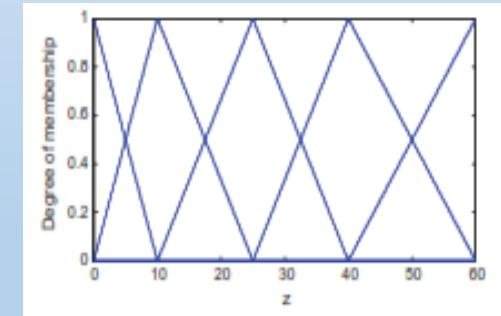
- Dos mediciones que corresponden a lodo y aceite y se controla el tiempo de lavado en función del grado de lodo y aceite de la pieza.
- Se consideran 3 niveles entre 0 y 100 para cada tipo de suciedad de la pieza. Para el lodo: SD poco lodoso, MD medio lodoso, LD muy lodoso, y para el aceite con VS poco aceitoso, MG medio aceitoso y LG muy aceitoso.
- Para el tiempo que es la variable a controlar (lazo abierto) se tienen cuatro niveles en el intervalo de [0-60]: VS muy corto, S corto, L largo, muy largo.
- Las MF consideradas son:



Lodoso



Aceitoso



Tiempo



# Diseño de Reglas

Washing time, $z$		Mud, $x$		
Axunge, $y$	NG	VS*	MD	LD
	MG	S	M	L
	LG	M	L	VL

Modelo

Tomando en cuenta las MF se puede ver que hay casos que corresponden a tiempos muy pequeño. Por ejemplo

$$\text{Si } x=60, y=70, x=\frac{60}{0} + \frac{60}{\cancel{4}/5} + \frac{60}{\cancel{1}/5}; y=\frac{70}{0} + \frac{70}{\cancel{3}/5} + \frac{70}{\cancel{2}/5}$$

$$\begin{aligned}\mu(z) &= \max \left\{ \min\left(\frac{3}{5}, \mu_M(z)\right), \min\left(\frac{1}{5}, \mu_L(z)\right), \min\left(\frac{2}{5}, \mu_L(z)\right), \min\left(\frac{1}{5}, \mu_{VL}(z)\right) \right\}, \\ &= \max \left\{ \min\left(\frac{3}{5}, \mu_M(z)\right), \min\left(\frac{2}{5}, \mu_L(z)\right), \min\left(\frac{1}{5}, \mu_{VL}(z)\right) \right\}\end{aligned}$$

$$\mu_M(z) = \frac{3}{5} \Rightarrow z_1 = 19, z_2 = 31 \Rightarrow z_{av} = 25$$

Washing time, $z$		Mud, $x$		
Axunge, $y$	NG	SD	MD (4/5)	LD (1/5)
	MG (3/5)	0	$\min\left(\frac{3}{5}, \mu_M(z)\right)$	$\min\left(\frac{1}{5}, \mu_L(z)\right)$
	LG (2/5)	0	$\min\left(\frac{2}{5}, \mu_L(z)\right)$	$\min\left(\frac{1}{5}, \mu_{VL}(z)\right)$

R1 : IF SD&NG THEN VS

R2: IF SD&S THEN S

:

R9 : IF LD&LG THEN VL

Washing time, $z$		Mud, $x$		
Axunge, $y$	NG	SD	MD (4/5)	LD (1/5)
	MG (3/5)	0	$\mu_M(z)$	$\mu_L(z)$
	LG (2/5)	0	$\mu_L(z)$	$\mu_{VL}(z)$

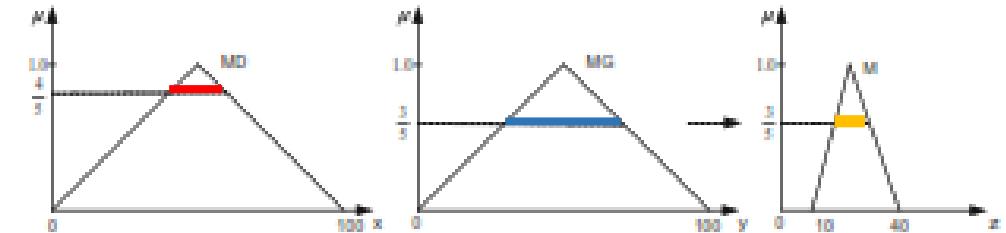
Washing time, $z$		Mud, $x$		
Axunge, $y$	NG	SD	MD (4/5)	LD (1/5)
	MG (3/5)	0	$\frac{3}{5}$	$\frac{1}{5}$
	LG (2/5)	0	$\frac{2}{5}$	$\frac{1}{5}$

# Inferencia Gráfica

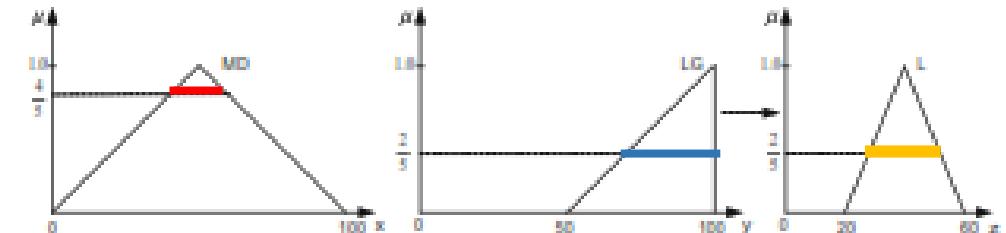
Washing time, $z$		Mud, $x$		
		SD	MD (4/5)	LD (1/5)
Axunge, $y$	NG	0	0	0
	MG (3/5)	0	$\mu_M(z)$	$\mu_L(z)$
	LG (2/5)	0	$\mu_L(z)$	$\mu_{VL}(z)$

Washing time, $z$		Mud $x$		
		SD	MD (4/5)	LD (1/5)
Axunge, $y$	NG	0	0	0
	MG (3/5)	0	$\min\left(\frac{1}{3}, \mu_M(z)\right)$	$\min\left(\frac{1}{3}, \mu_L(z)\right)$
	LG (2/5)	0	$\min\left(\frac{2}{3}, \mu_L(z)\right)$	$\min\left(\frac{1}{3}, \mu_{VL}(z)\right)$

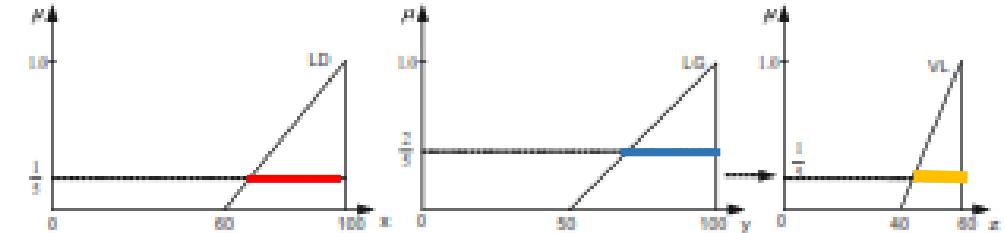
Washing time, $z$		Mud, $x$		
		SD	MD (4/5)	LD (1/5)
Axunge, $y$	NG	0	0	0
	MG (3/5)	0	3/5	1/5
	LG (2/5)	0	2/5	1/5



(a) First fuzzy rule



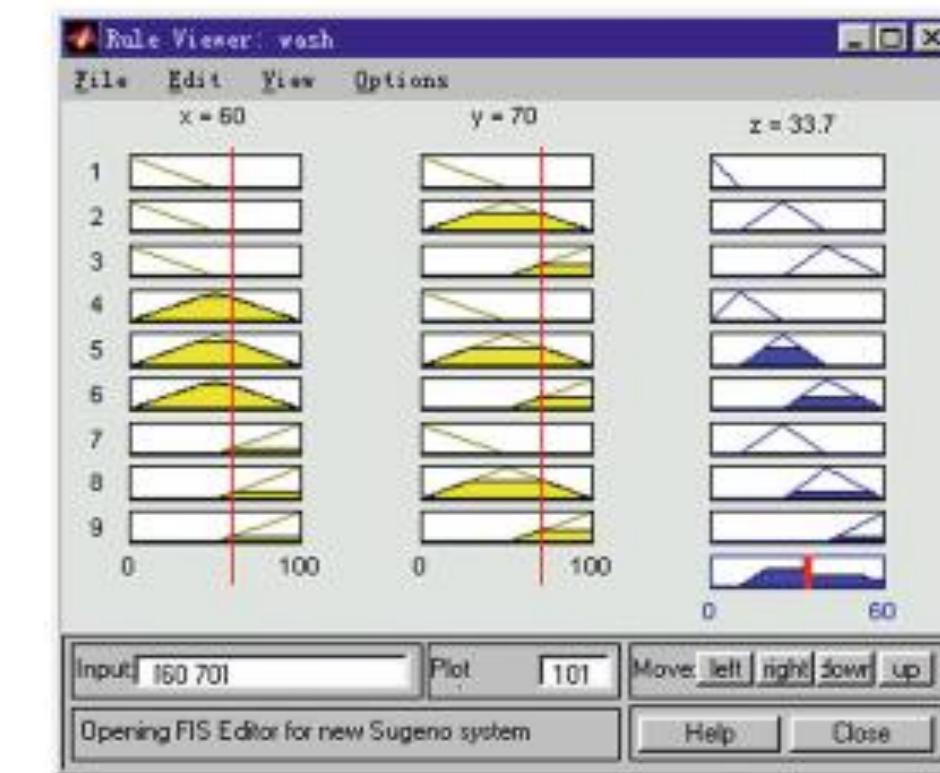
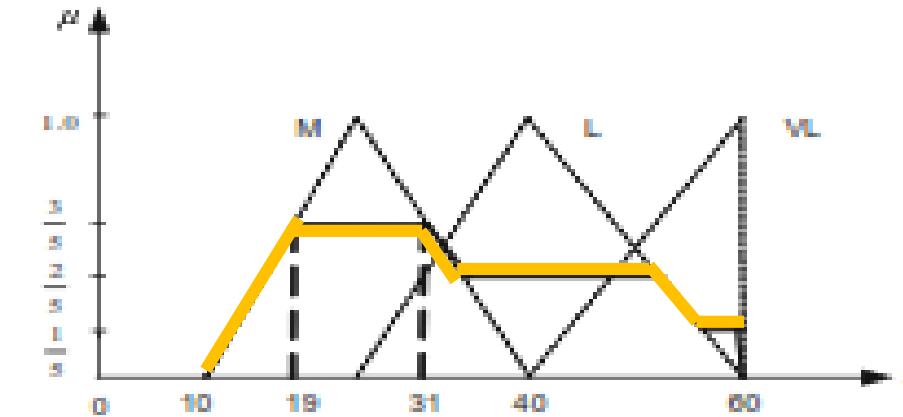
(b) Second fuzzy rule



(c) Third fuzzy rule



# Cuantificación Global





# Tarea 2 . Control Borroso del Tanque

- Considere el modelo lineal del tanque de agua

$$\frac{H(s)}{Q(s)} = \frac{20}{s+1} \text{ en el punto de operación } H_0 = 1 \text{ [m]; } Q_0 = 0.05 \left[ \frac{m^3}{s} \right]$$

con un nivel deseado de 1 [m] y en donde  $Q(s)$  negativo implica que se extrae flujo por la válvula y positivo que se introduce agua por arriba.

- Implementar el controlador borroso presentado en el cap. 3 del libro con un periodo de muestreo  $T=0.05$  [s].
- Considere los valores iniciales por equipos

Equipo 1	Equipo 2	Equipo 3
[2.5 3.2 0.5];	[1.5 0.5 3.];	[2.7 0 0.7]

y simule el comportamiento del nivel del tanque con el control borroso.

- ¿Le parecen satisfactorio el desempeño del controlador obtenido?
- ¿Cómo sugiere mejorar el desempeño de la regulación de nivel?



# Sintonización del PI con Lógica Borrosa

- Considere la ley de control

$$u(k) = k_p e(k) + k_i T \sum_{j=0}^k e(j) \quad \text{con } T \text{ el periodo de muestro, } e(k) = y_d(k) - y(k)$$

$$ec(k) = \Delta e(k) = e(k) - e(k-1)$$

- Regla para la parte proporcional

$ec$	N	Z	P
e			
N	N	N	N
Z	N	P	P
P	P	P	P

$$e(k) > 0 \Rightarrow \Delta k_p > 0$$

$$e(k) < 0 \Rightarrow \Delta k_p < 0$$

$$e(k) \approx 0 \exists \begin{cases} \Delta e < 0 \uparrow \text{sobreceso} \Rightarrow \Delta k < 0 \\ \Delta e \approx 0 \Rightarrow \Delta k > 0 \\ \Delta e > 0 \Rightarrow \Delta k > 0 \end{cases}$$

- Regla para la parte integral

$ec$	N	Z	P
e			
N	Z	Z	Z
Z	P	P	P
P	Z	Z	Z

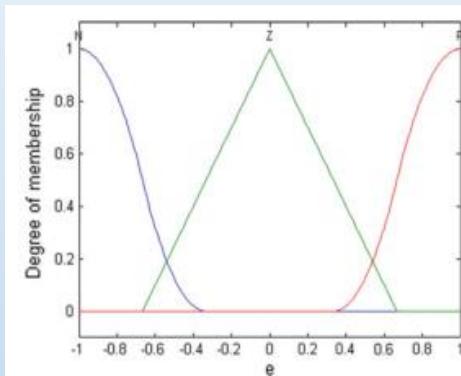
Considerando valida la técnica de separación se elige:

- Para e pequeños y cualquier ec se mantiene constante la ganancia  $k_i$
- Para e muy pequeños y cualquier ec se reduce la ganancia integral.

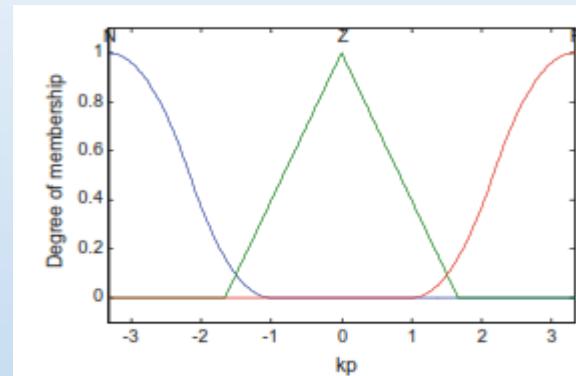


# Ejemplo del PI Borroso

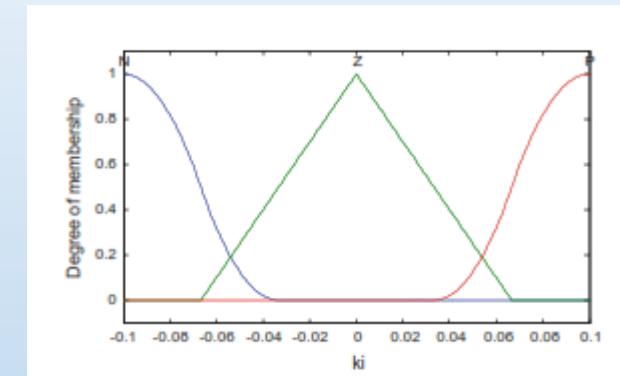
Sea:  $G(s) = \frac{133}{s^2 + 25s}$ ;  $T = 0.001$  [s] con  $y_r = 1.0$ . Diseñe un controlador PI borroso considerando las 9 reglas establecidas anteriormente.



MF del error & su derivada



MF de kp

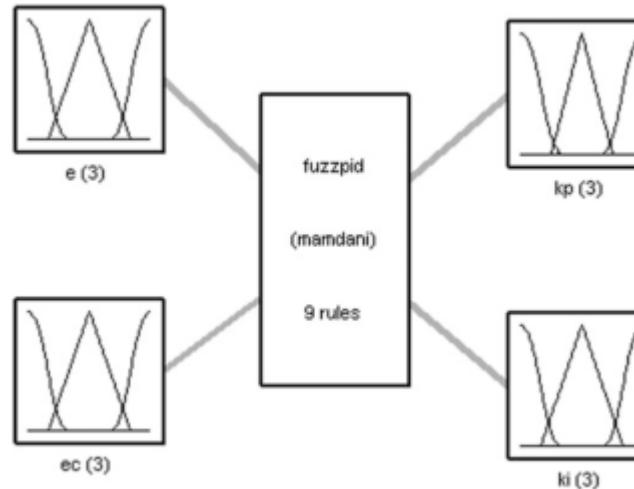


MF de ki

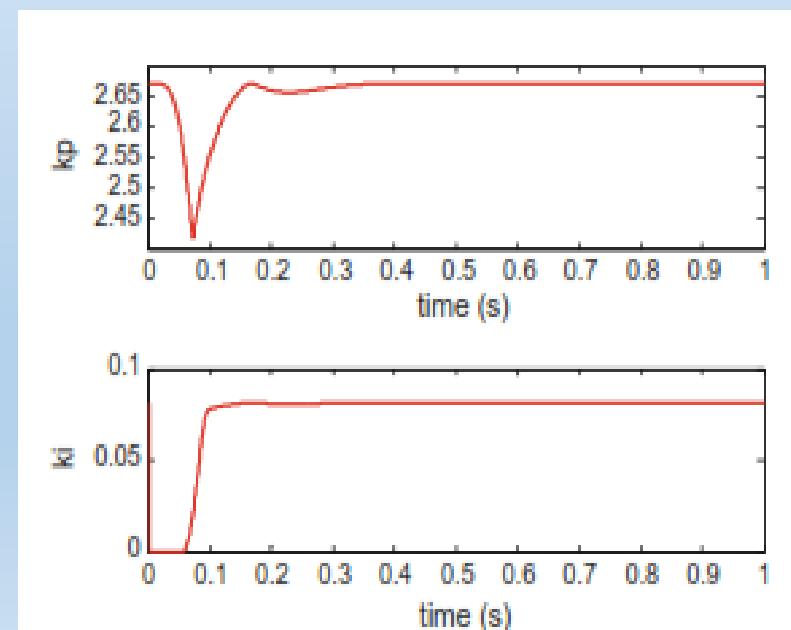
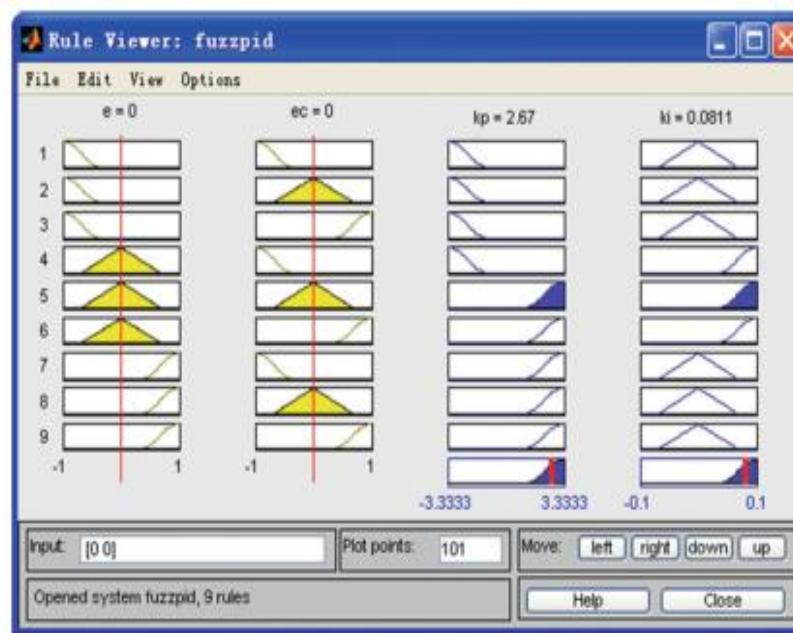
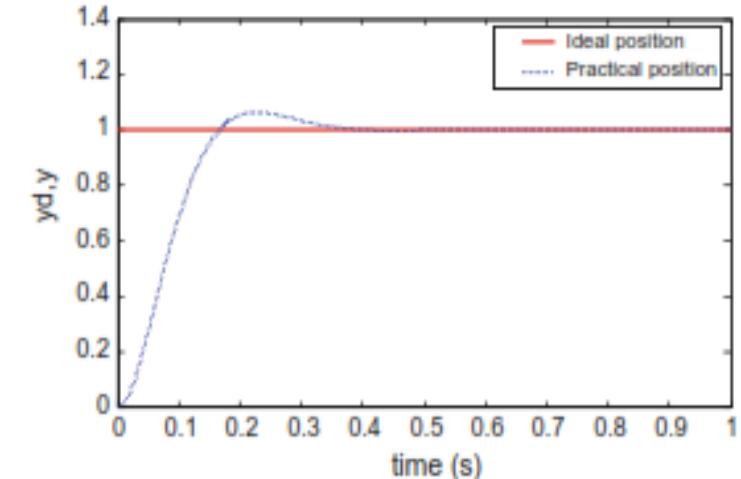
1. If ( $e$  is N) and ( $ec$  is N) then ( $kp$  is N)( $ki$  is Z) (1)
2. If ( $e$  is N) and ( $ec$  is Z) then ( $kp$  is N)( $ki$  is Z) (1)
3. If ( $e$  is N) and ( $ec$  is P) then ( $kp$  is N)( $ki$  is Z) (1)
4. If ( $e$  is Z) and ( $ec$  is N) then ( $kp$  is N)( $ki$  is P) (1)
5. If ( $e$  is Z) and ( $ec$  is Z) then ( $kp$  is P)( $ki$  is P) (1)
6. If ( $e$  is Z) and ( $ec$  is P) then ( $kp$  is P)( $ki$  is P) (1)
7. If ( $e$  is P) and ( $ec$  is N) then ( $kp$  is P)( $ki$  is Z) (1)
8. If ( $e$  is P) and ( $ec$  is Z) then ( $kp$  is P)( $ki$  is Z) (1)
9. If ( $e$  is P) and ( $ec$  is P) then ( $kp$  is P)( $ki$  is Z) (1)



# PI Borroso



System fuzzpid: 2 inputs, 2 outputs, 9 rules



# Modelo de Control Takagi-Sugeno

La diferencia de este modelo de control con respecto al de Mandami es que la salida de control no es borrosa sino que es constante o una función afín del estado. Es decir:

$$u = cte \quad \text{o} \quad u = mx + b$$

y por tanto es un valor exacto.

Por tanto este modelo es adecuado para sistemas continuos por pedazos (piecewise) como son los aviones, misiles, robots.

