Solución del problema LQR de Horizonte Finito

Juan Pablo Requez
juanrequez@gmail.com
jprequez@ucla.edu.ve
jrequez@unexpo.edu.ve

Un modelo dinámico puede ser descrito a través de un sistema de ecuaciones diferenciales. Estos modelos pueden ser modelos lineales o no lineales. Vamos a concentrarnos en los modelos lineales.
En general, la representación de un sistema de ecuaciones diferenciales lineales para describir a un sistema se conoce como *representación en variable de estado*. Esto se representa como (1):

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t) \tag{1}$$

y el sistema se dice que es *Lineal Invariante en el Tiempo (LTI, Linear Time Invariant)*.
La *salida del sistema y(t)* es una combinación lineal de los *estados* y de las *entradas*, según (2)

$$y(t) = Cx(t) + Du(t) \tag{2}$$

Pero solo nos interesan sistemas donde D=0 y por lo tanto

$$y(t) = Cx(t) \tag{3}$$

Nótese que (3) **no es una ecuación diferencial**. Estas ecuaciones reciben los nombres que se indican en la Tabla 1.

Tabla 1. ecuaciones de un sistema lineal en espacio de estados

Ecuación	Nombre
$\frac{dx(t)}{dt} = Ax(t) + Bu(t)$	Ecuación de estados
$y(t) = Cx(t)$	Ecuación de salida

En la Tabla 2 se observa el nombre que le corresponde a los elementos de las ecuaciones de un sistema en espacio de estados. Recuerda que *A*, *B* y *C* pueden ser, o no, dependientes del tiempo, pero *x*, *y* y *u* son siempre dependientes del tiempo.

Tabla 2. Elementos de un sistema lineal en espacio de estados

Matriz/vector	Tamaño	Nombre
A	$n \times n$	matriz de estados
B	$n \times m$	matriz de entradas
C	$p \times m$	matriz de salida
x(t)	$n \times 1$	vector de estados
u(t)	$m \times 1$	vector de entradas
y(t)	$p \times 1$	vector de salidas

Es necesario, para terminar de describir el sistema en espacio de estados, indicar que se debe conocer la condición inicial, $x(t_0) = x_0$.

En general, cada estado de un sistema representa a aquellas variables que cambian temporalmente de forma dependiente a la entradas, y por lo general pueden reconocerse en las expresiones físicas y químicas que describen al sistema porque ellas aparecen derivadas respecto al tiempo.
En resumen, considérese el sistema lineal (4)

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{4}$$

con condición inicial conocida, $x(t_0)$, y con las matrices $A \in R^{n\times n}, B \in R^{n\times m}, C \in R^{p\times n}$. Este sistema se dice que está representado en *espacio de estados*, y un diagrama de bloques para él se muestra en la Figura 1

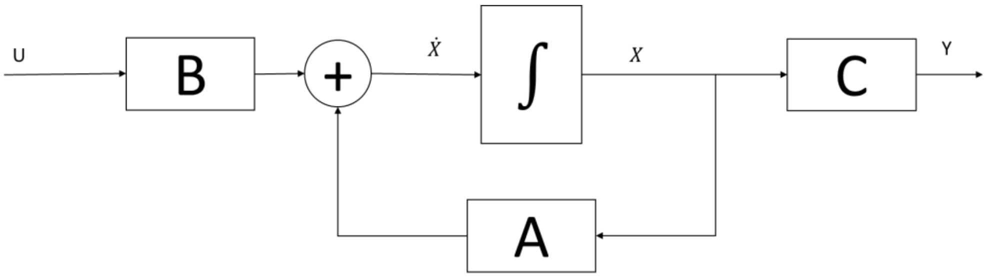


Figura 1. Diagrama de bloques del sistema lineal en espacio de estados

Como veremos a continuación, cuando se trabaja con un sistema lineal variante en el tiempo y se elige un funcional cuadrático, el problema de control óptimo se convierte en uno relativamente sencillo de resolver, y en uno de realimentación de estados, en donde las entradas se pueden escribir como una combinación lineal de los estados, como se muestra en la Figura 2

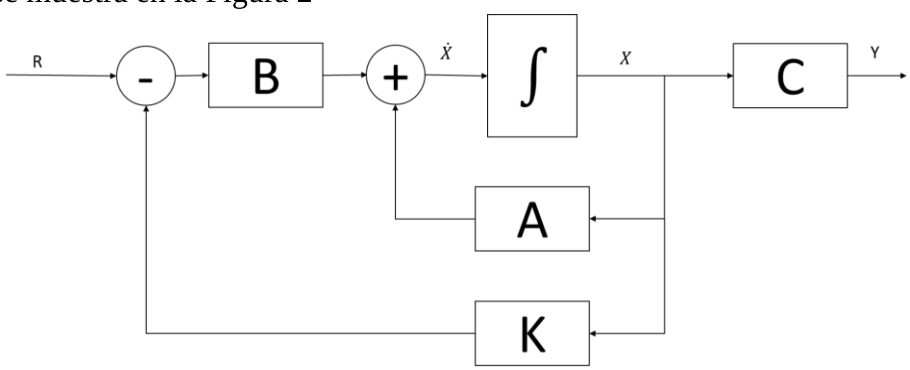


Figura 2. Diagrama de bloques del sistema lineal en espacio de estados en lazo cerrado

I. Problema del Regulador Lineal Cuadrático – Horizonte finito

Decimos que un controlador es un *regulador* si el objetivo del sistema de control es eliminar el efecto de perturbaciones, mientras que decimos que un controlador es un *seguidor (tracker, en inglés)* si su objetivo es alcanzar un valor de referencia o set point. Ambos problemas son diferentes, ya que uno desea que las perturbaciones de la salida desaparezcan mientras que el otro intenta que la salida alcance un valor deseado.

Decimos que un sistema es lineal si, *bueno, es lineal.* 😊

Decimos que un funcional objetivo es *cuadrático* si este puede escribirse a través de formas cuadráticas, como las estudiadas en la semana 2. Luego, el problema del *Regulador Lineal Cuadrático* es un problema de un diseño de controladores que busca eliminar el efecto de las perturbaciones sobre estados descritos a través de un modelo lineal y usando un funcional objetivo cuadrático. Esto quedará claro en breve.

