



Optimización dinámica de procesos

Prof. Cesar de Prada

Dpt. Ingeniería de Sistemas y
Automática

Universidad de Valladolid, España



Operación óptima (económica)

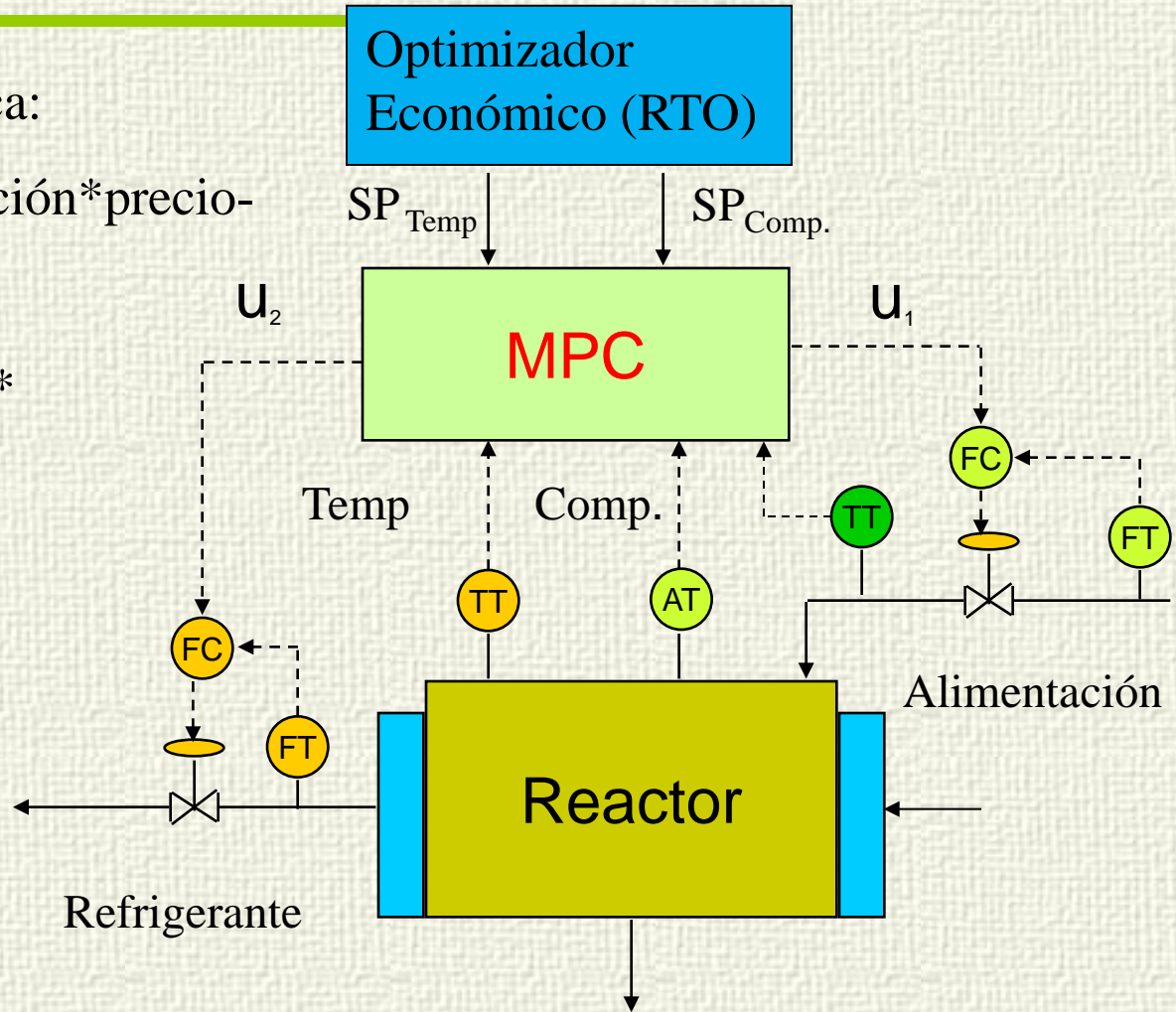


Función de costo económica:

(Flujo producto*concentración*precio -
- materia prima*precio -
-Flujo refrigerante*precio)*
tiempo

Max Beneficio

**Con las mismas
restricciones y el modelo
estacionario del MPC**





Ejemplo: Reactor químico



$$\text{Max } J = q c_B$$

$$V \frac{dc_A}{dt} = q(c_{Ai} - c_A) - V\beta e^{-E/RT} c_A$$

$$V\rho c_p \frac{dT}{dt} = q\rho c_p (T_i - T) - Vkc_A H - UA(T - T_r)$$

$$V_r \rho_r c_{pr} \frac{dT_r}{dt} = F_r \rho_r c_{pr} (T_{ri} - T_r) + UA(T - T_r)$$

$$c_B = c_{Ai} - c_A$$

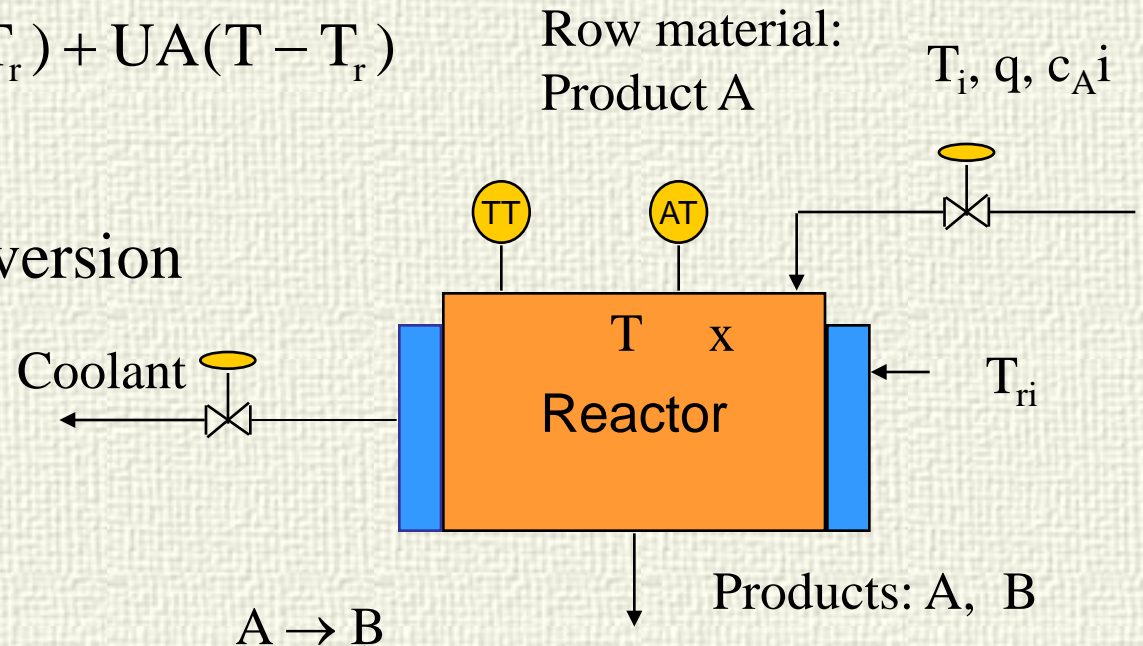
$$x = c_B / c_{Ai} \quad x \text{ conversion}$$

$$T_{\min} \leq T \leq T_{\max}$$

$$x_{\min} \leq x \leq 1$$

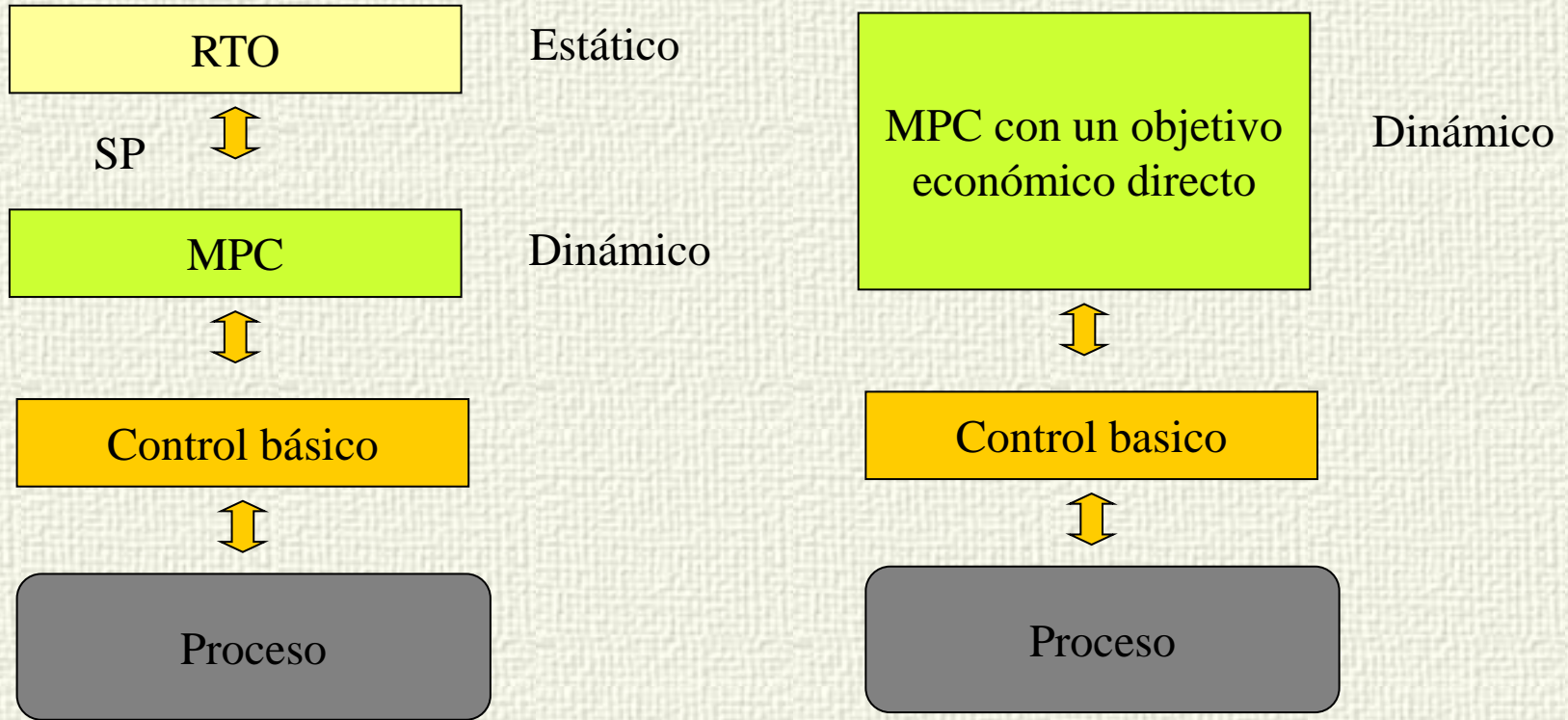
$$q_{\min} \leq q \leq q_{\max}$$

$$F_{r\min} \leq F_r \leq F_{r\max}$$





Operación dinámica de procesos óptima (económica)



$$\min_u J(x, u) \quad \min_u J = \int_0^T (x - SP)^2 dt$$

$$f(x, u) = 0 \quad F(\dot{x}, x, u) = 0$$

$$g(x, u) \leq 0 \quad g(x, u) \leq 0$$

$$\min_u J = \int_0^T L(x, u) dt \quad \text{Economic cost function}$$

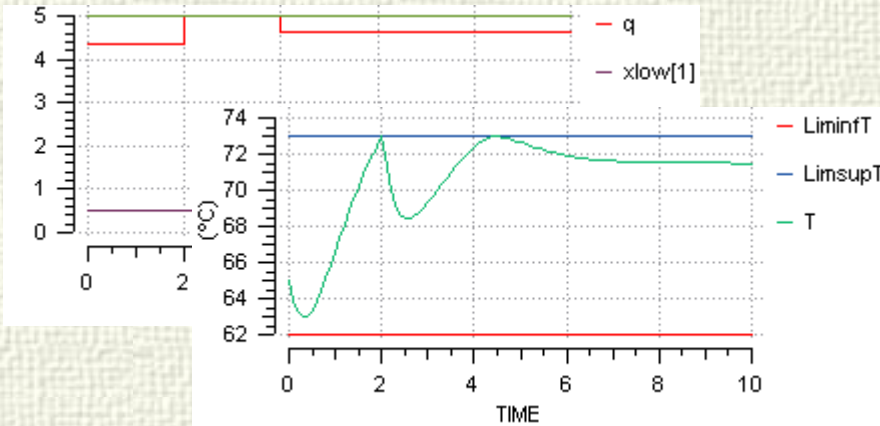
$$F(\dot{x}, x, u) = 0$$

$$g(x, u) \leq 0$$



Transitorio óptimo

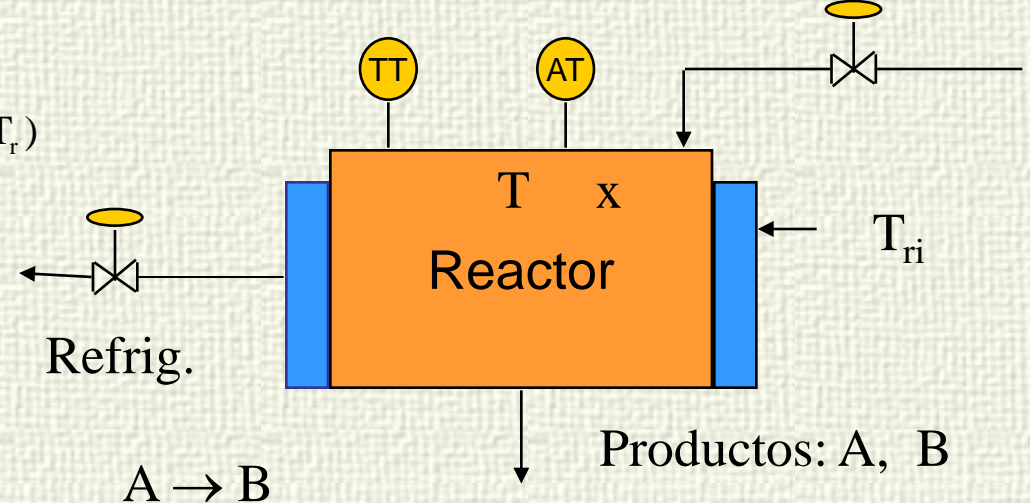
¿Como llegar al punto de operación óptimo en mínimo tiempo respetando las restricciones?



$$V\rho c_p \frac{dT}{dt} = q\rho c_p (T_i - T) + Vkc_A H - UA(T - T_r)$$

$$V_r \rho_r c_{pr} \frac{dT_r}{dt} = F_r \rho_r c_{pr} (T_{ri} - T_r) + UA(T - T_r)$$

$$V \frac{dc_A}{dt} = q(c_{Ai} - c_A) - V\beta e^{-E/RT} c_A$$





Parametrización de un modelo dinámico



Modelo de producción de un producto P por microorganismos X que se alimentan de un sustrato S