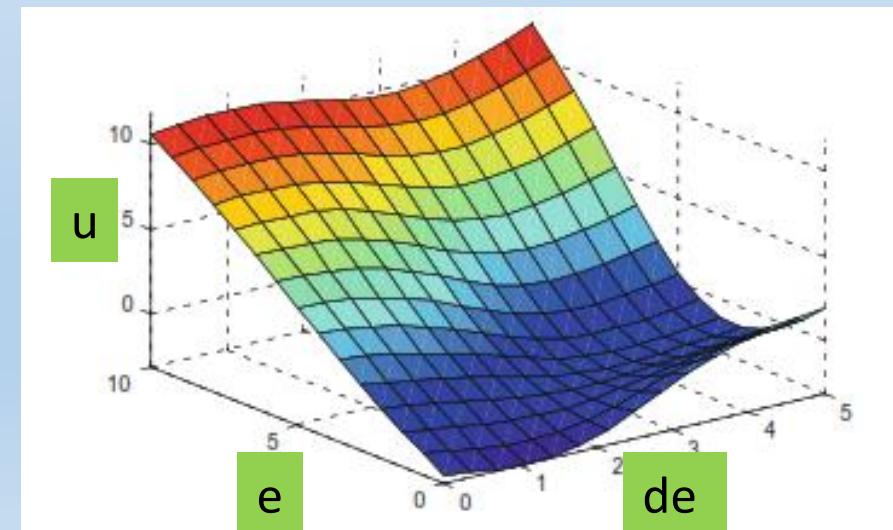
Ejemplo:

*IF e es P & de es N THEN  $u = 3e - 0.1de$*

*IF e es P & de es P THEN  $u = e + de$*

*:*

*IF e es Z & de es P THEN  $u = 0.0001de$*

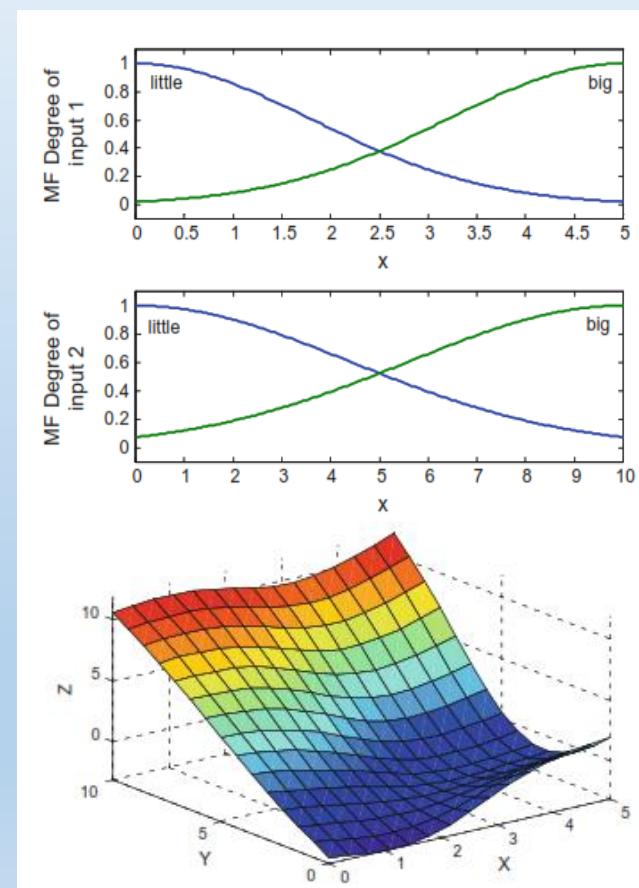




# Script

```
%T-S type fuzzy model
clear all;
close all;
ts2=newfis('ts2','sugeno'); Define el sistema ts2
ts2=addvar(ts2,'input','x',[0 5]); 1. entrada x [0 5]
ts2=addmf(ts2,1,'little','gaussmf',[1.8 0]); con mf Gaussiana
ts2=addmf(ts2,1,'big','gaussmf',[1.8 5]);
ts2=addvar(ts2,'input','y',[0 10]); 2. entrada y [0 10]
ts2=addmf(ts2,2,'little','gaussmf',[4.4 0]);con mf Gaussiana
ts2=addmf(ts2,2,'big','gaussmf',[4.4 10]);
ts2=addvar(ts2,'output','z',[-3 15]) 1. salida z [-3 15]
ts2=addmf(ts2,'output',1,'first area','linear',[-1 1 -3]); con la 1. área -x+y-3
ts2=addmf(ts2,'output',1,'second area','linear',[1 1 1]); con la 2. área x+y+1
ts2=addmf(ts2,'output',1,'third area','linear',[0 -2 2]);
ts2=addmf(ts2,'output',1,'fourth area','linear',[2 1 -6]);
rulelist=[1 1 1 1 1; Forma la matriz descripción de las reglas
    1 2 2 1 1;      con M+N+2 columnas, la 4. c es el peso
    2 1 3 1 1;      y la 5. con 1=and, 2=or
    2 2 4 1 1];
ts2=addrule(ts2,rulelist); suma reglas a ts2
showrule(ts2);
figure(1);
subplot 211;
plotmf(ts2,'input',1);
xlabel('x'),ylabel('MF Degree of input 1');
subplot 212;
plotmf(ts2,'input',2);
xlabel('x'),ylabel('MF Degree of input 2');
figure(2);
gensurf(ts2);
xlabel('x'),ylabel('y'),zlabel('z');
```

If  $X$  is small and  $Y$  is small then  $Z = -x + y - 3$   
If  $X$  is small and  $Y$  is big then  $Z = x + y + 1$   
If  $X$  is big and  $Y$  is small then  $Z = -2y + 2$   
If  $X$  is big and  $Y$  is big then  $Z = 2x + y - 6$





# Objetivo

Encontrar una ley de control que asigne la dinámica del proceso

$$\dot{x} = Ax + Bu$$

cuando

- Se usa una ley de control de tipo  $u = f(x)$
- Se considera disponible todo el vector de estado para LTIS
- Se asume que la pareja  $(A, B)$  es controlable

Chen (1984) Ref: Chi-Tsong Chen. Linear Systems Theory and Design, 1984



# Descripción del sistema con una entrada

Sea  $\dot{x} = Ax + bu$ , el cual puede describirse como

$$\dot{\bar{x}} = \bar{A}\bar{x} + \bar{b}\bar{u}$$

donde  $\bar{x} = Px$  y  $P^{-1} = Q = ( q_1 \ q_2 \ \dots \ q_n )$ . Si  $Q$  está formada por los vectores de la matriz de controlabilidad

$$q_n = b$$

$$q_{n-i} = Aq_{n-i+1} + \alpha_i q_n$$

donde  $i = 1, 2, \dots, n-1$ ,  $\det(sI - A) = s_n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} + \alpha_{n-1} s + \alpha_n$  y las matrices están dadas por:

$$\bar{A} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -\alpha_n & -\alpha_{n-1} & -\alpha_{n-2} & \dots & -\alpha_2 & -\alpha_1 \end{pmatrix}; \quad \bar{b} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$



## Retroalimentación de Estados

Sea la ley de control

$$u = -\bar{K}\bar{x} + r, \quad *$$

entonces el sistema en lazo cerrado

$$\dot{\bar{x}} = (\bar{A} - \bar{b}\bar{K})\bar{x} + \bar{b}r$$

es controlable para cualquier vector  $K$  si el sistema original  $(A, B)$  es controlable.

Es decir, la controlabilidad de  $(A, B)$  es invariantes bajo una retroalimentación de la forma (\*).

Sin embargo, se puede perder la observabilidad, si existen polos asignados con la ley de control iguales a los ceros del sistema con  $y = \bar{C}\bar{x}$ .

Hecho: Si el sistema es controlable con (\*), entonces los valores propios de  $(\bar{A} - \bar{b}\bar{K})$  se pueden asignar arbitrariamente.

# Algoritmo

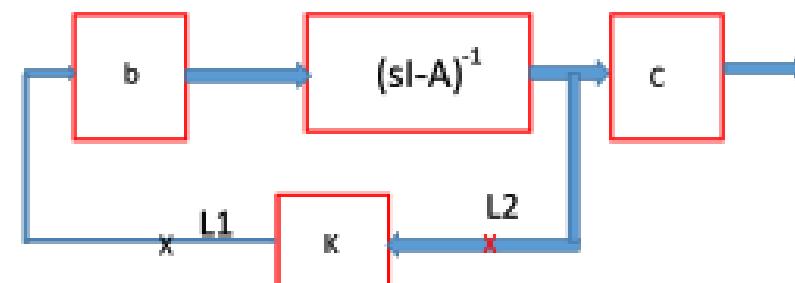
Sean los polos deseados:  $(\lambda_{1d}, \lambda_{2d}, \dots, \lambda_{nd})$

- ➊ Calcular  $\det(sI - A) = s^n + \alpha_1 s^{n-1} + \dots + \alpha_n$
- ➋ Calcular el polinomio deseado  
 $(s - \lambda_{1d})(s - \lambda_{2d}) \dots (s - \lambda_{nd}) = s^n + \bar{\alpha}_{1d}s^{n-1} + \dots + \bar{\alpha}_{nd}$
- ➌ Calcular la matriz  $Q$  y  $P = Q^{-1}$
- ➍  $\bar{K} = -(\alpha_n - \bar{\alpha}_{nd} \quad \alpha_{n-1} - \bar{\alpha}_{(n-1)d} \quad \alpha_{n-2} - \bar{\alpha}_{(n-2)d} \quad \dots \quad \alpha_1 - \bar{\alpha}_{1d})$
- ➎  $K = \bar{K}P$

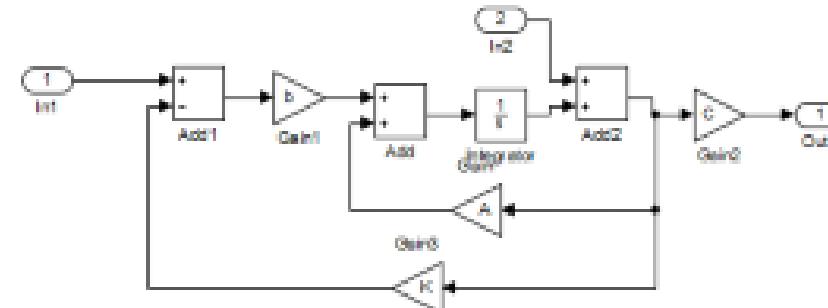
Propiedades:

$$I + L_1 = 1 + K(sI - A)^{-1}b = \frac{\det(sI - A + bK)}{\det(sI - A)}; \quad I + L_2 = (sI - A)^{-1}bK$$

$$S_1 = (I + L_1)^{-1} = (1 + K(sI - A)^{-1}b)^{-1} \quad S_2 = (I + L_2)^{-1}$$



# Propiedades del Control



;

$$E(s) = S_1(s)R(s), \quad Y_r(s) = C(sI - A)^{-1}bE(s)$$

$$\begin{aligned} Y_r(s) &= C(sI - A)^{-1}bS_1(s)R(s) = C(sI - A - bK)^{-1}bR(s) = \\ &= C(sI - A)^{-1}b \frac{\det(Aa(s))}{\det(Ac(s))} R(s) = \frac{Cadj(sI - A)b}{\det(Ac(s))} R(s) \end{aligned}$$

$$Y_d(s) = (I + (sI - A)^{-1}bK)^{-1}D(s) = S_2(s)D(s)$$

Matriz de R a U :  $T_1 = (1 + K(sI - A)^{-1}b)^{-1}K(sI - A)^{-1}b$

Matriz de R a X:  $T_2 = (I + (sI - A)^{-1}bK)^{-1}(sI - A)^{-1}bK$

- El sistema tiene los polos en el polinomio característico deseado
- Los ceros del sistema no se modifican.



# Control T-S para un Sistema Nolineal con Asignación de Polos

Sea un sistema no lineal descrito por pedazos como un sistema lineal

$$\dot{x}(t) = A_i x(t) + B_i u(t) \quad \forall i = 1, \dots, r$$

con  $x(t) = [x_1 \quad \dots \quad x_n]^T$ ,  $A_i$  una matriz de  $(n \times n)$ ,  $B_i$  un vector de  $(n \times 1)$  y cada estado borroso  $x_j$  tiene  $M_j^i$  como MF.

La regla T-S para cada  $i$  está dada por

IF  $x_1(t) \in M_1^i$  and  $x_2(t) \in M_2^i, \dots$  and  $x_n(t) \in M_n^i$ , THEN

$$\dot{x}(t) = A_i x(t) + B_i u(t)$$

La retroalimentación de estado del sistema no lineal  $u(t)$  se reduce a la regla

IF  $x_1(t) \in M_1^r$  and  $x_2(t) \in M_2^r$  and  $\dots$   $x_n(t) \in M_n^r$  THEN  $u(t) = K_i x(t)$  para cada  $i = 1, \dots, r$



# Compensación Distribuida Paralela PDC

Ejemplo:

Para 4 reglas la premisa del estado  $x_1$ , las funciones de pertenencia son:

$$w_1 = M_1^1(x_1), \quad w_2 = M_1^2(x_1), \quad w_3 = M_1^3(x_1), \quad w_4 = M_1^4(x_1),$$

La salida del sistema borroso se puede escribir como

$$\dot{x}(t) = \frac{\sum_{i=1}^r w_i (A_i x(t) + B_i u(t))}{\sum_{i=1}^r w_i} \quad ***$$

con  $w_i = \prod_{k=1}^n M_k^i(x_k(t))$  la MF de la regla  $i$ .

y por tanto para el modelo \*\*\* la retro de estado se reduce a  $u(t) = \frac{\sum_{i=1}^r w_i K_i x(t)}{\sum_{i=1}^r w_i}$



# PID con Takagi-Sugeno Reglas

Semana 8

Sistema de aire acondicionado



# Tarea del Control T-S para el tanque

- Considere el modelo no lineal del tanque con  $h(t)$  el nivel y

$$S \frac{dh(t)}{dt} + V_2 \sqrt{h(t)} = V_1 q_e(t)$$

con  $V_1$  % de apertura de la válvula de entrada

y  $V_2$  % de apertura de la válvula de salida

$q_e(t) = 1$  el flujo de entrada

$q_s(t) = V_2 \sqrt{h(t)}$  el flujo de salida

$S = 1$  el área transversal del tanque

$$e(t) = h_0(t) - h(t)$$

- Implementar el controlador borroso T-S con las siguientes leyes de control:

$$V_1 = \begin{cases} 0.25q_e(t) & \text{si } 0 < e(t) < 0.1 \\ 0.5q_e(t) & \text{si } 0.1 < e(t) < 0.5 \\ 0.75q_e(t) & \text{si } 0.5 < e(t) < 1 \\ q_e(t) & \text{si } 1 < e(t) \end{cases}$$

$$V_2 = \begin{cases} 0.25\sqrt{h(t)} & \text{si } -0.1 < e(t) < 0 \\ 0.5\sqrt{h(t)} & \text{si } -0.5 < e(t) < -0.1 \\ 0.75\sqrt{h(t)} & \text{si } -1 < e(t) < -0.5 \\ \sqrt{h(t)} & \text{si } e(t) < -1 \end{cases}$$



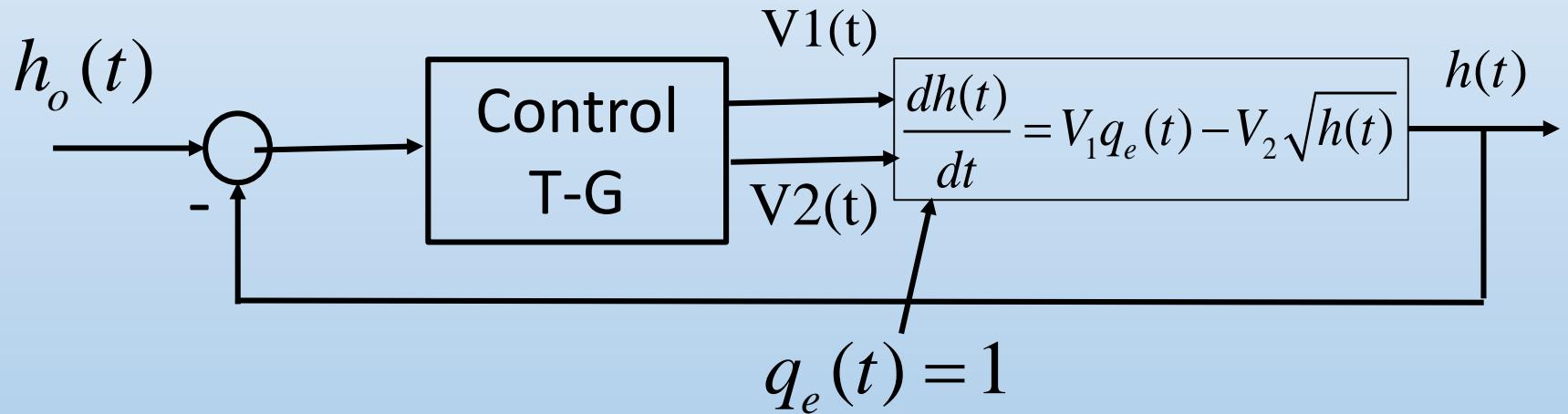
# Control T-S para el tanque

Entregar el 17 de noviembre

Seleccione una pareja fija de leyes de control

$$V1(t), V2(t) \text{ con } q_e=1.$$

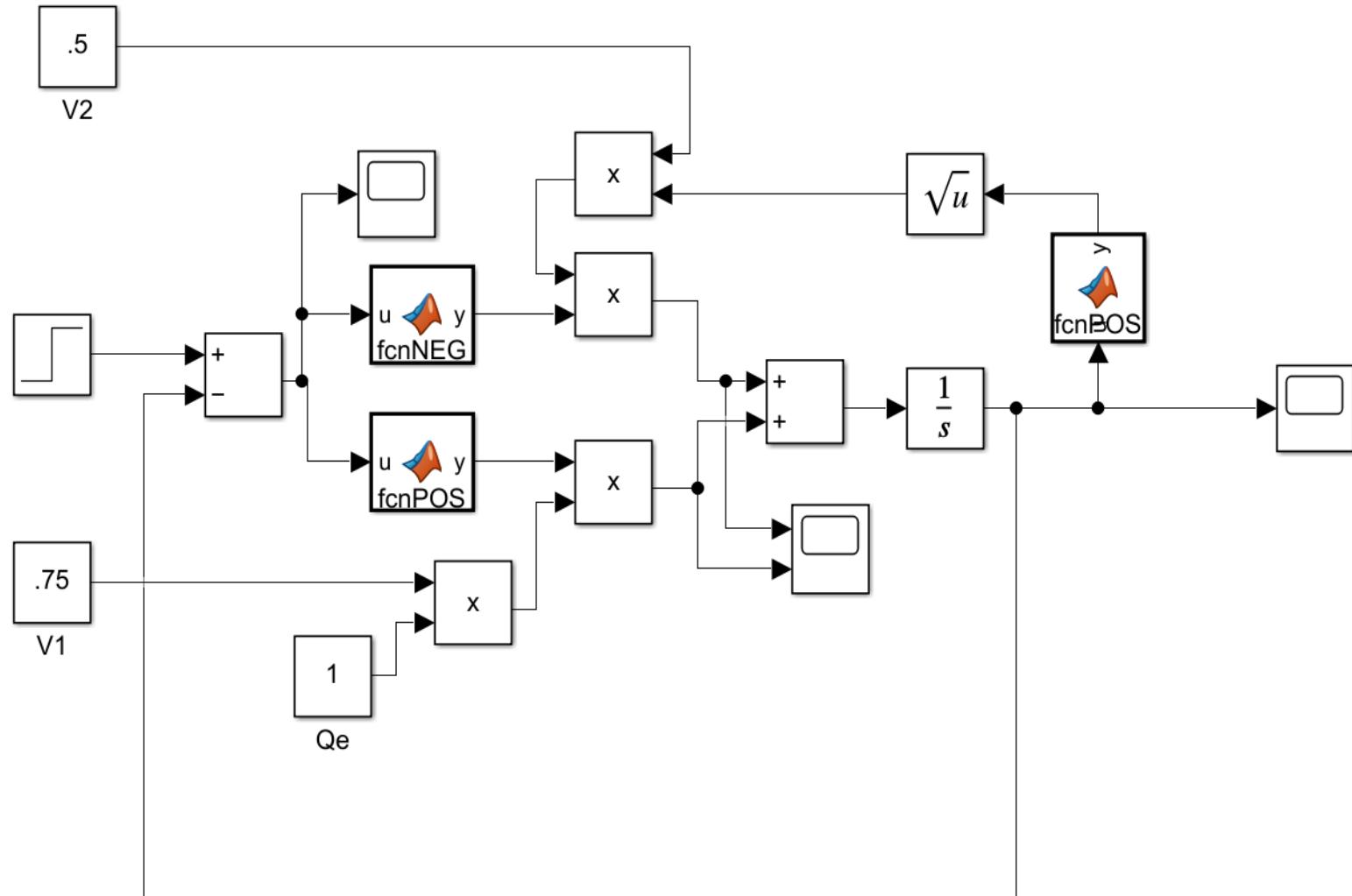
y compare el desempeño con el controlador T-S



El archivo de simulink del modelo se llama [Tarea3BorrosoT\\_S.mdl](#)

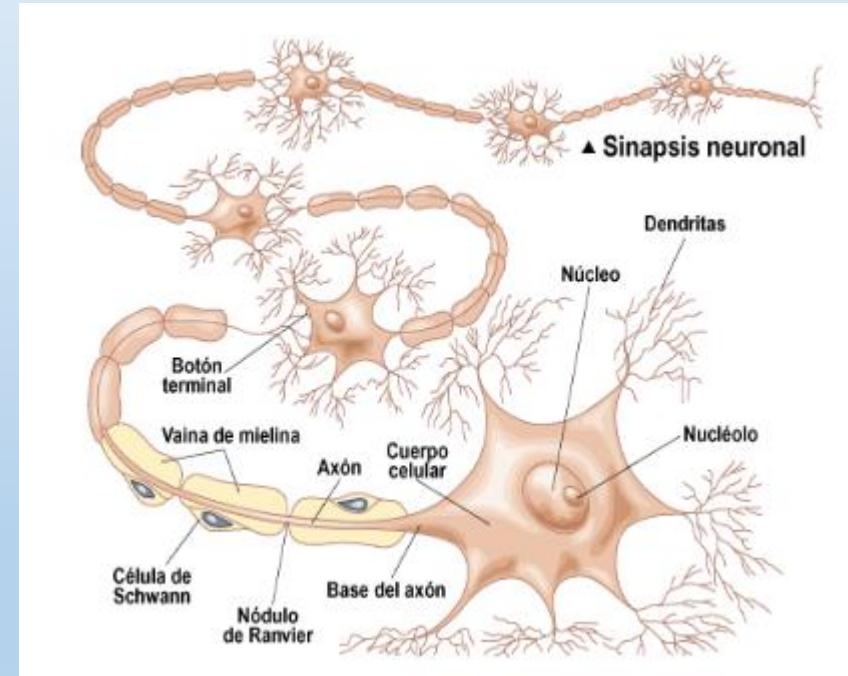


# Modelo no lineal del tanque



# Red Neuronal Biológica, RNB

Una neurona biológica es una célula especializada que transmite información a otros células o neuronas (señales electro-químicas). Las neuronas tienen muchas ramificaciones para conectarse a otras neuronas (dendritas=entradas y axones=salidas) y la información Es codificada en una taza de pulso o de disparo.

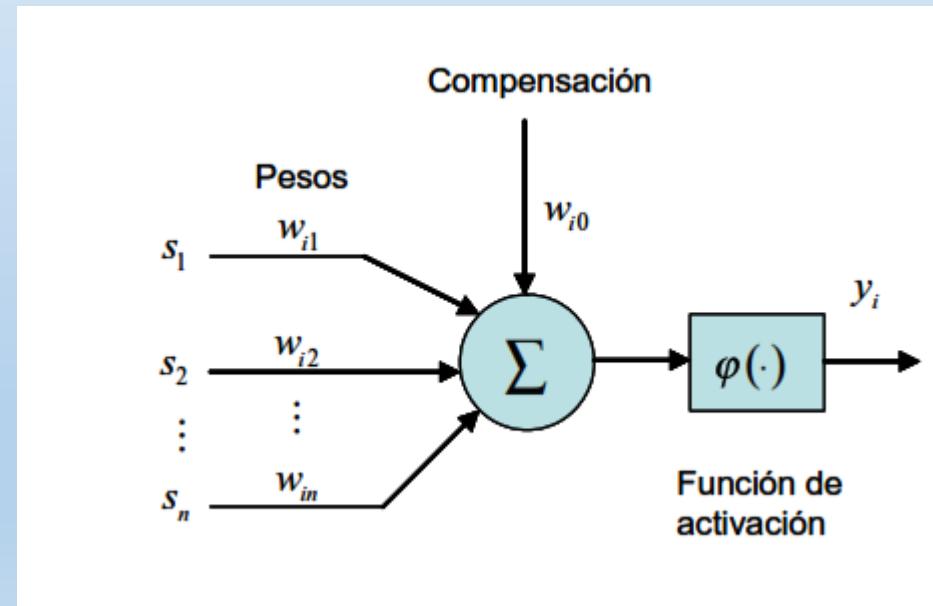




# Redes Neuronales Artificiales

Las redes neuronales artificiales son un modelo simplificado de RNB y se clasifican dentro del grupo de sistemas inteligentes. Se dicen inteligentes por tener la capacidad de adaptarse al medio o capacidad de entrenamiento. Están formadas por elementos simples interconectados con una cierta estructura. Como son las neuronas perceptrones (Rosenblatt 1962)

Se usan para clasificadores y reconocer patrones o generar funciones.