Sea un sistema lineal variante en el tiempo

$$\frac{dx(t)}{dt} = A(t)x(t) + B(t)u(t), \quad x(t_0) = x_0 \tag{5}$$

considerando un índice de desempeño cuadrático

$$J = \frac{1}{2}x^T(t_f)F(t_f)x(t_f) + \frac{1}{2}\int_{t_0}^{t_f} x^T(t)Q(t)x(t) + u^T(t)R(t)u(t)\,dt \tag{6}$$

donde

u(t)	sin restricciones
t_f	fijo (conocido)
$x(t_f)$	libre (desconocido)

y

Q(t)	$n \times n$	Simétrica y **semidefinida positiva**
R(t)	$m \times m$	Simétrica y **definida positiva**
F(t)	$n \times n$	Simétrica y **semidefinida positiva**

A este problema se le dice *problema del regulador lineal cuadrático*, o simplemente *regulador lineal cuadrático* (LQR, por las siglas para *Linear Quadratic Regulator)* puesto que se controlan directamente los estados (y no hay dependencia del problema de una referencia o set point), la ecuación de estado es lineal y el índice es cuadrático. La segunda parte del nombre es *Horizonte finito* porque el tiempo final t_f de funcionamiento del controlador es conocido y es un número finito. La palabra *horizonte* en control es usada para indicar el tiempo requerido para el funcionamiento de un controlador. Cuando el índice de desempeño es calculado hasta infinito (es decir $t_f \to \infty$) se dice que el problema es de *horizonte infinito* y es muchísimo ás sencillo que el problema de horizonte finito, aunque tal cosa no sea nuestra experiencia previa de nuestros años de estudio de cálculo. Estudiaremos el problema de horizonte infinito en una de las semanas próximas.

El problema LQR puede resolverse usando el razonamiento indicado en el problema de control óptimo usando el principio de mínimo de Pontryagin. Hay algunos detalles de la implementación de este método, en particular por una transformación que se usa para lograr que la representación sea de lazo cerrado. Estos detalles están disponibles en la bibliografía. A este respecto, la deducción de (Naidu, 2003) es bastante agradable y fácil de entender. La de (Wiberg, 1971) es bastante breve, y aunque usa otra notación diferente a la propuesta en este documento, es satisfactoria.

A. Solución del problema LQR-Horizonte infinito.

Sea un sistema lineal variante en el tiempo

$$\frac{dx(t)}{dt} = A(t)x(t) + B(t)u(t), \qquad x(t_0) = x_0 \tag{7}$$

considerando un índice de desempeño cuadrático

$$J = \frac{1}{2}x^T(t_f)F(t_f)x(t_f) + \frac{1}{2}\int_{t_0}^{t_f} x^T(t)Q(t)x(t) + u^T(t)R(t)u(t)\,dt \tag{8}$$

donde

u(t)	sin restricciones
t_f	fijo (conocido)
$x(t_f)$	libre (desconocido)

y

Q(t)	$n \times n$	Simétrica y **semidefinida positiva**
R(t)	$m \times m$	Simétrica y **definida positiva**
F(t)	$n \times n$	Simétrica y **semidefinida positiva**

Entonces, la solución del problema está dado por

$$u(t) = -R^{-1}(t)B^T(t)P(t)x(t) \tag{9}$$

donde P(t) es una matriz definida positiva, simétrica, de tamaño $n \times n$, que es solución de la *ecuación diferencial matricial de Riccati*

$$\frac{dP(t)}{dt} = -(P(t)A(t) + A^T(t)P(t) - P(t)B(t)R^{-1}(t)B^T(T)P(t) + Q(t)) \tag{10}$$

que posee condición final

$$P(t_f) = F(t_f) \tag{11}$$

y la trayectoria óptima de los estados está dada por

$$\frac{dx(t)}{dt} = (A(t) - B(t)R^{-1}(t)B^T(t)P(t))x(t) \tag{12}$$

y el costo óptimo está dado por

$$J = \frac{1}{2}x^T(0)P(0)x(0) \tag{13}$$

Al respecto de la solución, debemos aclarar algunas cosas:

- La parte más difícil del problema es resolver la ecuación diferencial de Riccati. Véase que P es una matriz, y por lo tanto, todos sus elementos son variantes en el tiempo y son solución de un sistema de ecuaciones diferenciales.
- La ecuación diferencial de Riccati no tiene $n \times n$ incógnitas (que corresponderían a $n \times n$ elementos de la matriz) porque esta solución es simétrica para todo *t*, luego, solo es necesario calcular los elementos de la diagonal principal y todos los que están por encima de ella. De esta forma, el problema tiene en realidad $\frac{n\times(n+1)}{2}$ incógnitas.
- Contrario al problema general (explicado en el formalismo hamiltoniano del principio del mínimo de Pontryagin), este problema NO es TPBVP. (o bueno, no lo es en el sentido estricto). Como la ecuación diferencial de Riccati no depende de x(t) (revísala), su solución puede hallarse independiente de la solución de los estados. Luego el problema separa la solución de los estados del cálculo de P(t), que puede hacerse de forma completamente independiente, pero hacia atrás en el tiempo. Esto no es un problema, ya que la dirección de resolución de una ecuación diferencial es complemente arbitraria por cualquier método numérico o analítico, da lo mismo resolverla hacia adelante que hacia atrás, desde el punto de vista procedimental, quiero decir.
- Como la solución de P(t) no depende de los estados, entonces es natural que uno primero resuelva la ecuación diferencial de Riccati y luego construya el resto. El resto puede ser armado directamente por sustitución, y en nuestro caso, implementado directamente en simulación, sin tener que resolver alguna cosa más.
- Como se muestra en la ecuación (9), la relación entre *u* y *x* está dada por una expresión matricial. Esta expresión puede representarse como (14), lo que convierte a (9) en (15).

$$K(t) = R^{-1}(t)B^T(t)P(t) \tag{14}$$

$$u(t) = -K(t)x(t) \tag{15}$$

- A la matriz *K(t)* se le conoce como *ganancia de Kalman*, y es una ganancia de realimentación del vector de estados.
- Aunque el sistema sea invariante en el tiempo (es decir, A, B, Q, R, F sean constantes), P(t) no será una matriz constante, puesto que es la solución de una ecuación diferencial.
- En general, aunque el problema está descrito para tiempo inicial t_0, lo más común en los problemas de control es utilizar $t_0 = 0$, siempre que el sistema sea invariante en el tiempo, claro.
- La pregunta de un lector preocupado puede ser ¿Y los multiplicadores de Lagrange?. Esta es una buena pregunta. La resolución del problema LQR presentada requiere escribir a los multiplicadores de Lagrange según

$$\lambda(t) = P(t)x(t) \tag{16}$$

Y de aquí se obtiene la ecuación diferencial de Riccati.