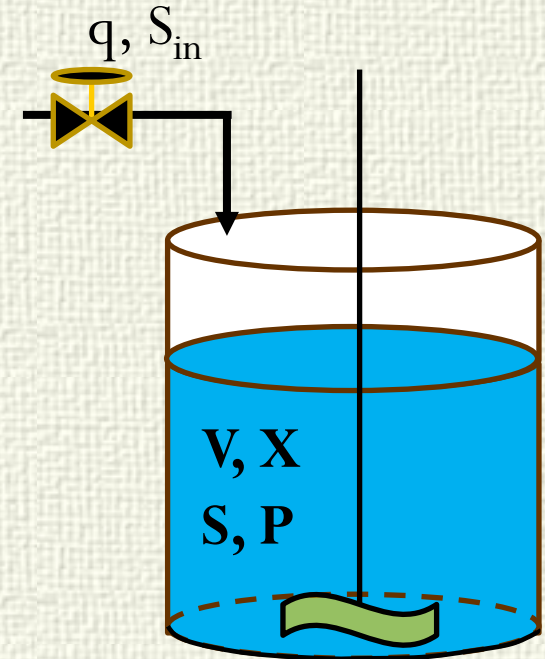
$$\dot{X} = \mu X - \sigma X \quad X(0) = X_0$$

$$\dot{S} = -\frac{\mu X}{Y_x} + \frac{\sigma X}{Y_s} + qS_{in} \quad S(0) = S_0$$

$$\dot{P} = vP \quad P(0) = P_0$$

$$\dot{V} = q \quad V(0) = V_0$$

$$\mu = \frac{\mu_m S}{K_m + S + \frac{S^2}{K_i}} \quad v = \frac{v_m S}{K_0 + S}$$

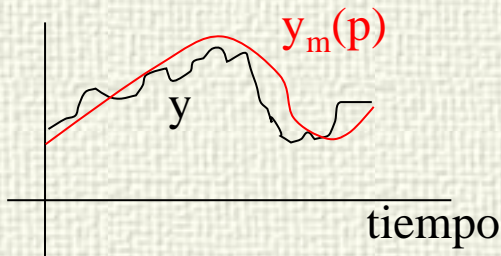




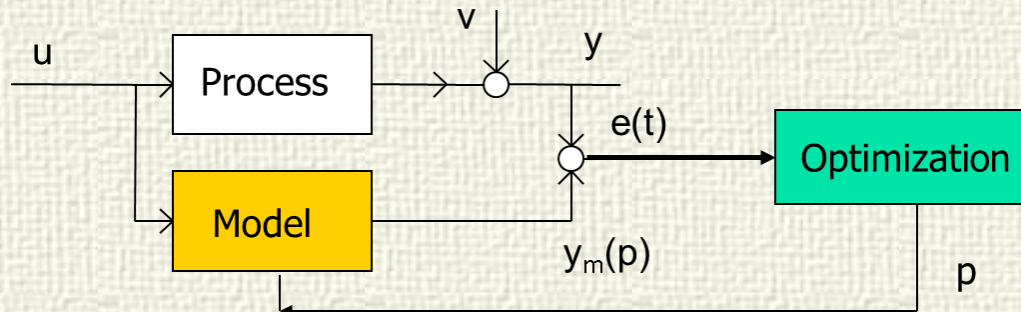
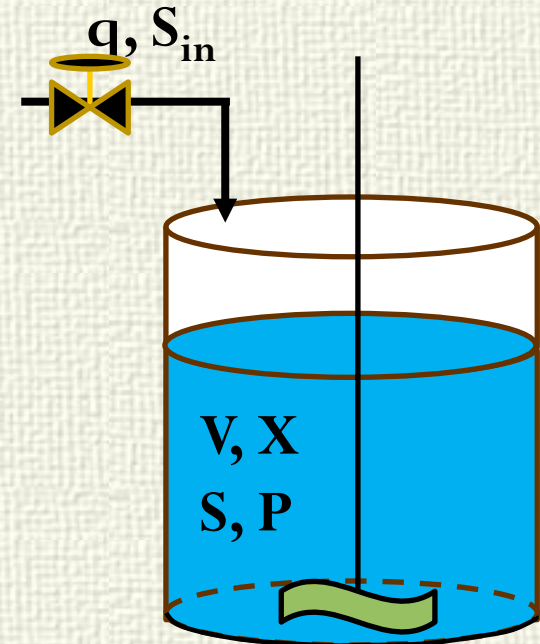
Parametrización de un modelo dinámico



Datos experimentales



$$u(t) \in U \quad q = u(t)$$





Parametrización de un modelo dinámico



$$\min_{\mu_m, K_0, v_m, K_i, X_0} J(u) = \int_0^T [(P - P_{\text{meas}})^2 + (s - s_{\text{meas}})^2] dt$$

s. t.:

$$\dot{X} = \mu X - \sigma X \quad X(0) = X_0$$

$$\dot{S} = -\frac{\mu X}{Y_x} + \frac{\sigma X}{Y_s} + q S_{\text{in}} \quad S(0) = S_0$$

$$\dot{P} = v P \quad P(0) = P_0$$

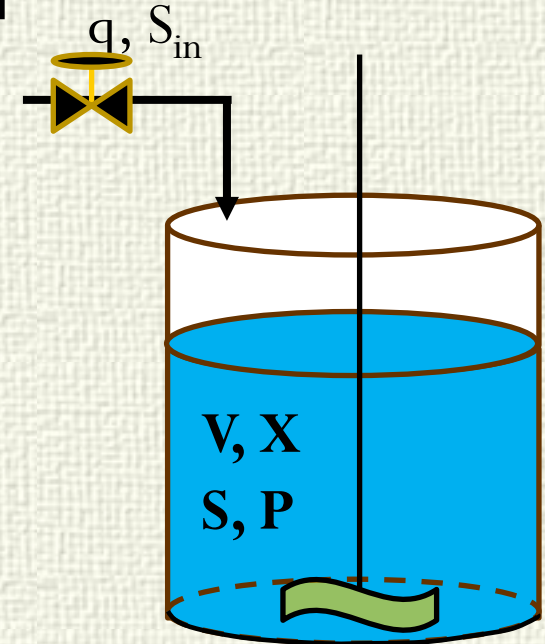
$$\dot{V} = q \quad V(0) = V_0$$

$$\mu = \frac{\mu_m S}{K_m + S + \frac{S^2}{K_i}} \quad v = \frac{v_m S}{K_0 + S}$$

$$q = u(t)$$

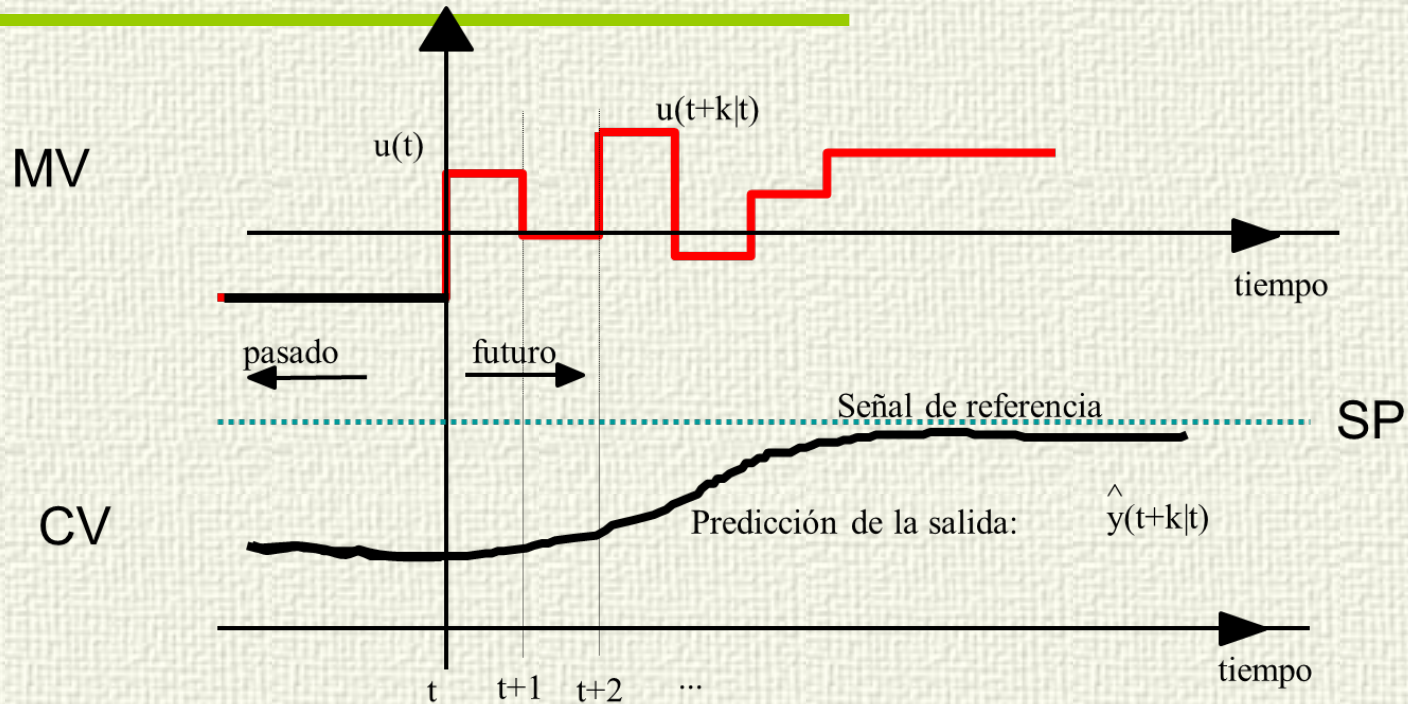
$$K_0 > 0, \mu_m > 0, \dots$$

$$X(t) \leq X^{UP}$$





Control Predictivo (MPC)



$$\min_{u(t), u(t+1), \dots} J = \sum_{j=N1}^{N2} [\hat{y}(t+j) - w(t+j)]^2 + \sum_{j=0}^{Nu-1} [\beta \Delta u(t+j)]^2$$

Sujeto al modelo dinámico del proceso

Cumpliendo unas restricciones sobre las MV y CV (OP/PV)



Maximizar la producción de P en un tiempo dado



$$\min_{u(t)} J = -P(t_f)$$

s. t.:

¿Como alimentar el reactor
respetando restricciones y
maximizando P?

$$\dot{X} = \mu X - \sigma X$$

$$X(0) = X_0$$

$$\dot{S} = -\frac{\mu X}{Y_x} + \frac{\sigma X}{Y_s} + qS_{in}$$

$$S(0) = S_0$$

$$\dot{P} = vP$$

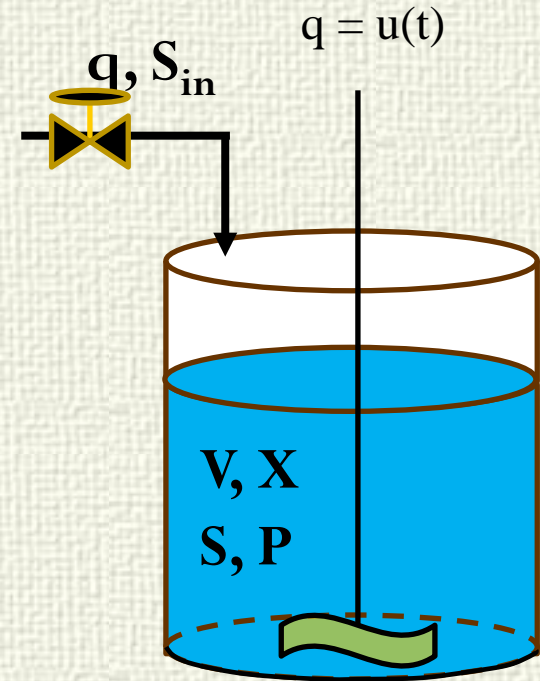
$$P(0) = P_0$$

$$\dot{V} = q$$

$$V(0) = V_0$$

$$\mu = \frac{\mu_m S}{K_m + S + \frac{S^2}{K_i}}$$

$$v = \frac{v_m S}{K_0 + S}$$



$$u(t) \in U$$

$$X(t) \leq X^{UP}$$

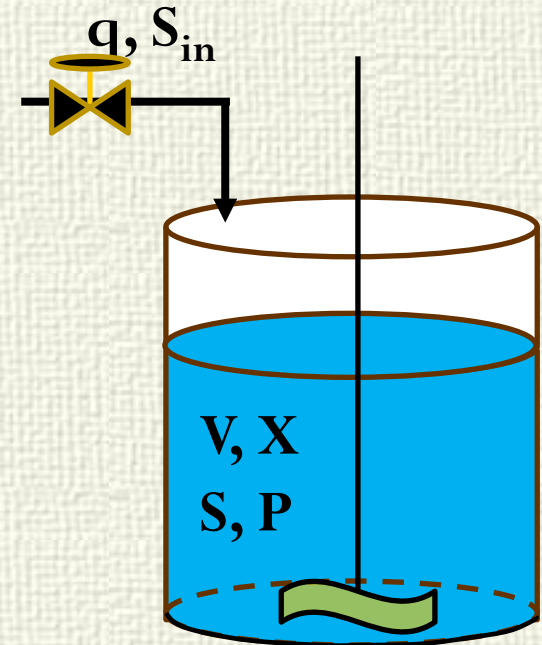
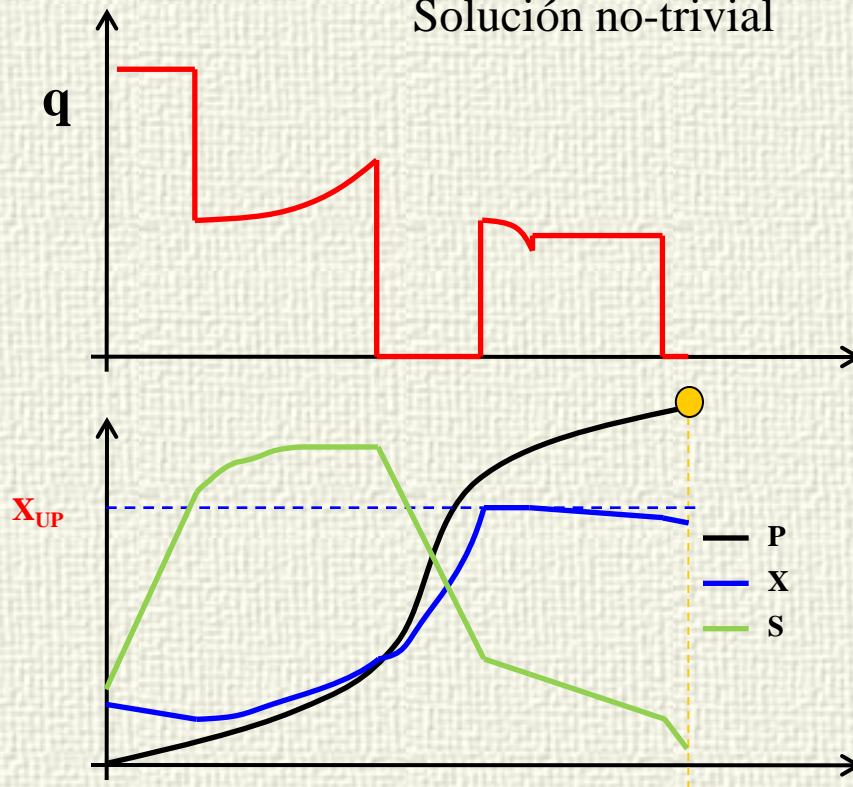
Path
constraint



Maximizar la producción de P en un tiempo dado



Solución no-trivial





Optimización dinámica (DO)

$$\min_{\mathbf{u}(t), \mathbf{x}_0, t_f} J(\mathbf{u}) = \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}) dt$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z}), \quad \mathbf{x}(t_0) = \mathbf{x}_0$$

$$\mathbf{h}(\mathbf{x}, \mathbf{u}, \mathbf{z}) = \mathbf{0}$$

$$\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{z}) \leq \mathbf{0}$$

DAE

✓ Muchos tipos:

- ✓ Problemas de valor inicial
- ✓ Problemas TPBV
- ✓ Problemas de tiempo mínimo
- ✓ DAE u ODE
- ✓ Híbridos
- ✓ Coste algebraico o integral
- ✓

Dynamic Optimization (DO)

Son problemas mas intensivos en cálculo que los NLP

Algunas de las restricciones son ecuaciones diferenciales

Las decisiones se toman a lo largo del tiempo



Optimización dinámica

$$\min_u J(u) = \int_0^T L(x, u) dt$$

Función de coste J

x estados

$$F(\dot{x}, x, u) = 0$$

Modelo DAE

u variables

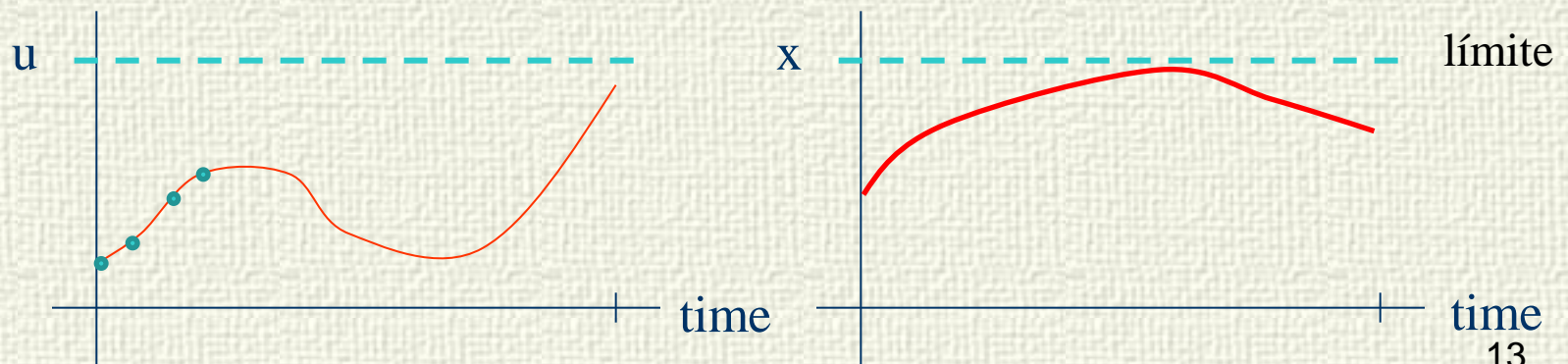
$$g(x, u) \leq 0$$

Restricciones

de decisión

u puede ser un conjunto de parámetros o un conjunto de variables que evolucionan a lo largo del tiempo

Dos problemas: Número infinito de variables de decisión y de restricciones





Métodos de solución

$$\min_u J(u) = \int_0^T L(x, u) dt$$

$$F(\dot{x}, x, u) = 0$$

$$g(x, u) \leq 0$$

Función de coste J

Modelo DAE

Restricciones

x estados

u variables
de decisión

Métodos indirectos

Se calculan las condiciones necesarias de optimalidad mediante cálculo de variaciones



Problema de contorno en dos puntos

Métodos Directos

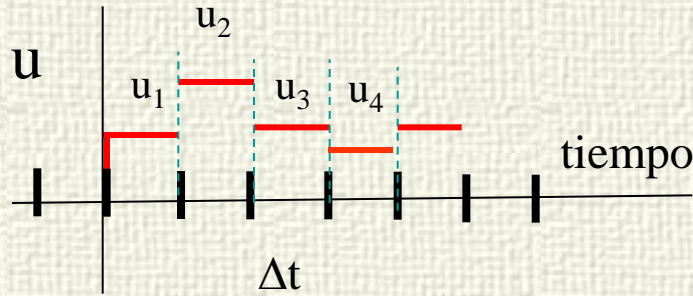
Se aproxima la solución mediante discretización de las variables dependientes del tiempo



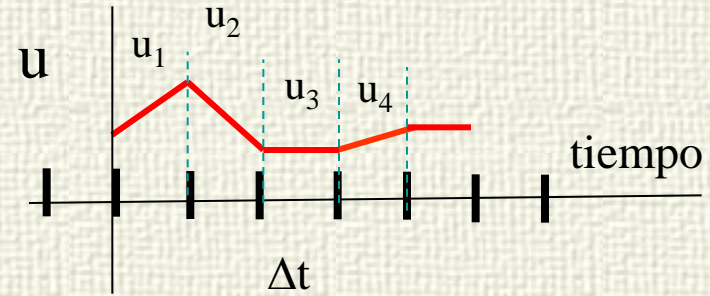
Programación no-lineal NLP



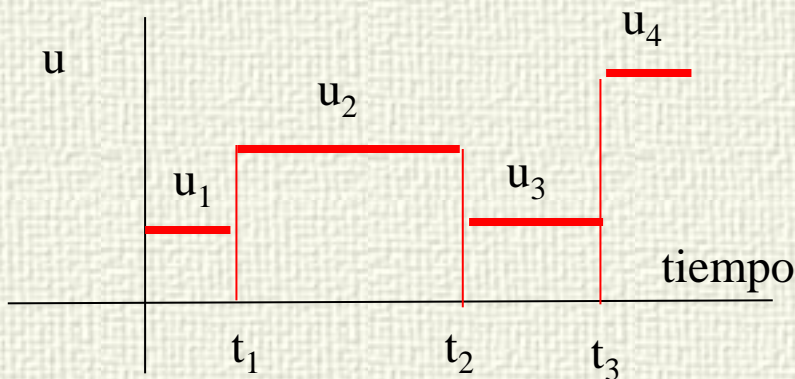
Parametrización del vector de control (CVP)



$$u_i = p_i$$



$$u_i = p_i t + b_i$$



$$p_i = u_i, t_i$$

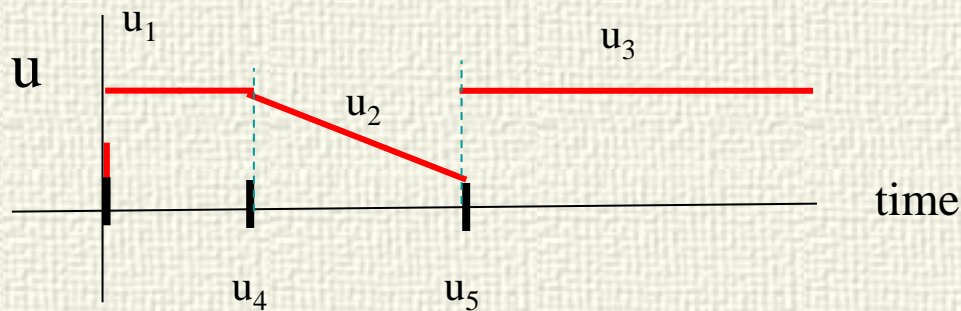
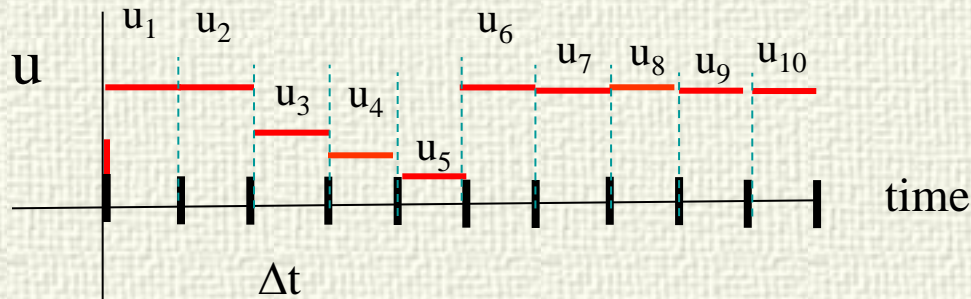
$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$



Parametrización



1. Comenzar con una parametrización sencilla
2. Refinarla incrementando los parámetros hasta identificar patrones
3. Redefinir la parametrización en base a los patrones para reducir el número de parámetros



Métodos de solución

Se aplica una parametrización $u(p)$

$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$

→ NLP

Enfoque secuencial

CVP mas resolver las ecuaciones DAE externamente mediante simulación

Enfoque simultaneo

Convertir el problema en uno NLP de gran tamaño mediante su discretización total



Enfoque secuencial

- ✓ El optimizador solo considera a $u(p)$ como variables del problema
- ✓ El modelo DAE se resuelve rigurosamente
- ✓ Dificultades con las restricciones de camino, sistemas inestables y el cálculo de gradientes

Optimizador NLP de $J(u(p))$
con respecto a p



Valores de
 p



Valores de
 $J(u), g(u)$

Simulador dinámico que
calcula los valores de x, z
solución del DAE, así como
de $J(x,u), g(x,u,z)$

$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$

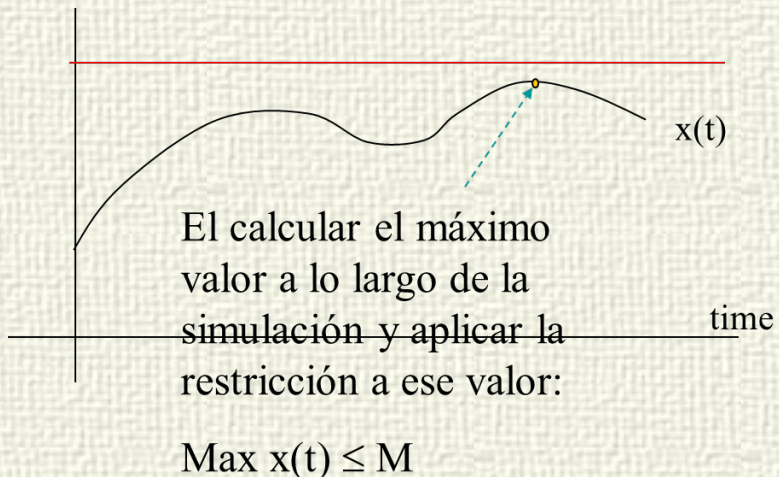
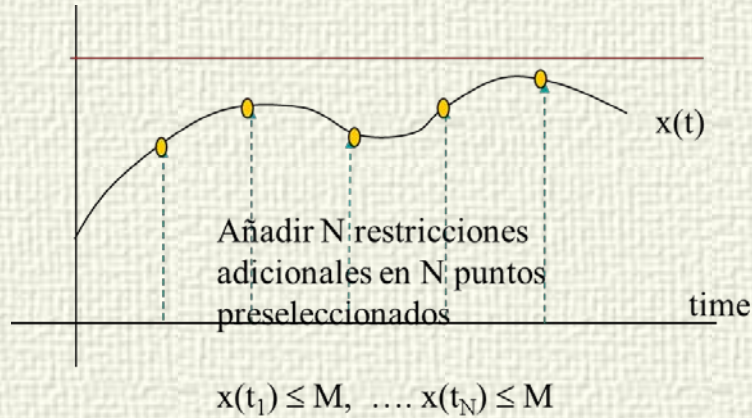
Solución con
software NLP

Múltiples llamadas al simulador desde el código NLP

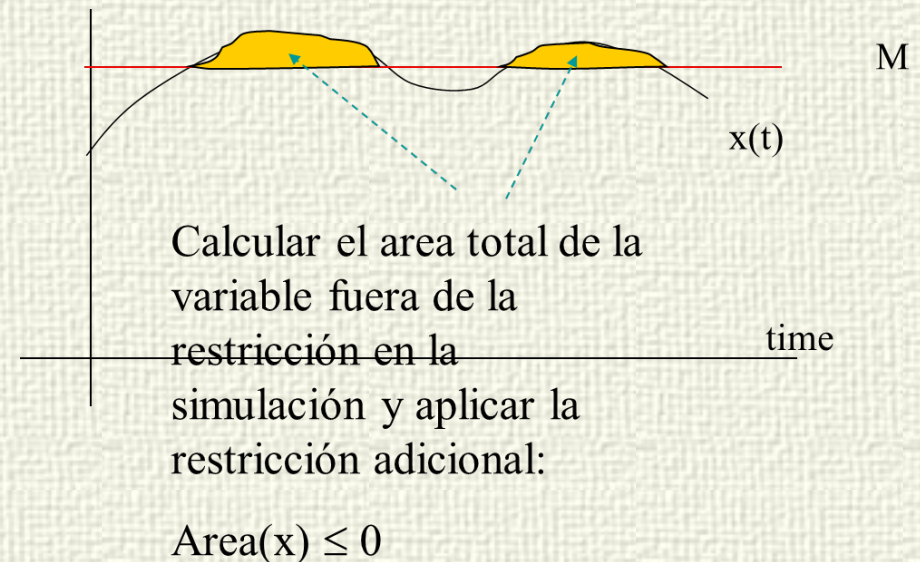


Restricciones de camino

Path Constraints



Diversas estrategias





Gradientes, ecuaciones de sensibilidad



$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$

Normalmente en NLP se requieren los gradientes

¿Diferencias finitas?

$$\frac{dJ}{dp_i} = \int_0^T \left(\frac{\partial L}{\partial x} \frac{\partial x}{\partial p_i} + \frac{\partial L}{\partial u} \frac{\partial u}{\partial p_i} \right) dt$$

$$\frac{dg_i}{dp_i} = \frac{\partial g_i}{\partial x} \frac{\partial x}{\partial p_i} + \frac{\partial g_i}{\partial u} \frac{\partial u}{\partial p_i}$$

$$s = \frac{\partial x}{\partial p} \quad \text{sensibilidades}$$

Pueden obtenerse integrando el sistema extendido / Sistema adjunto:

$$F(\dot{x}, x, u) = 0$$

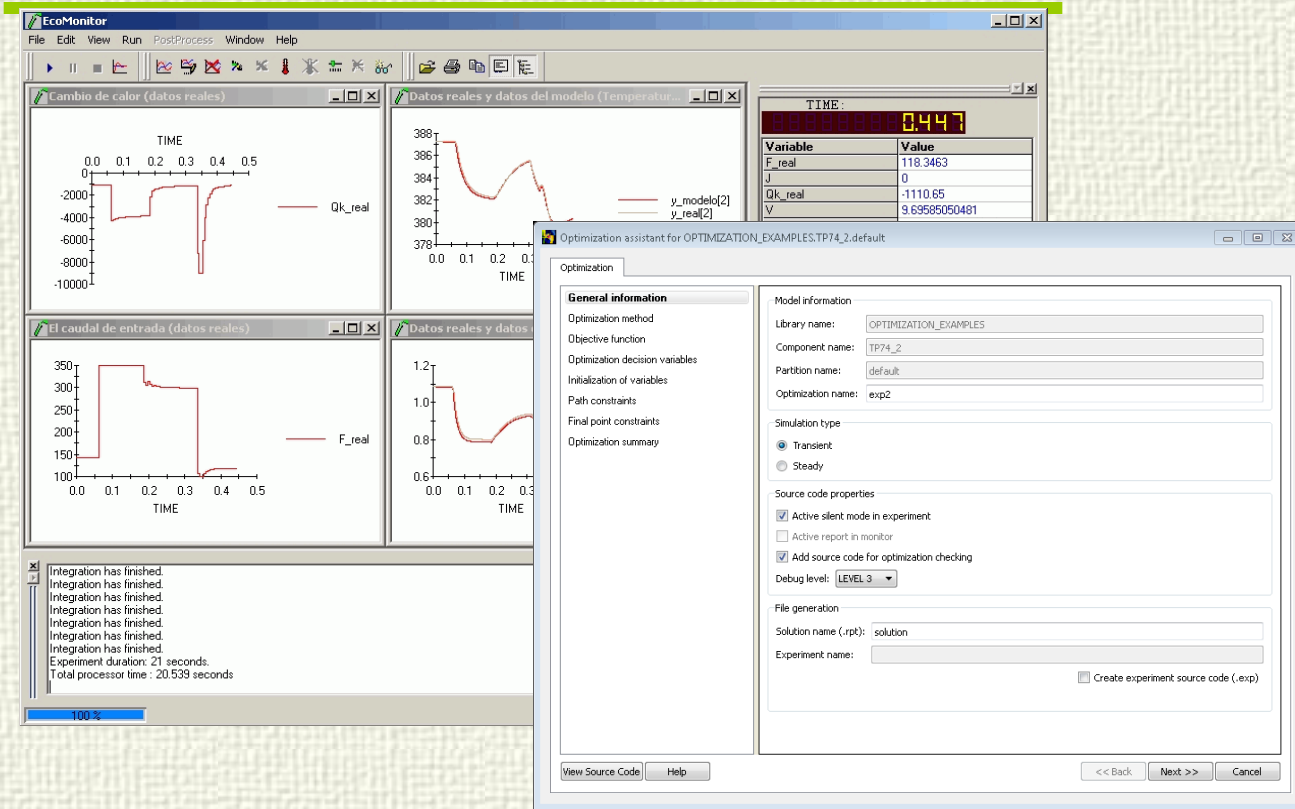
$$\frac{\partial F}{\partial \dot{x}} \frac{ds_i}{dt} + \frac{\partial F}{\partial x} s_i + \frac{\partial F}{\partial u} \frac{\partial u}{\partial p_i} = 0$$

Comparten el Jacobiano del sistema original

Integradores como DASPK3.0, IDAS.. dan las sensibilidades



Software



Entornos de simulación unidos a solvers NLP

Asistentes para la definición del problema y generación automática de código de optimización

EcosimPro, gProms, Dymola,..