En control se usan como modelos para la identificación o el control de sistemas



# Redes Neuronales

1943 McCulloch & Pitts propusieron el modelo de computación una neurona.

1949 Hebb's The organization of Behavior.

1948 Wiener and Rosenblueth Cybernetic: or Control and Communication in the Animal and the Machine.

1957 John von Neumann. The computer and the brain.

1962 Rosenblatt: Neurona = el perceptron.

1982 Red Hopfield.

1986 Back propagation, retropropagación BP-NN.

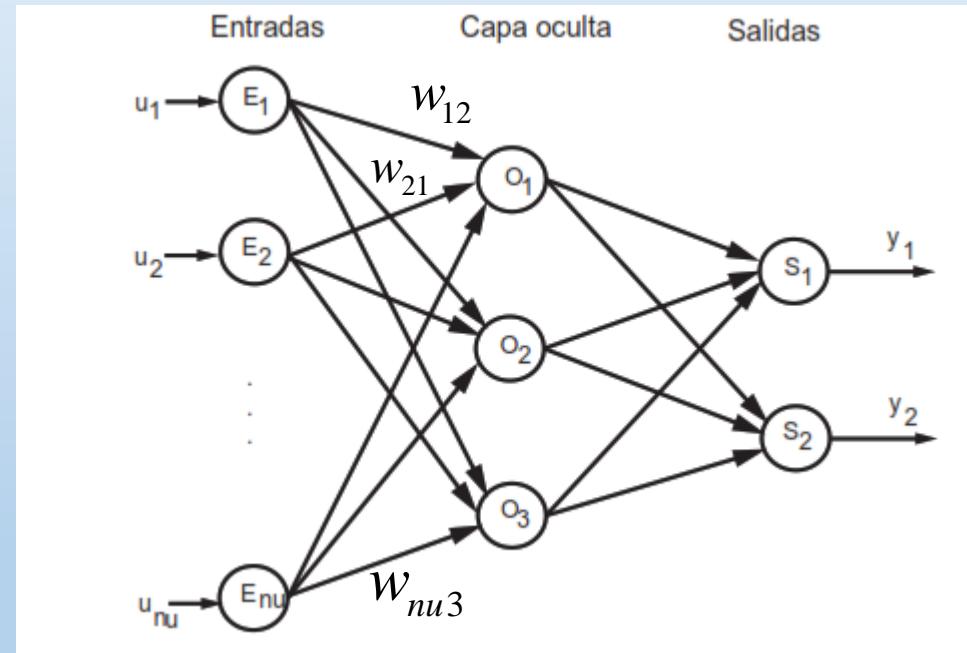
1988 RBF Radial back propagation. Una función radial. Cualquier función puede aproximarse con una RBF.



# Modelo Simple de una Red Neuronal

$$Net_i = \sum_j w_{ij}x_j + s_i - \theta_i,$$

$$y_i = f(Net_i),$$

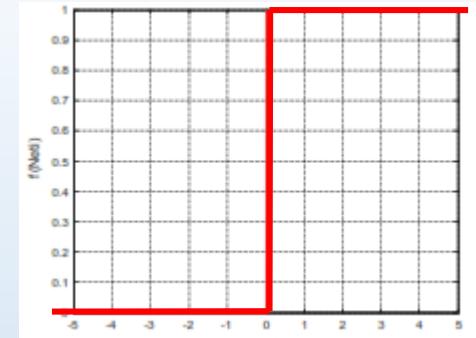


Estructura de una RN Retropropagación (NN Backpropagation)

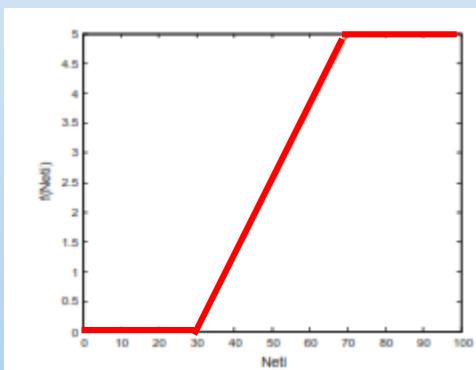


# Función característica nolineal

Valor de Umbral:  $f(Net_i) = \begin{cases} 1 & Net_i > 0 \\ 0 & Net_i \leq 0 \end{cases}$

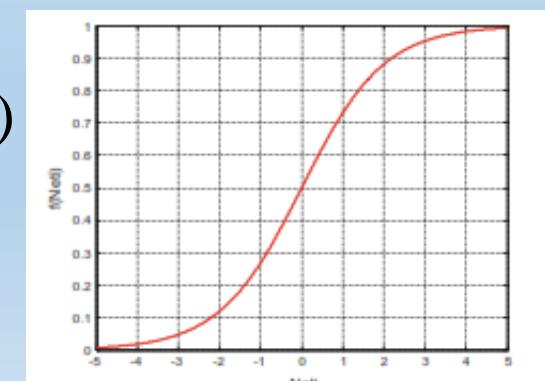


Función lineal:  $f(Net_i) = \begin{cases} 0 & Net_i \leq Net_{i0} \\ k_{Neti} & Net_{i0} < Net_i < Net_{i1} \\ f_{\max} & Net_i \geq Net_{i1} \end{cases}$



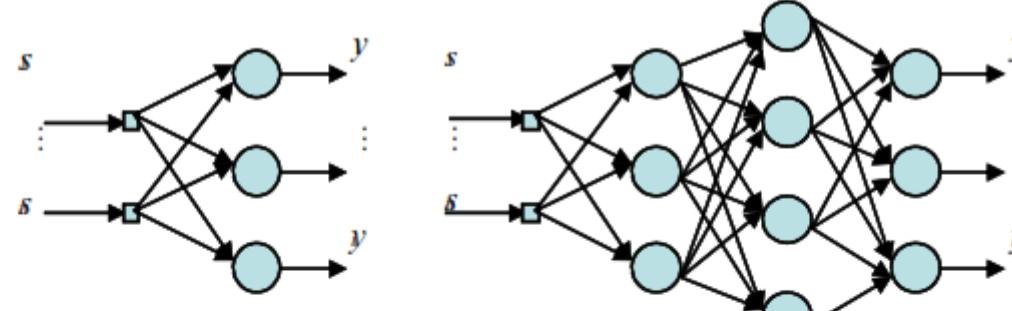
Función No lineal (Sigmoidal o Gaussiana)

$$f(Net_i) = \frac{1}{1 + \exp(-\frac{Net_i}{T})}$$

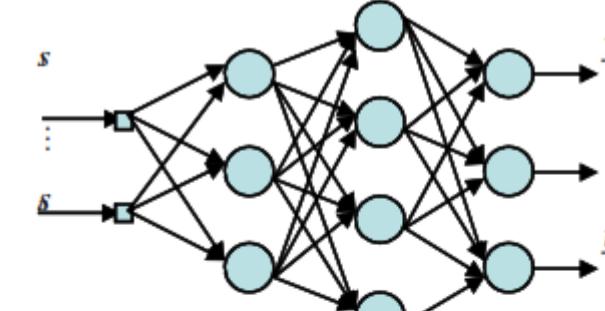




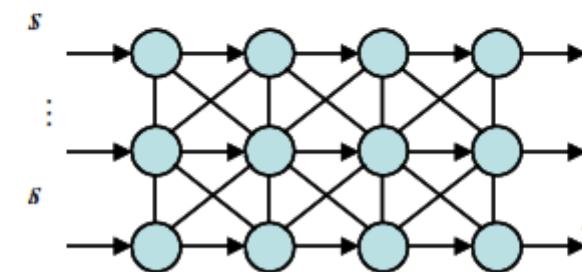
# Estructuras de las RN



Una sola capa

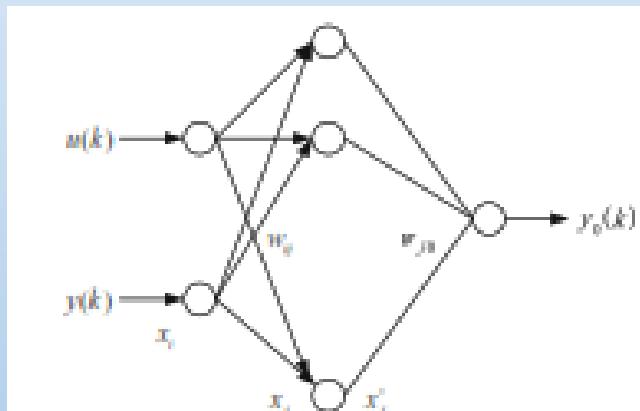
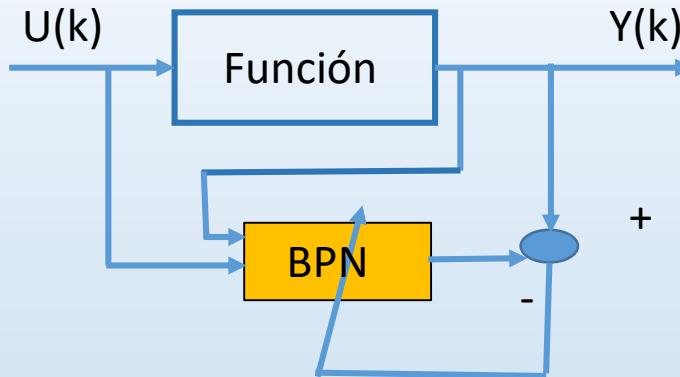


Múltiples capas



Malla

# Aproximación de la BPN



Red como aproximador

1) Alimentación hacia adelante

$$x_j = \sum_i w_{ij} x_i \quad \text{con} \quad x_j' = f(x_j) = \frac{1}{1 + \exp(-x_j)}$$

$$\frac{\partial(x_j')}{\partial x_j} = \frac{\exp(-x_j)}{(1 + \exp(-x_j))^2} = x_j'(1 - x_j')$$

2) La capa de salida  $y_o(k) = \sum_j w_{jo} x_j'$

3) El error  $y(k) - y_n(k)$

4) La función a minimizar  $J(k) = \frac{1}{2} e^2(k)$

5) Aprendizaje  $w_{jo}(k+1) = w_{jo}(k) + \Delta w_{jo}(k)$

$$\begin{aligned} \Delta w_{jo}(k) &= \eta \frac{\partial E(k)}{\partial w_{jo}(k)} = \eta e(k) \frac{\partial y_o(k)}{\partial w_{jo}(k)} \\ &= \eta e(k) x_j' \end{aligned}$$



# Continua la aproximación de la BPN

6) Ajuste de la etapa primera

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij}(k)$$

$$\Delta w_{ij}(k) = -\eta \frac{\partial E(k)}{\partial w_{ij}(k)} = -\eta e(k) \frac{\partial y_o(k)}{\partial w_{ij}(k)}$$

usando la regla de la cadena

$$\frac{\partial y_o(k)}{\partial w_{ij}(k)} = \frac{\partial y_o(k)}{\partial x_j'(k)} \frac{\partial x_j'(k)}{\partial x_j(k)} \frac{\partial x_j(k)}{\partial w_{ij}(k)} = w_{jo} \frac{\partial x_j'(k)}{\partial x_j(k)} x_i(k)$$

con  $\frac{\partial(x_j')}{\partial x_j} = \frac{\exp(-x_j)}{(1 + \exp(-x_j))^2} = x_j'(1 - x_j')$

$$= w_{jo} x_j'(k)(1 - x_j'(k)) x_i(k)$$



# Resumen

## Última capa

$$w_{jo}(k+1) = w_{jo}(k) + \Delta w_{jo}(k) + \alpha(w_{jo}(k) - w_{jo}(k-1))$$

## Primera capa

$$w_{ij}(t-1) = w_{ij}(t) + \Delta w_{ij}(t) + \alpha(w_{ij}(t) - w_{ij}(t-1))$$

donde  $\eta \in [0-1]$  es la razón de aprendizaje y

$\alpha \in [0-1]$  es el factor de la velocidad del algoritmo

Con el Jacobiano de la función aproximada:

$$\frac{\partial y(k)}{\partial u(k)} \approx \frac{\partial y_o(k)}{\partial u(k)} = \sum_j w_{jo} \dot{x}_j (1 - \dot{x}_j) w_{1j}$$



# Ejemplo de BP NN

Semana 10

- Considere la función (planta)

$$y(k) = u(k)^3 + \frac{y(k-1)}{1 + y(k-1)^2} \quad \text{señal de entrada } u(k) = 0.5 \sin(6\pi k)$$

Sea la entrada a la red el vector  $[u(k) \ y(k)]$ ,

con  $\eta=0.5$ ,  $\alpha=0.05$  y valores aleatorios

de los pesos entre  $[-1 \ 1]$ .

# Esquema de aprendizaje (2,6,1)

Semana 10

$$u(k) = 0.5 \sin(6\pi t)$$

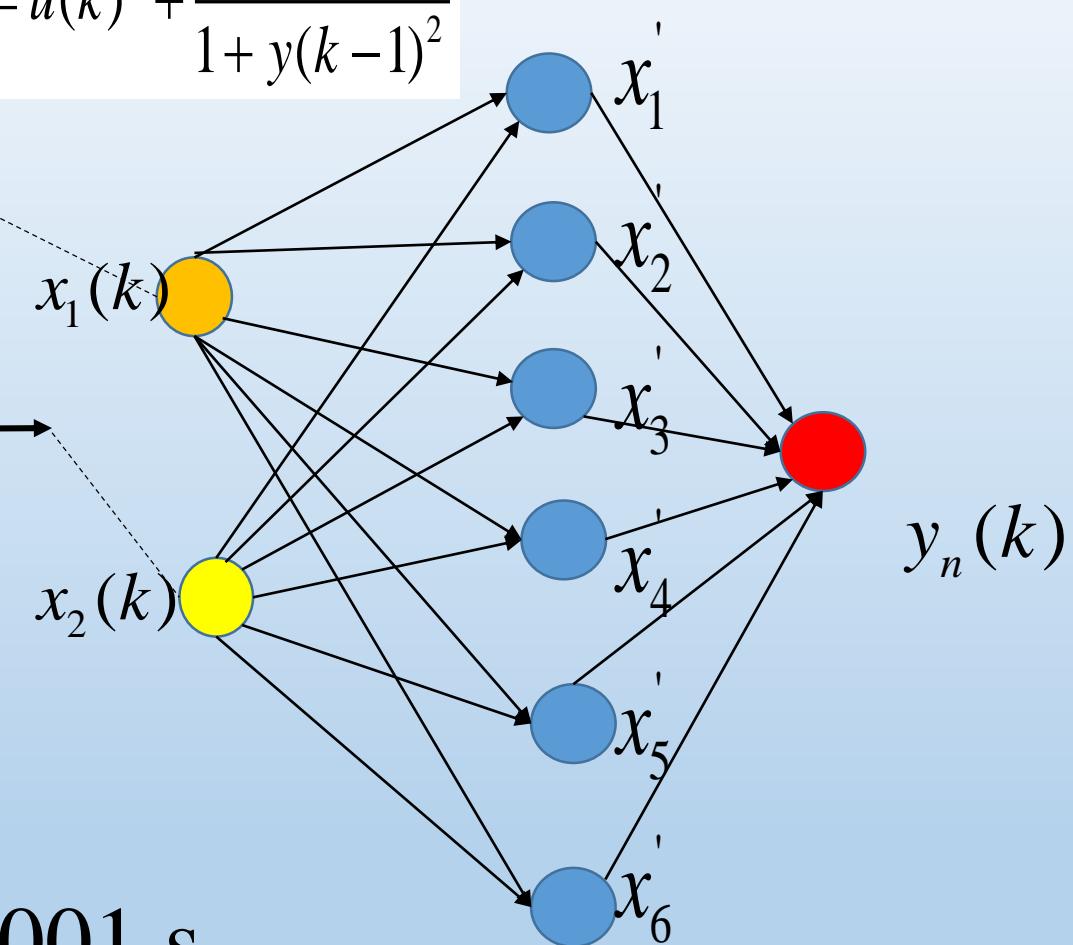
$$u^3$$

$$\frac{y}{1+y^2}$$

$$1/z$$

$$y(k) = u(k)^3 + \frac{y(k-1)}{1+y(k-1)^2}$$

$$y(k)$$



$$1/z = \text{retraso de } 0.001 \text{ s}$$



# Código

## Simulation program: chap7\_1.m

```
%BP approximation
clear all;
close all;

xite=0.50;
alfa=0.05;

wjo=rand(6,1);
wjo_1=wjo;wjo_2=wjo_1;

wij=rand(2,6);
wij_1=wij;wij_2=wij;

dwij=0*wij;

x=[0,0]';

u_1=0;
y_1=0;

I=[0,0,0,0,0,0]';
Iout=[0,0,0,0,0,0]';
FI=[0,0,0,0,0,0]';
```

Inicia variables

```
ts=0.001;
for k=1:1:1000

    time(k)=k*ts;
    u(k)=0.50*sin(3*2*pi*k*ts);
    y(k)=u_1^3+y_1/(1+y_1^2);

    x(1)=u(k);
    x(2)=y(k);

for j=1:1:6
    I(j)=x''*wij(:,j);
    Iout(j)=1/(1+exp(-I(j)));
end

yo(k)=wjo'*Iout;      % Output of NN1 networks

e(k)=y(k)-yo(k);    % Error calculation

wjo=wjo_1+(xite*e(k))*Iout+alfa*(wjo_1-wjo_2);
```

Calcula la  
la función a  
aproximar

```
for j=1:1:6
    FI(j)=exp(-I(j))/(1+exp(-I(j)))^2;
end

for i=1:1:2
    for j=1:1:6
        dwij(i,j)=e(k)*xite*FI(j)*wjo(j)*x(i);
    end
end

wij=wij_1+dwij+alfa*(wij_1-wij_2);
```

Calcula la  
Etapa OUT

Calcula la  
Etapa IN



# Jacobiano y Gráficas

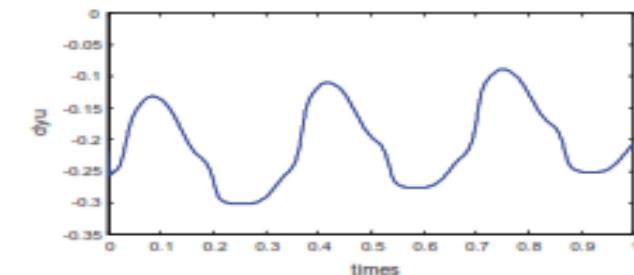
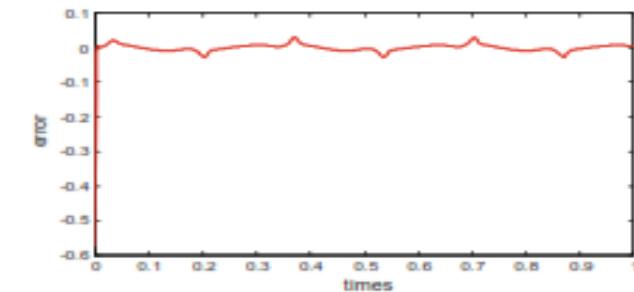
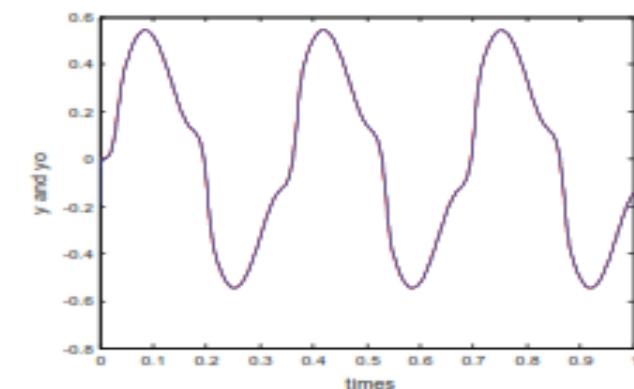
```
%Jacobiano
yu=0;
for j=1:1:6
    yu=yu+wjo(j)*wij(1,j)*FI(j);
end
dyu(k)=yu;

wij_2=wij_1;wij_1=wij;
wjo_2=wjo_1;wjo_1=wjo;
u_1=u(k);
y_1=y(k);
end

figure(1);
plot(time,y,'r',time,yo,'b');
xlabel('times');ylabel('y and yo');

figure(2);
plot(time,y-yo,'r');
xlabel('times');ylabel('error');

figure(3);
plot(time,dyu);
xlabel('times');ylabel('dyu');
```





# Tarea de BP

Cambiar el modelo

$$y(k) = u(k)^3 + \frac{y(k-1)}{1 + y(k-1)^2}$$

Por el modelo no lineal del tanque y seleccionando un muestro de 0.01s  
Entrenar la BP con el método del gradiente y suponer una señal de entrada senoidal.

$$h(k+1) = h(k) + 0.5q(k) - u(k)\sqrt{h(k)}$$

NO la pedí porque requería dos entradas



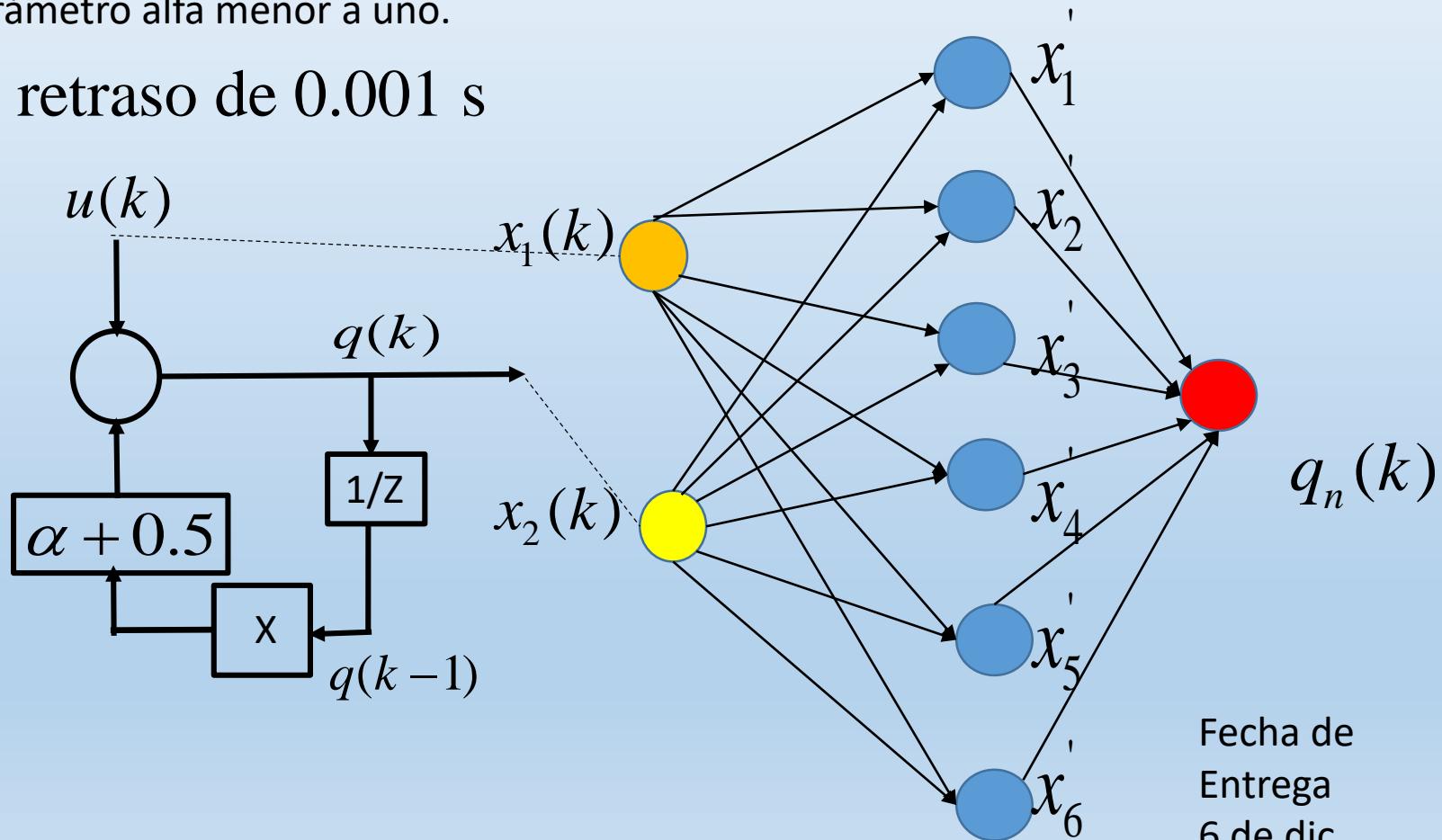
# Tarea de BP-NN (2,6,1)

Sea la función que corresponde al modelo de un ducto

Diseñe un BP-NN que genere dicho sistema con el parámetro alfa menor a uno.

$1/z =$  retraso de 0.001 s

$$q(k) = \alpha q(k-1)^2 + u(k)$$
$$u(k) = \sin(\pi 0.1(k-1))$$



# Neurona con Función de Base Radial RBF

Semana 11

- La diferencia entre BP y RBF es que la función de activación es una Gaussina en vez de la S simplificando el algoritmo y la carga computacional.