La solución de este problema la resumo en la Tabla 3, donde omito la dependencia temporal de las matrices del modelo y del índice de desempeño. La dependencia temporal de P(t) la incluyo, para remarcar que no hay forma que esta sea constante en esta formulación. Esta tabla sirve para una consulta más sencilla.

Tabla 3. Resumen LQR – Horizonte Finito

Solución del problema LQR – Horizonte Finito	
Modelo de la planta	$\frac{dx}{dt} = Ax + Bu$
Índice de desempeño	$J = \frac{1}{2}x^T(t_f)F(t_f)x(t_f) + \frac{1}{2}\int_{t_0}^{t_f} x^TQx + u^TRu\,dt$
Condiciones iniciales (conocidas)	$x(t_0) = x_0$
Tiempo final (conocido)	t_f
Ecuación diferencial de Riccati	$\frac{dP(t)}{dt} = -(P(t)A + A^TP(t) - P(t)BR^{-1}B^TP(t) + Q)$ $P(t_f) = F(t_f)$
Ganancia de realimentación	$K(t) = R^{-1}B^TP(t)$
Ecuación de control	$u(t) = -K(t)x(t)$
Ecuación de estados	$\frac{dx}{dt} = (A - BR^{-1}B^TP(t))x$
índice de desempeño óptimo	$J = \frac{1}{2}x^T(0)P(0)x(0)$

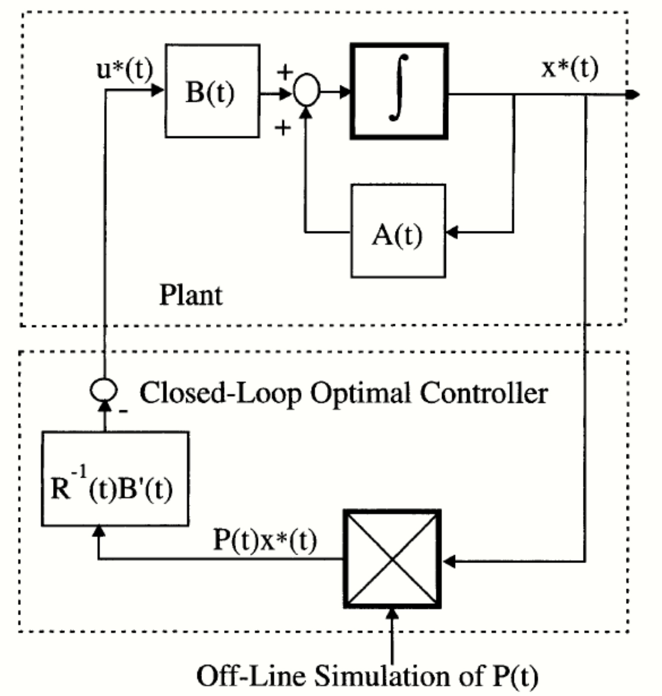


Figura 3. Diagrama de bloques del sistema lineal en espacio de estados en lazo cerrado (LQR, Horizonte Finito) (Naidu, 2003)

En la Figura 3 se muestra que es necesario hacer la simulación de P(t) antes de implementar el sistema de control de realimentación del vector de estados. Este diagrama es esencial para la implementación que realizaremos en simulación, pero debe considerarse que en la implementación física a nivel industrial es necesario poseer algún tipo de almacenamiento de datos para almacenar los valores de P(t), y alguna forma de interpolar entre estos valores ya que no puede tenerse de forma continua.

Procedimiento 3. Resolución del problema LQR de horizonte finito.

Dada la planta y el índice de desempeño mostrados en la Tabla 3, con condiciones iniciales conocidas $x(t_o) = x_0$ y tiempo final conocido t_f.

1. Resuelva la ecuación diferencial de Riccati a través de un método numérico.
2. Escriba la ganancia de realimentación K(t)
3. Sustituya la ganancia de realimentación en la ecuación de control u(t)
4. Determine la trayectoria de x(t) a través de la ecuación de estados a través de un método numérico.
5. Calcule el índice de desempeño óptimo.

Observaciones del procedimiento 3:

- En general, estos problemas no pueden ser resueltos de forma analítica porque la ecuación diferencial de Riccati es complicada y no lineal.
- Nuestro interés en esta asignatura no es la obtención de estas soluciones de manera explícita y analítica. Para la ejecución del procedimiento 3, usaremos los métodos numéricos disponibles en Matlab y la simulación dinámica de Simulink. Esto se presenta en los ejemplos siguientes. También, aunque la

solución es la misma para sistemas LTV y LTI, nos concentraremos en problemas LTI únicamente en los ejercicios, aunque la solución presentada no diferencia entre ellos.

Ejemplo 1. Control de nivel en un tanque abierto

Considere el conjunto de tanques que se muestran en la Figura 4. En este proceso todos los tanques están abiertos a la atmósfera y la temperatura es constante. Las aperturas de las válvulas se mantienen constante. Las variables de control son el flujo de entrada F_i y el flujo extraído f_0.

