

Caso de estudio: modelado, simulación, control de tanque de mezclado

Antonio Sala
UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Mini-curso en vídeo

VERSIÓN DEL DOCUMENTO:

Fecha y hora:	13/02/2025 [08:30]
Núm. objetos de aprendizaje:	31
Tiempo total lecciones en video:	04:42:35
	[propio (31): 04:42:35, terceros (0): 00:00:00]

©2025, Antonio Sala Piqueras, DISA-UPV.

Se reservan todos los derechos de uso, copia, y propiedad intelectual de los materiales de mi autoría en este documento y los materiales audiovisuales/PDF/código enlazados desde el mismo, tanto respecto a su uso *verbatim* o como base de obras derivadas. La copia o redistribución de contenidos de terceros deberá respetar los derechos de uso que sus autores hayan establecido.

Presentación

El presente PDF presenta una colección de vídeos que abordan el modelado, simulación y control de un tanque de mezclado (hay modelos de diferente complejidad, unos con composición interna homogénea –perfectamente agitado–, y otros con dinámica del propio mezclado interno). Se parte desde lo más básico, y se acaba simulando desacoplamiento, feedforward, ratio, antiwindup, override, etc. resolviendo, por tanto, el problema de forma “completa”, a falta de implementación.

Este mini-curso corresponde a la lista de reproducción de YouTube:

https://www.youtube.com/watch?v=_9L1mGvq5QQ&list=PLDzpfREU0p7sRVSe6FRa6aiFeBiY0fmx_

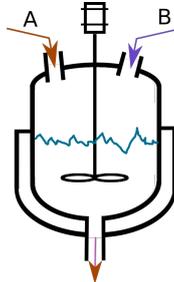
* Por defectos en el software (realizado por mí) que procesa una base de datos asociada para generar este PDF, los enlaces a vídeos externos salen con [??]. Haciendo click en el acrónimo al lado del número de vídeo se accede a la página web del mismo donde los enlaces están correctamente insertados.

Modelado y Linealización

[1: modmix]	<p>Modelado de un tanque de mezclado y obtención de representación interna (Matlab)</p> <p>Materiales: [CÓD.: ModeladoTanqueMezclado.mlx] [PDF]</p>	<p>**  10:11</p> <p>[YouTube ▶]</p>
-------------	--	--

Nota: Se recomienda al alumno que, como ejercicio propuesto, intente resolver este problema de modelado antes de consultar la solución.

Este vídeo discute el modelado de un tanque de mezclado representado en la siguiente figura:



Donde dos caudales $q_{a,in}$ y $q_{b,in}$ de componentes A y B son mezclados en el tanque.

El objetivo es obtener un modelo de un sistema dinámico que permita calcular el nivel y concentraciones en función del tiempo. Una vez el modelo está escrito y se comprueba que está correctamente planteado, pasa a obtenerse una representación interna $\dot{x} = f(x, u)$, $y = h(x, u)$.

* El uso de Matlab es poco relevante: sólo se utiliza el comando `solve` para despejar determinadas variables (ponerlas “a la izquierda de un signo de igualdad”) a partir de unas ecuaciones escritas en forma y orden arbitrario. Pero, realmente, el modelo es sencillo y dichas operaciones deberían poder ser realizadas manualmente sin casi ninguna dificultad.

* El modelo no incluye dinámicas asociadas a la “medida” de la concentración o nivel. En aplicaciones reales, si la velocidad del agitador es poca y la viscosidad de los reactivos es alta, puede haber un gran retraso hasta que la medida de la concentración en la tubería de salida coincide con la concentración “media” del tanque aquí modelada. Del mismo modo, turbulencias, salpicaduras, ebullición, etc. podrían introducir mucho ruido en un sensor de nivel, que por tanto debería incluir un *filtro* adecuado. Estas consideraciones están fuera de los objetivos de este vídeo, pero se resaltan para indicar que el modelo aquí obtenido podría no ser lo suficientemente realista en muchas aplicaciones (un modelo más complejo del mismo tanque se aborda en los vídeo [14] y [15]).

[2: odemix]	<p>Simulación de un tanque de mezclado con Matlab (integración numérica ode45)</p> <p>Materiales: [CÓD.: ModelSimulacionTanqueMezclado.mlx] [PDF]</p>	<p>**  10:18</p> <p>[YouTube ▶]</p>
-------------	--	--

En este vídeo se simula la respuesta temporal del tanque de mezclado cuyo modelo no lineal se desarrolló en el vídeo [1]. Para que sea más “autocontenido”, los dos primeros minutos resumen rápidamente esos pasos del modelado; si se ha visionado anteriormente el vídeo [1], puede comenzarse éste en el minuto [02:40].

En la ecuación de salida, en vez de incorporar todas las señales posibles, como se hizo en el vídeo antes referido, ahora sólo se desea obtener las trayectorias (respuesta temporal) del nivel del tanque h y de la concentración de componente B , x_B . Por ello se selecciona sólo un subconjunto de los resultados de `solve`.

La simulación pasa primero por convertir el modelo de la Symbolic Toolbox en una función ordinaria (numérica) de Matlab, con el comando `matlabFunction`. Ello se realiza tanto con las ecuaciones de estado como de salida.

Un segundo paso genera una función del tiempo interpolando entre los puntos (arbitrarios) de los perfiles de entrada $q_{a,in}$ y $q_{b,in}$, con el comando `interp1`. Con ello ya se tiene todo preparado para ejecutar `ode45` y obtener las trayectorias del estado (v_a, v_b) y con dichas trayectorias sustituirlas (ya no hace falta más integración numérica) en las ecuaciones de salida.

Los sistemas que `ode45` puede simular podrían ser más complejos e integrar, por ejemplo, “controladores”; un bucle cerrado de control se simula con `ode45` en el vídeo [??]. Asimismo, en modelados físicos complejos de elevado orden o constantes de tiempo muy diferentes (stiff), es posible que otros integradores numéricos diferentes a `ode45` sean más eficientes computacionalmente (el vídeo [??] compara `ode45` con `ode15s`).