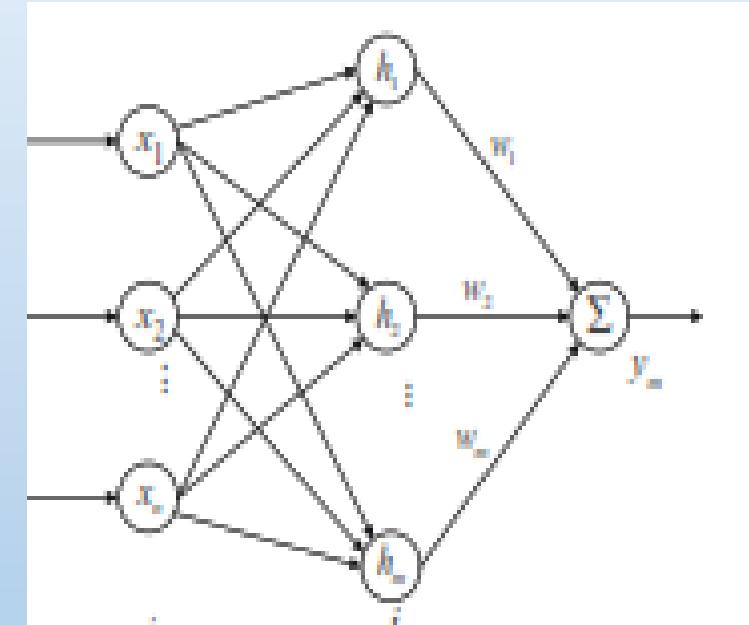
$$h_j = \exp\left(-\frac{\|x - c_j\|^2}{2b_j^2}\right)$$

con  $c = \begin{bmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \cdots & \vdots \\ c_{n1} & \cdots & c_{nm} \end{bmatrix}$  el valor de la media

de la FG de la RN j para la entrada  $i=1,\dots,n$   
b=ancho de la función Gaussiana

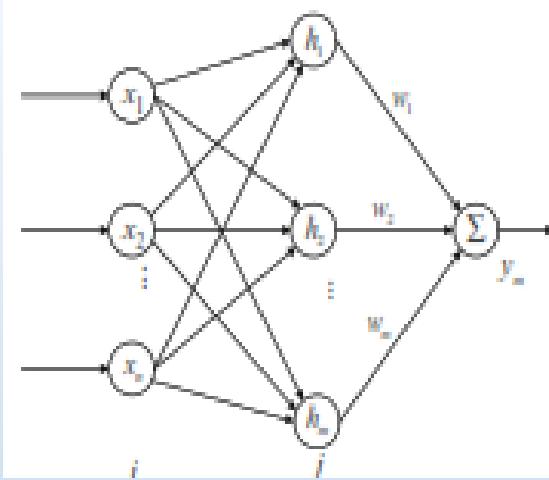
$$w = [w_1 \quad \cdots \quad w_m]^T$$

$$y = w^T h = w_1 h_1 + w_2 h_2 + \cdots + w_m h_m$$

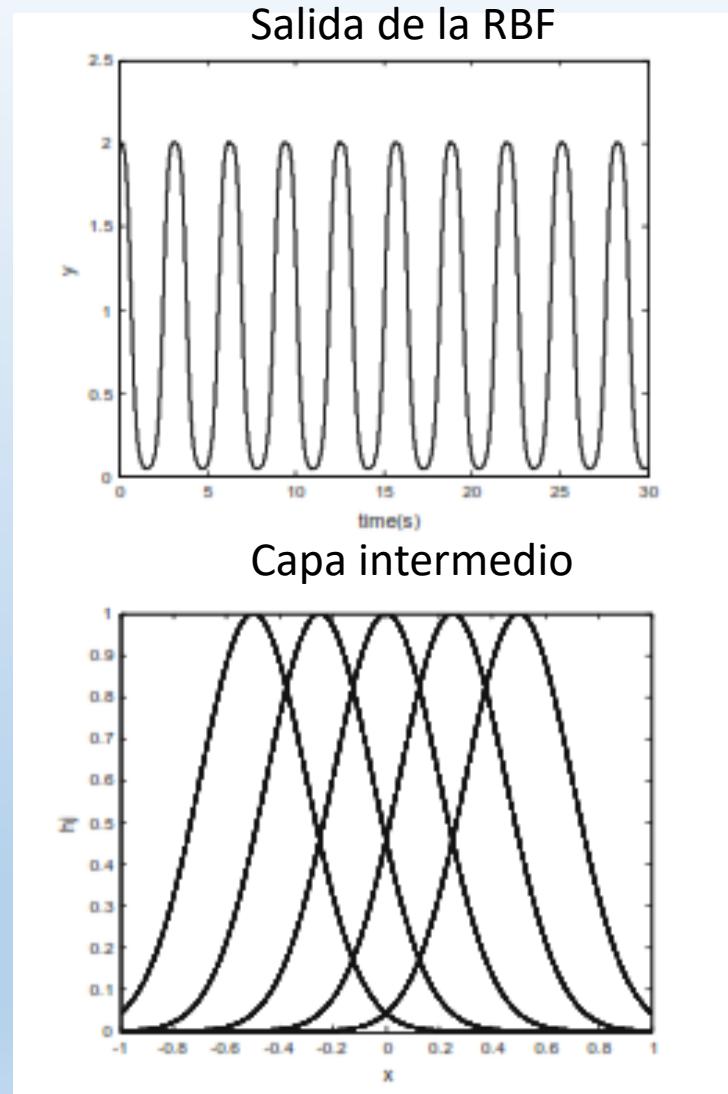
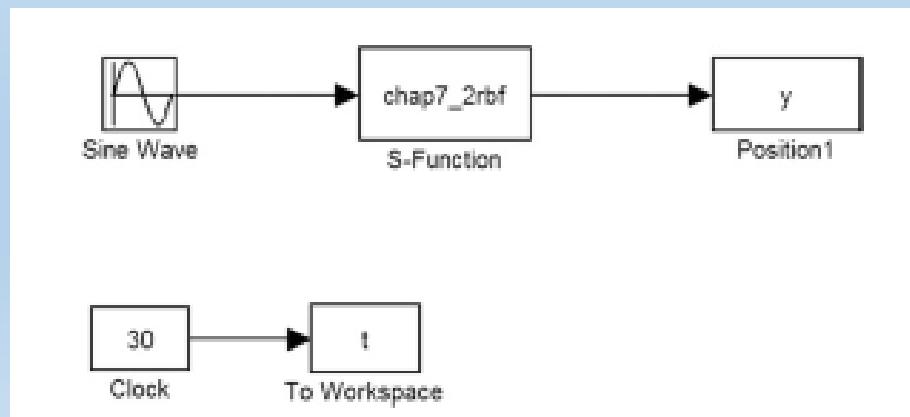




# Estructura (1,5,1) de la RBF para $\sin(t)$

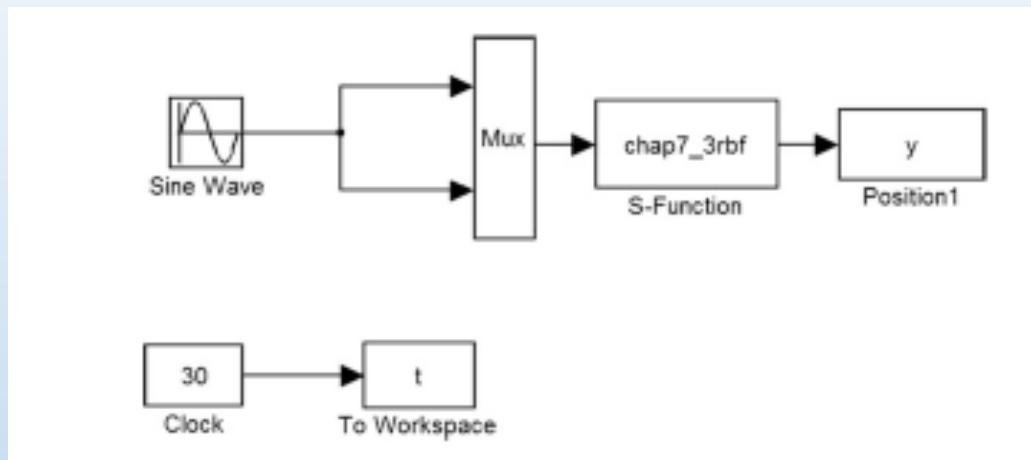


```
%RBF-1  
c=[ -0.5 -0.25 0 0.25 0.5]; %ci,j  
b=[0.2 0.2 0.2 0.2 0.2]'; %bj
```



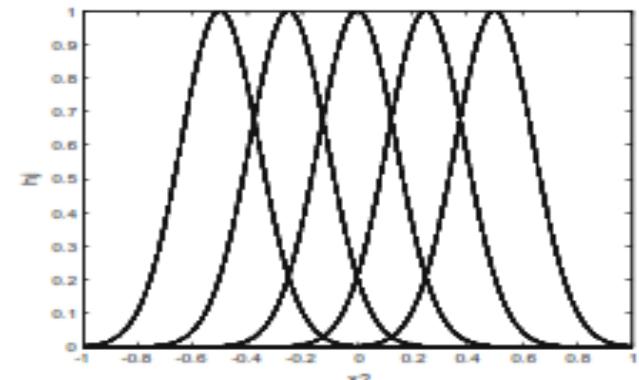
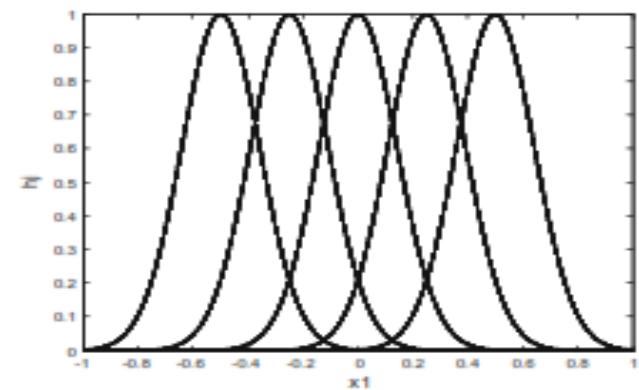
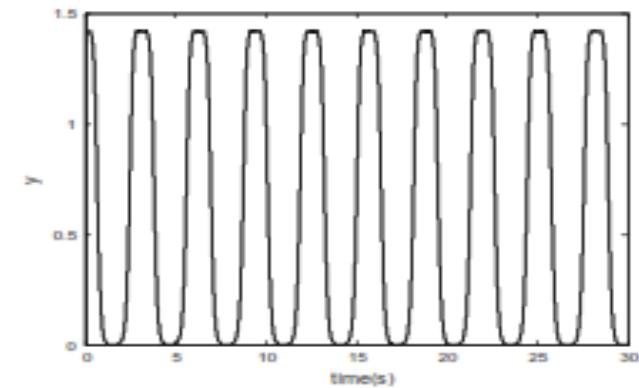
# Estructura (2,5,1) de la RBF para $\sin(t)$

Ambas entradas son  $\sin(t)$



```
% RBF-1  
c=[-0.5 -0.25 0 0.25 0.5;  
-0.5 -0.25 0 0.25 0.5]; %cij  
b=[0.2 0.2 0.2 0.2 0.2]'; %bj
```

En RBF los parámetros  $c_{ij}$  y  $b_j$  deben elegirse en términos de los valores de las entradas. Por tanto, se deben ajustar los valores usando también el método de gradiente descendiente.



# Neurona con Función de Base Radial RBF

- La diferencia entre BP y RBF es que la función de activación es una Gaussina en vez de la S simplificando el algoritmo y la carga computacional.

$$h_j = \exp\left(-\frac{\|x - c_j\|^2}{2b_j^2}\right)$$

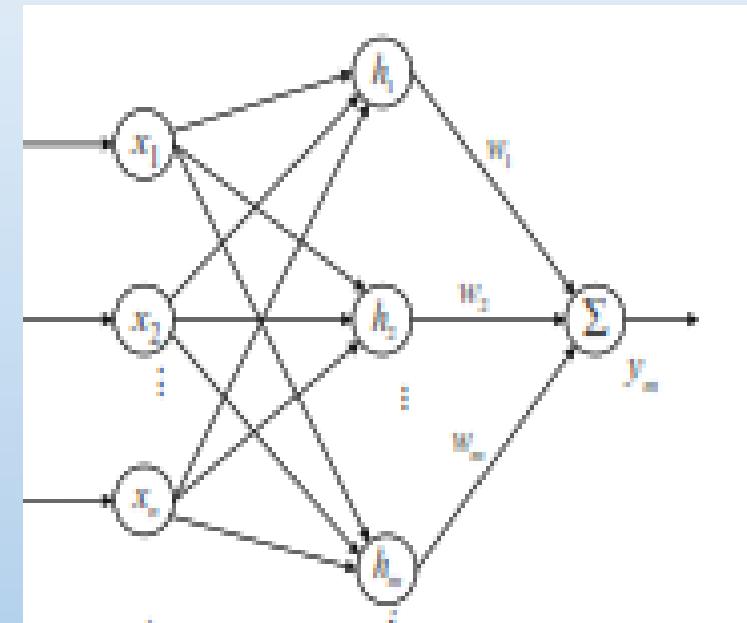
con  $c = \begin{bmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \cdots & \vdots \\ c_{n1} & \cdots & c_{nm} \end{bmatrix}$  el valor de la media

de la FG de la RN j para la entrada  $i=1, \dots, n$

$b_j$  = ancho de la función Gaussiana

$$w = [w_1 \quad \cdots \quad w_m]^T$$

$$y = w^T h = w_1 h_1 + w_2 h_2 + \cdots + w_m h_m$$





# Método del gradiente para RBF

Semana 12

$$E(t) = \frac{1}{2} (y(t) - y_m(t))^2$$

$$\Delta w_j(t) = -\eta \frac{\partial E}{\partial w_j} = \eta(y(t) - y_m(t))h_j$$

$$w_j(t) = w_j(t-1) + \Delta w_j(t) + \alpha(w_j(t-1) - w_j(t-2))$$

$$\Delta b_j = -\eta \frac{\partial E}{\partial b_j} = \eta(y(t) - y_m(t))w_j h_j \frac{\|\mathbf{x} - \mathbf{c}_j\|^2}{b_j^3}$$

$$b_j(t) = b_j(t-1) + \Delta b_j + \alpha(b_j(t-1) - b_j(t-2))$$

$$\Delta c_{ji} = -\eta \frac{\partial E}{\partial c_{ji}} = \eta(y(t) - y_m(t))w_j \frac{x_j - c_{ji}}{b_j^2}$$

$$c_{ji}(t) = c_{ji}(t-1) + \Delta c_{ji} + \alpha(c_{ji}(t-1) - c_{ji}(t-2))$$



# Ejemplo: tomados del libro el resto del capítulo

- Se ve el efecto de los parámetros de la función  $h(x)$  y  $c$  y  $b$  y del número de capas de la NN



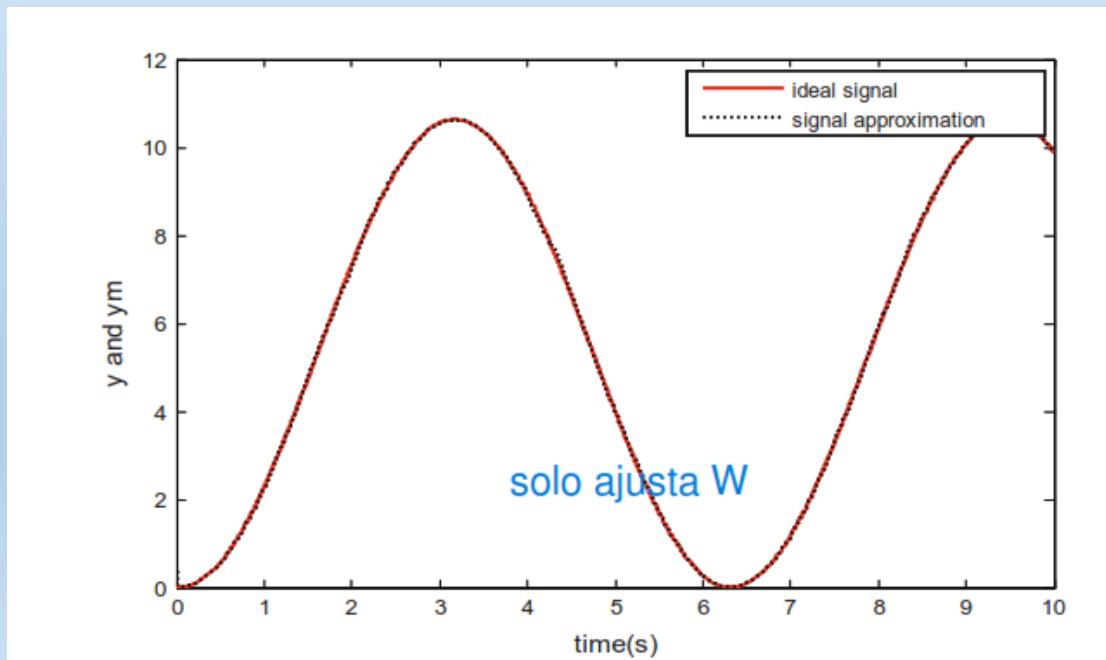
# Aproximar con RBF (2-5-1) la Función de Transferencia

$$G(s) = \frac{ent(s)}{sal(s)} = \frac{133}{s^3 + 25s}, \quad ent(t) = \sin(t), \quad ent = [0 \ 1], \quad sal = [0 \ 10]$$

$$x_1 = ent, \quad x_2 = sal, \quad \alpha = 0.05, \quad \eta = 0.5$$

$$b_j = 1.5, \quad c_j = \begin{bmatrix} -1 & -.5 & 0 & 0.5 & 1 \\ -10 & -5 & 0 & 5 & 10 \end{bmatrix}$$

Solo se ajustaron las w y las  $c_j$  y b se quedaron fijas



# Efecto de los parámetros c y b (2-5-1)

$$y(k) = u(k)^3 + \frac{y(k-1)}{1 + y(k-1)^2}$$

$$\alpha = 0.05, \quad \eta = 0.3$$

$$u(k) = \sin(k)$$

Izquierda : solo se ajustó w

Derecha : se ajustan w, c y b

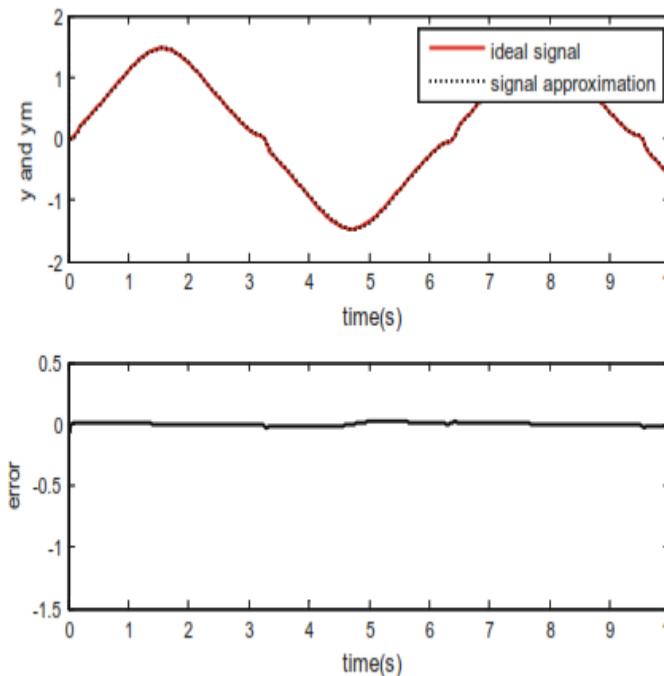
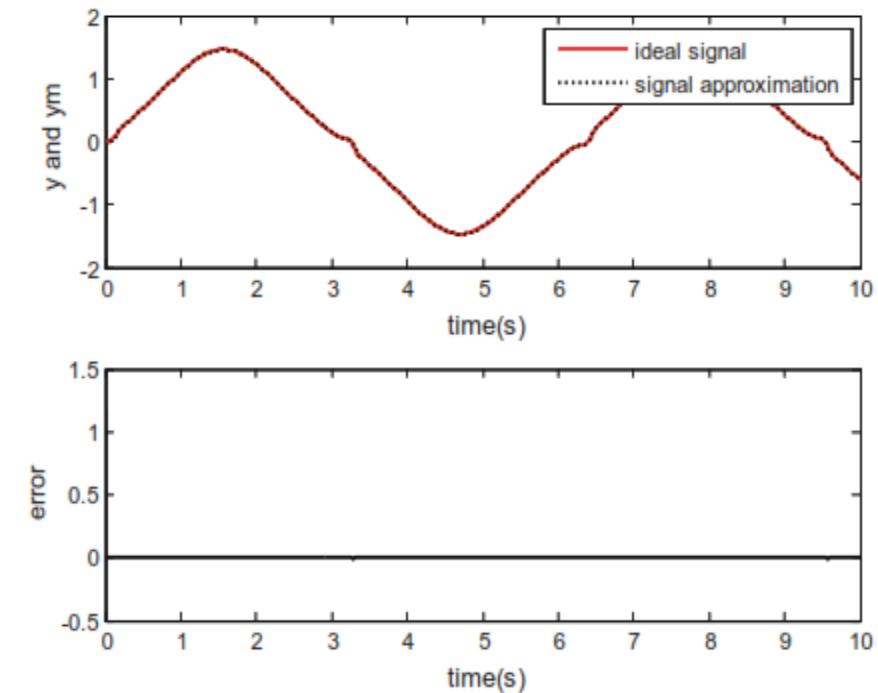


Fig. 7.19 RBF neural network approximation by only updating w ( $M = 1$ )



20 RBF neural network approximation by updating w, b, c ( $M = 2$ )

# Efecto de los parámetros $c$ y $b$ (2-5-1)

$$u = 0.5 \sin(2\pi t)$$

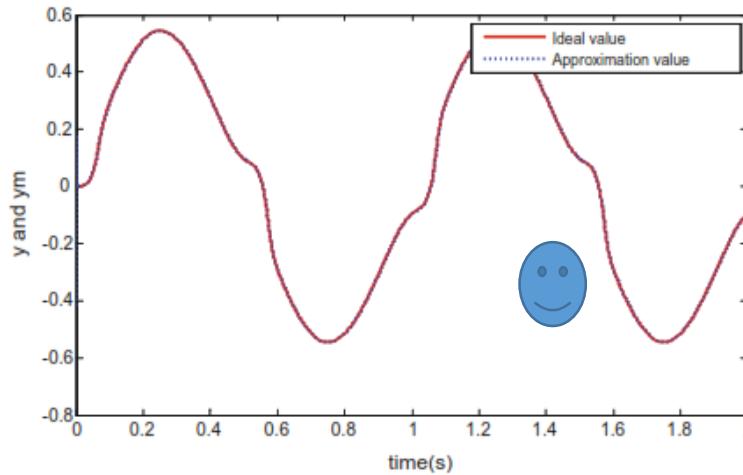


Fig. 7.22 RBF approximation with moderate  $b_j$  and  $c_j$  ( $M_b = 1$ ,  $M_c = 1$ )

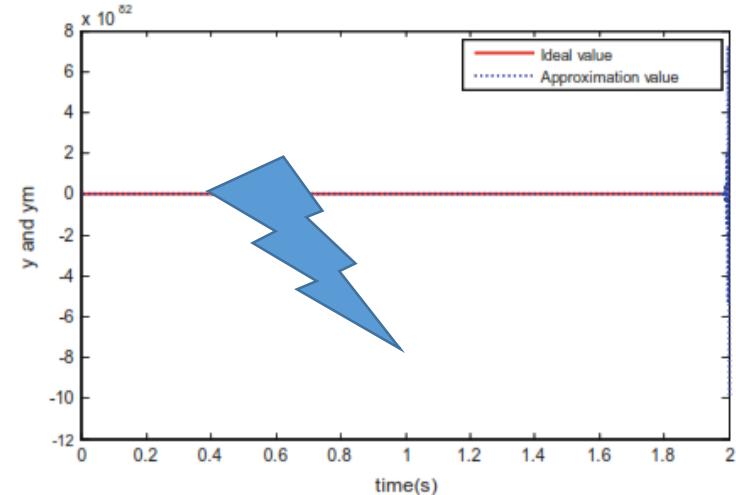


Fig. 7.24 RBF approximation with moderate  $b_j$  and improper  $c_j$  ( $M_b = 1$ ,  $M_c = 2$ )