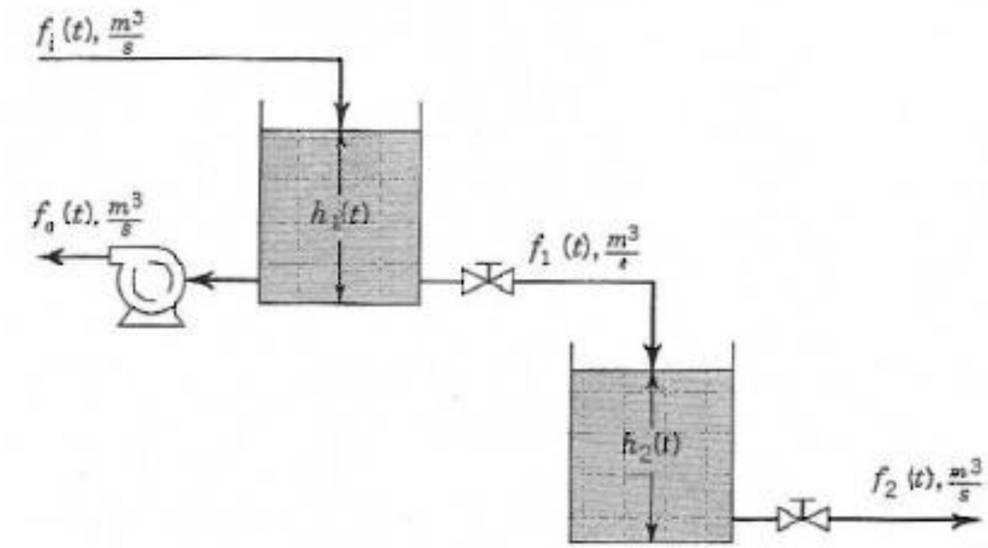


Figura 4. sistema de nivel del ejemplo 1

El modelo del nivel en variables de desviación puede representarse como

$$\frac{d}{dt}H_1 = -0.1107H_1 + F_i - F_o$$

$$\frac{d}{dt}H_2 = 0.1107H_1 - 0.1107H_2$$

Se desea implementar una estrategia de control que minimice la variación del nivel y además minimice los cambios de F_i y f_0:

$$J = \frac{1}{2}\int_0^{100} 10x_1^2 + 15x_2^2 + 0.1u_1^2 + u_2^2 dt$$

desde una posición inicial de 0.1 para cada nivel.

Solución:

Digamos que

$$x_1 = H_1 \qquad x_2 = H_2$$

$$u_1 = F_o \qquad u_2 = F_i$$

Con esto

$$\frac{d}{dt}x_1 = -0.1107x_1 + u_2 - u_1$$

$$\frac{d}{dt}x_2 = 0.1107x_1 - 0.1107x_2$$

en forma matricial:

$$\begin{pmatrix} \frac{d}{dt}x_1 \\ \frac{d}{dt}x_2 \end{pmatrix} = \begin{pmatrix} -0.1107 & 0 \\ 0.1107 & -0.1107 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

Donde observamos que

$$\frac{d}{dt}X = A \cdot X + B \cdot U$$

$$A = \begin{pmatrix} -0.1107 & 0 \\ 0.1107 & -0.1107 \end{pmatrix}, \qquad B = \begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}$$

$$X(0) = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}$$

El índice de desempeño puede escribirse como

$$J = \frac{1}{2}\left(\int_0^{100} 10x_1^{\ 2} + 15x_2^{\ 2} + 0.1u_2^{\ 2} + u_1^{\ 2}\ dt\right)$$

y usando la representación de formas cuadráticas

$$J = \frac{1}{2} \cdot \int_0^{100} X^T \cdot \begin{pmatrix} 10 & 0 \\ 0 & 15 \end{pmatrix} \cdot X + U^T \cdot \begin{pmatrix} 0.1 & 0 \\ 0 & 1 \end{pmatrix} \cdot U\ dt$$

Luego se tiene que

$$Q = \begin{pmatrix} 10 & 0 \\ 0 & 15 \end{pmatrix} \qquad R = \begin{pmatrix} 0.1 & 0 \\ 0 & 1 \end{pmatrix} \qquad F = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Véase que F es semidefinida positiva. ¿Por qué?
Por último:

$$t_f = 100$$

Paso 1: Véase el problema de resolver la ecuación diferencial matricial de Riccati. La variable de la ecuación

$$\frac{dP(t)}{dt} = -(P(t)A + A^T P(t) - P(t)BR^{-1}B^T P(t) + Q)$$

debe ser una matriz P, $n \times n$, que en este caso es

$$P = \begin{pmatrix} p_{11} & p_{12} \\ p12 & p_{22} \end{pmatrix}$$

Esto requiere escribir

$$\begin{pmatrix} \frac{d}{dt}p_{11} & \frac{d}{dt}p_{12} \\ \frac{d}{dt}p12 & \frac{d}{dt}p_{22} \end{pmatrix} = -\left[\begin{pmatrix} p_{11} & p_{12} \\ p12 & p_{22} \end{pmatrix}\cdot\begin{pmatrix} -0.1107 & 0 \\ 0.1107 & -0.1107 \end{pmatrix} + \begin{pmatrix} -0.1107 & 0 \\ 0.1107 & -0.1107 \end{pmatrix}^T\cdot\begin{pmatrix} p_{11} & p_{12} \\ p12 & p_{22} \end{pmatrix} - \begin{pmatrix} p_{11} & p_{12} \\ p12 & p_{22} \end{pmatrix}\cdot\begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}\cdot\begin{pmatrix} 0.1 & 0 \\ 0 & 1 \end{pmatrix}^{-1}\cdot\begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}^T\cdot\begin{pmatrix} p_{11} & p_{12} \\ p12 & p_{22} \end{pmatrix} + \begin{pmatrix} 10 & 0 \\ 0 & 15 \end{pmatrix}\right]$$

y al simplificar

$$\begin{pmatrix} \frac{d}{dt}p_{11} & \frac{d}{dt}p_{12} \\ \frac{d}{dt}p12 & \frac{d}{dt}p_{22} \end{pmatrix} = \begin{pmatrix} 11.0\cdot p_{11}^{2} + 0.2214\cdot p_{11} - 0.1107\cdot p12 - 0.1107\cdot p_{12} - 10.0 & 0.2214\cdot p_{12} - 0.1107\cdot p_{22} + 11.0\cdot p_{11}\cdot p_{12} \\ 0.2214\cdot p12 - 0.1107\cdot p_{22} + 11.0\cdot p12\cdot p_{11} & 0.2214\cdot p_{22} + 11.0\cdot p12\cdot p_{12} - 15.0 \end{pmatrix}$$

Lo que conduce a tres ecuaciones diferenciales (véase que P es simétrica, por lo tanto p_{12} aparece dos veces en la ecuación con expresiones idénticas, y una de ellas no es necesaria entonces)

$$\frac{d}{dt}p_{11} = 11.0\cdot p_{11}^{2} + 0.2214\cdot p_{11} - 0.1107\cdot p12 - 0.1107\cdot p_{12} - 10.0$$

$$\frac{d}{dt}p_{12} = 0.2214\cdot p_{12} - 0.1107\cdot p_{22} + 11.0\cdot p_{11}\cdot p_{12}$$

$$\frac{d}{dt}p_{22} = 0.2214\cdot p_{22} + 11.0\cdot p12\cdot p_{12} - 15.0$$

Estas ecuaciones están acopladas y son no lineales (cuadráticas, para ser preciso).