[3: slkmix]	<p>Modelo de tanque de mezclado: importación a Simulink</p> <p>Materiales: [CÓD.: slkmix.zip] [PDF]</p>	<p>**  08:03</p> <p>[YouTube ▶]</p>
-------------	--	--

Este vídeo importa a Simulink las ecuaciones de estado y de salida obtenidas en el vídeo [1]; por hacerlo un poco más autocontenido, un breve resumen de dicho modelado se discute hasta el instante [01:44] del vídeo.

La simulación en Simulink es una alternativa a la del `ode45` discutida en el vídeo [2]. Simulink permite una interfaz de usuario gráfica incorporando generadores de ondas, osciloscopios, interacciones con otros sistemas, etc. que podría resultar más laborioso de preparar si se generara manualmente un modelo para `ode45` en casos complejos (realmente Simulink acaba traduciendo todo a una función para el integrador numérico, pero de forma casi transparente al usuario).

La clave de la importación a Simulink es el comando `matlabFunctionBlock`, que traduce una expresión del Symbolic Toolbox a un bloque `Matlab Function` de código Matlab dentro de un modelo o subsistema “.slx”. Dos llamadas a ese comando crean ecuaciones de estado y de salida; luego, se pone un integrador tras las ecuaciones de estado (para transformar las derivadas del estado a los estados propiamente dichos), especificando una condición inicial arbitraria. Esos estados son “realimentados” a la ecuación de estados (dado que las derivadas del estado dependen de ellos mismos), e insertados en la ecuación de salida. Los adecuados bloques `Demux`, `Inport` y `Outport` completan el modelo, que se guarda como un *subsistema* para ser utilizado como un componente de otros modelos más complejos.

[4: slkmix2]	<p>Modelo de tanque de mezclado: simulación con Simulink de respuesta temporal</p> <p>Materiales: [CÓD.: slkmix.zip] [PDF]</p>	<p>*  04:18</p> <p>[YouTube ▶]</p>
--------------	---	---

Este vídeo incorpora a un modelo de simulación el subsistema generado en el vídeo [3], mediante el bloque **Subsystem Reference**. Luego, genera formas de onda arbitrarias para las entradas q_a y q_b (mediante el bloque **Repeating Sequence (Stair)**), visualizando las entradas y salidas mediante los bloques **Scope** oportunos.

Nota: Simulink puede también hacer linealización (Capítulo ??), como se discute en el vídeo [7] sobre este mismo modelo.

[5: linmix]	<p>Linealización de un modelo tanque de mezclado, simulación comparada con original (ode45, Matlab)</p> <p>Materiales: [CÓD.: LinealizTanqueMezclado.mlx] [PDF]</p>	<p>**  10:55</p> <p>[YouTube ▶]</p>
-------------	--	--

En este vídeo se linealiza el modelo del tanque de mezclado cuyo modelo no lineal de segundo orden se desarrolló en el vídeo [1]. Los detalles sobre la simulación de este tanque con **ode45** se discuten en el vídeo [2]. Los contenidos de modelado y simulación serán referidos muy brevemente en este vídeo, se recomienda al lector visionar los vídeos referidos para más detalle.

En la ecuación de salida, ahora buscaremos simular/linealizar un modelo de la concentración de componente A , x_A y nivel del tanque h .

El primer paso para linealizar consiste en seleccionar un punto de funcionamiento (equilibrio). Se selecciona $q_{a,in} = 3$, $q_{b,in} = 1$. Fijando estas dos variables y las derivadas a cero, el comando **vpasolve** calcula los valores de todo el resto de variables (volúmenes acumulados, concentraciones, caudal de salida y nivel) para que se mantenga un equilibrio constante.

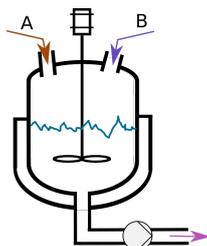
El comando **jacobian** calcula las derivadas parciales necesarias respecto a estados y entradas, tanto en ecuación de estado como de salida, para obtener una representación interna normalizada lineal.

La parte final del vídeo simula con **ode45** el modelo original y el modelo linealizado (éste último con entradas y condiciones iniciales incrementales). Deshaciendo el cambio de variable a incrementos de la solución linealizada se comparan ambas simulaciones, comprobando que el modelo linealizado tiene una exactitud razonable.

Como alternativa, que produce el mismo resultado, la linealización “automática” de un modelo Simulink de este tanque se aborda en el vídeo [7].

[6: linmixqs]	<p>Modelado, Linealización y Simulación de tanque de mezclado con 3 entradas (Matlab)</p> <p>Materiales: [CÓD.: LinealizTanqueMezclaQS.mlx] [PDF]</p>	<p>***  09:45</p> <p>[YouTube ▶]</p>
---------------	--	---

Este vídeo discute el modelado de un tanque de mezclado de dos componentes con extracción forzada, representado en la siguiente figura:



En este tanque, dos caudales $q_{a,in}$ y $q_{b,in}$ de componentes A y B son mezclados en el tanque. Intencionalmente, el sistema es *mu*y similar al del vídeo [1]: la diferencia ahora es que el caudal de salida ya no depende del nivel, sino que es fijado de forma “exógena” por bombas aguas abajo, con lo que aunque sea una “salida de material”, desde el punto de vista de teoría de sistemas, q_s es una variable de *entrada*. Por lo tanto, el modelo tiene una entrada *más* y, obviamente, una ecuación *menos* que el considerado en [1].

El modelado, representación interna y posterior linealización son paralelos a los desarrollos de los vídeos [1] y [5]. La diferencia es que en este sistema, para que exista equilibrio debe cumplirse que la suma de caudales de entrada y salida sea cero... entonces en equilibrio las tres entradas no son “independientes” y, por otro lado, si suman cero entonces cualquier nivel es un punto de equilibrio (equilibrio indiferente en Física, sistema marginalmente inestable en teoría de control). Estas cuestiones (existencia de infinitas soluciones en un sistema de ecuaciones) hacen que `solve` se pueda “liar” un poco... si sospechamos de esta situación, llamar a `solve` con la opción 'ReturnConditions' activada permite que el Symbolic Toolbox nos devuelva las infinitas soluciones de nivel que existen en este depósito, como se esperaba intuitivamente.