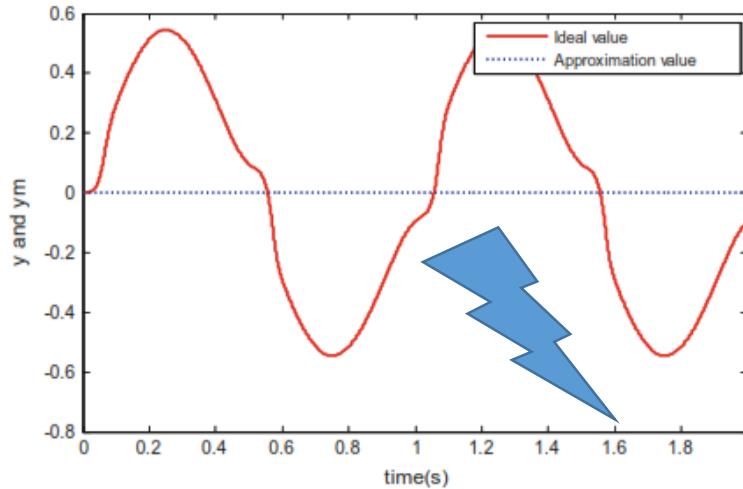


Fig. 7.23 RBF approximation with improper  $b_j$  and moderate  $c_j$  ( $M_b = 2$ ,  $M_c = 1$ )

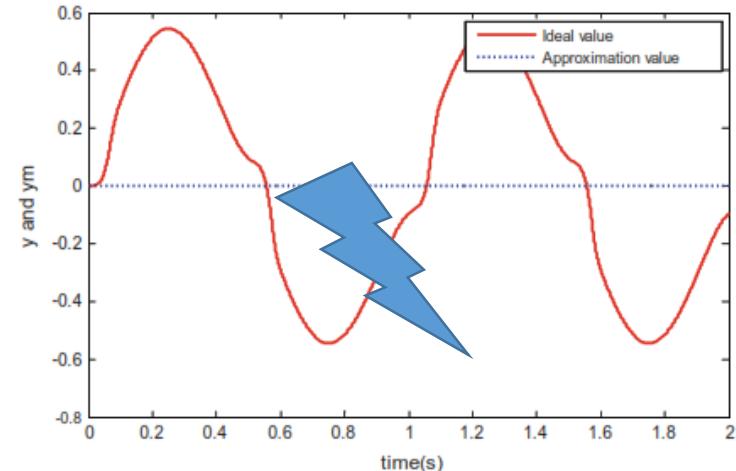
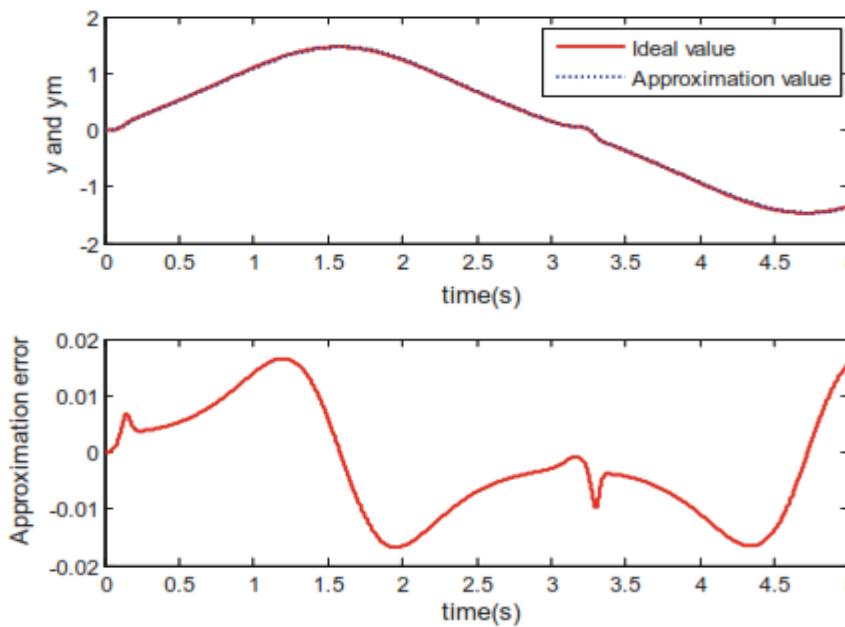
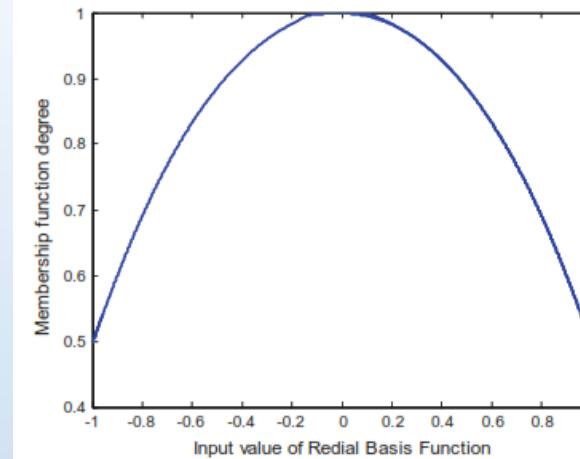


Fig. 7.25 RBF approximation with improper  $b_j$  and  $c_j$  ( $M_b = 2$ ,  $M_c = 2$ )



# Efecto de las capas RBF

Con una sola capa



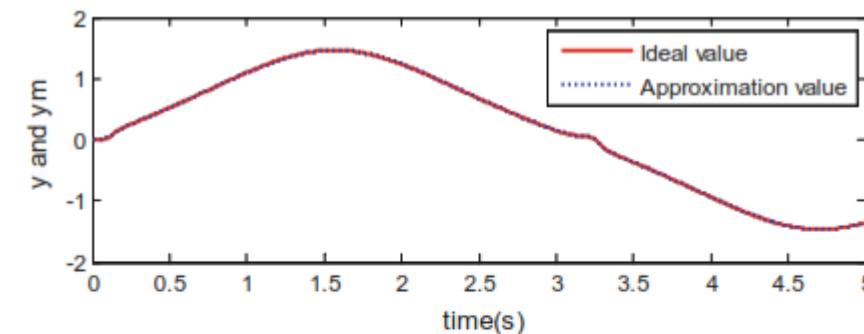
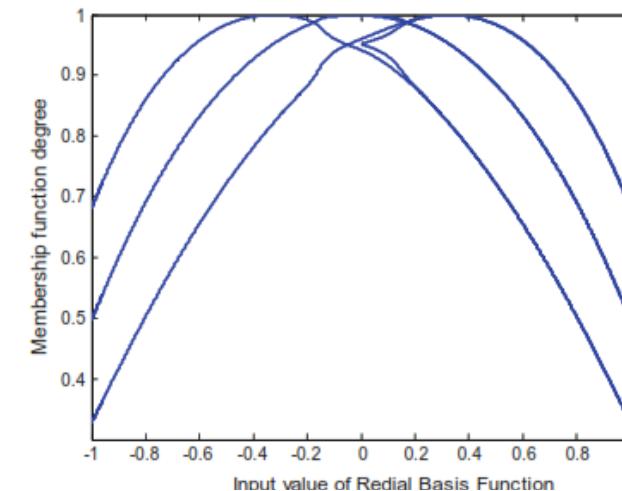
Max 0.02

Fig. 7.27 Approximation with only one hidden net ( $m = 1$ )

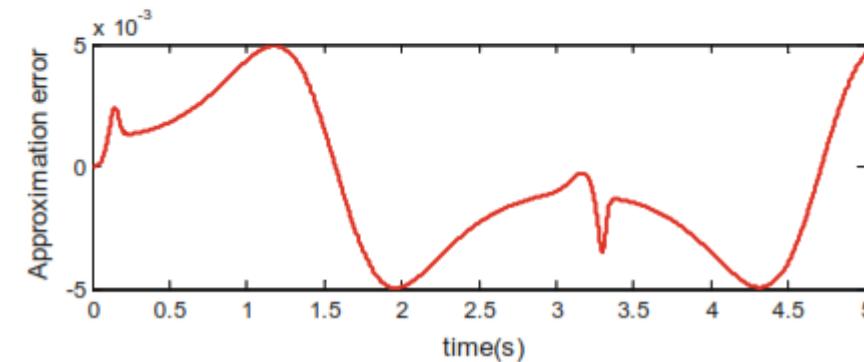


# Efecto de las capas RBF

Con tres capas



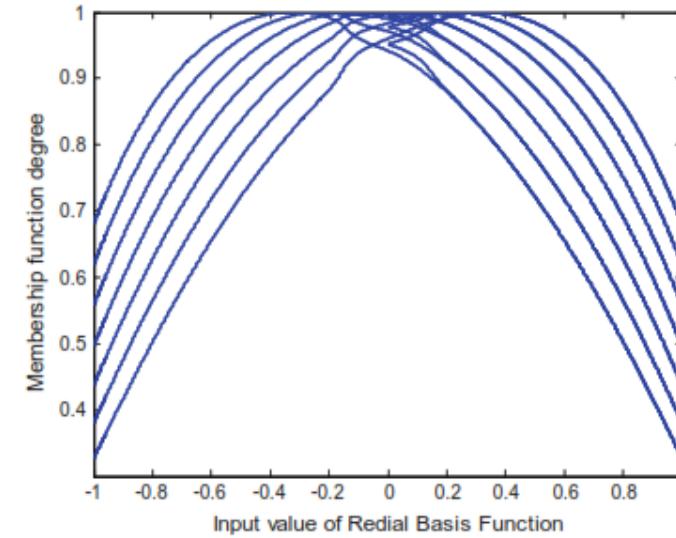
Max 0.005



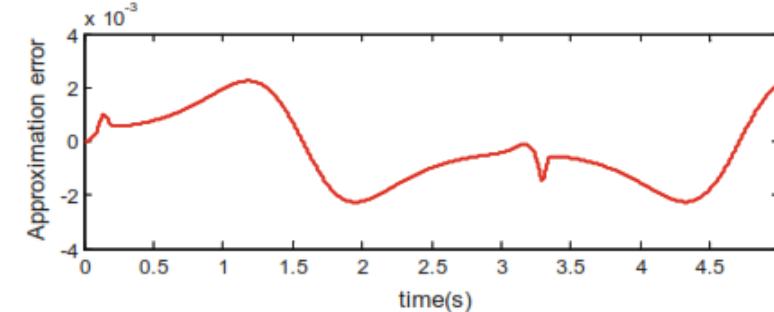
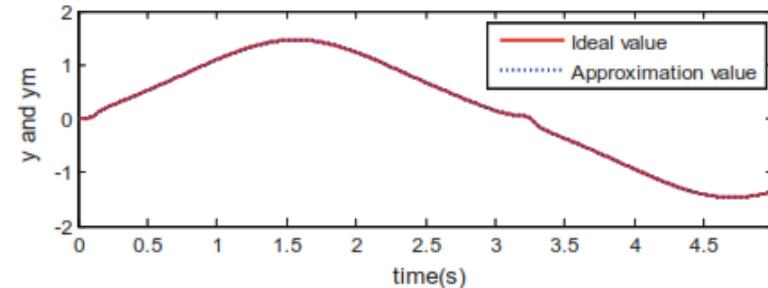


# Efecto de las capas RBF

Con ciete capas



Max 0.002





# Control Adaptable con RBF

- Razones para la aplicación de la RBF para control e identificación de sistemas.
- Al ser un generador de funciones universales, pueden ser entrenadas para hacer tareas de identificación y control.
- Reduce el uso de análisis matemáticos complejos que se usan tradicionalmente en control.
- Permite generar funciones no lineales complejas.
- Modelos del sistema a controlar a priori.
- El paralelismo de NN permite el manejo de multiprocesamiento con circuitos integrados y sistemas paralelos.
- Daños de una parte de la NN **no** altera el resto del procesamiento paralelo.

# Descripción del problema: caso continuo

Aproximar  $f(x)$  por  $\hat{f}(x)$

$$\ddot{x} = f(x, \dot{x}) + g(x, \dot{x})u \quad x \in R^2, u \in R$$

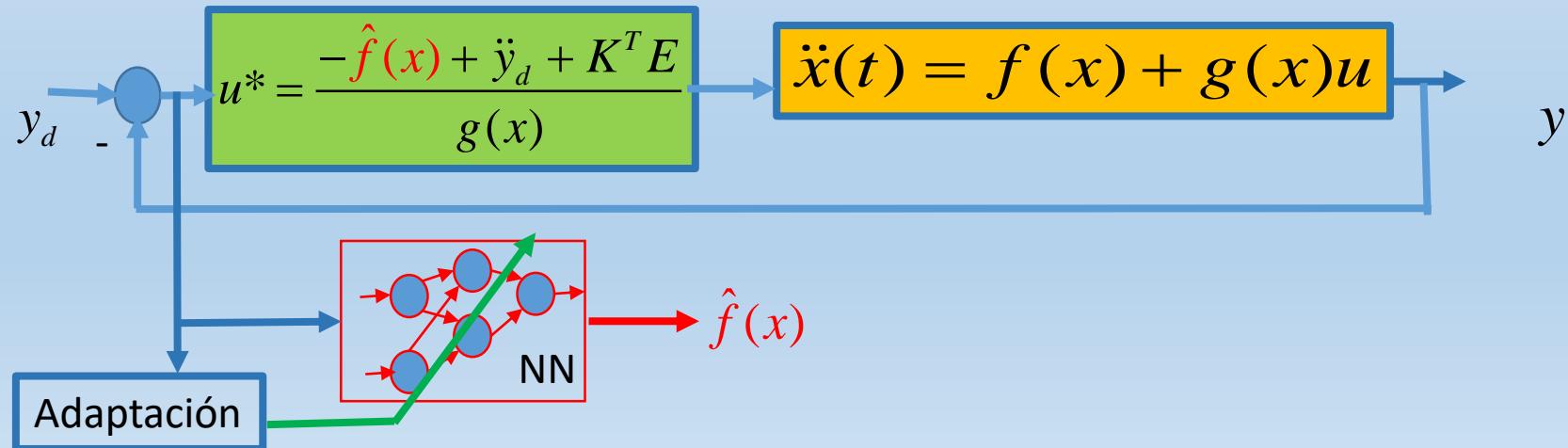
$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = f(x_1, x_2) + g(x_1, x_2)u \quad y = x_1$$

$$\text{Señal de Seguimiento } y_d \quad e = y_d - y, \quad E = [e \quad \dot{e}]^T$$

$$\text{Ley de control } u^* = \frac{-\hat{f}(x) + \ddot{y}_d + K^T E}{g(x)} \text{ con } K = [k_p \quad k_d]^T$$

Ecuación del error:  $\ddot{e} + k_d \dot{e} + k_p e = 0$ . Si las 2 raíces están RHLR  $\lim_{t \rightarrow \infty} e = 0$





# Descripción del problema: caso discreto

Aproximar  $f(x)$  por  $\hat{f}(x)$

$$y(k) = f(y(k-2), y(k-1)) + g(y(k-2), y(k-1))y(k), \quad \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \in R^2, u(k) \in R$$

$$x_1(k+1) = x_2(k)$$

$$x_2(k+1) = f(x_1(k), x_2(k)) + g(x_1(k), x_2(k))u(k) \quad y(k) = x_1(k)$$

Señal de Seguimiento  $y_d$ , Señal de error  $e(k) = y_d(k) - y(k) = y_d(k) - x_1(k)$

Ecuación del error deseada:  $e(k) + k_d e(k-1) + k_p e(k-2) = 0$  con  $\lim_{k \rightarrow \infty} e(k) = 0$

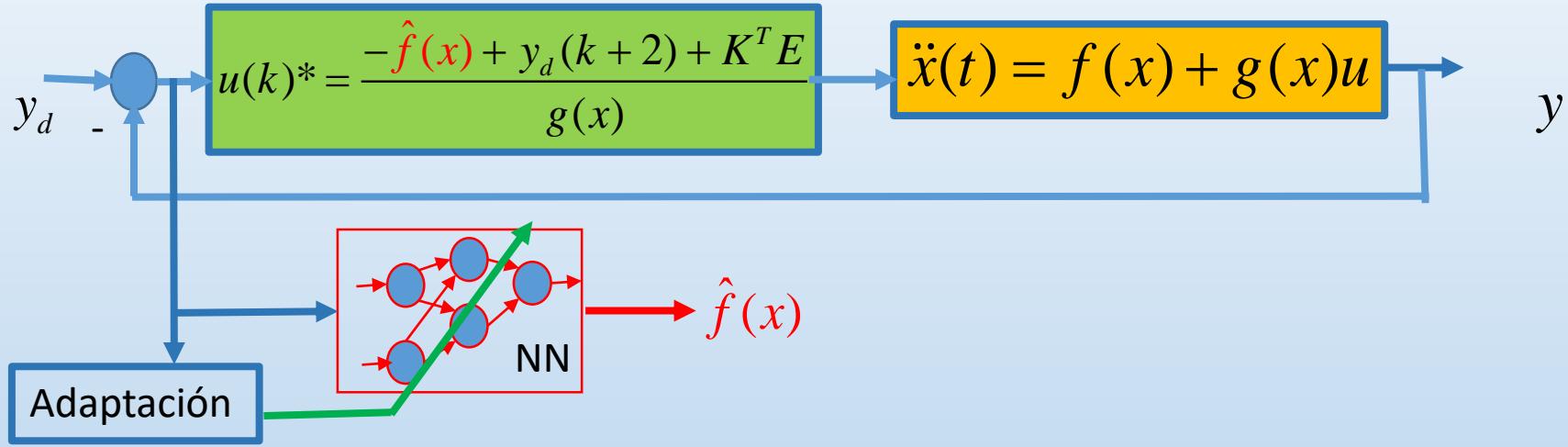
implica que las raíces de la ecuación del error deben estar dentro del círculo unitario.

$$E := [e(k) \quad e(k+1)]^T, \quad K = [k_p \quad k_d]^T, \quad e(k+2) = -k_d e(k+1) - k_p e(k).$$

$$\text{Ley de control } u(k)^* = \frac{x_2(k+1) - f(x(k))}{g(x)} = \frac{y_d(k+2) - e(k+2) - f(x(k))}{g(x)}$$
$$= \frac{-f(x(k)) + y_d(k+2) + K^T E}{g(x)}$$



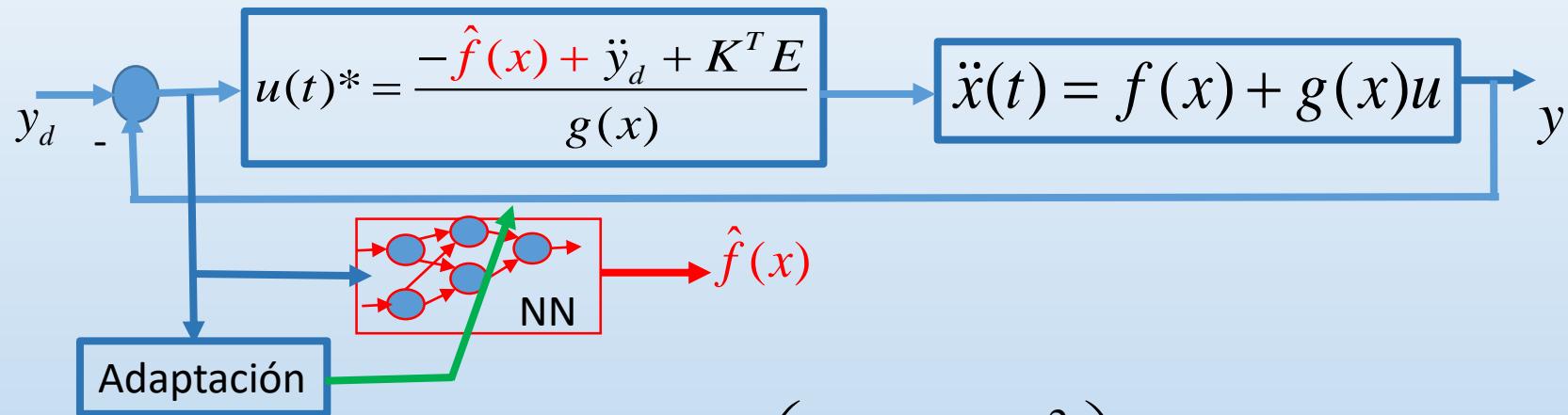
# Implementación del Controlador RBF: caso discreto





# Adaptación de la NN y su Estabilidad: caso continuo

Semana 14



$$h(\xi_n) = g\left( \frac{\|\xi_n - c_{ij}\|^2}{b_j} \right)$$

Se propone

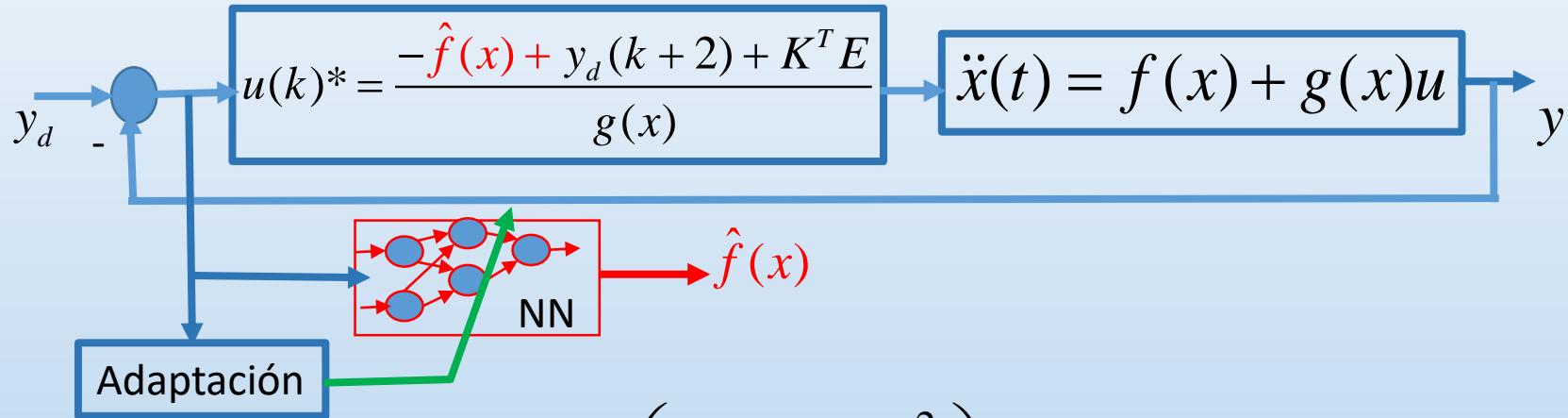
$$\hat{f}(x) = W^T h(\xi_n) + \varepsilon$$

$$\dot{W} = -\gamma E^T P b h(\xi_n)$$



# Adaptación de la NN y su Estabilidad: caso discreto

Semana 14



$$h(\xi_n) = g\left( \frac{\|\xi_n - c_{ij}\|^2}{b_j} \right)$$

Se propone  $\hat{f}(x) = W^T h(\xi_n) + \varepsilon$

$$W(k+1) = W(k) - \gamma E^T P b h(\xi_n)$$



# Condiciones de estabilidad: caso continuo

$$\dot{e}_1 = e_2$$

$$\dot{e}_2 = -k_p e_1 - k_d e_2 + \hat{f}(x) - f(x)$$

equivalentemente

$$\dot{E} = \Lambda E + B(\hat{f}(x) - f(x))$$

$$\text{con } \Lambda = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Se propone encontrar:

$$W^* = \arg \min_{W \in \Omega} [\sup |\hat{f}(x) - f(x)|] \quad \text{con } e_w \triangleq \hat{f}(x | w) - f(x)$$

Vincula el error de la salida y el error de la estimación de la función desconocida

$$\begin{aligned} \dot{E} &= \Lambda E + B(\hat{f}(x | \hat{W}) - \hat{f}(x | W^*) + e_w) \\ &= \Lambda E + \underbrace{B[(\hat{W} - W^*)^T h(x) + e_w]}_M = \Lambda E + BM \end{aligned}$$

$$\text{Función de Lyapunov } V \triangleq \frac{1}{2} E^T P E + \frac{(\hat{W} - W^*)^T (\hat{W} - W^*)}{2\gamma} = V_{1y} + V_{2w} \geq 0 \quad \text{y} \quad \dot{V} < 0$$

Suponga  $\Lambda^T P + P\Lambda = -Q$  con  $Q > 0$

Es decir, si se conoce  $f(x)$  el control estabiliza la planta



# Condiciones de estabilidad: caso discreto

$$e_1(k+1) = e_2(k)$$

$$e_2(k+1) = -k_p e_1(k) - k_d e_2(k) + \hat{f}(x(k)) - f(x(k))$$

$$E(k+1) = \Lambda E(k) + B(\hat{f}(x(k)) - f(x(k)))$$

$$\text{con } \Lambda = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Se propone encontrar:

$$W^* = \arg \min_{W \in \Omega} [\sup |\hat{f}(x) - f(x)|] \quad \text{con } e_w \triangleq \hat{f}(x|w) - f(x)$$

Vincula el error de la salida y el error de la estimación de la función desconocida

$$\begin{aligned} E(k+1) &= \Lambda E(k) + B(\hat{f}(x|\hat{W}) - \hat{f}(x|W^*) + e_w(k)) \\ &= \Lambda E(k) + \underbrace{B[(\hat{W} - W^*)^T h(\xi_n) + e_w(k)]}_{M(k)} = \Lambda E(k) + BM(k) \end{aligned}$$

Función de Lyapunov

$$V(k) \triangleq \frac{1}{2} E(k)^T P E(k) + \frac{(\hat{W}(k) - W^*)^T (\hat{W}(k) - W^*)}{2\gamma} = V_{1y} + V_{2w} \geq 0 \quad \text{y} \quad \Delta V(k) < 0$$

Suponga  $\Lambda^T P \Lambda - P = -Q$  con  $Q > 0$

Es decir, se asume que si se conoce  $f(x)$  el control estabiliza la planta.