Muy importante el cálculo de sensibilidades para la calidad de la solución.
Errores relativos Simulación / Optimización



Ejemplo: Reactor químico

$$\text{Max } J = q c_B$$

$$V \frac{dc_A}{dt} = q(c_{Ai} - c_A) - V\beta e^{-E/RT} c_A$$

$$V\rho c_p \frac{dT}{dt} = q\rho c_p (T_i - T) - Vkc_A H - UA(T - T_r)$$

$$V_r \rho_r c_{pr} \frac{dT_r}{dt} = F_r \rho_r c_{pr} (T_{ri} - T_r) + UA(T - T_r)$$

$$c_B = c_{Ai} - c_A$$

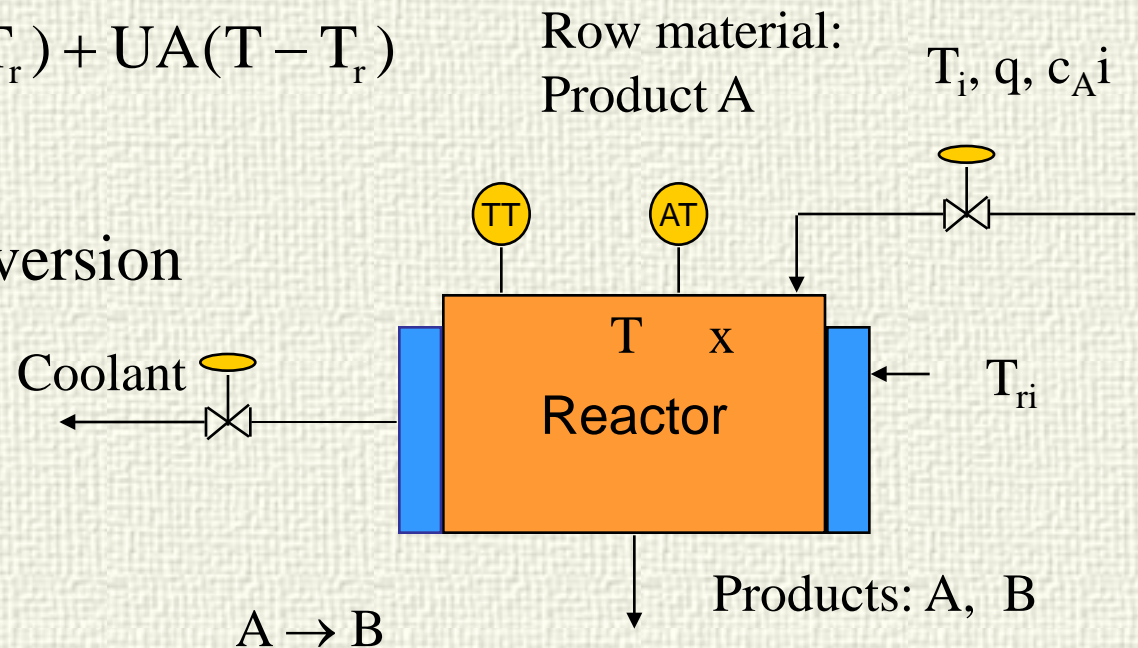
$$x = c_B / c_{Ai} \quad x \text{ conversion}$$

$$T_{\min} \leq T \leq T_{\max}$$

$$x_{\min} \leq x \leq 1$$

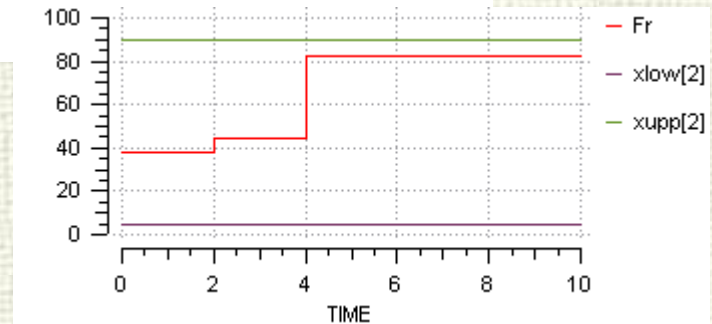
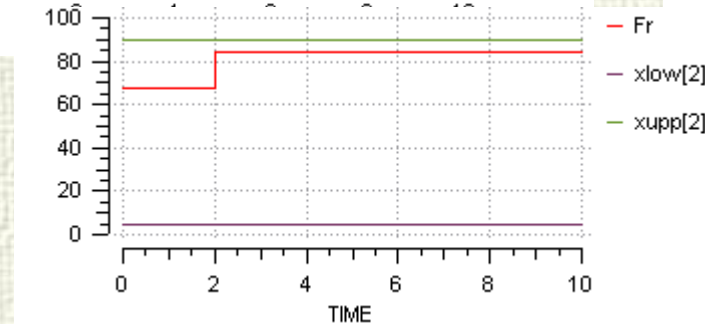
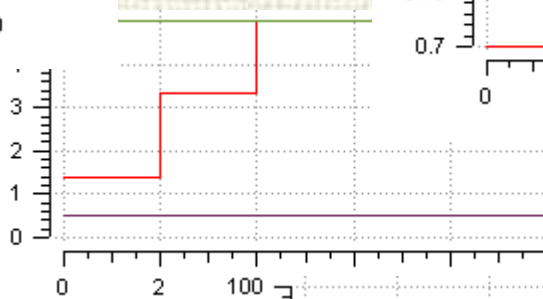
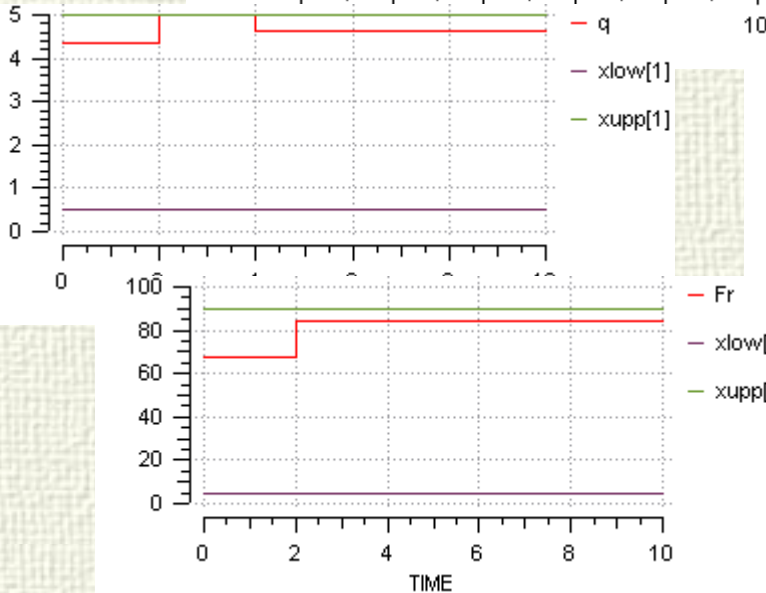
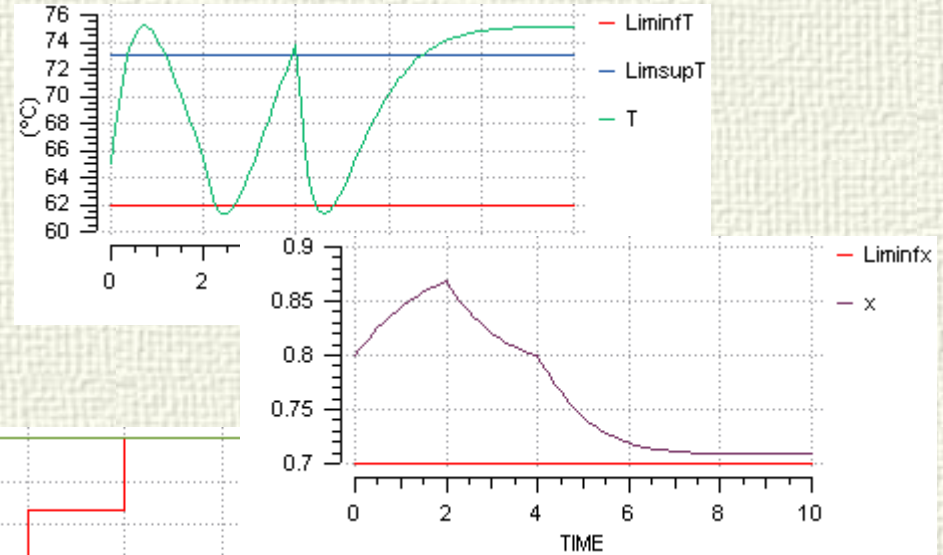
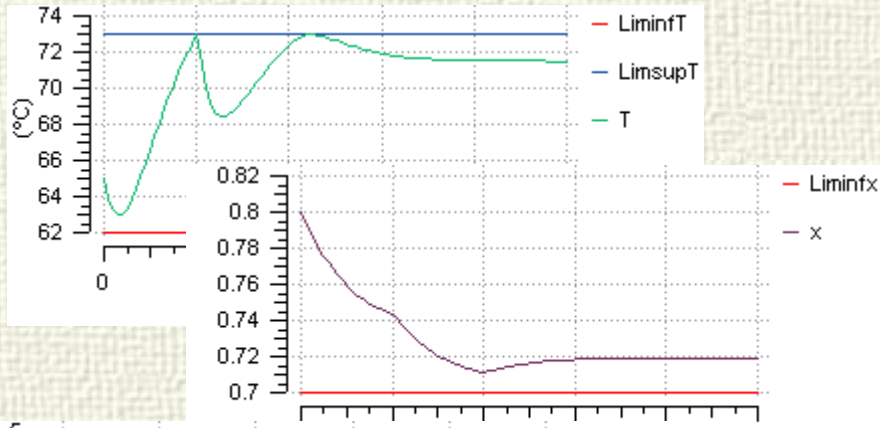
$$q_{\min} \leq q \leq q_{\max}$$

$$F_{r\min} \leq F_r \leq F_{r\max}$$





Con / sin sensibilidades





Software: Ecosimpro

The screenshot displays the Ecosimpro software interface. The main window is titled "EcoMonitor" and contains four subplots showing simulation data over time (0.0 to 0.5). The top-left plot, titled "Cambio de calor (datos reales)", shows the heat flow Q_k_{real} fluctuating between approximately -10000 and -2000. The top-right plot, titled "Datos reales y datos", shows a variable starting at 388 and decreasing to about 382. The bottom-left plot, titled "El caudal de entrada (datos reales)", shows the input flow F_{real} starting at 150, peaking at 350, and then dropping to 100. The bottom-right plot, titled "Datos reales y datos", shows a variable starting at 1.2 and decreasing to about 0.8. A status bar at the bottom indicates "100%" completion and shows the file names "simul_estim_param_cstr" and "exp1".

An "Optimization assistant for OPTIMIZATION_EXAMPLES.TP74_2.default" dialog box is overlaid on the right side of the screen. It contains the following information:

- General information:** Optimization method, Objective function, Optimization decision variables, Initialization of variables, Path constraints, Final point constraints, Optimization summary.
- Model information:** Library name: OPTIMIZATION_EXAMPLES; Component name: TP74_2; Partition name: default; Optimization name: exp2.
- Simulation type:** Transient; Steady.
- Source code properties:** Active silent mode in experiment; Active report in monitor; Add source code for optimization checking; Debug level: LEVEL 3.
- File generation:** Solution name (.rpt): solution; Experiment name: [empty]; Create experiment source code (.exp).

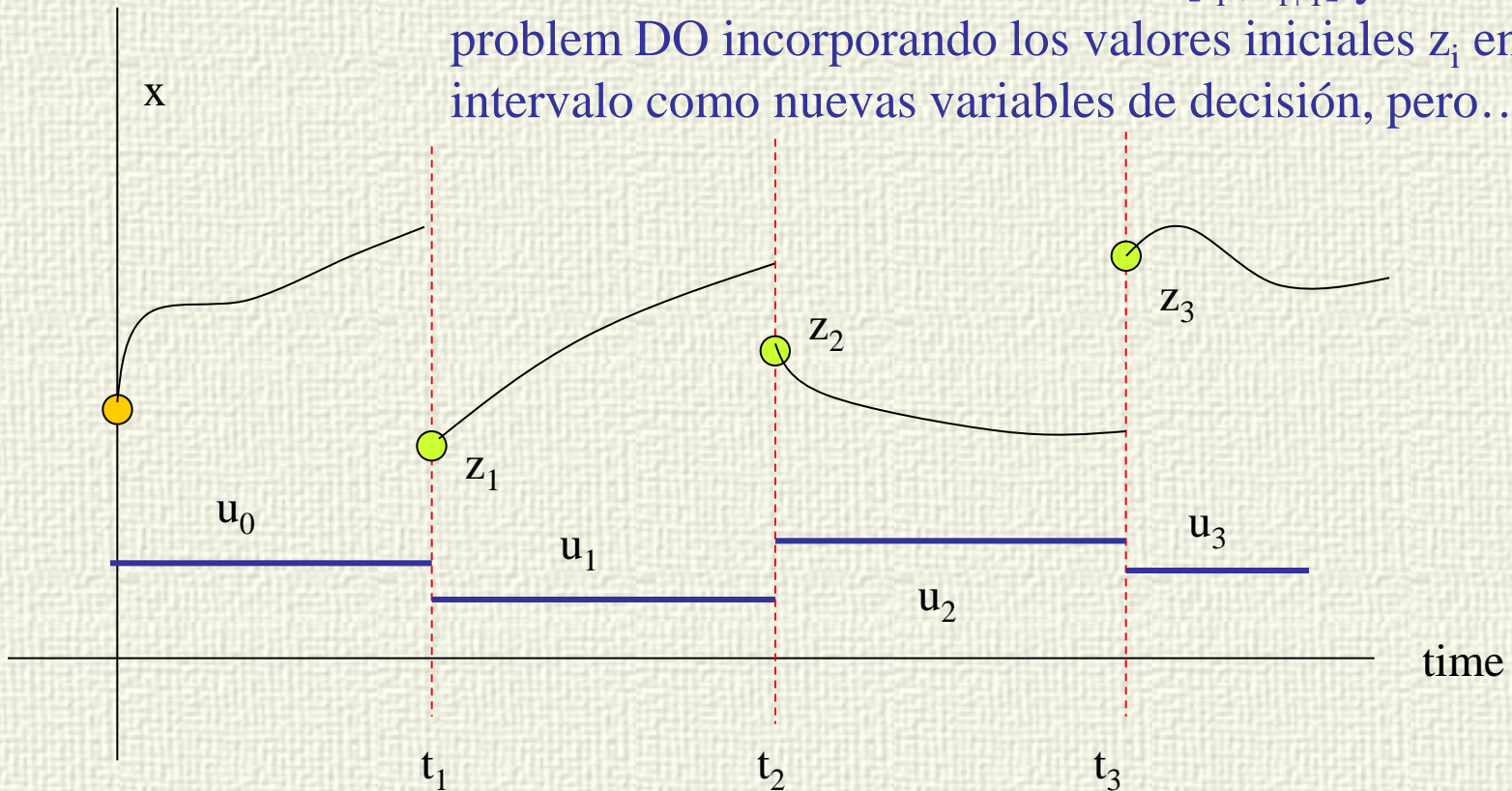
Buttons at the bottom of the dialog include "View Source Code", "Help", "<< Back", "Next >>", and "Cancel".