Una vez esa cuestión aclarada, todo es prácticamente igual que en los otros vídeos referidos, que abordaban el caso donde q_s era función del nivel h .

Nota: en muchas ocasiones, las entradas de “caudal” en procesos industriales lo son porque hay un sistema interno (control en cascada esclavo) de control de caudal, como el descrito en el vídeo [20], cuya visualización sería sólo aconsejable a estudiantes avanzados, dada su dificultad. ****

[7: slkmixLin] [Linealización de modelos Simulink: ejemplo tanque de mezclado](#) ***  08:44
Materiales: [[CÓD.:](#) LinealizTanqueSimulink.mlx] [[PDF](#)] [YouTube ▶]

En este vídeo se demuestra cómo utilizar el comando `linearize` que linealiza un modelo de Simulink en los que se han marcado determinadas señales como *Linear Analysis Points*. Estos puntos son extraídos con el comando `getlinio`, y los tiempos donde extraer el punto de operación son argumentos a `linearize`.

El vídeo compara la linealización obtenida con la linealización obtenida mediante `jacobian` de la Symbolic Toolbox, y se ve que coinciden. Por hacerlo autocontenido, los minutos [01:30] a [03:30] describen brevemente dicha linealización por jacobiano, cuyos detalles se presentan en el vídeo [5].

El vídeo también comprueba que la respuesta ante escalón de dos modelos linealizados en puntos diferentes exhibe grandes diferencias. Aunque el análisis en detalle queda fuera de los objetivos de este vídeo, ello puede causar problemas a los sistemas de control lineales en este tipo de sistemas, necesitando estrategias que aseguren tolerancia a error de modelado (control robusto) o de planificación de ganancia (cambiar controlador según el punto de funcionamiento).

Control (I)

[8: fdtapml] [Control multibucle: función de transferencia aparente \(ejemplo Matlab\)](#) ****  09:50
Materiales: [[CÓD.:](#) RGAmezclaFdTAParent.zip] [[PDF](#)] [YouTube ▶]

Este vídeo ilustra el concepto de función de transferencia “aparente” entre una entrada y una salida cuando se cierran otros bucles, que motiva la metodología de ganancia relativa para selección de emparejamiento discutida en el vídeo [??].

Se pone como ejemplo el modelo de tanque de mezclado, modelado en el vídeo [1], cuya linealización se discute en el vídeo [5].

Para ese proceso, se calculará la matriz de ganancia relativa (RGA) y se comprobará la función de transferencia aparentes entre $q_{a,in}$ y x_a cuando se cierra el bucle $q_{b,in} \rightarrow h$ y cómo la RGA predice el cambio de ganancia.

Del mismo modo, también se calculará y simulará la función de transferencia aparentes entre $q_{b,in}$ y x_a cuando se cierra el bucle $q_{a,in} \rightarrow h$.

[9: mbmix]	<p>Caso de estudio metodología ganancia relativa: control multibucle de un tanque de mezclado (Matlab)</p> <p>Materiales: [CÓD.: RGAmezclaPruebasMultibucle.zip] [PDF]</p>	<p>***  10:53</p> <p>[YouTube ▶]</p>
------------	---	---

Este vídeo ilustra el funcionamiento de la metodología de ganancia relativa para selección de emparejamiento discutida en el vídeo [??].

Se pone como ejemplo el modelo de tanque de mezclado, modelado en el vídeo [1], cuya linealización se discute en el vídeo [5]. Al modelo se le ha añadido un retardo en la salida de la concentración para aproximar dinámica del sensor y tiempo de mezclado. *Nota:* en aplicaciones reales, dependiendo de la velocidad de la agitación y de la viscosidad de los reactivos, dicha dinámica puede tener un retardo o constantes de tiempo que influyan durante una parte significativa del transitorio, y que dificulten el control preciso de la concentración.

Nota: si se desea una aproximación más sencilla a un proceso de mezclado, el emparejamiento RGA en un mezclado “estático” (instantáneo, sin dinámica) se discute en el vídeo [??]; se recomienda la visualización previa de ese vídeo para una mejor comprensión de los procesos de mezclado.

El concepto de “función de transferencia” aparente subyacente a la metodología se aborda en el vídeo [8], pero no es necesario para comprender los desarrollos aquí.

Para ese proceso, se calculará la matriz de ganancia relativa (RGA) y se harán pruebas sobre dos emparejamientos:

- El “malo”: $q_a \rightarrow x_a, q_b \rightarrow h$.
- El “bueno”: $q_a \rightarrow h, q_b \rightarrow x_a$.

Se diseñarán reguladores PD para el caso donde el resto de los bucles no esté conectado (o sea, con la función de transferencia “aislada” correspondiente). Se observará que el cambio en prestaciones cuando los dos bucles están conectados es mucho menor cuando se usa el emparejamiento recomendado por la RGA.

La simulación de estos bucles con dinámica del actuador y saturación se aborda en el vídeo [10], usando Simulink, donde se implementarán también de forma muy sencilla las estrategias *antiwindup* que completarían el diseño del control.

[10: mbmixaw]	<p>Control multibucle de un tanque de mezclado: dinámica actuador, saturación y antiwindup (Simulink)</p> <p>Materiales: [CÓD.: SimulinkMultibucleTanqueMezcla.zip] [PDF]</p>	<p>***  09:27</p> <p>[YouTube ▶]</p>
---------------	--	---

Este vídeo simula utilizando Simulink los controladores diseñados y analizados en el vídeo [9].

Para ese proceso, se harán pruebas sobre dos emparejamientos:

- El “malo”: $q_a \rightarrow x_a, q_b \rightarrow h$.
- El “bueno”: $q_a \rightarrow h, q_b \rightarrow x_a$.

La diferencia con el referido vídeo es que aquí se añadirá una dinámica de actuador $G_{act}(s) = 1/(0.02s + 1)$ y una saturación a ± 2 de ambos caudales. Esta dinámica y saturación empeorarán el comportamiento del emparejamiento no recomendado, y causarán el efecto *wind-up* en la acción integral en el bucle con el emparejamiento recomendado por la RGA.