## Continua caso continuo

$$\begin{aligned}\dot{V}_{1y} &= \frac{1}{2}(\dot{E}^T PE + E^T PE) = \frac{1}{2}(E^T P^T + MT)PE + \frac{1}{2}E^T P(AE + M) \\ &= -\frac{1}{2}E^T QE + E^T PM = -\frac{1}{2}E^T QE + (W - W^*)^T E^T PBh(x) + E^T PB e_w\end{aligned}$$

$$\dot{V}_{2w} = \frac{1}{2}(\hat{W} - W^*)^T \dot{\hat{W}}$$

$$\begin{aligned}\dot{V} &= \dot{V}_{1y} + \dot{V}_{2w} = -\frac{1}{2}E^T QE + E^T PB e_w + \frac{1}{\gamma}(\hat{W} - W^*)^T [\dot{\hat{W}} + \gamma E^T PBh(x)] \\ &= -\frac{1}{2}E^T QE + E^T PB e_w\end{aligned}$$

$$\|E\| \leq \frac{2\lambda_{\max}(PB)e_{w\max}}{\lambda_{\min}(Q)},$$

$\lambda_{\max}$  y  $\lambda_{\min}$  denota valor propio máx y mín de una matriz



# Continua caso discreto???



# Ejemplo: Sistema lineal

$$\dot{x}_1 = x_2$$

$$x_1(0) = \pi / 60 \quad x_2(0) = 0$$

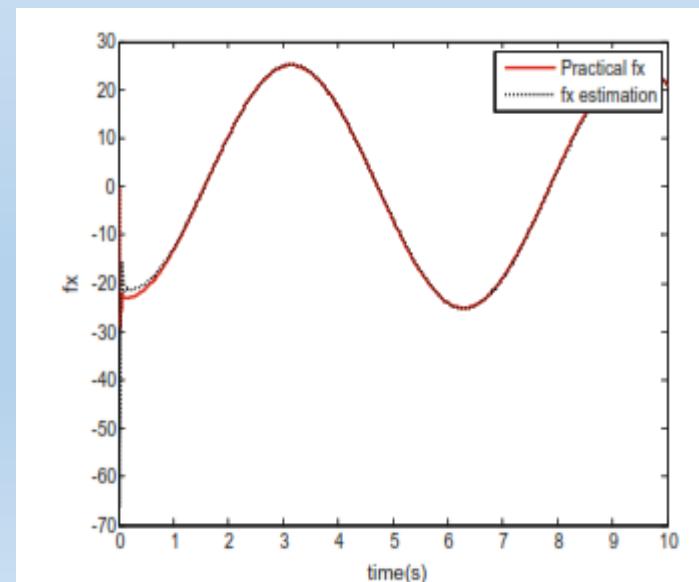
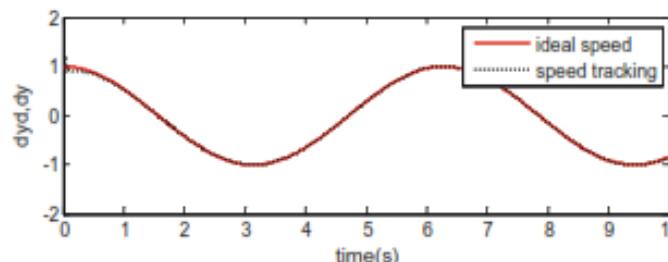
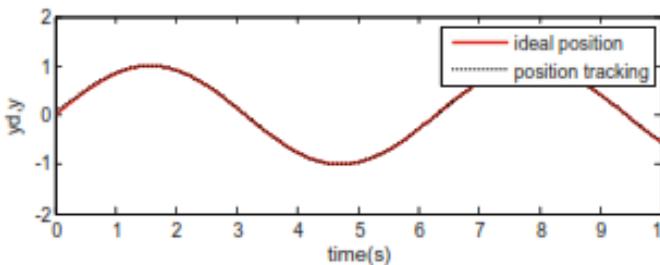
$$\dot{x}_2 = f(x) + g(x)u = -25x_2 + 133u$$

$$y = x_1$$

$$y_d = \sin t$$

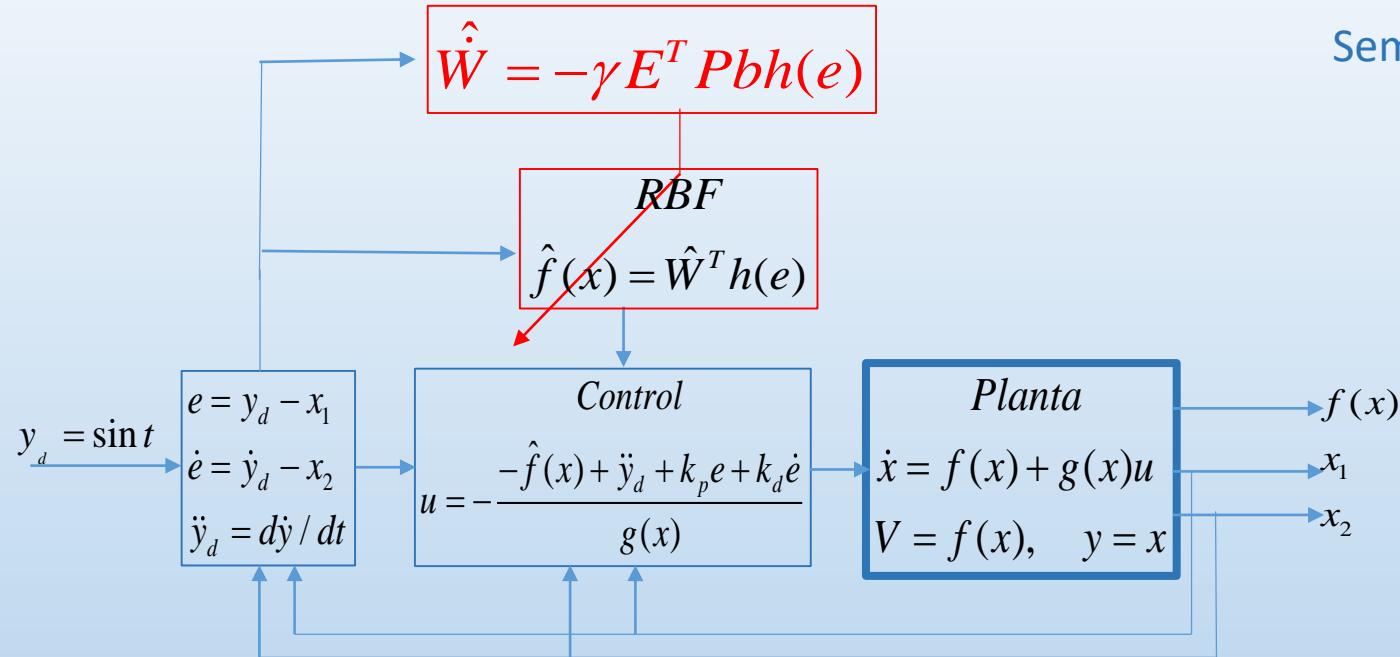
Para la RBF(2-5-1):  $c = [-1 \quad -0.5 \quad 0 \quad 0.5 \quad 1]$ ,  $b = 1.$ ,  $w = 0.$

Para el control:  $Q = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix}$ ,  $k_d = 50$ ,  $k_p = 30$ ,  $\gamma = 1000$ .



# Implementación del Control Adaptable con RBF

Semana 14



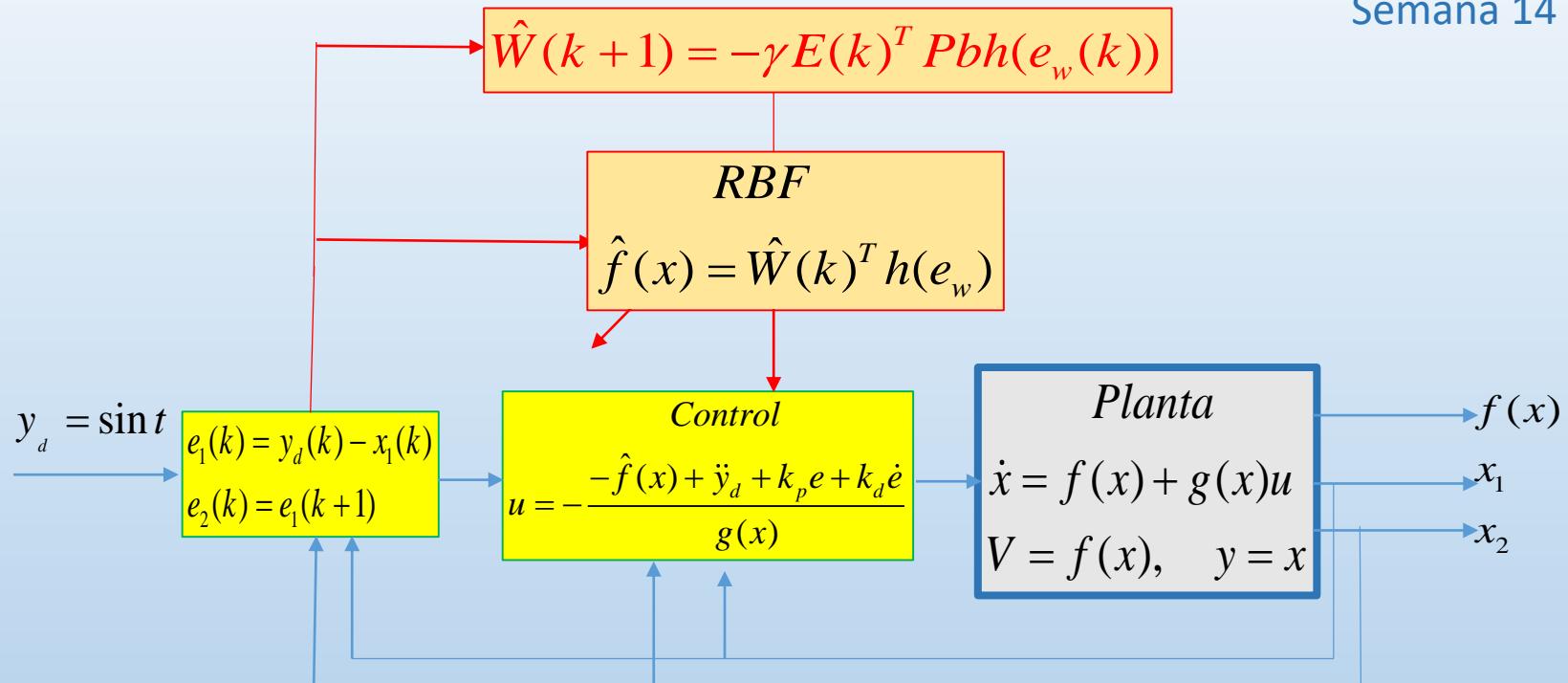
$$E = [e \quad \dot{e}] \quad b=1, \quad Q = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix} \quad \Lambda = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\Lambda^T P + P \Lambda = -Q \text{ con } Q > 0$$

$$c = [-1 \quad -0.5 \quad 0 \quad 0.5 \quad 1]$$

# Implementación del Control Adaptable con RBF

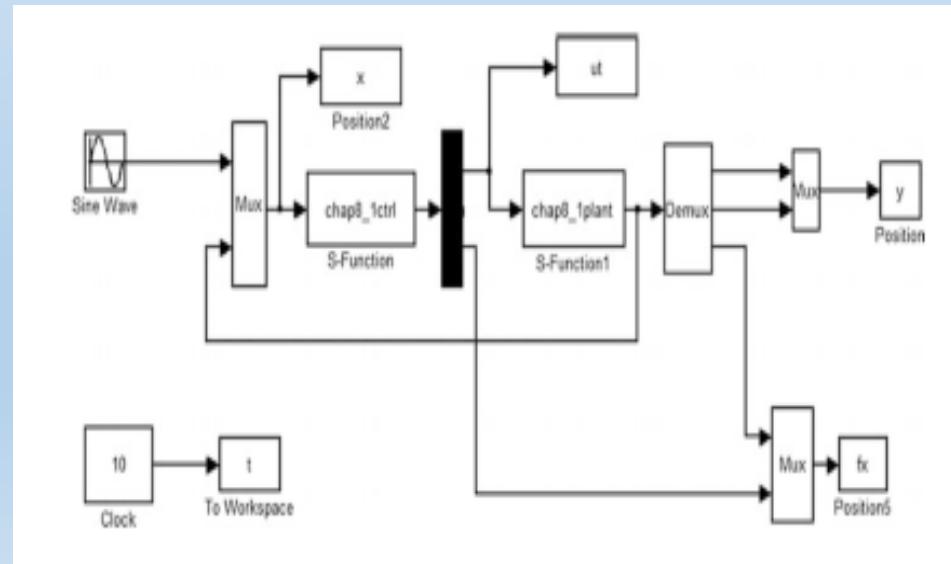
Semana 14



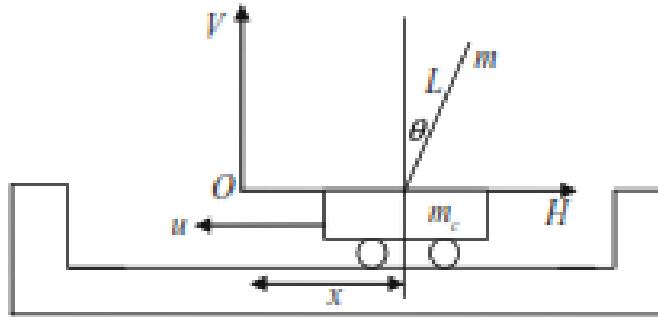


# Interpretación

- Se genera la serie de tiempo o función  $\hat{f}(t)$  más no se identifica la función y sus parámetros  $f(x)$
- Si no se conoce el estado, no se puede decir que se conoce  $f(x)$ .
- ¿Qué ocurre si se cambia la señal de excitación  $y_d(t)$ ?



# Sistema No lineal: Péndulo Invertido



$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x) + g(x)u\end{aligned}$$

$$f(x) = \frac{g \sin x_1 - m L x_2^2 \cos x_1 \sin x_1 / (m_c + m)}{L(4/3 - m \cos^2 x_1 / (m_c + m))}$$

$$g(x) = \frac{\cos x_1 / (m_c + m)}{L(4/3 - m \cos^2 x_1 / (m_c + m))}$$

$x_1$  = ángulo,  $x_2$  = velocidad angular,  $g$ =gravedad,  $m_c$  = masa del carro

$m$  = masa del péndulo,  $L = 0.5$  la mitad de la longitud del péndulo

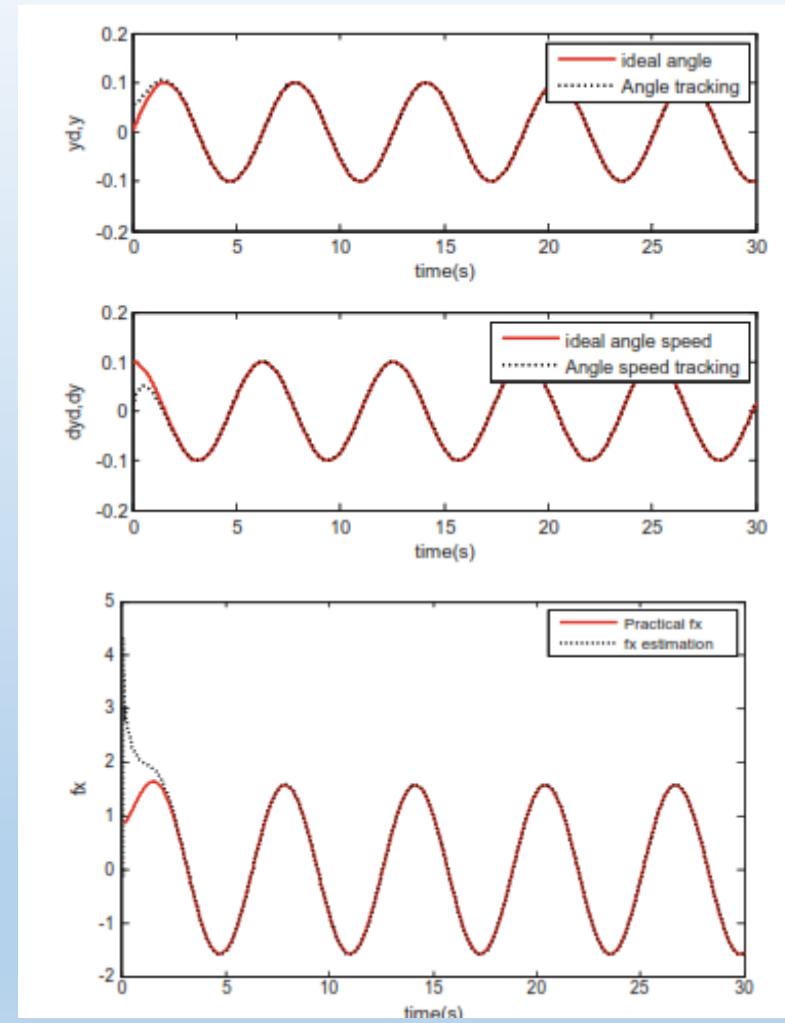
$u$ = control (par), señal de referencia deseada  $= 0.1 \sin t$

Se propone un RBF con  $c = [-2 \quad -1 \quad 0 \quad 1 \quad 2]$ ,  $b = 0.2$  y

un control con  $kd=50$ ,  $kp=30$  con  $\gamma=1200$ .

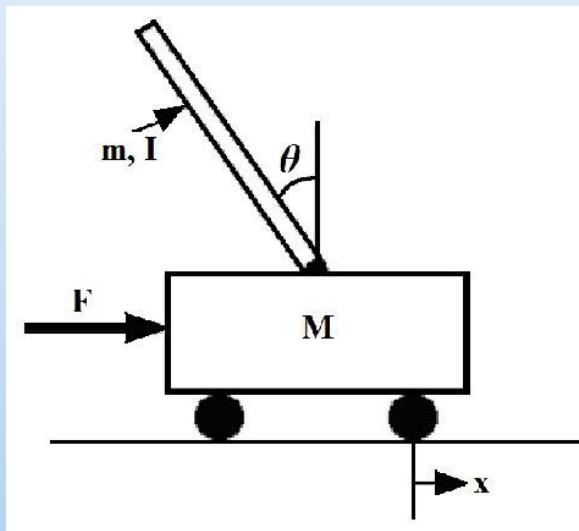


# Resultados





# Péndulo Invertido del II-UNAM





# Control con funciones y parámetros desconocidos

semana 15

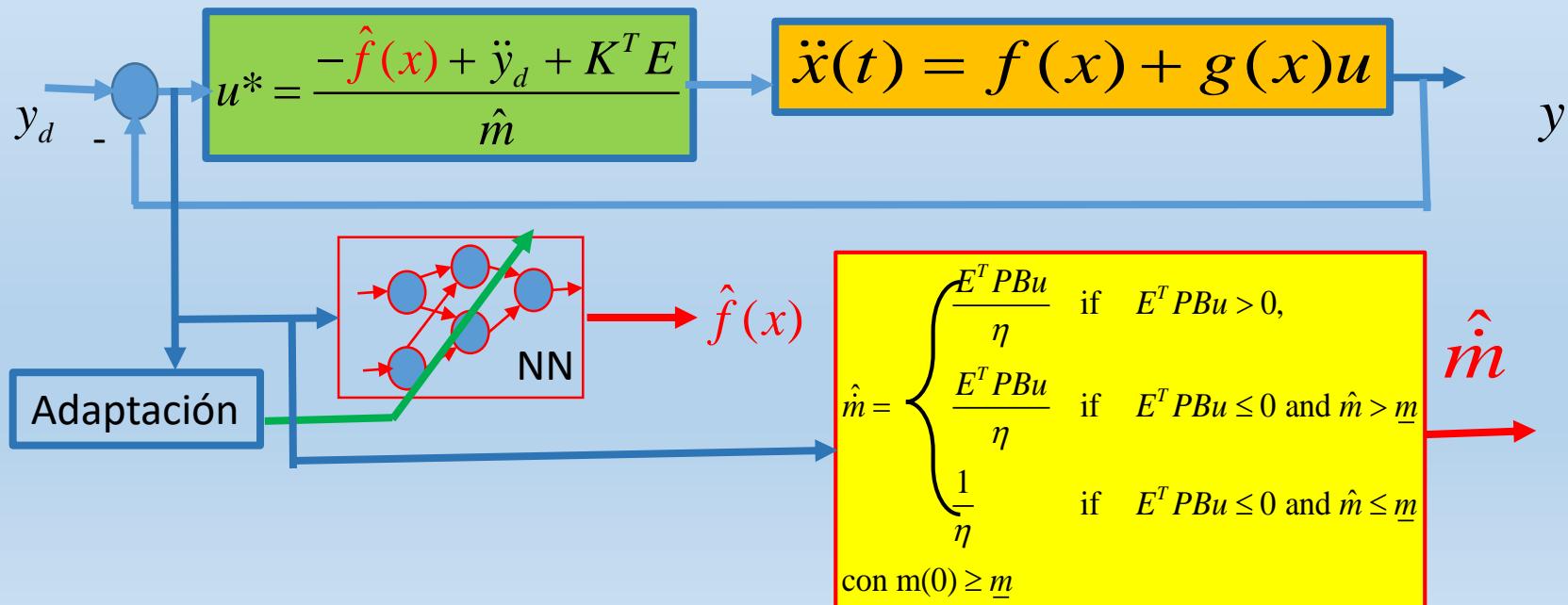
$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = f(x) + mu \quad y = x_1$$

Señal de Seguimiento  $y_d \quad e = y_d - y, \quad E = [e \quad \dot{e}]^T$

Ley de control  $u^* = \frac{-\hat{f}(x) + \ddot{y}_d + K^T E}{m}$  con  $K = [k_p \quad k_d]^T$

Ecuación del error:  $\ddot{e} + k_d \dot{e} + k_p e = 0$ . Si las 2 raíces están RHLR  $\lim_{t \rightarrow \infty} e = 0$





# Prueba de Estabilidad

A partir de la ecuación del error:

$$\ddot{e} = -K^T E + \hat{f}(x) - f(x) + (m - \hat{m})u$$

Se propone diseñar la RBF

$$\dot{E} = \Lambda E + B(\hat{f}(x) - f(x)) + (m - \hat{m})u$$

$$\text{con } \Lambda = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$W^* = \arg \min_{W \in \Omega} [\sup | \hat{f}(x) - f(x) |] \quad \text{con } e_w \triangleq \hat{f}(x | w) - f(x)$$

Ojo: La red solamente se preocupa por estimar la función, más no el parámetro

$$\begin{aligned} \dot{E} &= \Lambda E + B\{(\hat{f}(x | \hat{W}) - \hat{f}(x | W^*) + e_w + (m - \hat{m})u\} \\ &= \Lambda E + \underbrace{B[(\hat{W} - W^*)^T h(x) + e_w + (m - \hat{m})u]}_M = \Lambda E + M \end{aligned}$$

$$V \triangleq \frac{1}{2} E^T P E + \frac{(\hat{W} - W^*)^T (\hat{W} - W^*)}{2\gamma} + \frac{1}{2} \eta \tilde{m}^2 = V_{1y} + V_{2w} + V_3 \geq 0 \quad \text{y} \quad \dot{V} < 0$$