Simulation environments with optimization



Multiple shooting

Dividir el horizonte en n intervalos $[t_i, t_{i+1}]$ y resolver el problem DO incorporando los valores iniciales z_i en cada intervalo como nuevas variables de decisión, pero...



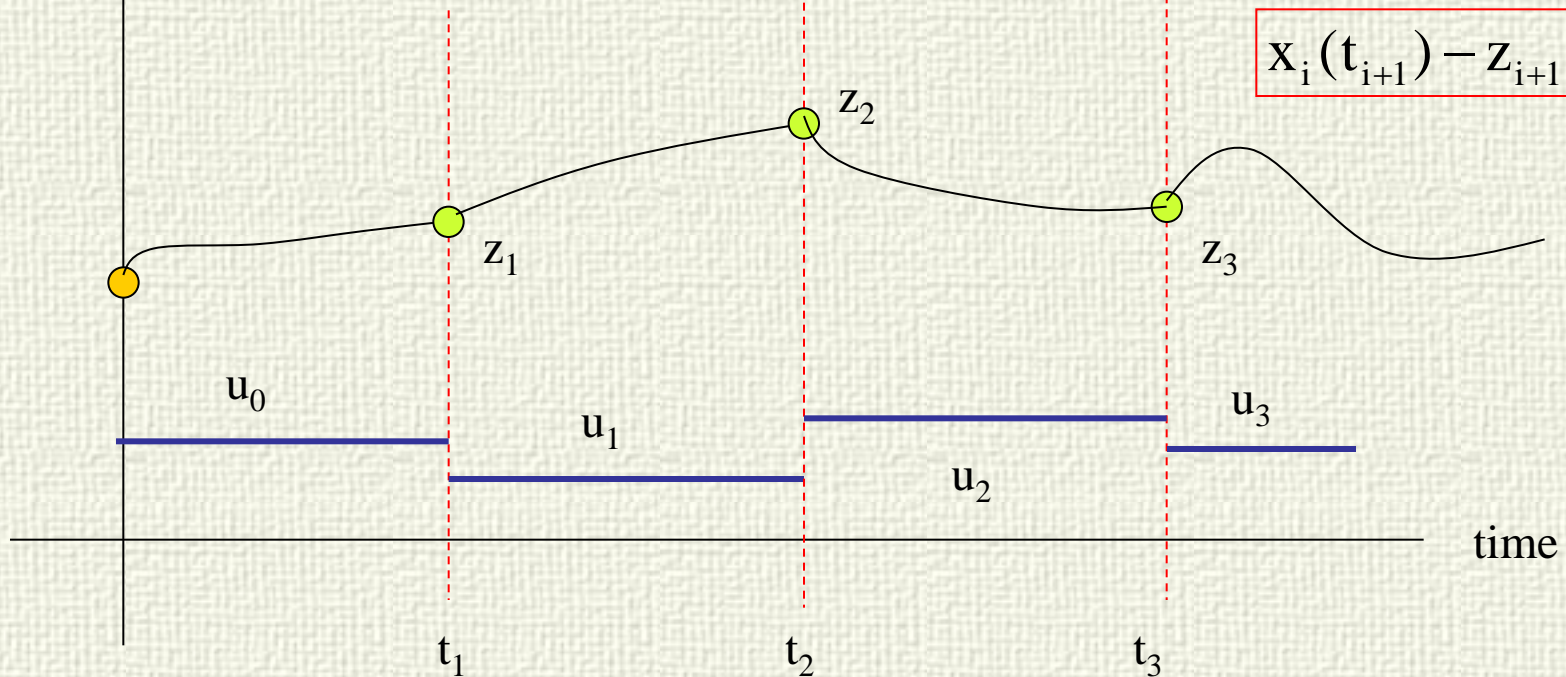
$$J = \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} L(x_i(t), u_i, z_i) dt$$

x_i x en el intervalo i



Multiple shooting

...imponiendo la condición de que el estado final de una etapa debe ser igual a inicial de la siguiente para asegurar su continuidad





Multiple shooting

✓ Ventajas:

- La inicialización de x puede estar mas cerca de la trayectoria deseada, lo que facilita la convergencia
- Las restricciones de camino pueden imponerse sobre z_i
- La evolución de la etapa i es independiente de la etapa $i+1$
- Facilita la paralelización
- Permite el uso de métodos secuenciales con sistemas inestables

✓ Desventajas:

- Mayor complejidad

$$J = \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} L(x_i(t), u_i, z_i) dt$$

$$v = \{u_0, z_1, u_1, \dots, z_i, u_i, \dots, z_{n-1}, u_{n-1}\}$$

$$\min_v J$$

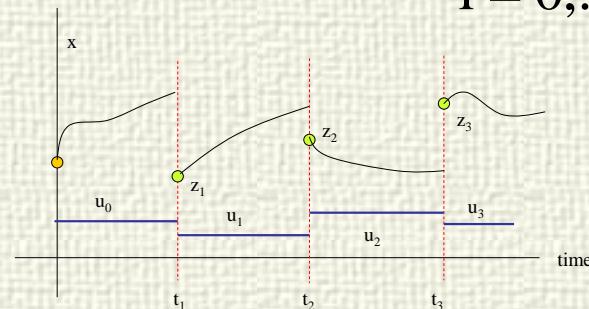
$$F_i(\dot{x}_i, x_i, u_i, t) = 0$$

$$x_i(t_i) = z_i$$

$$g_i(z_i, u_i, t_i) \leq 0$$

$$x_i(t_{i+1}) - z_{i+1} = 0$$

$$i = 0, \dots, n$$





MPC + MHE

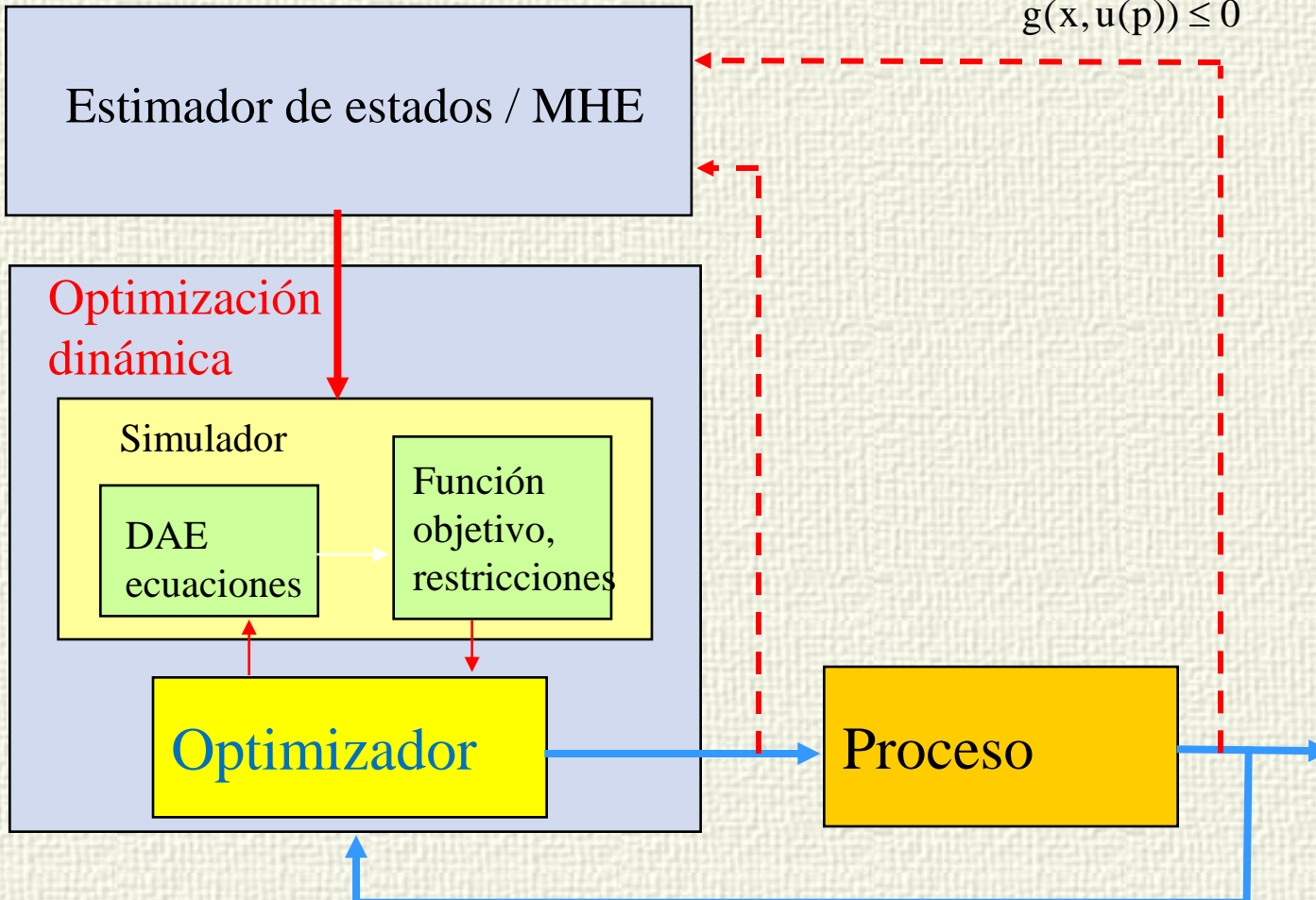


$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$

$$x(0) = x_p$$





Enfoque simultaneo

$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$

Discretizar
totalmente las
ecuaciones

$$F(\dot{x}, x, u(p)) = 0$$



$$F\left(\frac{x(t + \Delta t) - x(t)}{\Delta t}, x(t + \Delta t), u(p)\right) = 0$$

$$t = 0, \Delta t, 2\Delta t, \dots$$

$$\int_0^T L(x, u(p)) dt$$



$$\sum_{j=0}^N [L(x(j), u(j, p))] \Delta t$$

Sistema de ecuaciones algebraicas



Enfoque simultaneo

$$\min_p J(p) = \int_0^T L(x, u(p)) dt$$

$$F(\dot{x}, x, u(p)) = 0$$

$$g(x, u(p)) \leq 0$$



$$\min_{p,x} J = \sum_{j=0}^N [L(x(j), u(j,p))] \Delta t$$

$$F(x(1), x(0), u(0,p)) = 0$$

$$F(x(2), x(1), u(1,p)) = 0$$

$$F(x(3), x(2), u(2,p)) = 0$$

.....

$$F(x(N), x(N-1), u(N-1,p)) = 0$$

$$g(x(0), u(0,p)) \leq 0$$

$$g(x(1), u(1,p)) \leq 0$$

$$g(x(2), u(2,p)) \leq 0$$

$$g(x(N-1), u(N-1,p)) \leq 0$$

✓ El número de variables de decisión y ecuaciones se incrementa de acuerdo a la discretización.

✓ Facilita la imposición de restricciones de camino y el cálculo de gradientes

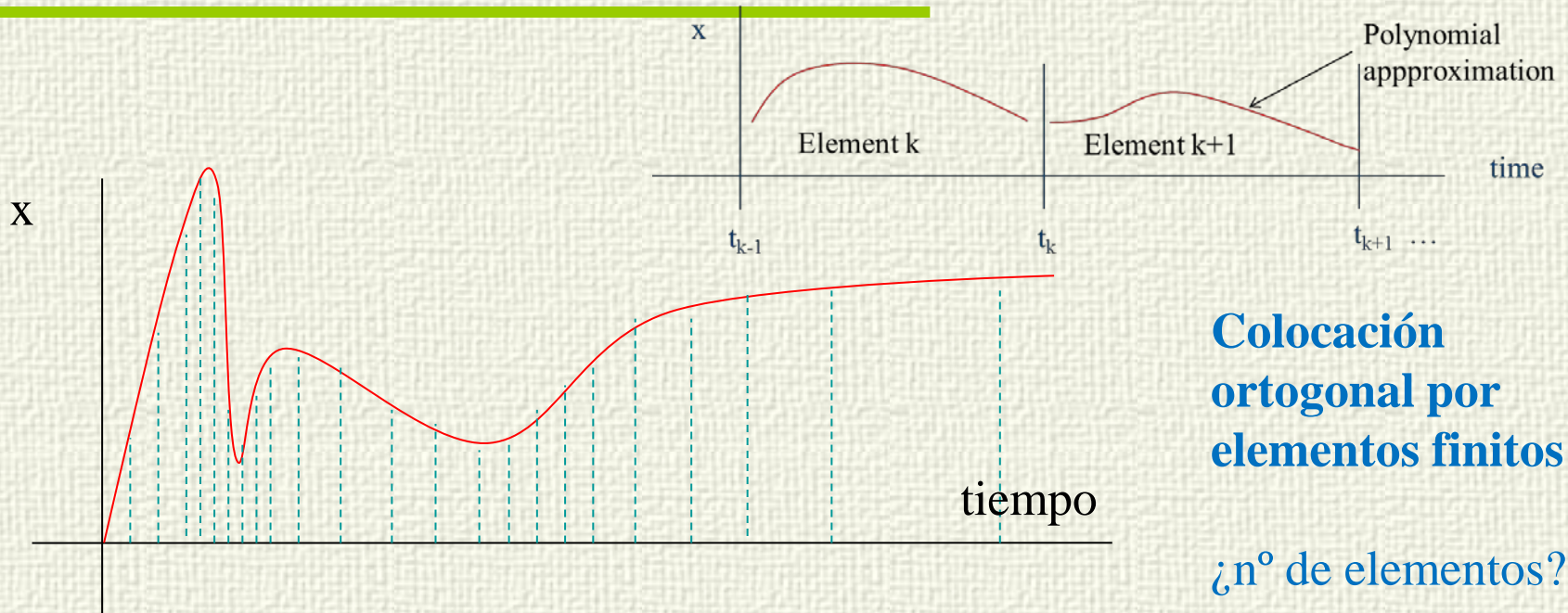
✓ Puede haber problemas con la discretización de las DAE

$x(i)$ y p son variables de decisión

Solución con software NLP



Discretización



**Colocación
ortogonal por
elementos finitos**

¿n° de elementos?