Por ello, se propone usar el “tracking input” del bloque PID de Simulink para implementar un antiwindup realimentando la salida “real” del actuador tras la saturación (los detalles teóricos detrás de esta metodología de antiwindup se abordan en los vídeos [??], [??] y [??]). Se comprueba que la respuesta es satisfactoria y ese sería, por tanto, el diseño final recomendado en este caso de estudio multibucle. Si se considerara que el acoplamiento (transitorio en una variable cuando se cambia la otra referencia) fuera no satisfactorio, entonces deberían considerarse otras opciones como el desacoplamiento (abordado en el vídeo [11]).

[11: mixdecaw]	Tanque de mezclado, caso de estudio: control en cascada + desacoplamiento inverso + antiwindup Materiales: [CasoEstudioMezcladoCascDecAW.pdf]	****  11:40 [YouTube ▶]
----------------	---	---

Este vídeo presenta un diseño “intuitivo” de un control inicial multibucle para un tanque de mezclado (ver detalles en vídeo [9]), al que se le incorpora un control en cascada de unas servoválvulas con sensor extra de caudal en cada una de ellas.

Una vez planteado ese diseño original, si la interacción entre los bucles no fuera satisfactoria, podría incluirse un desacoplamiento. Dado que el desacoplamiento en forma inversa está particularmente indicado si se dispone de sensor de las acciones de control, como es el caso aquí, se elegirá esa forma de implementar el desacoplamiento, con bloques SISO (los detalles teóricos aparecen en el vídeo [??], y las ventajas e inconvenientes en aplicaciones en el vídeo [??]).

El desacoplamiento en forma inversa también resulta especialmente conveniente si se desea implementar *antiwindup* cuando los actuadores de los bucles esclavos saturan (Nota: el antiwindup en el caso multibucle también se discute en el vídeo [10] que podría ser de interés visualizar antes de este). Se propone un PID con “tracking input” (detalles teóricos en vídeo [??]) y un diagrama de bloques específico para desacoplamiento y feedforward necesario para llevar a cabo ese antiwindup (que no era necesario en el multibucle del vídeo [10]), con lo que se completaría el diseño propuesto.

La simulación con Simulink de la estructura propuesta se presenta en el vídeo [12].

[12: mixdecawS]	Simulación de Tanque de mezclado: multibucle y desacoplamiento inverso + antiwindup (Simulink) Materiales: [CÓD.: SimDesacAntiWindup.zip] [PDF]	****  06:17 [YouTube ▶]
-----------------	---	---

Este vídeo simula en Simulink el diseño considerado en el vídeo [11], comparando la solución multibucle recomendada por la RGA con el desacoplamiento en forma inversa; se ha considerado una dinámica de actuadores (que podría ser originada, por ejemplo, como la dinámica en bucle cerrado de un control en cascada). El vídeo es bastante breve, dado que los conceptos teórico-prácticos subyacentes se han desarrollado en el vídeo [11] y la simulación del control multibucle (con dos posibles emparejamientos y antiwindup) se ha discutido también con bastante detalle en el vídeo [10].

[13: mixfl]	Linealización y desacoplamiento por realimentación del estado: ejemplo tanque de mezclado (Matlab) Materiales: [CÓD.: DesacoplRealimentTanqueMezclado.mlx] [PDF]	*****  08:02 [YouTube ▶]
-------------	--	--

Este vídeo discute la linealización y desacoplamiento por realimentación del estado en un tanque de mezclado (mediante elementos no lineales, al ser el proceso no lineal). Un ejemplo sencillo sin usar Matlab se aborda en el vídeo [??], cuya visualización previa se aconseja.

El tanque de mezclado ha sido modelado en detalle en el vídeo [1]. No obstante, el primer minuto y medio en el vídeo aquí discutido revisa rápidamente dicho modelo, por comodidad del lector, dado que, obviamente, el modelo no lineal es necesario conocerlo para poder programar la cancelación de sus no-linealidades. Si se quisiera evitar usar expresiones no lineales, el modelo linealizado y el desacoplamiento alrededor de un punto de funcionamiento usando dicho modelo linealizado se abordan, por ejemplo, en los vídeos [5] y [11], respectivamente. Si las referencias de concentración cambiasen mucho, el uso de ecuaciones no lineales sería ventajoso (por eso el *ratio control* se usa con profusión en procesos químicos); si todo fueran pequeños incrementos alrededor de un punto nominal de operación, entonces no habría demasiada diferencia entre el enfoque no-lineal general y el desacoplamiento linealizado.

Tras el modelado, se entra en el objetivo principal del vídeo que es linealización y desacoplamiento. En concreto, primero se calculan las expresiones $\frac{dx_A}{dt} = \phi_1(v_a, v_b, q_{ain}, q_{bin})$ y $\frac{dh}{dt} = \phi_2(v_a, v_b, q_{ain}, q_{bin})$ mediante la Symbolic Toolbox, multiplicando el Jacobiano de la ecuación de salida (**jacobian**) por la ecuación de estado.

Después, se intenta resolver el sistema de dos ecuaciones

$$\phi_1(v_a, v_b, q_{ain}, q_{bin}) = \nu_1, \quad \phi_2(v_a, v_b, q_{ain}, q_{bin}) = \nu_2$$

considerando ν_1 y ν_2 como unas nuevas entradas “artificiales” que serán calculadas con posterioridad, y considerando q_{ain} y q_{bin} como incógnitas a “despejar”. Con el comando **solve** se consigue, sin ningún tipo de problema, despejar expresiones $q_{ain} = \sigma_1(v_a, v_b, \nu_1, \nu_2)$ y $q_{bin} = \sigma_2(v_a, v_b, \nu_1, \nu_2)$. Usando esas expresiones como ley de control no lineal, se consigue $\frac{dx_A}{dt} = \nu_1$, $\frac{dh}{dt} = \nu_2$ transformando el tanque de mezclado en dos sistemas lineales, desacoplados, de primer orden cada uno de ellos. Lo único que quedaría para completar el diseño serían, por ejemplo, un par de reguladores proporcionales para el proceso $\frac{1}{s}$ (integrador puro), no discutidos en este vídeo.