Función de Lyapunov

Suponga  $\Lambda^T P + P\Lambda = -Q$  con  $Q > 0$ ,  $\tilde{m} = m - \hat{m}$ ,  $\eta > 0$

Es decir, si se conoce  $f(x)$  y  $m$  el control estabiliza la planta



$$\dot{V} = \dot{V}_{1y} + \dot{V}_{2w} + \dot{V}_3 = -\frac{1}{2} E^T QE + E^T PBe_w + \frac{1}{\gamma} (\hat{W} - W^*)^T [\dot{\hat{W}} + \gamma E^T PBh(x)] \\ + \tilde{m}(E^T PBu - \eta \dot{\hat{m}})$$

Se selecciona la ley de adaptación:  $\dot{\hat{W}} = -\gamma E^T PBh(x)$ ,  $\hat{f}(x) = \hat{W}^T h(x) + \varepsilon$  y además para que el último término sea negativo

$$\hat{m} = \begin{cases} \frac{E^T PBu}{\eta} & \text{if } E^T PBu > 0, \\ \frac{E^T PBu}{\eta} & \text{if } E^T PBu \leq 0 \text{ and } \hat{m} > \underline{m} \\ \frac{1}{\eta} & \text{if } E^T PBu \leq 0 \text{ and } \hat{m} \leq \underline{m} \end{cases}$$

con  $m(0) \geq \underline{m}$

$$= -\frac{1}{2} E^T QE + E^T PBe_w$$

# Ejemplo con m desconocida

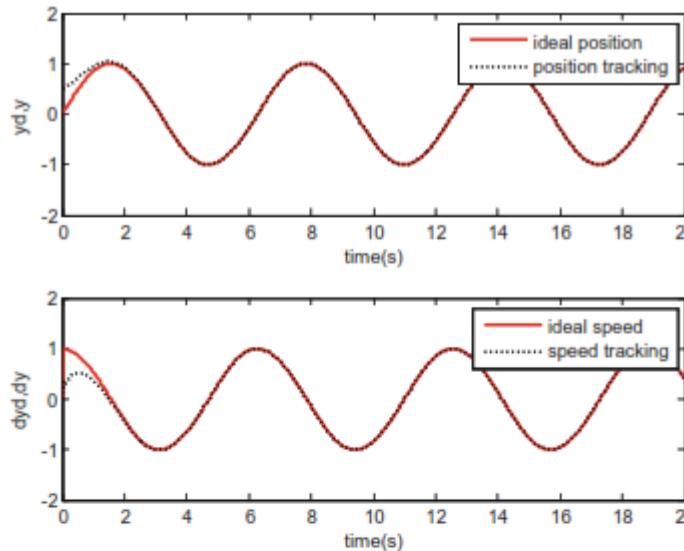


Fig. 8.7 Position and speed tracking

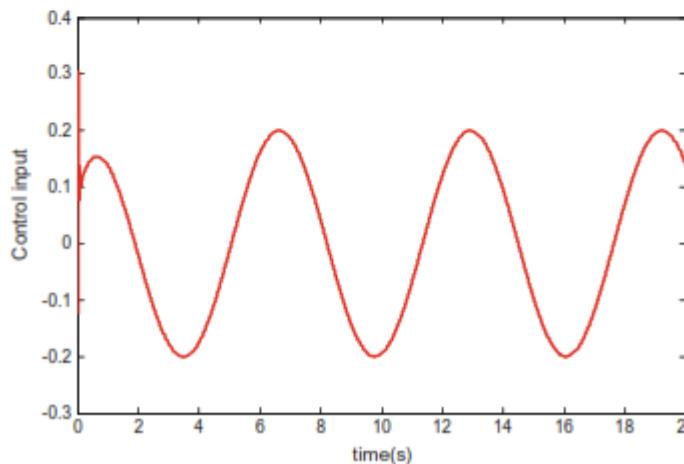
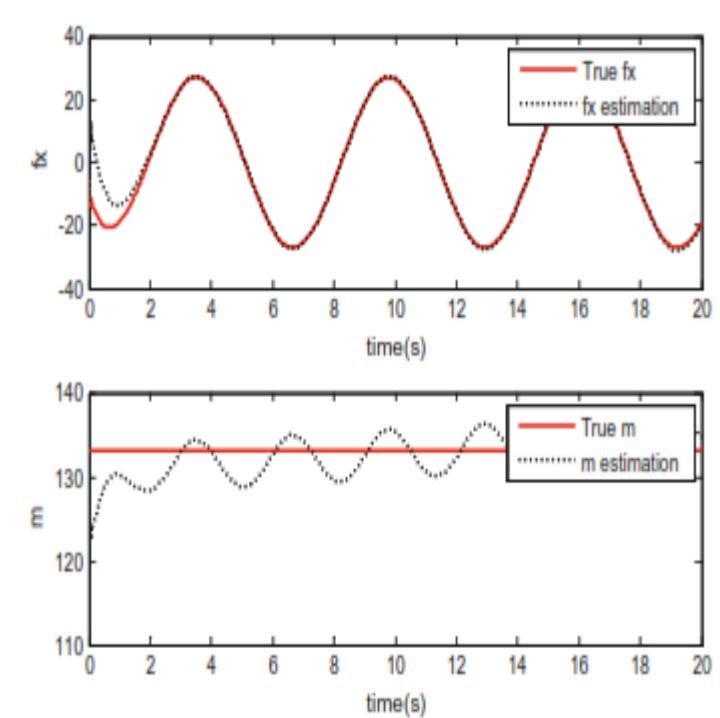


Fig. 8.8 Control input

El parámetro se queda oscilando  
La función  $f(x)$  se estima bien.



# Control por Modos Deslizantes SMC

- Un aspecto intrínseco de los SMC es su naturaleza de acción discontinua, cuya función principal es la de comutar entre dos estructuras diferentes, de tal manera que se establece un tipo de trayectoria en el espacio de estado llamado SM en cada canal de la retroalimentación.
- Esto permite suponer que se tiene un desempeño superior el cual incluye insensibilidad a variaciones en los parámetros y rechazo a perturbaciones.
- Control por modos deslizantes permite manejar incertidumbres definidas en conjuntos compactos (cualquier espacio que contenga un número finito de puntos)



# Marco de referencia

Las variables de estado se restringen a satisfacer un conjunto de ecuaciones (superficie de conmutación o variedad) y el diseño consta de dos pasos.

- a) Diseñar la superficie de deslizamiento tal que las variables de estado del sistema se restringen a tener una respuesta deseada.
- b) Construir un control retroalimentado conmutado (switched) que lleve las trayectorias de los estados a la superficie de deslizamiento. La construcción de la superficie estable se basa en la teoría de Lyapunov.



# Modos deslizantes adaptables SMC con NN

- Condición de Emparejamiento Matching: las incertidumbres deben estar en el espacio rango de la entrada de control para asegurar la propiedad de invariancia del comportamiento del sistema durante el modo deslizante.
- Control para que el sistema  $\dot{x} = Ax + Bu + p(t)$  siga una referencia a pesar de  $p(t)$  desconocida.

$$y = Cx(t)$$

$p(t)$  desconocida.

- Una variable deslizante (caracterizada por un polinomio estable Alfa)

$$s(x) = \alpha^T x = \sum_{i=1}^n \alpha_i x_i = \sum_{i=1}^{n-1} \alpha_i x_i + x_n \quad \text{Alfa} = \begin{bmatrix} \alpha_1 & \dots & \alpha_{n-1} & 1 \end{bmatrix}^T$$



## Considere un sistema de 2. Orden

$J\ddot{\theta}(t) = u(t) + d(t)$ ,  $J$  = momento de inercia,  $\theta$ =ángulo,  
 $u(t)$  la señal de control, y  $d(t) < D$  la perturbación.

Con el error  $e(t) = \theta(t) - \theta_d(t)$ , su derivada  $\dot{e}(t) = \dot{\theta}(t) - \dot{\theta}_d(t)$   
y la función del SM  $s(t) = ce(t) + \dot{e}(t)$   $c>0$   
tal que la raíz  $\lambda$  de polinomio  $(c + \lambda)$  este en el RHCP.

$$\text{Si } s(t) = 0, \Rightarrow ce(t) + \dot{e}(t) = 0$$

$$\Rightarrow e(t) = e(0)e^{-ct}$$



Suponga la función de Lyapunov  $V(t) = \frac{s^2(t)}{2}$  y

$\therefore \dot{V}(t) = s(t)\dot{s}(t) \leq 0$  con la derivada de la superficie

$$\dot{s}(t) = c\dot{e}(t) + \ddot{e}(t) = c\dot{e}(t) + \ddot{\theta}(t) - \ddot{\theta}_d(t)$$

$$= c\dot{e}(t) + \frac{u(t) + d(t)}{J} - \ddot{\theta}_d(t)$$

$$s(t)\dot{s}(t) = s(t) \left( c\dot{e}(t) + \frac{u(t) + d(t)}{J} - \ddot{\theta}_d(t) \right) < 0$$

Seleccionando:  $u(t) := J(-c\dot{e}(t) + \ddot{\theta}_d(t)) - \eta s(t) - D \operatorname{sgn} s(t)$

$$\therefore s(t)\dot{s}(t) = s(t) \left( \frac{-\eta s(t) - D \operatorname{sgn} s(t)}{J} + \frac{d(t)}{J} \right)$$

$$\dot{V}(t) = \frac{1}{J} \left( \underbrace{-\eta s^2(t)}_{\text{neg}} - \underbrace{s(t)D \operatorname{sgn} s(t)}_{\text{neg}} + \underbrace{s(t)d(t)}_{\text{?}\pm} \right) \leq -\frac{\eta}{J} s^2(t) < -\frac{2\eta}{J} V(t)$$

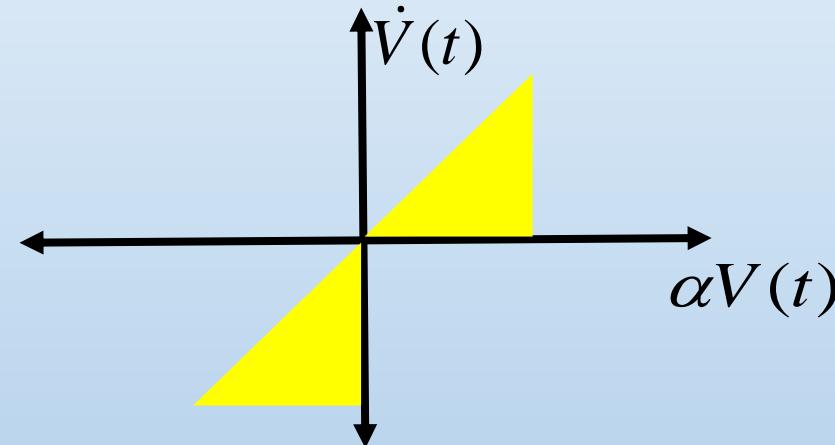


Por tanto aplicando el siguiente Lema a la desigualdad anterior

Lema: Sea  $f(t)$  y  $V(t):[0,\infty) \in R$ , entonces

$$\dot{V}(t) \leq -\alpha V(t) + f(t), \quad \forall t \geq t_0 \geq 0 \quad \Rightarrow$$

$$V(t) \leq e^{-\alpha(t-t_0)}V(t_0) + \int_{t_0}^t e^{-\alpha(t-\tau)}f(\tau)d\tau$$

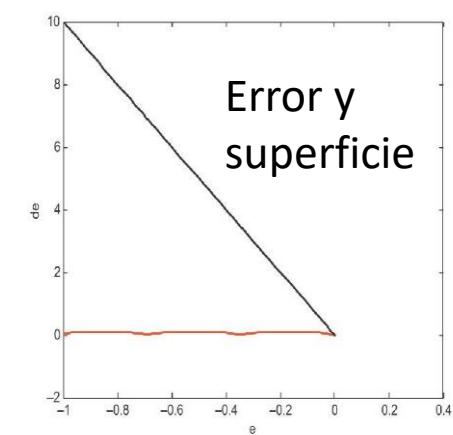
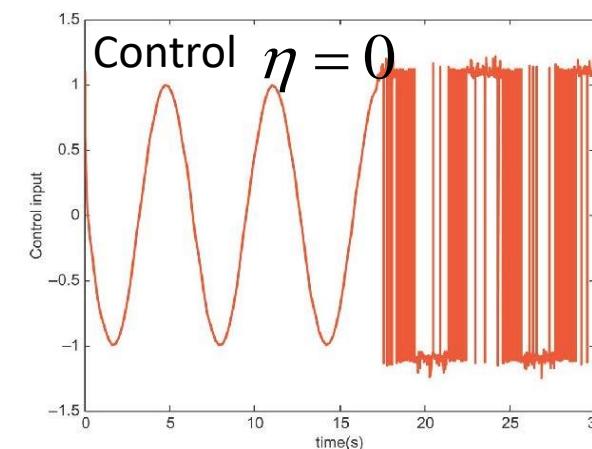
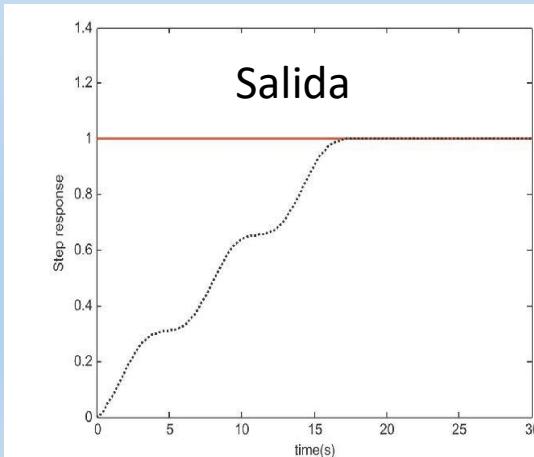
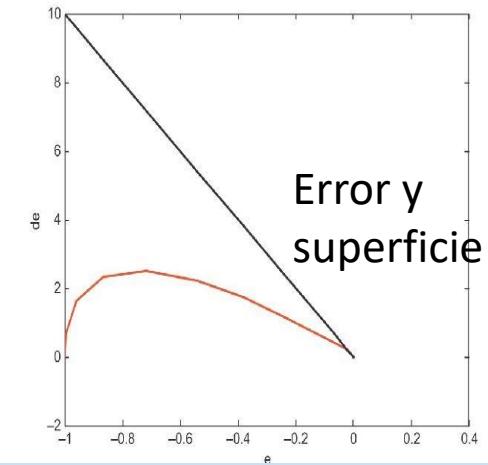
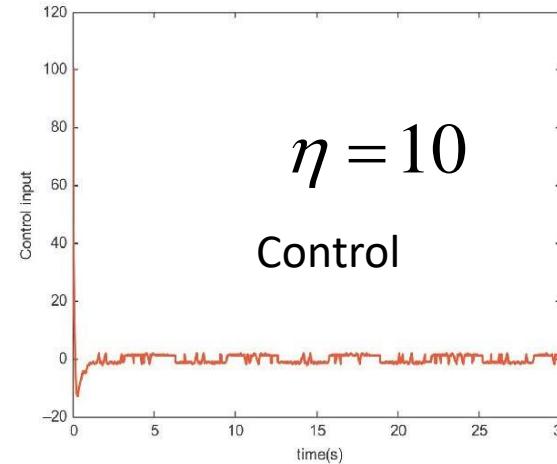
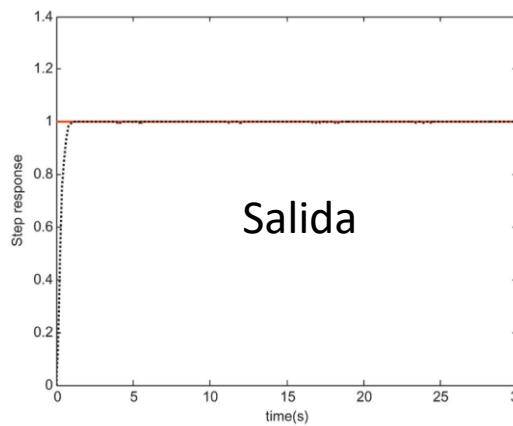


$$\dot{V}(t) \leq -\frac{2\eta}{J}V(t) \quad \Rightarrow \quad V(t) \leq e^{-\frac{2\eta}{J}(t-t_0)}V(t_0) \text{ está acotada}$$

La convergencia a las superficie del deslizamiento esta garantizada, si se selecciona adecuadamente D, en caso contrario el efecto de Chattering se manifiesta.



# Ejemplo: $J = 2$ , $d(t) = \sin(t)$ , $\theta_d = 1$ , $c = 10$ , $\eta = (0, 10)$ , $D = 1.1$



Tomado del otro libre de Liu



# SMC+RBF para un Sistema de 2.orden

$$\dot{x}_1(t) = x_2(t)$$

$$\dot{x}_2(t) = f(x) + bu(t) + d(t) \quad |d| \leq D$$

$$e = x_1 - x_d$$

1. Paso SM :  $s = \dot{e} + ce \quad c > 0 \quad \ddot{s} = \ddot{\dot{e}} + c\dot{e} = \ddot{x}_1 - \ddot{x}_d + c\dot{e}$

$$\dot{s} = \cancel{f(x)} + bu(t) + d(t) - \cancel{\ddot{x}_d} + c\dot{e}$$

Si  $f(x)$  y  $b$  son conocidas se propone

$$u = \frac{1}{b}(-\cancel{f(x)} + \ddot{x}_d - \cancel{c\dot{e}} - \eta \operatorname{sgn}(s))$$

$$\dot{s} = -\eta \operatorname{sgn}(s) + d(t) \Rightarrow s\dot{s} = -\eta s \operatorname{sgn}(s) + sd(t)$$

2. Paso. Considerando  $\eta \geq D$  y seleccionado

$$V(s) = \frac{1}{2}s^2, \quad V(s) = s\dot{s} \leq 0$$

Si  $f(x)$  es desconocida se puede usar una RBF para identificarla



# SMC+RBF Entrada de RBF $x^T = [x_1 \quad x_2]$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \color{red}f(x) + bu(t) + d(t)\color{black} \quad |d| \leq D$$

De acuerdo a lo visto en el capítulo anterior:

$$h_j = \exp\left(\frac{\|x - c_i\|^2}{2b_j^2}\right) \Rightarrow \hat{f}(x) = (W^*)^T h(x) + \varepsilon$$

$$\dot{s} = f + bu + d(t) - \ddot{x}_d + c\dot{e}, \quad u = \frac{1}{b} \left( -\hat{f}(x) + \ddot{x}_d - c\dot{e} - \eta \operatorname{sgn}(s) \right),$$

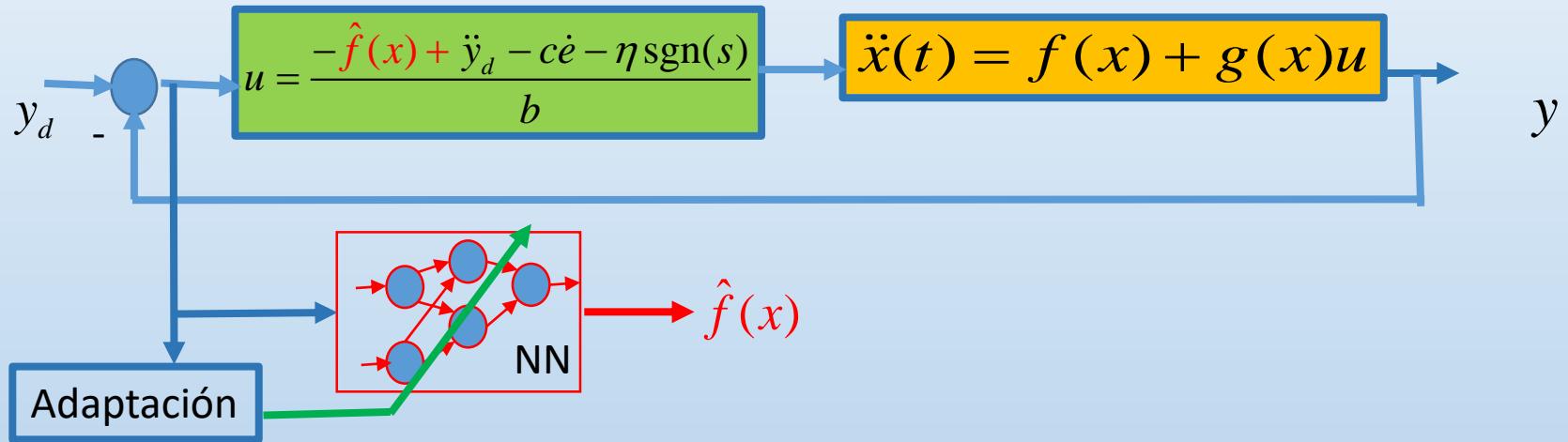
$$\dot{s} = f + \underbrace{(-\hat{f} + \ddot{x}_d - c\dot{e} - \eta \operatorname{sgn}(s))}_{u} + d(t) - \ddot{x}_d + c\dot{e}$$

$$\dot{s} = f - \hat{f} - \eta \operatorname{sgn}(s) + d(t) = \tilde{f} - \eta \operatorname{sgn}(s) + d(t)$$

$$\tilde{f} = \underbrace{(W^*)^T h(x) - \hat{W}^T h(x)}_{\tilde{W}^T h(x)} + \varepsilon = \tilde{W}^T h(x) + \varepsilon$$



# Si se desconoce $f(x)$





Definiendo la función de Lyapunov:  $L = \frac{s^2}{2} + \frac{\gamma}{2} \tilde{W}^T \tilde{W}$   $\gamma > 0$

$$\begin{aligned}\dot{L} &= s\dot{s} + \gamma \tilde{W}^T \dot{\tilde{W}} = s\left(\tilde{f} - \eta \operatorname{sgn}(s) + d(t)\right) - \gamma \tilde{W}^T \hat{\tilde{W}} \\ &= \tilde{s}\left(\tilde{W}^T h(x) + \varepsilon - \eta \operatorname{sgn}(s) + d(t)\right) - \gamma \tilde{W}^T \hat{\tilde{W}} \\ &= \tilde{W}^T \underbrace{\left(sh(x) - \gamma \hat{\tilde{W}}\right)}_0 + s\left(\varepsilon - \eta \operatorname{sgn}(s) + d(t)\right)\end{aligned}$$

$\therefore$  La adaptación de la RBF  $\hat{\tilde{W}} = \frac{sh(x)}{\gamma}$

$$\dot{L} = s\left(\varepsilon - \eta \operatorname{sgn}(s) + d(t)\right) = s\left(\varepsilon + d(t)\right) - (\eta s \operatorname{sgn}(s))$$

$$\eta \geq \varepsilon + D + \eta_0 > 0$$

Se puede aproximar:  $\dot{L} \leq -\eta s \operatorname{sgn}(s) \leq 0$

$$\int_0^t \dot{L} dt \leq -\eta_0 \int_0^t s \operatorname{sgn}(s) dt \quad L(t) - L(0) \leq -\eta_0 \int_0^t s \operatorname{sgn}(s) dt$$



Lema de Barbalat: Si  $\lim_{t \rightarrow \infty} \int_0^t f(\tau) d\tau$  existe y es finito, y

$f(t)$  es una función uniformemente continua  $\Rightarrow \lim_{t \rightarrow \infty} f(t) = 0$

$\Rightarrow V(t)$  está limitada, s y  $\tilde{W}$  también están limitadas  
además s está limitada y por tanto la integral

$\int_0^t s \operatorname{sgn}(s) dt$  también está limitada.