La implementación de la ecuación de Riccati fue ejecutada en Matlab, con un programa elaborado para resolverlo. Es un programa sencillo (que está disponible en el aula virtual) y está formado por la función "solveRiccati" que invoca a "difRiccati". Como se estudió previamente en *Modelaje y Simulación*, para resolver una ecuación diferencial en Matlab es conveniente usar dos programas. Ese paradigma es utilizado aquí. Intenta entender este código.

difRiccati	Ecuación diferencial
solveRiccati	Implementa la ecuación diferencial, el método numerico, y reacomoda la salida para tener la forma deseada en la que *t* va de 0 a t_f y no al contrario (recuerde se resuelve hacia atrás en el tiempo).

```
function [t,P,Pv]=solveRiccati(tf,A,B,Q,R,F)
%Halla P (dependiente del tiempo) a través de la resolución de la ecuación
%diferencial de Riccati. Los terminos de entrada son autoexplicativos.
%Pv es la versión vectorial de P
%P es la versión matricial de P
%t es el tiempo
%rango de tiempo
tspan=[tf 0];
%condición final
Fend=F(:);
```

```
[t,Pv]=ode15s(@(t,P) difRiccati(t,P,A,B,Q,R),tspan,Fend);

t=flip(t);
Pv=flip(Pv);
[n,~]=size(A);
P=reshape(Pv',n,n,[]);
```

```
function dP=difRiccati(t,P,A,B,Q,R)
%P entra como vector
P=reshape(P,size(A)); %convierte P de vector a matriz de tamaño nxn

dP=-P*A-A'*P-Q+P*B*inv(R)*B'*P;
%dP es una matriz
dP=dP(:); %convierte dP de matriz a vector de tamaño n^2 x 1
```

La evolución temporal de los valores de P se muestra en la Figura 5. Fíjate que todos se comportan de forma similar, pero en diferentes escalas y velocidades.

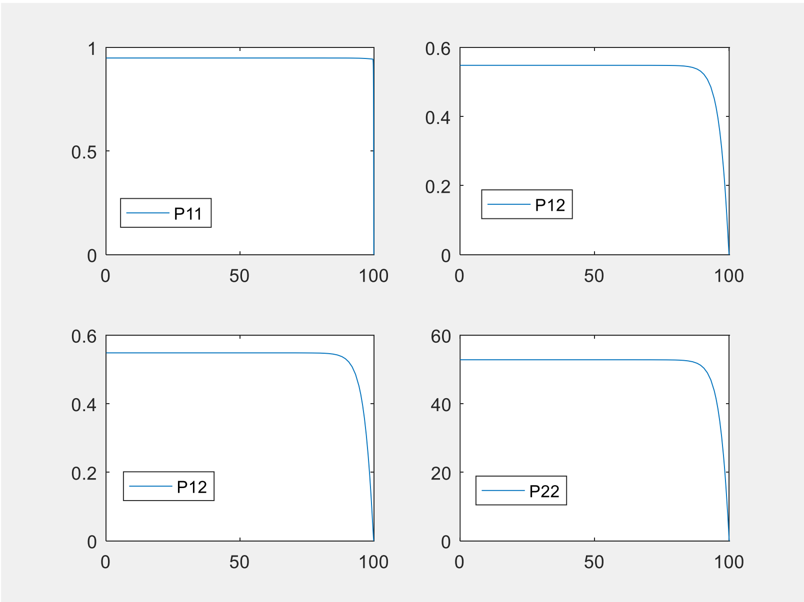


Figura 5. Evolución temporal de los valores de P (Riccati)

El resto de los pasos es mostrado en la simulación.
Este sistema de control es implementado a través de un diagrama de bloques en simulink

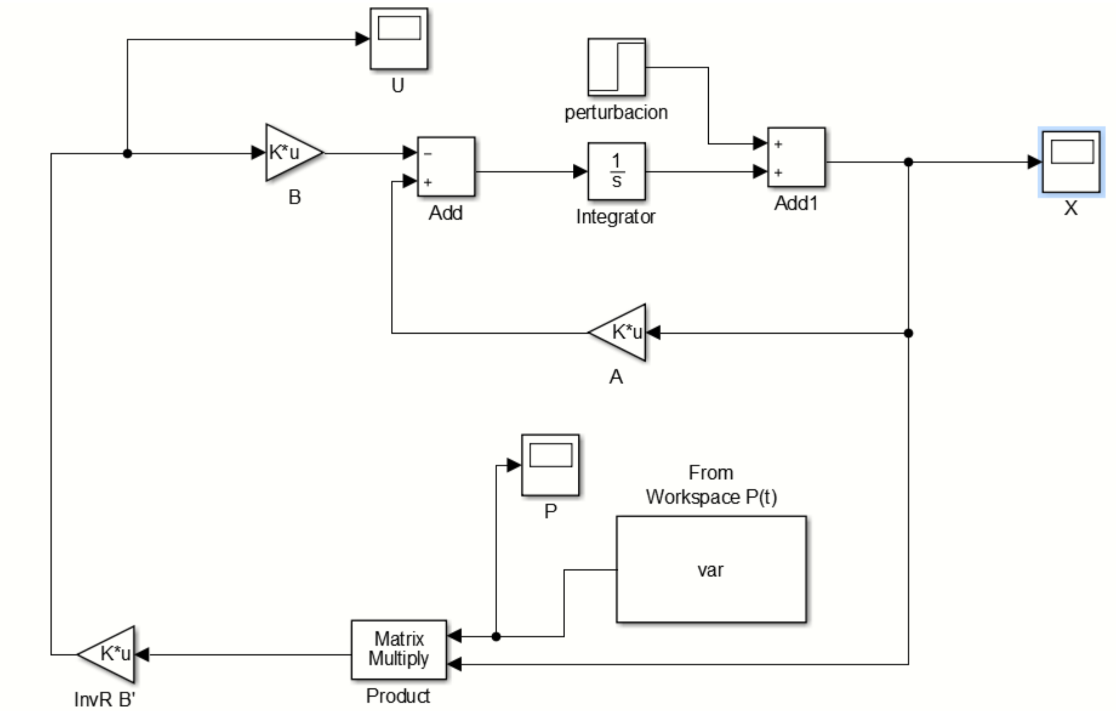


Figura 6. Diagrama de simulación del sistema de control

La matriz P debe suministrarse al diagrama de bloques desde el Workspace, a través del bloque llamado "from workspace". Como P tiene un valor matricial para cada unidad de tiempo utilizado, este bloque es de especial cuidado. En el video adjunto a este material hay más detalles al respecto. P también debe reescribirse, del resultado obtenido de usar solveRiccati, según

```
var.time=t;
var.signals.values=P;
var.signals.dimensions=[n n];
```