Otro ejemplo de este tipo de manipulaciones para un problema de control cinemático de robots aparece en el vídeo [??], y la misma idea aplicada a procesos lineales (para desacoplar) se discute en el vídeo [??].

Nótese que en un caso general, el número de derivadas de las salidas a hacer hasta que se pueda plantear el sistema de ecuaciones a resolver con **solve** puede ser diferente de uno (aunque en este caso ha sido así para ilustrar un ejemplo sencillo), e incluso puede ser diferente en unas salidas u otras, como ilustra el vídeo [??] arriba mencionado; eso puede ocurrir tanto en un caso lineal como en un caso no lineal. La metodología acaba transformando un proceso de m entradas y m salidas a **blkdiag**($1/s^{\rho_1}, 1/s^{\rho_2}, \dots, 1/s^{\rho_m}$), siendo ρ_i lo que se denomina *grado relativo* de la salida i ; en este caso sencillo ha sido $\rho_1 = 1, \rho_2 = 1$.

Modelado con agitación imperfecta

[14: tmrmod1]	Modelo procesado de mezclado en un tanque (1): división en dos volúmenes con concentración diferente	****	☰	09:42
	Materiales: [CÓD.: ModeladoTanqueMezclado2.mlx] [PDF]			[YouTube ▶]

En este vídeo se modela un tanque de mezclado, similar al del vídeo [1], pero incorporando una dinámica adicional en el mezclado (concentración interior del tanque no homogénea).

Este vídeo discute aspectos preliminares; en concreto, los detalles de cómo modelar la concentración no homogénea mediante la división del volumen interno en dos volúmenes (cada uno la mitad del total) y cómo calcular los caudales que deben intercambiarse entre suministros de entrada, salida y entre los dos sub-volúmenes. El modelo físicamente perfecto dividiría en muchos más *elementos finitos* e incorporaría ecuaciones de mecánica de fluidos, pero dicho modelado está fuera de los objetivos del presente material.

En este vídeo sólo se considera el volumen “total” y no el volumen de componente “A” que influirá en las concentraciones; el modelo completo incluyendo concentraciones se aborda en el vídeo [15], continuación de éste.

[15: tmrmod2]	Modelo tanque de mezclado con dinámica de mezclado (2): modelo en representación interna	****  10:02
Materiales: [CÓD.: ModeladoTanqueMezclado2.mlx] [PDF]		[YouTube ▶]

En este vídeo se modela un tanque de mezclado, similar al del vídeo [1], pero incorporando una dinámica adicional en el mezclado (concentración interior del tanque no homogénea).

La concentración no homogénea se modela dividiendo el volumen del depósito en dos mitades, que intercambian un caudal q_{12} entre ellos (los detalles de la determinación de la fórmula de dicho caudal se aborda en el vídeo [14]). Obviamente, el modelado más fiel sería simulando ecuaciones en derivadas parciales (Navier-Stokes, etc.) pero, vaya, eso complica mucho la simulación y en muchos casos se prefieren aproximaciones de orden bajo (por ejemplo, las aproximaciones de primer orden de un intercambiador de calor en el vídeo [??]).

La parte fundamental de la dinámica se ha modelado en el Symbolic Toolbox: tras proponer las variables intervinientes y las ecuaciones que las relacionen, se analiza primero si el modelo está bien planteado como sistema de ecuaciones diferenciales y, a continuación, se obtendrá la representación interna normalizada mediante el comando `solve`.

Las ecuaciones de estado y de salida, las saturaciones, retardo de transporte y dinámica de válvulas se integrarán todas en bloques Simulink.

[16: tmrmod3]	Modelo tanque de mezclado (v2): integración en Simulink, dinámica adicional, saturación y retardo	***  10:26
Materiales: [CÓD.: TanqueDeMezclado2S.slx] [PDF]		[YouTube ▶]

En este vídeo se integran en un subsistema de Simulink las ecuaciones de estado y de salida que se dedujeron a partir de leyes de la Física en el vídeo [15].

Se incorpora, además, la saturación de determinadas entradas, el límite inferior (cero) de los volúmenes (umbrales en el integrador), dinámica de las válvulas, así como retardo de transporte (medida de concentración aguas abajo, con retardo inversamente proporcional al caudal).

[17: tmrsim]	Tanque de mezclado no homogéneo: simulación	***  05:45
Materiales: [CÓD.: tanquemezcladoSimularLinealizar.slx] [PDF]		[YouTube ▶]

En este vídeo se simula el subsistema codificado en Simulink en el vídeo [16], integrándolo en un modelo completo de Simulink donde las salidas se visualizan con bloques `scope` y las entradas son generadas por bloques `repeating sequence`. Se comprueba el efecto de la dinámica de concentración simulando la concentración superior, inferior y aguas abajo tras el retardo.

[18: tmrlin]	Linealización tanque (V2) con dinámica de mezclado y retardo (Simulink)	***  06:18
Materiales: [CÓD.: LinealizarTanqueV2.zip]		[YouTube ▶]

Este vídeo linealiza el modelo de tanque de mezclado con retardo de transporte discutido en los vídeos [15] y [16], usando el comando `linearize` de Simulink. El desarrollo es muy parecido al del vídeo [7], pero ahora se establecen diferentes opciones para asegurar que el resultado es un modelo en tiempo continuo y que *no* se aproxima el retardo mediante Padé, que sería la opción por defecto. Para comprobar el grado de no-linealidad, se linealiza en siete puntos de funcionamiento diferentes, y se observan las diferencias en la respuesta ante incrementos escalón de entradas / en frecuencia en los distintos puntos de operación. El cuarto de los siete puntos de operación será el que se escoja como punto nominal para el diseño de controladores.