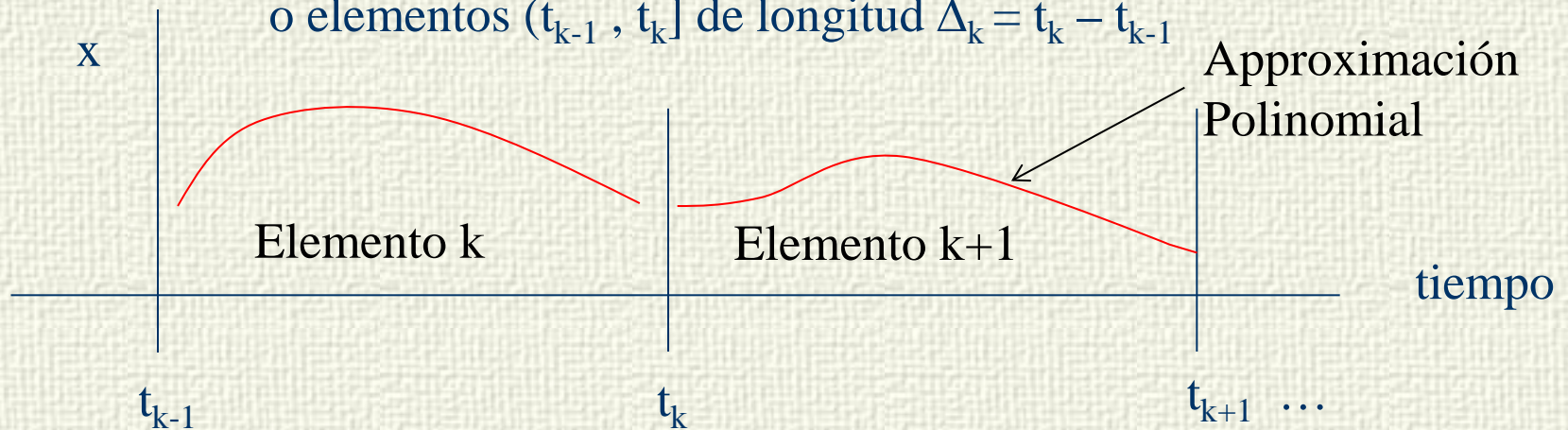
La integración de sistemas stiff usa métodos de paso y estructura variable para mantener el error de integración bajo cotas.

El uso de métodos de paso fijo obliga a usar un gran número de intervalos, resultando en un alto número de ecuaciones y variables y no garantiza la calidad



Colocación en elementos finitos

El horizonte temporal se divide en K intervalos o elementos $(t_{k-1}, t_k]$ de longitud $\Delta_k = t_k - t_{k-1}$



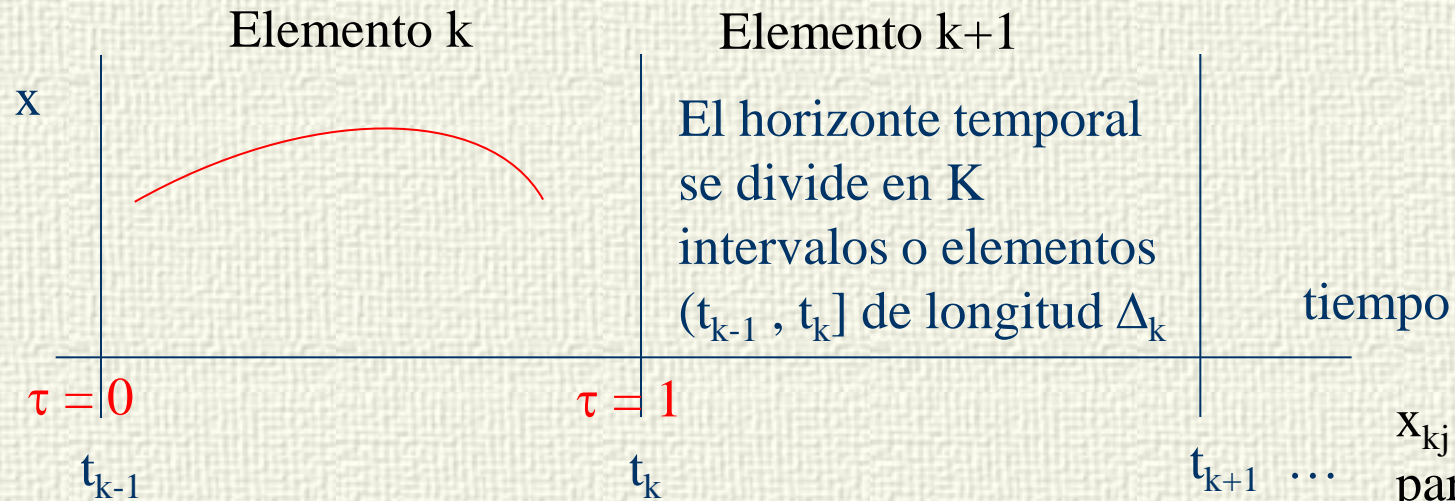
En cada intervalo $(t_{k-1}, t_k]$ la solución x se aproxima por un polinomio, que es una aproximación continua suave dentro del elemento, pero permite discontinuidades en los perfiles de la señal de control entre elementos

Hay muchos tipos de aproximaciones polinómicas que pueden usarse

El número K de elementos no tiene que ser grande necesariamente



Colocación en elementos finitos



x_{kj}
parámetros a
determinar

Una posibilidad es aproximar la evolución temporal de las variables mediante una combinación lineal de polinomios conocidos $P_j(\tau)$ de orden P . Normalmente se utilizan polinomios de interpolación de Lagrange

$$\mathbf{x}(t) \approx \sum_{j=0}^P P_j(\tau) \mathbf{x}_{kj}$$

$k = 1, \dots, K$

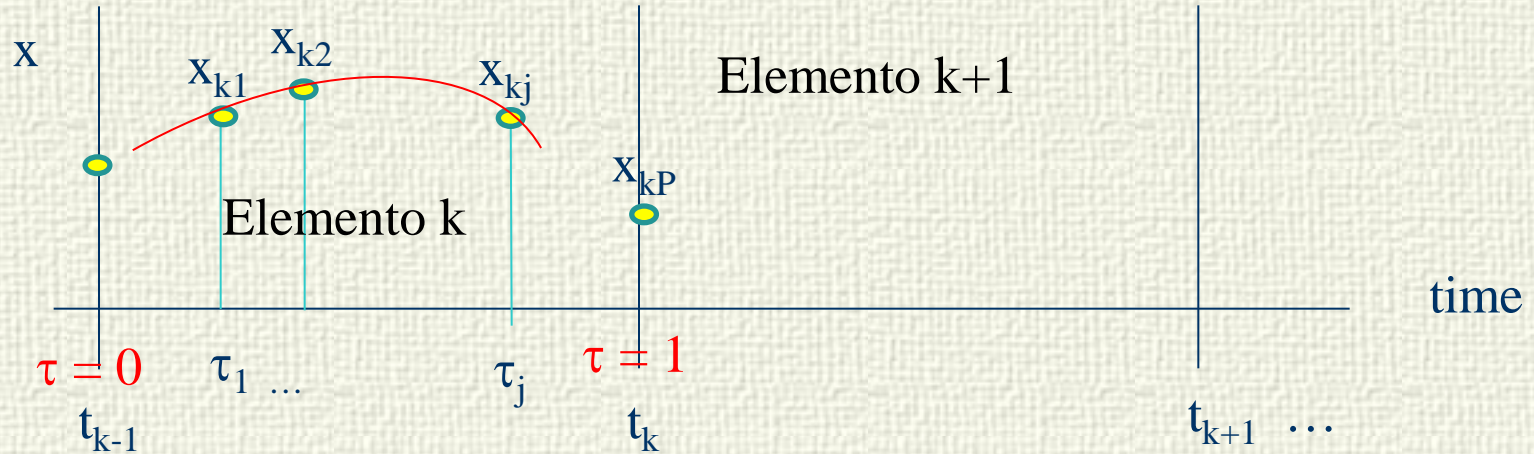
$$t = t_{k-1} + \tau \Delta_k \quad \tau \in (0, 1]$$

$$\dot{\mathbf{x}}(t) \approx \sum_{j=0}^P \frac{\dot{P}_j(\tau) \mathbf{x}_{kj}}{\Delta_k}$$

τ Tiempo
normalizado



Polinomios de interpolación de Lagrange



$$\mathbf{x}(t) \approx \sum_{j=0}^P P_j(\tau) \mathbf{x}_{kj}$$

$$t = t_{k-1} + \tau \Delta_k \quad \tau \in (0,1]$$

$$\dot{\mathbf{x}}(t) \approx \sum_{j=0}^P \frac{\dot{P}_j(\tau) \mathbf{x}_{kj}}{\Delta_k}$$

$$P_j(\tau) = \prod_{i=0, i \neq j}^P \frac{\tau - \tau_i}{\tau_j - \tau_i}$$

$$\mathbf{x}(t_{kj}) = \mathbf{x}(t_{k-1} + \tau_j \Delta_k) = \mathbf{x}_{kj}$$

Los parámetros x_{kj} tienen un significado claro usando los polinomios de Lagrange

Se seleccionan $P+1$ puntos de interpolación
 $\tau_0 = 0, \tau_1, \dots, \tau_P$
 $\tau_i < \tau_{i+1}$



Polinomios de interpolación de Lagrange



$$P_j(\tau) = \prod_{i=0, i \neq j}^P \frac{\tau - \tau_i}{\tau_j - \tau_i}$$

$$P_0 = \frac{\tau - \tau_1}{\tau_0 - \tau_1} \frac{\tau - \tau_2}{\tau_0 - \tau_2} \frac{\tau - \tau_3}{\tau_0 - \tau_3} =$$

$$P_1 = \frac{\tau - \tau_0}{\tau_1 - \tau_0} \frac{\tau - \tau_2}{\tau_1 - \tau_2} \frac{\tau - \tau_3}{\tau_1 - \tau_3} =$$

$$P_2 = \frac{\tau - \tau_0}{\tau_2 - \tau_0} \frac{\tau - \tau_1}{\tau_2 - \tau_1} \frac{\tau - \tau_3}{\tau_2 - \tau_3} =$$

$$P_3 = \frac{\tau - \tau_0}{\tau_3 - \tau_0} \frac{\tau - \tau_1}{\tau_3 - \tau_1} \frac{\tau - \tau_2}{\tau_3 - \tau_2} =$$

$$\mathbf{x}(t) \approx \sum_{j=0}^P P_j(\tau) \mathbf{x}_{kj}$$

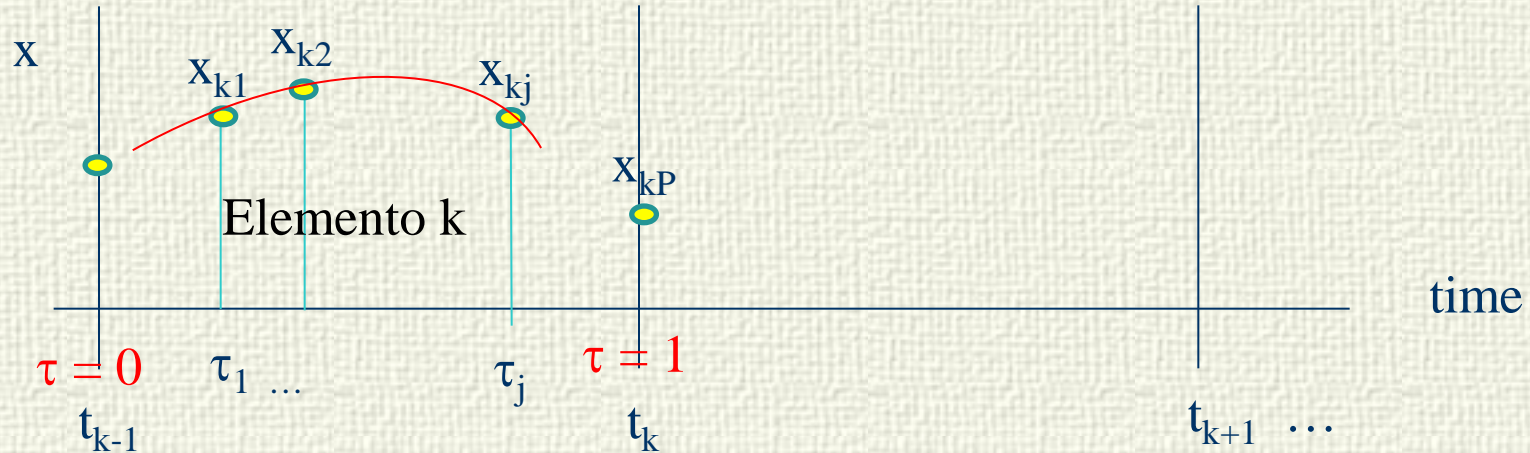
Si t coincide con τ_j , p.e. τ_2 , todos los P_j son cero excepto P_2 que vale 1, luego:

$$\mathbf{x}(t_{kj}) = \mathbf{x}(t_{k-1} + \tau_j \Delta_k) = \mathbf{x}_{kj}$$

Los coeficientes \mathbf{x}_{kj} coinciden con los valores de la variable \mathbf{x} en el punto τ_j del elemento k



Colocación en elementos finitos



Las ecuaciones DAE deben cumplirse en los puntos de colocación τ_i :
Esto proporciona un conjunto de ecuaciones que permiten calcular los valores de los coeficientes desconocidos x_{ki}

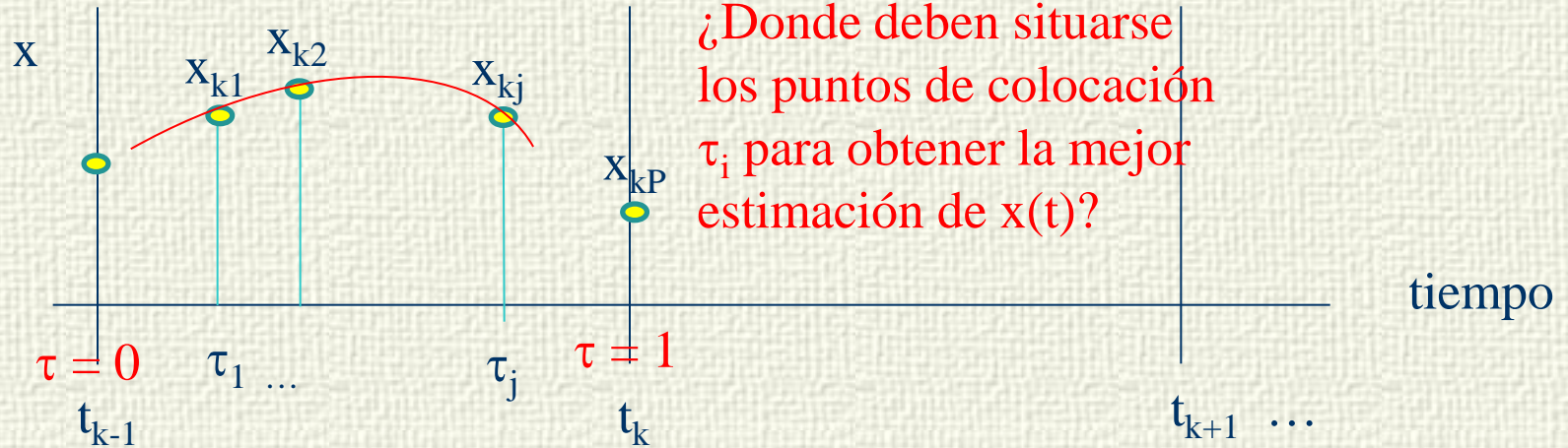
$$F(\dot{x}, x, u(p)) = 0$$

$$F\left(\sum_{j=0}^P \frac{\dot{P}_j(\tau_i) \mathbf{x}_{kj}}{\Delta_k}, \mathbf{x}_{ki}, u(p)\right) = 0 \quad k = 1, \dots, K$$

Los $P+1$ puntos de colocación se sitúan en instante fijos τ_i en cada elemento k . Existen diferentes métodos para escoger su posición



Colocación ortogonal



$$F\left(\sum_{j=0}^P \frac{\dot{P}_j(\tau_i) \mathbf{x}_{kj}}{\Delta_k}, \mathbf{x}_{ki}, u(p)\right) = 0 \quad \begin{matrix} k = 1, \dots, K \\ i = 1, \dots, P \end{matrix}$$

Para reducir P, se escogen polinomios ortogonales:

Las ecuaciones no se imponen en $\tau_0 = 0$, en su lugar, se utilizan, bien las condiciones iniciales en $t = 0$, o se impone la continuidad de los estados entre elementos. Esto da suficientes ecuaciones adicionales para calcular todos los \mathbf{x}_{kj}

$$\int_0^1 P_j(\tau) P_i(\tau) d\tau = 0 \quad i \neq j$$



Colocación ortogonal

Shifted Gauss-Legendre and Radau roots as collocation points.