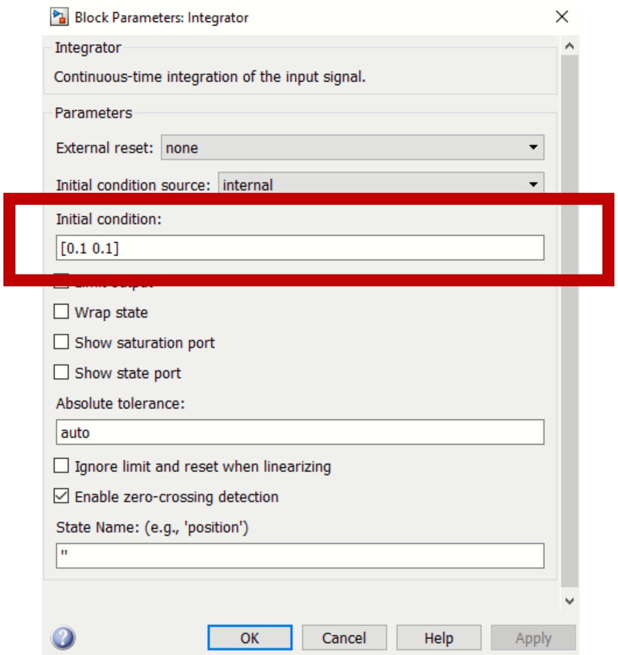
Esto se encarga de que Matlab entienda que hay un valor de tiempo para cada matriz en el arreglo de P, y que cada matriz es de $n \times n$, que en este caso es 2x2.

El bloque denominado "perturbación" posee salida 0. No está haciendo ninguna función en el diagrama, pero lo coloco para recordar que este sistema es un regulador.

El bloque integrador tiene condiciones iniciales diferentes de cero, ya que el diseño fue hecho considerando que

$$X(0) = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}$$

Pero este diseño no depende explícitamente de la condición inicial, luego puede usarse con otras condiciones iniciales:



Paso 2:
En el diagrama de bloques se ve que se calcula $K(t) = R^{-1}B^{T}P(t)$ multiplicando P por la matriz indicada invR B'.

Paso 3.
El resultado de la multiplicación anterior, al multiplicarse por X, da U

.
Paso 4.
Los valores de la trayectoria de los estados se obtienen por simulación dinámica. Estos se presentan en la Figura 7. La trayectoria de la entrada se presenta en la Figura 8

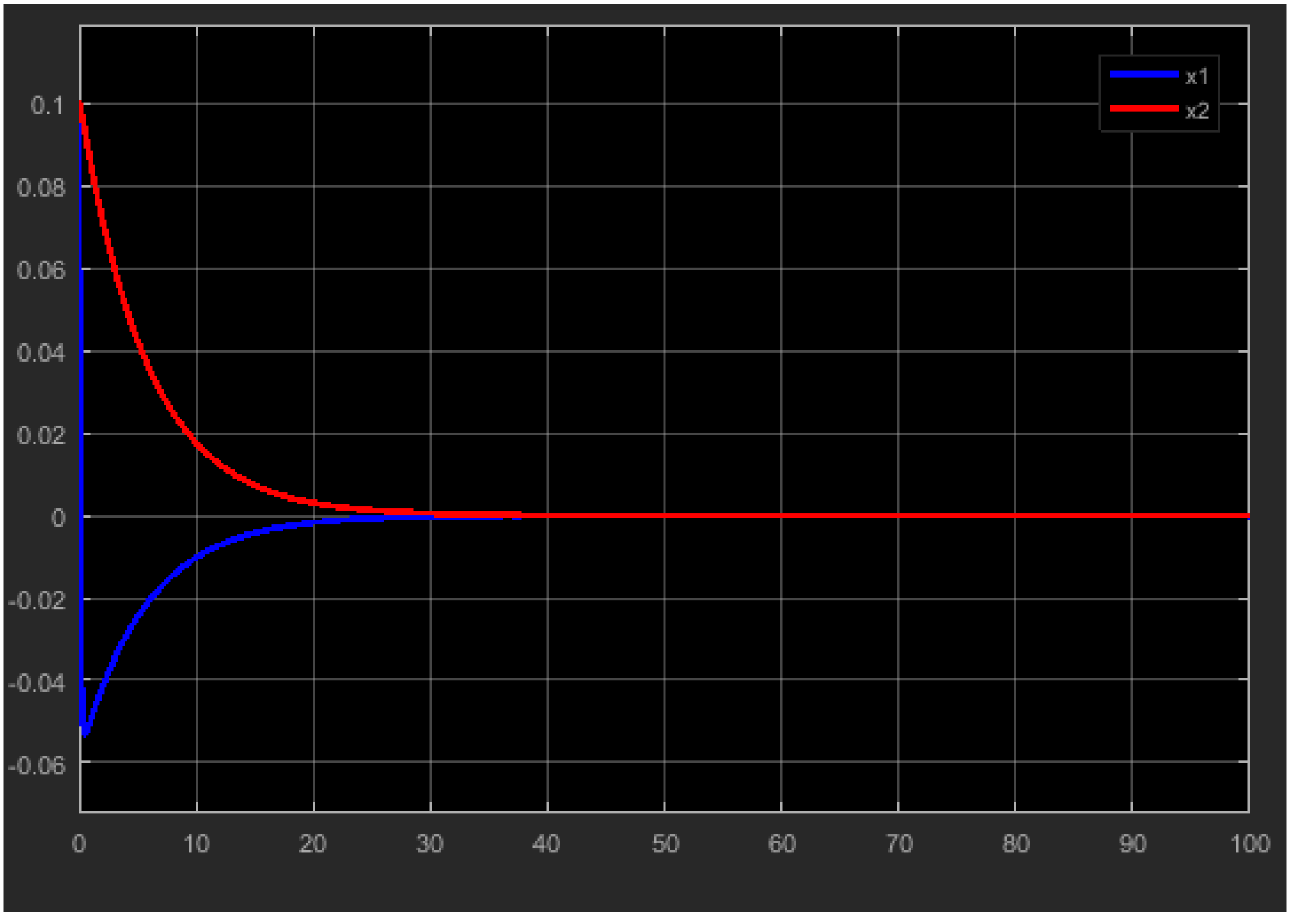


Figura 7. Trayectoria de los estados (respuesta del sistema)