Control (II)

[19: tmplant]	Tanque de mezclado (V2): planteamiento del problema de control	***  06:59
Materiales: [MixingV2planteam.pdf]		[YouTube ▶]

En este vídeo se plantean los rasgos generales de la estructura de control a diseñar para el proceso de tanque de mezclado modelado en el vídeo [16], y linealizado en el vídeo [18]. Básicamente, dado que hay tres salidas (Q_s , $x_{a,out}$, h), y dos actuadores (V_{out} , q_B), se planteará un control en cascada con sensor extra de modo que Q_s sea controlado con V_{out} y se disponga de dos nuevas variables manipuladas (Q_b , ref_{Q_s}) para controlar $x_{a,out}$ y h , respectivamente, en una estrategia multibucle.

En el diseño de control se analizará la mejora de prestaciones si se incorporara desacoplamiento, y si se dispusieran de sensores de caudal de entrada q_A y de pureza del componente A ($x_{a,in}$) para hacer prealimentación (feedforward).

Tras el planteamiento del problema discutido aquí, el diseño en concreto y su simulación se abordan en otros vídeos de la colección, comenzando por el [20] y [21].

Otro caso de estudio de un problema de mezclado+calentamiento, resuelto por “sentido común” (sin modelos) se aborda en los vídeos [??] y [??], cuya visualización se aconseja. El mezclado estático se aborda en detalle en el vídeo [??].

[20: tmcas]	Tanque de mezclado (V2): diseño de bucle cascada esclavo de control de caudal	****  11:00
Materiales: [CÓD.: DisenyoControlTanqueNominalEsclavoCasc.mlx] [PDF]		[YouTube ▶]

Este vídeo discute el diseño para el modelo de tanque de mezclado linealizado en el vídeo [18], del control en cascada (esclavo) de caudal, de modo que una vez cerrado el bucle de control en cascada, de modo que las variables manipuladas sean caudal de salida ref_{Q_s} y caudal q_B , según se había esbozado en el vídeo [19].

En pocas palabras, se alimenta al comando `pidtune` de la Control System Toolbox con el modelo entre V_{out} y Q_s , y se deja que diseñe un PI con las opciones por defecto, de modo similar al que un operador de planta activaría la opción de *autosintonía* de un PID industrial.

El objetivo del vídeo, no obstante, es analizar las características del modelo en bucle abierto y en bucle cerrado resultante, para comprender mejor lo que está sucediendo.

En efecto, el modelo del proceso tiene un *cerro* en el origen: no se puede mantener un caudal de salida constante sin que la entrada (abertura de válvula) crezca hasta infinito. En el vídeo se explica la razón física de ello (hay que abrir más conforme el depósito se va vaciando), y el efecto de esta cancelación polo-cero en el bucle cerrado.

El análisis se hace tanto en tiempo (step) como en frecuencia (**bodemag**).

Una vez verificado el comportamiento del bucle cerrado, se genera un nuevo modelo **G2** cuyas entradas sean perturbaciones de caudal y composición en el *wild feed* y referencias de caudales de *B* en el *controlled feed* y de caudal de salida (al control aquí diseñado). Este modelo se usará en fases posteriores para multibucle, desacoplamiento y prealimentación.

Nota: en muchas aplicaciones, habría sido suficiente (y aconsejable) suponer que el controlador en cascada se comporta “suficientemente bien” para poder asumir que la variable manipulada es el caudal de salida, sin detallar el control interno ni cerrar el bucle con un modelo complejo del proceso... modelando directamente de una forma similar a la realizada en el vídeo [6], y eliminando los aspectos relacionados con la válvula de salida del modelo: el objetivo “práctico” del control en cascada es *simplificar* y no *complicar* las cosas; el objetivo de los profesores de control es enseñar cosas difíciles (complicando, a veces, las cosas sin que estrictamente fuese necesario en la práctica).

[21: tmrml]	Tanque de mezclado (V2): control multibucle (con cascada preexistente)	***  12:52 [YouTube ▶]
Materiales: [CÓD.: tmmb.zip]		

Este vídeo aborda el diseño del control multibucle del tanque de mezclado cuyo modelo linealizado se obtuvo en el vídeo [18], suponiendo un bucle de control en cascada de caudal de salida preexistente (diseñado en el vídeo [20]), siguiendo los pasos a la solución global del problema esbozados en el vídeo [19].

El emparejamiento seleccionado es nivel \rightarrow caudal de salida, concentración \rightarrow caudal de disolvente q_b .

Dado que, en bucle cerrado, de referencia de caudal de salida a nivel se tiene integrador, se diseña con **pidtune** un regulador *proporcional* que consigue una dinámica satisfactoria ante referencia. No obstante, tiene error de posición ante perturbaciones en la entrada según se justifica cualitativamente y se verifica en simulación (un bucle cerrado estable con integradores debe hacer cero la entrada al integrador... que sólo es el error si el integrador está en el regulador). Por ello, al final del vídeo, se rehace el diseño y se sintoniza un regulador **PI**.

El control de concentración se hace, también con **pidtune**, con un **PI**.

Al final del vídeo, se usa el comando **connect** para simular el bucle cerrado resultante, tanto ante cambios de referencia como ante perturbaciones en respuesta ante escalón (el estudio del error en bucle cerrado en el dominio de la frecuencia se aborda en el vídeo [24]).

[22: tmff1]	Tanque de mezclado (V2): prealimentación	****  09:47 [YouTube ▶]
Materiales: [CÓD.: TanqueFFyDCStatic.zip] [PDF]		

Nota: En la grabación del vídeo existía un error (bastante irrelevante) en lo que respecta a prealimentación/desacoplamiento en el modelo del control en cascada. Ha sido corregido en el archivo .zip adjunto de materiales.

Este vídeo discute el feedforward (en ganancia estática) para el modelo de tanque de mezclado linealizado en el vídeo [18], una vez cerrado el bucle de control en cascada, de modo que las variables manipuladas son caudal de salida ref_{Q_s} y caudal q_B . Los

sensores para prealimentación serán el caudal y la pureza del reactivo entrante por la alimentación izquierda ($Q_a, x_{a,in}$). Obviamente, la decisión sobre si se compran o no dichos sensores podría venir aconsejada según la mejora de prestaciones con/sin prealimentación en simulaciones.