Degree	P	Legendre Roots	Radau Roots
1		0.500000	1.000000
2		0.211325 0.788675	0.333333 1.000000
3		0.112702 0.500000 0.887298	0.155051 0.644949 1.000000
4		0.069432 0.330009 0.669991 0.930568	0.088588 0.409467 0.787659 1.000000
5		0.046910 0.230765 0.500000 0.769235 0.953090	0.057104 0.276843 0.583590 0.860240 1.000000

τ_0 es siempre = 0

Los puntos de colocación τ_i , $i = 1, \dots, P$ se toman como las raíces de polinomios tipo Gauss-Jacobi típicamente:

$$P_P^{\text{Legendre}}(\tau) = \sum_{j=0}^P (-1)^{P-j} \tau^j \gamma_j$$

$$\gamma_0 = 1$$

$$\gamma_j = \frac{(P-j+1)(P+j)}{j^2}$$

Da mas exactitud

$$P_P^{\text{Radau}}(\tau) = \sum_{j=0}^P (-1)^{P-j} \tau^j \gamma_j$$

$$\gamma_0 = 1$$

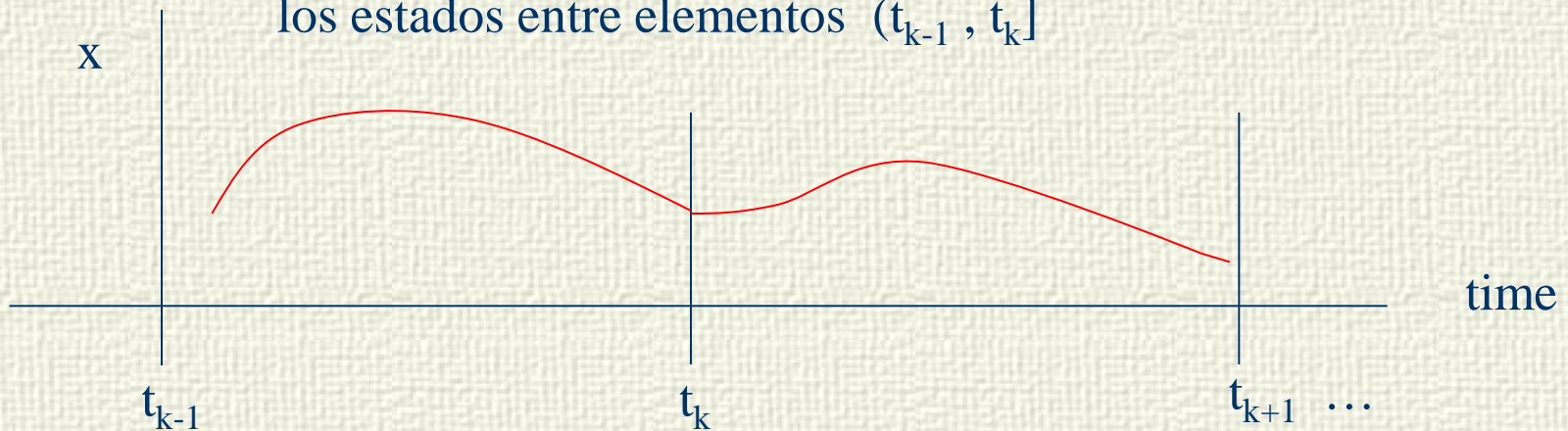
$$\gamma_j = \frac{(P-j+1)(P+j+1)}{j^2}$$

Da mas robustez



Colocación ortogonal

Se fuerza la continuidad de la evolución de los estados entre elementos $(t_{k-1}, t_k]$



$$\mathbf{x}(t_k) = \mathbf{x}_{k+1,0} = \mathbf{x}_{k,P}$$

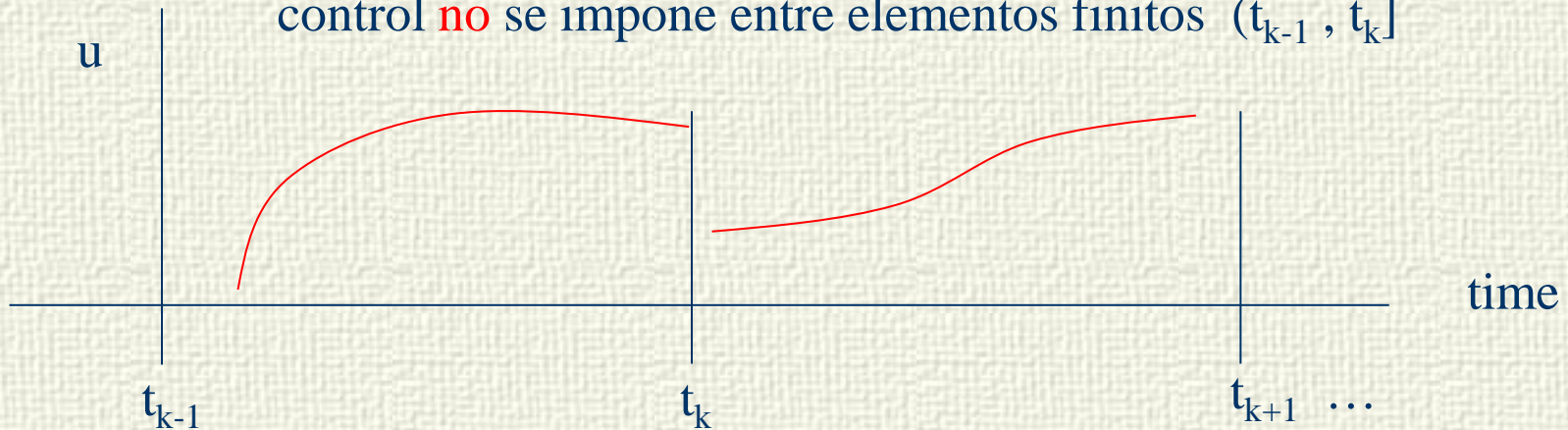
$$\mathbf{x}(t_0) = \mathbf{x}_{10} = \mathbf{x}_0$$

Los métodos simultáneos de solución son adecuados para sistemas inestables



Colocación ortogonal

La continuidad de los perfiles de las señales de control **no** se impone entre elementos finitos $(t_{k-1}, t_k]$



$$\mathbf{u}(t) \approx \sum_{j=1}^P \bar{P}_j(\tau) \mathbf{u}_{kj}$$

$$\bar{P}_j(\tau) = \prod_{i=1, i \neq j}^P \frac{\tau - \tau_i}{\tau_j - \tau_i}$$

$$t = t_{k-1} + \tau \Delta_k \quad \tau \in (0, 1]$$

Las variables de control, si se desea, pueden también aproximarse por polinomios de interpolación de Lagrange en cada elemento finito $(t_{k-1}, t_k]$



Ejemplo

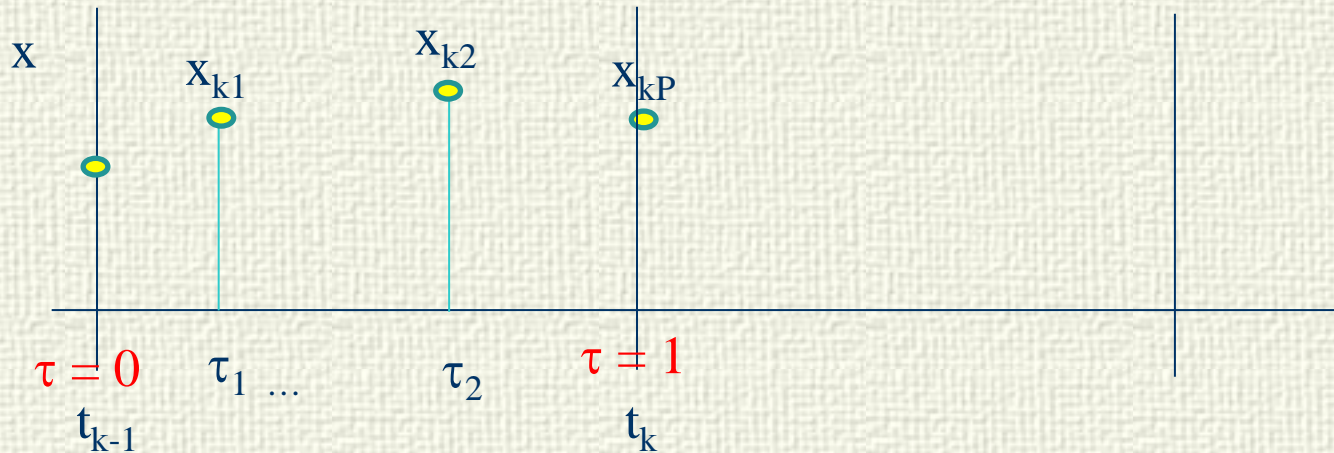
Integrar entre $t = 0$ y 1

$$\dot{x} = x^2 - 2x + 1 \quad x(0) = -3$$

Se selecciona $K = 2$ elementos finitos de igual tamaño

$$\Delta_k = (1 - 0)/2 = 0.5$$

$P = 3$, 4 puntos de colocación



Los puntos de colocación de Radau para $P = 3$ son:

$$\tau_0 = 0 \quad \tau_1 = 0.155051 \quad \tau_2 = 0.644949 \quad \tau_3 = 1$$



Example

$$P_j(\tau) = \prod_{i=0, i \neq j}^P \frac{\tau - \tau_i}{\tau_j - \tau_i}$$

Los puntos de colocación de Radau para $P=3$ son :
 $\tau_0 = 0$ $\tau_1 = 0.155051$ $\tau_2 = 0.644949$ $\tau_3 = 1$

$$P_0 = \frac{\tau - \tau_1}{\tau_0 - \tau_1} \frac{\tau - \tau_2}{\tau_0 - \tau_2} \frac{\tau - \tau_3}{\tau_0 - \tau_3} = -10\tau^3 + 18\tau^2 - 9\tau + 1$$

$$P_1 = \frac{\tau - \tau_0}{\tau_1 - \tau_0} \frac{\tau - \tau_2}{\tau_1 - \tau_2} \frac{\tau - \tau_3}{\tau_1 - \tau_3} = 15.5808 \tau^3 - 25.6296\tau^2 + 10.0488\tau$$

$$P_2 = \frac{\tau - \tau_0}{\tau_2 - \tau_0} \frac{\tau - \tau_1}{\tau_2 - \tau_1} \frac{\tau - \tau_3}{\tau_2 - \tau_3} = -8.9141\tau^3 + 10.2963\tau^2 - 1.3821\tau$$

$$P_3 = \frac{\tau - \tau_0}{\tau_3 - \tau_0} \frac{\tau - \tau_1}{\tau_3 - \tau_1} \frac{\tau - \tau_2}{\tau_3 - \tau_2} = 3.3333\tau^3 - 2.6667\tau^2 + 0.3333\tau$$

$$x(t_{k-1} + \tau_j \Delta_k) = x_{kj} \quad x(t) \approx \sum_{j=0}^P P_j(\tau) x_{kj} \quad t = t_{k-1} + \tau \Delta_k \quad \tau \in (0,1]$$



Example

$$\dot{\mathbf{x}}(t) \approx \sum_{j=0}^P \frac{\dot{P}_j(\tau) \mathbf{x}_{kj}}{\Delta_k}$$

Los puntos de colocación de Radau para $P=3$ son :
 $\tau_0 = 0$ $\tau_1 = 0.155051$ $\tau_2 = 0.644949$ $\tau_3 = 1$

$$\dot{P}_0(\tau) = -30\tau^2 + 36\tau - 9$$

$$\dot{x} = x^2 - 2x + 1 \quad x(0) = -3$$

$$\dot{P}_1(\tau) = 46.7423\tau^2 - 51.2592\tau + 10.0488$$

$$\dot{P}_2(\tau) = -26.7423\tau^2 + 20.5925\tau - 1.3821$$

$$\dot{P}_3(\tau) = 10\tau^2 - 5.3333\tau + 0.3333$$

$$\sum_{j=0}^3 \frac{\dot{P}_j(\tau) \mathbf{x}_{kj}}{0.5} = \mathbf{x}^2 - 2\mathbf{x} + 1$$

$$\mathbf{x}(t_{k-1} + \tau_j \Delta_k) = \mathbf{x}_{kj}$$

$$k = 1, 2$$

$$t = t_{k-1} + \tau \Delta_k \quad \tau \in (0, 1]$$



Ejemplo

$$\dot{\mathbf{x}} = \mathbf{x}^2 - 2\mathbf{x} + 1 \quad \mathbf{x}(0) = -3 \quad \longrightarrow \quad \sum_{j=0}^3 \frac{\dot{\mathbf{P}}_j(\tau) \mathbf{x}_{kj}}{0.5} = \mathbf{x}^2 - 2\mathbf{x} + 1 \quad k = 1, 2$$

$$\text{En los puntos de colocación } \tau_i : \quad \sum_{j=0}^3 \frac{\dot{\mathbf{P}}_j(\tau_i) \mathbf{x}_{kj}}{0.5} = \mathbf{x}_{ki}^2 - 2\mathbf{x}_{ki} + 1 \quad \begin{array}{l} k = 1, 2 \\ i = 1, \dots, 3 \end{array}$$