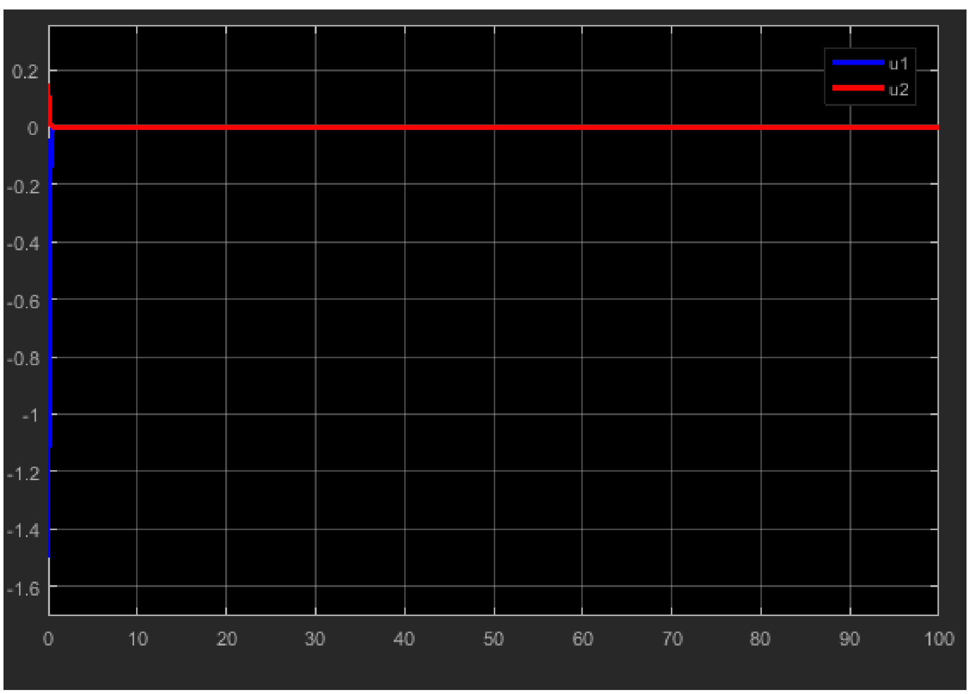


Figura 8. Trayectoria de la entrada

La entrada posee un cambio muy pequeño al inicio del problema, y luego regresa a cero rápidamente. Esto es uno de los efectos del controlador óptimo, intenta que las variables tengan la menor desviación posible de cero para evitar que el índice de desempeño sea mayor. Un acercamiento de la trayectoria de la entrada en el primer segundo se muestra en la Figura 9.

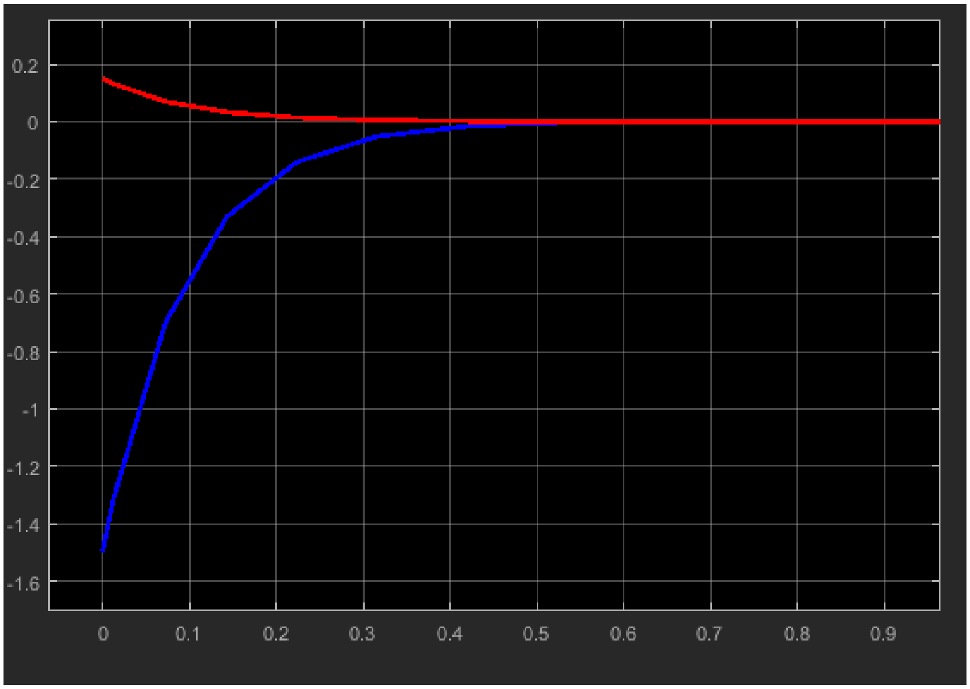


Figura 9. Trayectoria de la entrada (acercamiento)

Como las condiciones iniciales son conocidas y conociendo el valor de P en 0:

P0=P(:,:,1)

P0 =

0.9492 0.5483
0.5483 52.8138

entonces índice de desempeño es

x'*P0*x

ans =

0.5486

Ejemplo 2: Una corrida más sencilla en Matlab.