Sin asumir ningún desacoplamiento preexistente, la solución se obtiene con inversas de matrices (teoría en vídeo [??]). Como el proceso tiene integradores, debe calcularse la prealimentación con los valores de las matrices de ganancia a una frecuencia suficientemente pequeña pero no cero, y comprobar que cambiando dicha frecuencia no cambian apreciablemente las matrices obtenidas (esto es, que existe el límite cuando $s \rightarrow 0$ de las matrices de prealimentación). En este vídeo, se ha usado el comando `evalfr` para llevar a cabo este paso.

Se analiza el significado físico de la matriz de prealimentación, comparándola con el cálculo de nuevos puntos de funcionamiento constante ante los nuevos valores de las variables de entrada no manipuladas: para pequeños incrementos, el resultado linealizado coincide con dicho cálculo de puntos de funcionamiento. Para incrementos grandes, se plantea la conveniencia de hacer un feedforward con los cálculos no lineales exactos como, de hecho, se hace en muchas aplicaciones mediante ratio control. En este caso de estudio, esto se realiza, por ejemplo, en el vídeo [27].

La prealimentación lineal asumiendo desacoplamiento preexistente, y la comparación con los resultados de este vídeo, se discuten en el vídeo [23].

[23: tmdcff2]	Tanque de mezclado (V2): desacoplamiento y prealimentación desacoplada	****  10:31
Materiales: [Cód.: TanqueFFyDCStatic.zip] [PDF]		[YouTube ►]

Este vídeo discute el desacoplamiento (en ganancia estática) para el modelo de tanque de mezclado linealizado en el vídeo [18], una vez cerrado el bucle de control en cascada, con variables manipuladas caudal de salida ref_{Q_s} y caudal q_B , que se supone controlan concentración de salida y nivel, respectivamente, en una estructura multibucle.

En efecto, aunque el control de nivel (Q_s) no influye sobre la concentración (en régimen permanente), el de concentración (q_B) sí influye sobre nivel. Por ello, un desacoplamiento que incremente en 1 unidad Q_s cuando se incrementa en la misma cantidad q_B podría ser aconsejable.

La segunda parte del vídeo discute la prealimentación con los sensores de caudal y pureza del reactivo entrante por la alimentación izquierda ($Q_a, x_{a,in}$).

Sin suponer desacoplamiento, las inversas de matrices asociadas se han discutido en el vídeo [22]. Con desacoplamiento preexistente, sólo hay que dividir por la *diagonal* de la matriz de transferencia, resultando, quizás, más sencillo de explicar a no expertos en control multivariable. Se analiza el significado físico de las diferentes ganancias de feedforward en el caso desacoplado, así como la relación con las del vídeo [22]. La extensión al modelo no lineal (en equilibrio) de estos cálculos con la matriz de ganancia se aborda en el vídeo [27].

[24: tmrffdyn]	Tanque de mezclado (V2): prealimentación y desacoplo dinámicos	****  09:47
Materiales: [Cód.: TanqueFFyDCDynamic.zip] [PDF]		[YouTube ►]

Este vídeo discute los aspectos dinámicos del diseño de prealimentación y desacoplamiento para el modelo de tanque de mezclado lineal (vídeo [18]). El diseño puramente estático (bloques feedforward y desacoplamiento constantes) ha sido discutido en los vídeos [22] y [23]... es importante destacar que la conclusión de este vídeo será que esos diseños constantes son muy razonables.

Los sensores para prealimentación eran el caudal y la pureza del reactivo entrante por la alimentación izquierda ($q_w, x_{a,in}$). Se supone preexistente un control en cascada diseñado en el vídeo [20].

En primer lugar, se hace un análisis sobre si prealimentación y desacoplo son realmente necesarios y hasta qué rapidez (ancho de banda), examinando el diagrama de Bode de magnitud (**bodemag**) de los errores ante perturbaciones y referencias. Se comprueba que con un ancho de banda de aproximadamente 10 rad/s sería suficiente.

Sin asumir ningún desacoplamiento preexistente, la solución de la prealimentación se obtiene con inversas de matrices (teoría en vídeo [??]); en este caso, con la inversa de *matrices de transferencia*. Dibujando su diagrama de Bode, se comprueba que no es realizable (respuesta tiende a infinito) pero que hasta 10 rad/s es aproximadamente constante, lo cual justifica la conveniencia de la solución *estática* antes mencionada.

En cuanto al desacoplamiento (en forma inversa), se obtiene un bloque con dinámica realizable (aunque, de igual modo, dado que no hay cambios relevantes hasta 10 rad/s, sería muy razonable implementar el desacoplamiento constante estático).

[25: tmsimtodolin] 06:25	Tanque de mezclado (V2): simulación de multibucle, cascada, desacoplamiento y prealimentación (modelo linealizado)	***  [YouTube ▶]
Materiales: [CÓD.: Tmsimulinktodoconlineal.zip]		

Este vídeo implementa en un diagrama de bloques de Simulink la estructura de control completa planteada en el vídeo [19], sobre el modelo linealizado. Incorpora:

- Modelo linealizado del tanque (4 entradas, 3 salidas), del vídeo [18],
- Control en cascada (esclavo) de caudal de salida, manipulando válvula de salida, vídeo [20],
- Control multibucle de nivel y concentración, vídeo [21], manipulando referencia a control de caudal de salida y caudal entrante de disolvente q_b , respectivamente.
- Prealimentación estática (matriz, vídeo [22]), con sensores de caudal y concentración del *wild feed*, y
- desacoplamiento (dinámico, vídeo [24]).

Mediante los adecuados bloques **Manual Switch**, se simulan distintas combinaciones de desacoplamiento o feedforward, bien presente o bien ausente.

[26: tmsimbcnl]	Tanque de mezclado V2: simulación no lineal del control linealizado	***  08:27 [YouTube ▶]
Materiales: [CÓD.: simulaffydcNOlineal.slx]		

Este vídeo implementa en un diagrama de bloques de Simulink la estructura de control completa planteada en el vídeo [19], sobre el modelo linealizado, que se simuló en el vídeo [25] con dicho modelo.