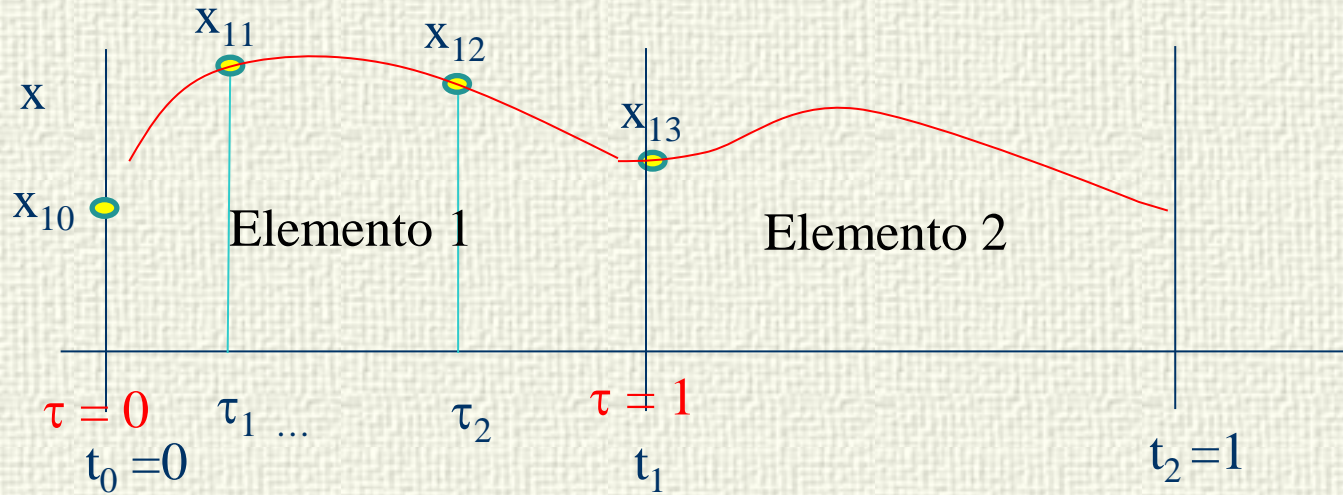
$$\begin{aligned} &(-30\tau_i^2 + 36\tau_i - 9)\mathbf{x}_{10} + (46.7423\tau_i^2 - 51.2592\tau_i + 10.0488)\mathbf{x}_{11} + \\ &+ (-26.7423\tau_i^2 + 20.5925\tau_i - 1.3821)\mathbf{x}_{12} + (10\tau_i^2 - 5.3333\tau_i + 0.3333)\mathbf{x}_{13} = \\ &= 0.5(\mathbf{x}_{1i}^2 - 2\mathbf{x}_{1i} + 1) \quad i = 1, 2, 3 \end{aligned}$$

$$\begin{aligned} &(-30\tau_i^2 + 36\tau_i - 9)\mathbf{x}_{20} + (46.7423\tau_i^2 - 51.2592\tau_i + 10.0488)\mathbf{x}_{21} + \\ &+ (-26.7423\tau_i^2 + 20.5925\tau_i - 1.3821)\mathbf{x}_{22} + (10\tau_i^2 - 5.3333\tau_i + 0.3333)\mathbf{x}_{23} = \\ &= 0.5(\mathbf{x}_{2i}^2 - 2\mathbf{x}_{2i} + 1) \quad i = 1, 2, 3 \end{aligned}$$

8 incógnitas, 6 ecuaciones



Ejemplo



$$\mathbf{x}(t_k) = \mathbf{x}_{k+1,0} = \mathbf{x}_{k,P} = \sum_{j=0}^P P_j(1) \mathbf{x}_{k,j}$$

$$\mathbf{x}(0.5) = \mathbf{x}_{20} = \mathbf{x}_{13} = \sum_{j=0}^3 P_j(1) \mathbf{x}_{1j}$$

$$\mathbf{x}(t_0) = \mathbf{x}_{10} = \mathbf{x}_0$$

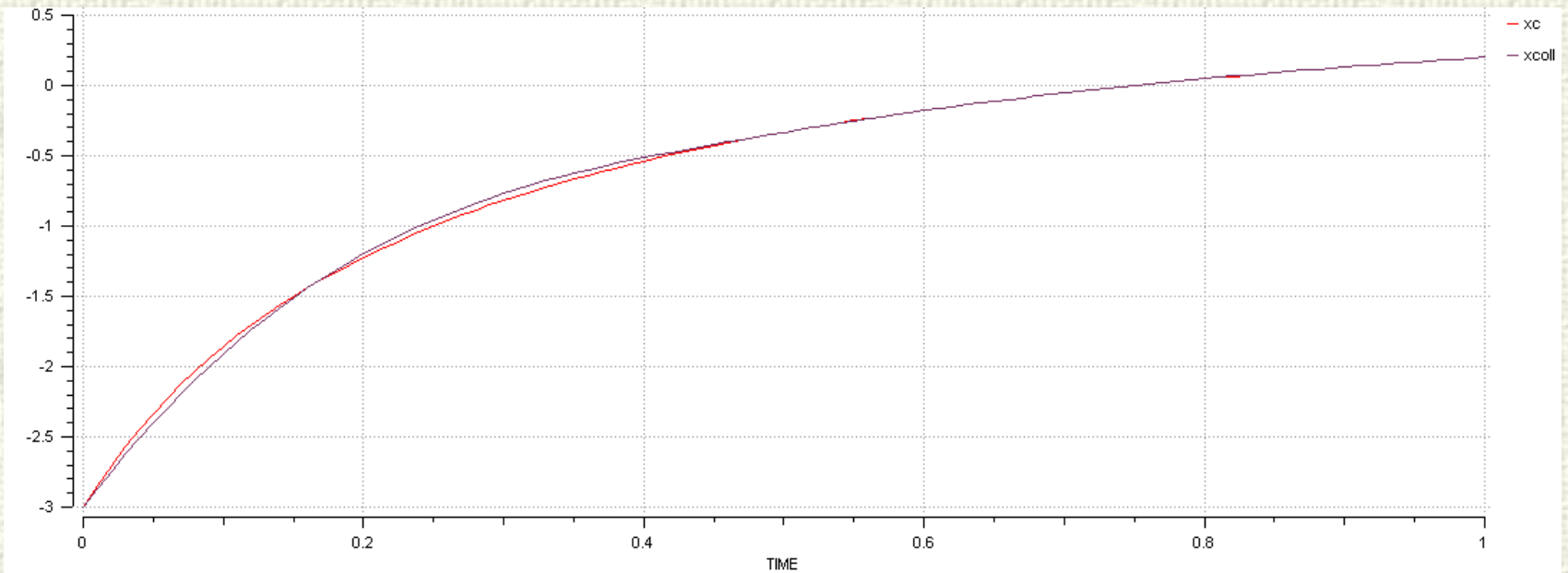
$$\mathbf{x}(0) = \mathbf{x}_{10} = -3$$

8 incógnitas, 8 ecuaciones

La continuidad de estados entre elementos y las condiciones iniciales proporcionan las ecuaciones restantes



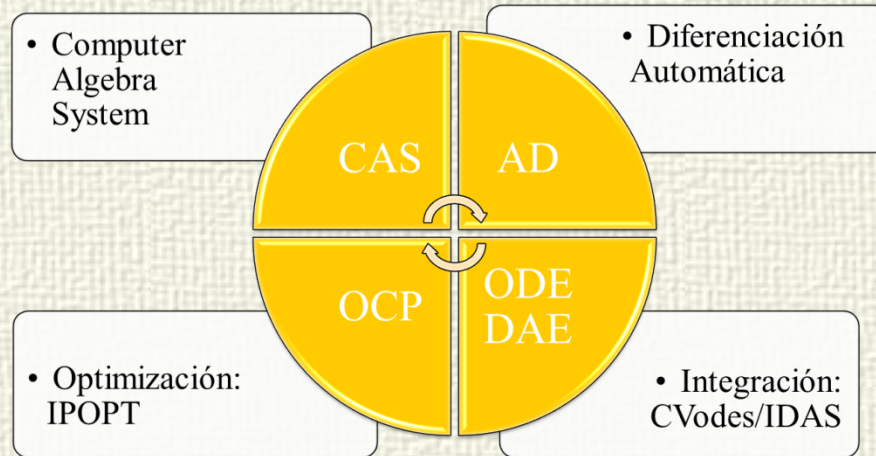
Ejemplo



Solución exacta / solución mediante colocación ortogonal



Software



Solución eficiente de problemas de gran escala

Entornos de modelado y optimización
GAMS, AIMMS, XPRESS, Gurobi,...
pueden usarse tras la discretización

CasADi es un entorno simbólico para optimización numérica que facilita la discretización e implementa diferenciación automática (gradientes y Hessianos).

Genera código C e implementa interfaces a códigos DAE y de optimización como SUNDIALS, IPOPT etc.

Se gestiona desde una interfaz con Python



Diferenciación Automática



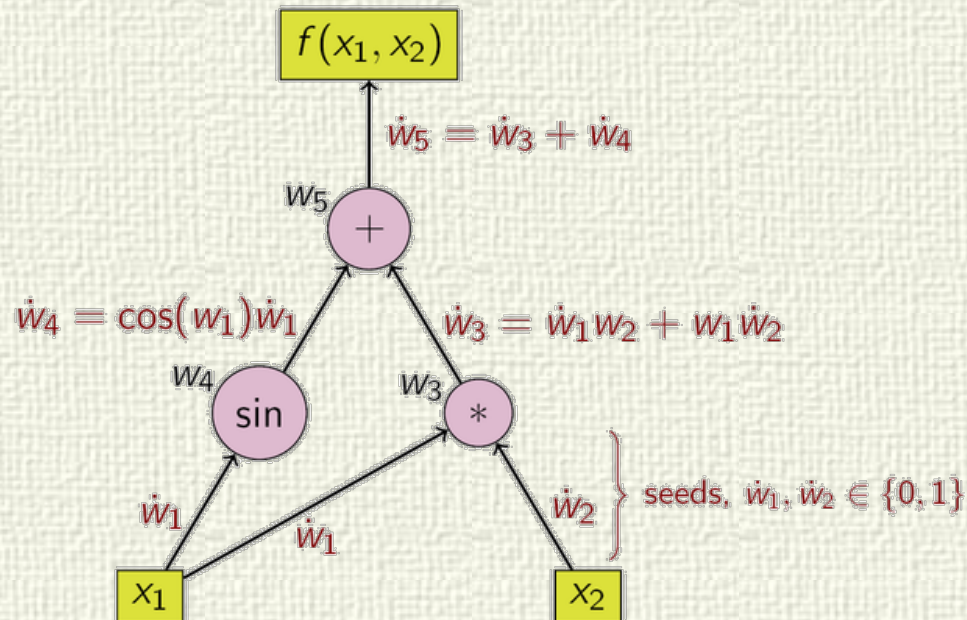
Ejemplo::

$$f = x_1 x_2 + \sin(x_1)$$

$$? \frac{\partial f}{\partial x_1} ?$$

Assignment	Derivatives
$w_1 = x_1$	$w_1' = 1$ (<i>seed</i>)
$w_2 = x_2$	$w_2' = 0$ (<i>seed</i>)
$w_3 = w_1 w_2$	$w_3' = w_1' w_2 + w_1 w_2' = x_2$
$w_4 = \sin(w_1)$	$w_4' = \cos(w_1) w_1' = \cos(x_1)$
$w_5 = w_3 + w_4$	$w_5' = w_3' + w_4' = x_2 + \cos(x_1)$

Forward propagation
of derivative values





Estabilidad EMPC

$$\min_v J(v) = \Phi(z_N) + \sum_{j=0}^{N-1} L(z_j, v_j)$$

$$z_{j+1} = f(z_j, v_j) \quad z_0 = x_k$$

$$z_j \in X \quad v_j \in U \quad z_N \in X_f$$

$$u_k = v_0^*$$

$$x_{k+1} = f(x_k, u_k)$$

Aunque el enfoque de garantía de estabilidad es similar al del caso NMPC, normalmente la función de costo debe modificarse con términos de regularización (desviaciones cuadráticas respecto al óptimo en estado estacionario) para asegurar el cumplimiento de ciertas hipótesis.

El estudio de estabilidad del EMPC difiere del NMPC usado para control en que, en este último caso, se conoce el objetivo al que se quiere llegar en el futuro, lo que no ocurre en el caso del EMPC

Para estabilizar las soluciones en lazo cerrado, se suele añadir una penalización terminal Φ definida positiva, como en el caso del NMPC



Estabilidad NMPC

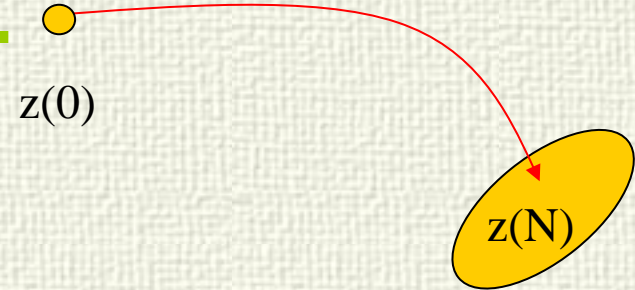
$$\min_v J(v) = \Phi(z_N) + \sum_{j=0}^{N-1} L(z_j, v_j)$$

$$z_{j+1} = f(z_j, v_j) \quad z_0 = x_k$$

$$z_j \in X \quad v_j \in U \quad z_N \in X_f$$

$$u_k = v_0^*$$

$$x_{k+1} = f(x_k, u_k)$$



Normalmente se acompaña de una restricción adicional que fuerza el estado en T a estar dentro de una región terminal que contiene al óptimo, en la cual se supone existe una ley de control $u = c(x)$ que mantendría la evolución del sistema dentro de la región terminal. Se supone además que:

$$\Phi(x_{k+1}) - \Phi(x_k) \leq -L(x_k, c(x_k)) \quad \forall x \in X_f$$

$$\alpha_1(|x|) \leq L(x_k, u_k) \leq \alpha_2(|x|)$$

Con α_1, α_2 funciones positivas definidas

Existen métodos para el diseño de la región terminal y la ley de control asociada, pero pueden no ser fáciles de aplicar



Estabilidad EMPC

Con EMPC, en el cual un objetivo estable no se conoce de antemano, la suposición

$$\alpha_1(|\mathbf{x}|) \leq L(\mathbf{x}_k, \mathbf{u}_k) \leq \alpha_2(|\mathbf{x}|)$$

no tiene, en general, por qué cumplirse. Para que se cumpla, y poder aplicar así las condiciones de garantía de estabilidad del NMPC, hay que modificar la formulación del problema añadiendo términos de regularización en la función de costo,

$$\min_{\mathbf{v}} \Phi(\mathbf{z}_N) + 0.5 \left\| \mathbf{z}_N - \mathbf{z}^* \right\|_P^2 + \sum_{j=0}^{N-1} L(\mathbf{z}_j, \mathbf{v}_j) + 0.5 \left\| \begin{array}{c} \mathbf{z}_j - \mathbf{z}^* \\ \mathbf{v}_j - \mathbf{v}^* \end{array} \right\|_Q^2$$

que involucran el óptimo $(\mathbf{z}^*, \mathbf{v}^*)$ en estado estacionario calculado mediante:

$$\min_{\mathbf{z}, \mathbf{v}} L(\mathbf{z}, \mathbf{v})$$

$$\mathbf{z} = \mathbf{f}(\mathbf{z}, \mathbf{v}) \quad \mathbf{z} \in \mathbf{X} \quad \mathbf{v} \in \mathbf{U}$$



Operación y Control de procesos

- ✓ Del seguimiento de consignas y el rechazo de perturbaciones a operar una planta dinámicamente con un objetivo económico
- ✓ Marco: NMPC económico/
Operación óptima de procesos
- ✓ El desarrollo de este tipo de sistemas es complejo y a veces específico, pero hay herramientas y desarrollos que lo empiezan a hacer posible