Para el sistema descrito por las matrices

$$A = \begin{bmatrix} 0 & 1 \\ -2 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix} \qquad R = \frac{1}{4} \qquad F = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix}$$

tf=5

Se muestra la implementación en Matlab

El código que es necesario ejecutar es

```
A=[0 1;-2 1];
B=[0;1];
Q=[2 3;3 5];
R=1/4;
F=[1 0.5;0.5 2];
tf=5;
[t,P,Pv]=solveRiccati(tf,A,B,Q,R,F);
var.time=t;
var.signals.values=P;
var.signals.dimensions=[2 2];
```

Puede verse en la Figura 10 la evolución temporal de los coeficientes de P. Solo aparecen tres líneas, pero son 4 coeficientes de P. ¿Por qué?.

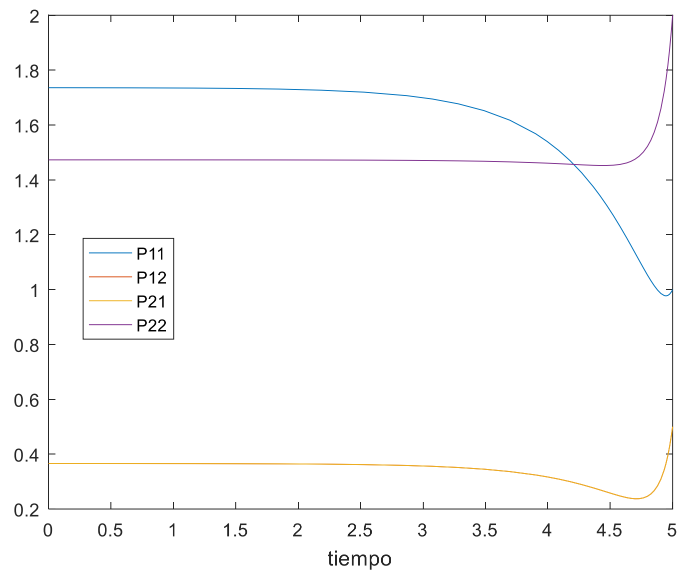


Figura 10. Evolución de los coeficientes de P en el tiempo

El diagrama de bloques es esencialmente el mismo utilizado en el ejemplo 1, pero ahora no se considera la condición inicial sino que se permite que la perturbación entre en t=0 con un valor de 2. ***El tiempo de simulación debe ser 5.*** Puede notarse que ahora la respuesta es mas lenta. Esto dependerá por supuesto del sistema y de su dinámica. Las otras gráficas pueden obtenerse para este proceso, tal como en el ejemplo anterior.

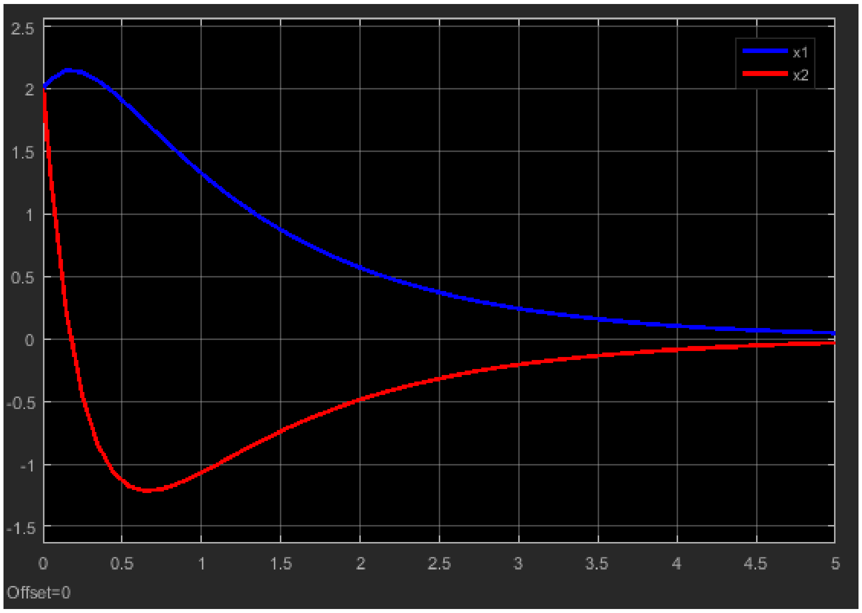


Figura 11. Evolución de los estados del ejemplo 2

II. Resumen

El procedimiento de diseño de un controlador de realimentación de vector de estados para un sistema que minimice un criterio cuadrático en un tiempo finito es de relativa facilidad de implementación, pero requiere la solución de una ecuación diferencial matricial hacia atrás en el tiempo. Esto implica que debe hacerse la implementación de esta solución antes de ejecutar la simulación y tenerse guardada en memoria. Si la aplicación es una aplicación industrial, es necesario poseer suficiente memoria disponible en el sistema de control para manejar esta información. Además, este sistema de control solo tiene significado en el rango de tiempo de trabajo, entre t_0 y t_f, y si se desea usar más allá del tiempo final, es necesario implementar un procedimiento que permita establecer un ciclo con el controlador.

Aunque en su descripción existe la definición de las condiciones iniciales, las condiciones iniciales no aparecen de forma explícita en ningún punto del diseño porque no forman parte de la solución de la ecuación de Riccati. Esto hace que el mismo sistema de control pueda usarse con otras perturbaciones o condiciones iniciales.

III. Referencias

Anderson, B. D., & Moore, J. B. (1971). *Linear Optimal Control.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

Anderson, B. D., & Moore, J. B. (1989). *Optimal Control - Linear Quadratic Methods.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

Athans, M. (1971). The Role and Use of the Stocastic Linear-Quadratic-Gaussian Problem in Control System Design. *Transactions on Automatic Control*, 529-552.

Bennett, S. (1996). A Brief History of Automatic Control. *IEEE Control Systems*, 17-25.

Bryson, A. E. (1996). Optimal control 1950 to 1985. *IEEE Control Systems*, 26-33.

Kalman, R. E. (1960). Contributions to the theory of Optimal Control. *Bolet{in de la Sociedad Matemática Mexicana*, 102-119.

Kalman, R. E. (1960). On the general theory of control systems. *Proceedings of the First IFAC Congress of Automatic Control* (págs. 481-492). Moscow: Butterworths.

Naidu, D. S. (2003). *Optimal Control Systems.* Boca Raton: CRC Press LLC.

Wiberg, D. M. (1971). *Theory and Problems of State Space and Linear Systems.* McGraw-Hill.