En este vídeo se simula sobre el modelo no lineal construido en el vídeo [16], incorporando los adecuados pasos desde variables en unidades absolutas a incrementales y viceversa. Al igual que en el vídeo del modelo lineal, mediante los adecuados bloques **Manual Switch**, se simulan distintas combinaciones de desacoplamiento o feedforward presente o ausente.

Se observan los siguientes comportamientos que la simulación lineal no consideraba:

1. Dinámicas del transitorio diferentes si se cambia el punto de funcionamiento (referencias).

2. Efectos perjudiciales de la saturación (mala cancelación de perturbaciones extremas e inestabilidad interna –windup– de los reguladores PI cuyo actuador satura).

En el primer caso, podría plantearse una planificación de ganancia. En el segundo caso, debería tenerse en cuenta la presencia de saturación mediante antiwindup o/y cambios en el desacoplamiento y feedforward. Las soluciones a estos problemas *no* entran dentro de los objetivos de este vídeo.

[27: tm2ffnl]	Tanque de mezclado: prealimentación, desacoplamiento, 2GL, ratio control con modelo no lineal Materiales: [CÓD.: tm2ffnl.zip] [PDF]	****  11:48 [YouTube ▶]
---------------	---	---

Este vídeo diseña, para el caso de estudio del tanque de mezclado, usando un modelo no lineal, una estructura que calcula el punto de equilibrio de las variables manipuladas q_b y ref_{q_s} en función de las perturbaciones y referencias. Con ello, se unifica el cálculo (en régimen permanente) de bloques que en control lineal serían prealimentación, control con 2 grados de libertad y desacoplamiento, según lo discutido desde un punto de vista genérico en el vídeo [??], cuya visualización se aconseja para poder extraer conclusiones generales importantes en ingeniería de control, más allá de este problema concreto. Este vídeo, por tanto, sería la generalización al caso no lineal (pero sólo en régimen permanente, equilibrio constante) de los cálculos lineales de prealimentación y desacoplamiento vistos en los vídeos [22] y [23].

El cálculo está dividido en dos bloques **Matlab Function** de Simulink:

- Cálculo del caudal de disolvente en equilibrio en función de q_w , $x_{a,in}$ y $ref_{x_{a,out}}$, equivalente a prealimentación + 2 GL,
- Cálculo del caudal de salida dados q_w y q_b (siendo éste determinado por el punto de funcionamiento y bucle de concentración). Esto sería equivalente a prealimentación y desacoplamiento dado que q_b puede no ser igual al que el primer cálculo haya decidido, debido a transitorios del bucle de concentración y posibles saturaciones.

[28: tm2aw]	Tanque de mezclado no lineal: diseño del antiwindup Materiales: [CÓD.: tm2ffnl.zip] [PDF]	***  08:22 [YouTube ▶]
-------------	---	--

Este vídeo discute la elección de las distintas estrategias de *antiwindup* en el control en cascada esclavo (se elige **clamping**), y en los PI de nivel y concentración (debido al cálculo de punto de funcionamiento externo, variable, y ser maestros de controles en cascada, se elige **tracking-mode antiwindup**). La teoría del antiwindup con entrada externa de seguimiento se discute en los vídeos [??] y [??], y la sintonía de la dinámica del tracking en el vídeo [??], cuya visualización previa se recomienda. El funcionamiento de la estrategia decidida se comprueba en simulación en la parte final del vídeo.

[29: tm2orm]	Tanque de mezclado: necesidad de override para proteger de desbordamientos (motivación) Materiales: [CÓD.: tmsolfinal.zip] [PDF]	**  03:24 [YouTube ▶]
--------------	--	---

Este vídeo simula la estructura de control completa diseñada en los vídeos [27] y [28] para un tanque de mezclado no lineal, ante fallos en la válvula de salida o referencias de concentración y nivel excesivamente bajas. Se observa que se pueden alcanzar niveles muy diferentes al punto de funcionamiento y se motiva, por tanto, la conveniencia de añadir un control “override” de modo que si el nivel sale de unos límites entonces el actuador q_b pase a controlar nivel en vez de concentración. Los detalles de ese diseño se ven en el vídeo [30].

[30: tm2orpi] Tanque de mezclado: override nivel mínimo y máximo (control proporcional) ****  12:03
Materiales: [[CÓD.:](#) tmsolfinal.zip] [[PDF](#)] [YouTube ▶]

Este vídeo detalla la lógica de “*override*” en el tanque de mezclado para intentar no superar unas referencias de nivel máximo y mínimo. Básicamente, en situación normal el actuador q_b (caudal de disolvente) será usado para controlar concentración, pero si el control de nivel no cumple su objetivo (por saturación o fallo en la válvula de salida, o por una referencia de usuario fuera de límites, tal y como se simuló en el vídeo de motivación [29]) entonces se producirá el *override* y q_b pasará a controlar nivel.

Se proponen dos controladores de nivel proporcionales, y el vídeo discute las razones que justifican la propuesta de lógica de conmutación entre los tres posibles modos de control (overflow, underflow, normal). Se simula el resultado y se observa que existe un “error de posición” de modo que los límites del *override* son sólo aproximados. La incorporación de acción integral en los reguladores con *override* podría ser razonable, pero eso complica el escenario *antiwindup*; estas ideas se discuten en el vídeo [31].

[31: tm2orpi] Tanque de mezclado: override con acción integral y opciones antiwindup ****  10:34
Materiales: [[CÓD.:](#) tmsolfinal.zip] [[PDF](#)] [YouTube ▶]

En este vídeo se plantea usar reguladores PI para que cuando el *override* en el tanque de mezclado se active, que se alcancen las referencias de seguridad (aunque podrían ser brevemente rebasadas en el transitorio) debido a la acción integral (para evitar los problemas de error de posición discutidos en el vídeo [30] con *override* proporcional). Ello requiere pensar en diferentes opciones *antiwindup* (adicionales a las del bucle de concentración) y se prueban *clamping* y *back-calculation* (con distintas constantes K_t) así como PI con entrada externa de seguimiento (*tracking mode*) que, conceptualmente, sería la solución teóricamente mejor para seguir al controlador activo en vez de simplemente esperar en saturación a ser activado, aunque también tiene sus inconvenientes prácticos.