De acuerdo con el Lema de Barbalat

cuanto t tiende al infinito el error y su derivada  $\rightarrow 0$



## Ejemplo:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = f(x) + 100u + d(t)$$

$$\text{con } f(x) = x_1 + x_2 \text{ y } d(t) = \cos t, x_d = \sin t, \quad x(0) = \begin{pmatrix} 0.2 \\ 0 \end{pmatrix}$$

1) Para el SMC se selecciona  $c=10$ ,  $\eta=10$ ,  $\gamma=0.01$

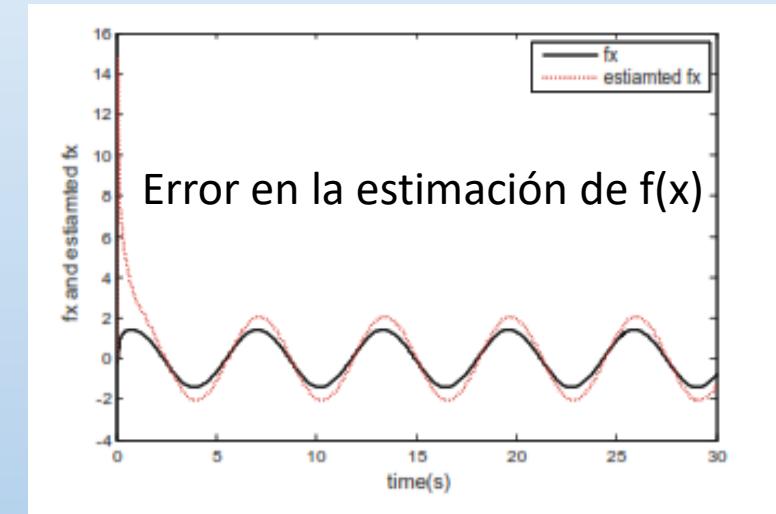
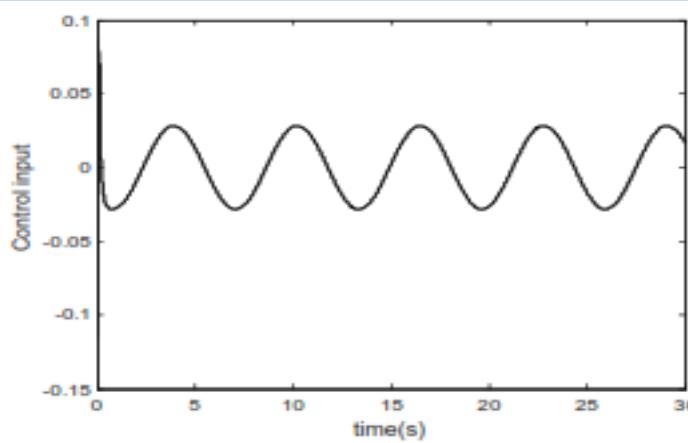
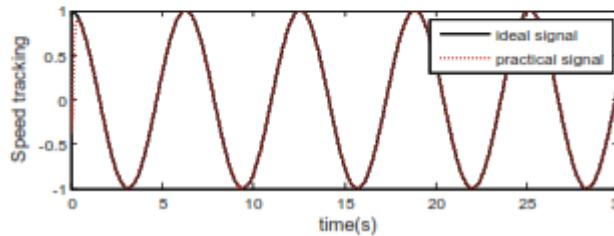
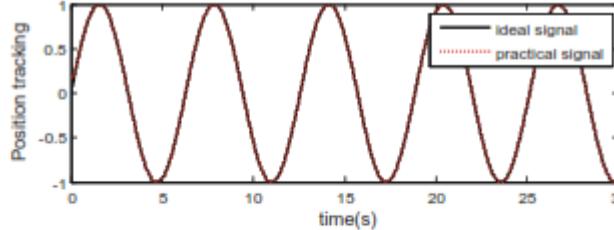
Para eliminar el chattering se usa la función saturación en lugar del  $ssign(s)$ , con  $\Delta=.05$

$$sat(s) = \begin{cases} 1 & s > \Delta \\ ks & ssign(s) \leq \Delta \quad k = 1/\Delta \\ -1 & s < -\Delta \end{cases}$$

2) Para la estructura de la RBF (2-5-1) con  $c=[-1 \quad -0.5 \quad 0 \quad 0.5 \quad 1]$ ,  $b_j = 5$



# Comportamiento





# RBF implementación discreta con T=1 ms motivación

$$\dot{x}_1 = x_2$$

$\dot{x}_2 = f(x) + u$ ; con  $f(x)$  = desconocida, error:  $e = x_1 - x_d$ ,  $\dot{e} = x_2 - \dot{x}_d$

1) Diseño del modo de deslizamiento:  $\sigma = \lambda e + \dot{e}$ ,  $\dot{\sigma} = \lambda \dot{e} + \ddot{e}$  con  $\lambda > 0$

$$\dot{\sigma} = \lambda \dot{e} + f(x) + u - \ddot{x}_d \quad \text{con} \quad u = -\lambda \dot{e} - \hat{f}(x) + \ddot{x}_d - \eta \sigma \operatorname{sgn}(\sigma)$$

2) Estimación  $\hat{f}(x) = (W^*)^T h(x) + \varepsilon$  con  $h_j(x) = \exp\left(\frac{\|x - c_i\|^2}{2b_j^2}\right)$

$i$  = #de entradas,  $j$  = # de nodos de la capa escondida,

$W^*$  = peso ideal de la RBF,  $\varepsilon < \varepsilon_N$  error de RBF

3) Prueba de estabilidad  $V = \frac{\sigma^2}{2} + \frac{\tilde{W}^T \tilde{W}}{2\gamma} \quad \gamma > 0$ ,  $\tilde{W} = \hat{W} - W^*$

$$\dot{V} = \varepsilon \sigma - \eta \sigma \operatorname{sgn}(\sigma) + \tilde{W}^T \underbrace{\left( \frac{\dot{\tilde{W}}}{\gamma} - \sigma h(x) \right)}_0$$

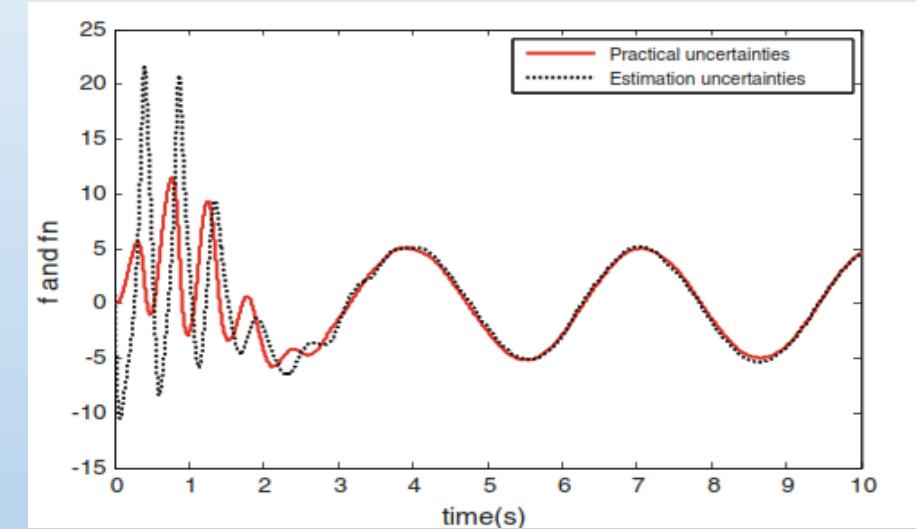
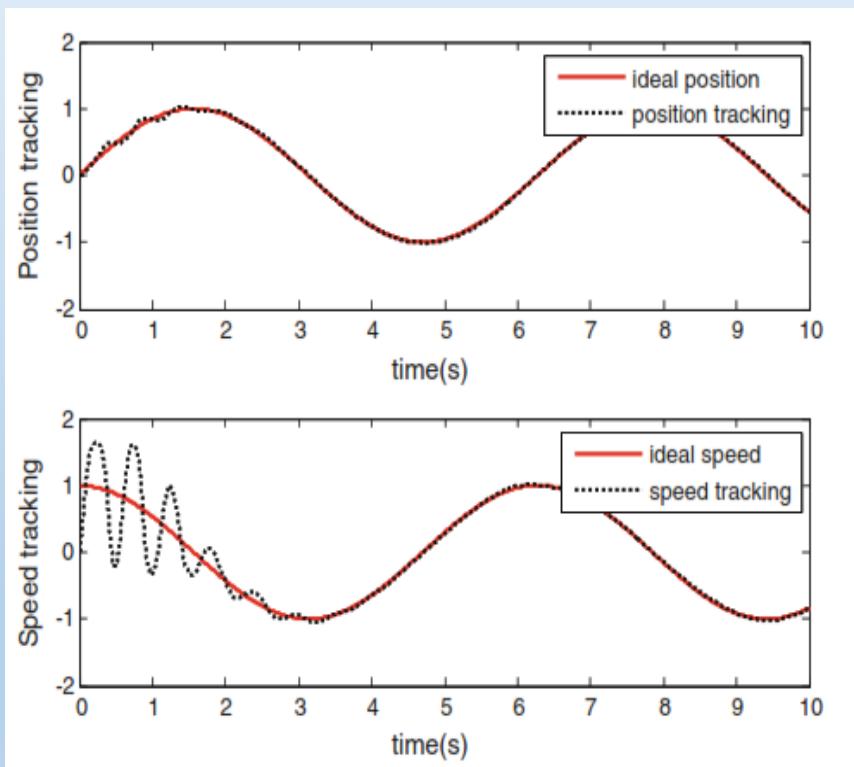
NOTA: Si  $\dot{V} = 0 \Rightarrow$  la RBF no converge a  $W^*$

# Implementación

$$f(x) = 10x_1 x_2, \quad x_d = \sin t, \quad x(0) = [0.5 \ 0]^T, \quad \lambda = 200, \quad \eta = 0.20, \quad \gamma = 100$$

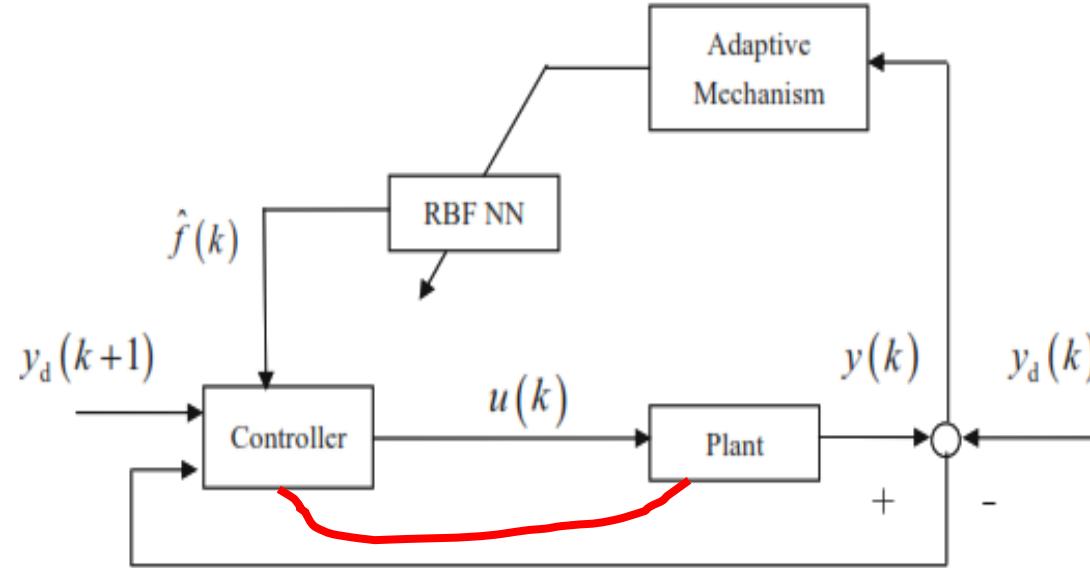
$$\text{RBF (2-5-1), } c_i = [-1 \ 0.5 \ 0 \ 0.5 \ 1], \quad b_j = 3, \quad W(0) = 0$$

discretizando el sistema con  $T_s = 1ms$ .



Diseñar el controlador y el análisis de estabilidad para sistemas discretos.

# Modelo discreto



$$y(k+1) = f(x(k)) + u(k), \quad x(k) = [y(k) \quad y(k-1) \quad \dots \quad y(k-n-1)]^T$$

Entrada  $u(k)$ , Salida  $y(k)$

1) Sea el control:  $u(k) = y_d(k-1) - f(x(k)) - c_1 e(k) \Rightarrow$

la ecuación del error se reduce a

$$e(k+1) + c_1 e(k) = 0 \quad |c_1| < 1$$

si se conoce  $f(x)$ .



# Cuando se desconoce $f(x(k))$

Se usa la RBF:  $\hat{f}(x(k)) = \hat{W}(k)^T h(x(k))$

Si esta entrenada  $f^*(x(k)) = \hat{f}(x(k), W^*(k)) - \Delta_f(x)$        $|\Delta_f(x)| < \varepsilon_f$

y

$$\begin{aligned}\tilde{f}(x(k)) &= f(x(k)) - \hat{f}(x(k)) = \hat{f}(x(k), W^*) - \Delta_f(x(k)) - \hat{W}(k)^T h(x(k)) \\ &= -\tilde{W}(k)^T h(x(k)) - \Delta_f(x(k)) \quad \text{con} \quad \tilde{W}(k) = W(k) - W^*\end{aligned}$$

Se tiene la ley de control

$$u(k) = y_d(k+1) - \hat{f}(x(k)) - c_1 e(k)$$

y la ecuación del error se reduce a:  $e(k+1) = \tilde{f}(x(k)) - c_1 e(k)$

$$e(k) + c_1 e(k-1) = \tilde{f}(x(k-1)) \quad \text{o} \quad e(k) = \Gamma^{-1}(z^{-1}) \tilde{f}(x(k-1)) \quad \text{o}$$

$$e(k) - \Gamma^{-1}(z^{-1}) \tilde{f}(x(k-1)) = 0$$

con  $\Gamma(z^{-1}) = 1 + c_1 z^{-1}$       y     $z^{-1}$  el operador retardo discreto

Se define con  $\beta > 0$

$$e_1(k) = \beta(e(k) - \Gamma^{-1}(z^{-1}) v(k))$$



Reescribiendo

$$e(k) = \Gamma^{-1}(z^{-1}) \tilde{f}(x(k-1)), \quad e_1(k) = \beta(e(k) - \Gamma^{-1}(z^{-1})v(k))$$

$$e_1(k) = \beta \Gamma^{-1}(z^{-1}) (f(x(k-1)) - v(k)) = \frac{\beta}{1 + c_1 z^{-1}} (\tilde{f}(x(k-1)) - v(k))$$

$\Rightarrow$

$$e_1(k-1) = \frac{\beta (\tilde{f}(x(k-1)) - v(k)) - e_1(k)}{c_1}$$

$$\Delta \hat{W}(k) = \begin{cases} \frac{\beta}{\gamma c_1^2} h(x(k-1)) e_1(k) & \text{if } |e_1(k)| > \varepsilon_f \\ 0 & \text{if } |e_1(k)| \leq \varepsilon_f \end{cases}$$

$$\text{con } \Delta \hat{W}(k) = \hat{W}(k) - \hat{W}(k-1).$$

LA PRUEBA DE ESTABILIDAD ESTÁ EN EL LIBRO

# Ejemplo:

$$y(k) = \underbrace{\frac{0.5 y(k-1)(1 - y(k-1))}{1 + \exp(-0.25 y(k-1))}}_{f(x(k-1))} + u(k-1) \quad \text{con } c_1 = -0.01$$

$RBF(1,9,1)$ ,  $y_d = \sin t$ .

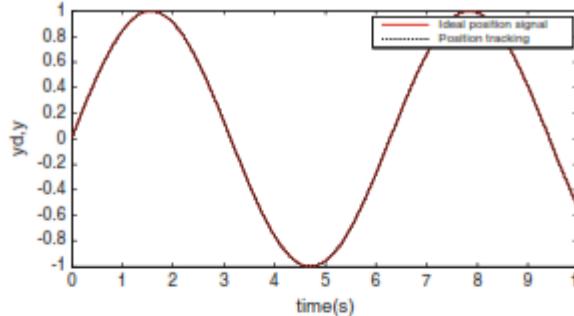


Fig. 10.4 Position tracking

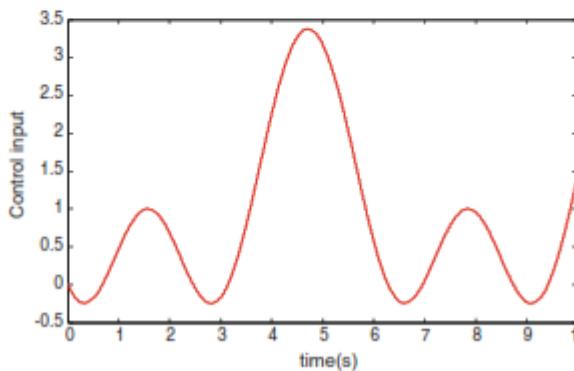


Fig. 10.5 Control input

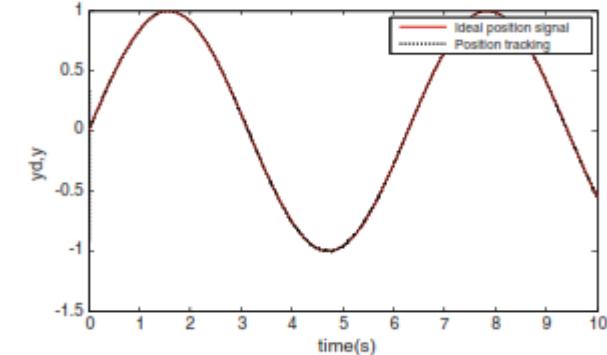
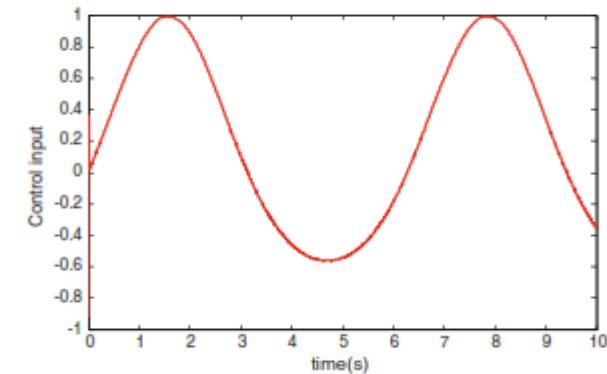
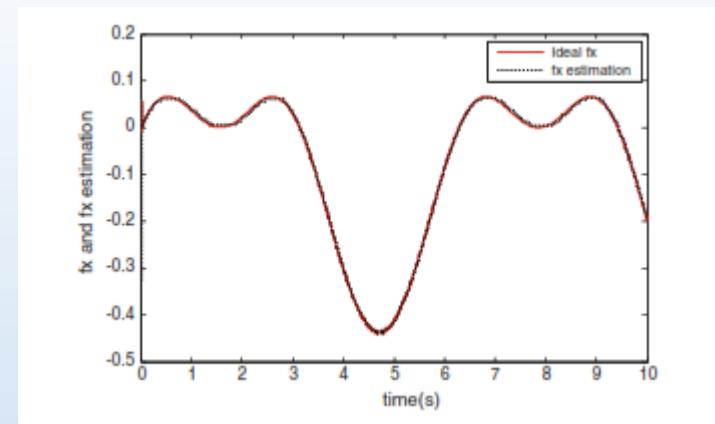


Fig. 10.6 Position tracking



Conociendo  $f(x)$

Estimando la RBF la  $f(x)$



Valor estimado de  $f(x)$



# Ejemplo: Sistema lineal

$$\dot{x}_1 = x_2$$

$$x_1(0) = \pi / 60 \quad x_2(0) = 0$$

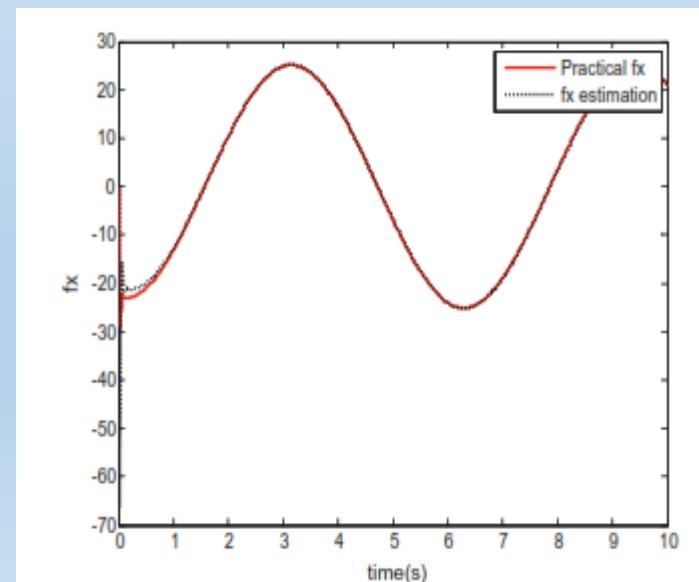
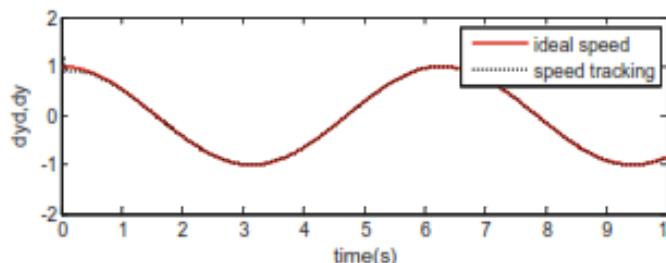
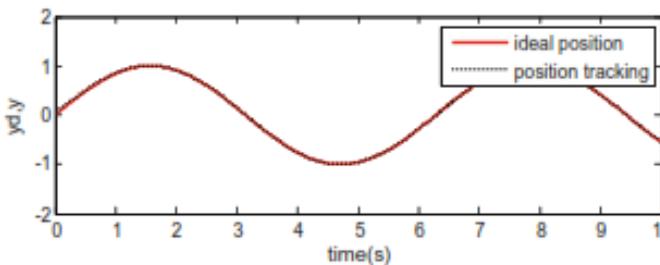
$$\dot{x}_2 = f(x) + g(x)u = -25x_2 + 133u$$

$$y = x_1$$

$$y_d = \sin t$$

Para la RBF(2-5-1):  $c = [-1 \quad -0.5 \quad 0 \quad 0.5 \quad 1]$ ,  $b = 1.$ ,  $w = 0.$

Para el control:  $Q = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix}$ ,  $k_d = 50$ ,  $k_p = 30$ ,  $\gamma = 1000$ .





# Tarea 5 control DI+RBF como estimador

Se entrega el 10 de enero de 2023

Considere el sistema:

$$\dot{x}_1 = x_2 \quad \text{con } x_1(0) = \pi / 60, \quad x_2(0) = 0,$$

$$\dot{x}_2 = f(x) + 15u \quad f(x) = -20x_2 - x_1,$$

$$y = x_1 \quad \text{la señal de seguimiento } y_d = 0.5(\cos t)(\sin 0.5t)$$

$$\text{y el control } u^* = \frac{-\hat{f}(x) - \ddot{y}_d + K^T E}{15} \quad \text{con } e = y_d - y, \quad E = [e \quad \dot{e}]^T.$$

- Seleccione los parámetros del control  $K = [k_p \quad k_d]^T$  tal que el sistema sea estable asumiendo  $f(x)$  conocida.
- Seleccione los parámetros  $b_1$  y  $c$  de una RBF con estructura (2 – 5 – 1) y entrene la red con la función gradiente tal que  $\hat{f}(x)$  sea estimada.
- Sustituya la  $\hat{f}(x)$  en la ley  $u^*$  y compare en simulación la diferencia del sistema cuando se conoce  $f(x)$  y con estimación.



# Tarea 6 control con SM+RBF como estimador

Consideré el sistema:

Se entrega el 17 de enero de 2023

$$\dot{x}_1 = x_2 \quad \text{con } x_1(0) = \pi / 60, \quad x_2(0) = 0,$$

$$\dot{x}_2 = f(x) + 15u + d \quad f(x) = -20x_2 - x_2 x_1, \quad \text{perturbación } d(t)=5$$

$$y = x_1 \quad \text{señal de seguimiento } y_d = 0.5(\cos t)(\sin 0.5t)$$

$$\text{con } e = y_d - y,$$

a) Diseñe un control de modo de deslizamiento asumiendo  $f(x)$  conocida:

$$\sigma = \lambda e + \dot{e}, \quad \dot{\sigma} = \lambda \dot{e} + \ddot{e} \quad \text{con } \lambda > 0$$

b) Seleccione los parámetros  $b_1$  y  $c$  de una RBF con estructura (2 – 5 – 1) y entrene la red con la función gradiente tal que  $\hat{f}(x)$  sea estimada. Se pueden usar los mismos parámetros que en la tarea 5.

c) Sustituya la  $\hat{f}(x)$  en el control por SM y compare en simulación la diferencia del sistema cuando se conoce  $f(x)$  y con estimación.