

Diseño y simulación de un controlador y regulador cuadrático lineal (LQR) con aplicación de filtros de Kalman (LQG), para complementar las materias de sistemas dinámicos y control en la Universidad Industrial de Santander.

Luis Felipe Carreño Sánchez, Leonardo Hermán Ramírez Plata

Trabajo de Grado para Optar al Título de Ingeniero Mecánico

Director

Jabid Eduardo Quiroga Méndez

PhD en Ingeniería Civil

Universidad Industrial de Santander

Facultad de Ingenierías Físico-Mecánicas

Escuela de Ingeniería Mecánica

Bucaramanga

2023

Tabla de contenido

	Pág.
Introducción-----	9
1. Objetivos-----	11
1.1 <i>Objetivo general</i> -----	11
1.2 <i>Objetivos específicos</i> -----	11
2. Marco teórico-----	12
2.1 <i>Ingeniería de control</i> -----	12
2.1.1 Teoría de control lineal-----	12
2.1.2 Control retroalimentado en lazo cerrado-----	14
2.2 <i>Sistemas lineales invariantes en el tiempo</i> -----	17
2.2.1 Linealización de sistemas no lineales-----	18
2.2.2 Respuesta natural del sistema-----	19
2.2.3 Respuesta forzada del sistema-----	20
2.2.4 Sistema de tiempo discreto-----	21
2.3 <i>Dinámica de sistemas</i> -----	23
2.3.1 Sistemas mecánicos con dos o más grados de libertad-----	23
2.3.2 Ecuaciones de movimiento de Lagrange-----	24
2.3.3 Coordenadas generalizadas-----	24
2.3.4 Lagrangiano-----	24
2.3.5 Principio de Hamilton-----	25
2.3.6 Ecuaciones de movimiento de Lagrange para sistemas conservativos-----	25
2.4 <i>Ingeniería de control moderna</i> -----	27
2.4.1 Controlabilidad-----	27
2.4.2 Observabilidad-----	30
2.4.3 Control óptimo: LQR (Linear Quadratic Regulator)-----	31
2.4.4 Estimador óptimo: LQR (Filtro de Kalman)-----	33
2.4.5 Control óptimo basado en sensores: LQG (Linear Quadratic Gaussian)-----	35
3. Metodología-----	37
3.1 <i>Modelo Matemático del Péndulo Invertido Lineal</i> -----	37
3.1.1 DCL del Péndulo Invertido Lineal-----	38
3.1.2 Ecuaciones de Movimiento-----	39
3.2 <i>Espacios de estados</i> -----	47
3.2.1 Variables de Estado-----	47
3.2.2 Linealización de las Ecuaciones de Movimiento-----	48
3.2.3 Ecuación de Estado-----	49

3.2.4 Ecuación de Salida -----	50
3.3 <i>Sistema del Péndulo Invertido en Ambiente Matlab</i> -----	50
3.3.1 Comportamiento no Lineal del Péndulo Invertido -----	51
3.3.2 Propiedades del Sistema de Control -----	52
3.4 <i>Diseño del Controlador</i> -----	54
3.4.1 Diseño del Controlador LQR -----	54
3.5 <i>Filtro de Kalman</i> -----	60
3.5.1 Observabilidad-----	61
3.5.2 Diseño del Observador-----	62
3.6 <i>Sistema del péndulo invertido en ambiente Simulink</i> -----	63
3.6.1 Controlador LQR -----	66
3.7 <i>Controlador LQG</i> -----	74
3.8 <i>Elaboración de Videos explicativos</i> -----	78
3.8.1 Video 1 -----	79
3.8.2 Video 2 -----	79
3.8.4 Video 3 -----	79
3.8.4 Video 4 -----	80
4.8.5 Video 5 -----	80
4.8.6 Video 6 -----	80
4. Análisis de resultados -----	80
4.1 <i>Matlab</i> -----	80
4.1.1 Resultados LQR -----	80
4.1.2 Comparación de las técnicas de control utilizadas -----	87
4.2 <i>Simulink</i> -----	92
4.2.1 Resultados LQR -----	92
5.2.2 Resultados LQG -----	94
5. Conclusiones-----	96
Referencias -----	97

Lista de Figuras

	Pág.
Figura 1 Diagrama de bloques de un sistema en lazo abierto.....	13
Figura 2 Diagrama de bloques de un sistema en lazo cerrado.....	14
Figura 3 Diagrama de bloques de un sistema retroalimentado en lazo cerrado.....	15
Figura 4 Control crucero de circuito abierto frente a circuito cerrado	17
Figura 5 Convolución para un sistema de entrada única, salida única (SISO)	21
Figura 6 Ubicación de polos en tiempo continuo frente a tiempo discreto	23
Figura 7 Diagrama de bloques de un controlador óptimo (LQR)	32
Figura 8 Diagrama de bloques del funcionamiento del filtro de Kalman (LQR)	35
Figura 9 Diagrama de bloques de un controlador (LQR) con aplicación de filtro de Kalman (LQG)	36
.....	
Figura 10 DCL del Péndulo Invertido Lineal	38
Figura 11 Diagrama de entrada y salidas del sistema	47
Figura 12 Código con asignación de valores a variables y matrices de estado (A) y entrada (B)	51
.....	
Figura 13 Código de la función Pendcart.....	52
Figura 14 Comando para determinar la estabilidad del sistema	52
Figura 15 Comando para determinar la controlabilidad del sistema	53
Figura 16 Código para determinar las matrices de ponderación Q y R	54
Figura 17 Comando LQR.....	55
Figura 18 Código de referencias, condiciones iniciales y vector de tiempo discreto.....	56
Figura 19 Comando para implementar la ley de control.....	56
Figura 20 Código para la comparación gráfica de la respuesta del controlador LQR	57
Figura 21 Gráfica de la comparación de respuestas del controlador LQR con la posición de otros	58
polos estables	
Figura 22 Código para la comparación grafica de la función de costo del controlador LQR	59
Figura 23 Gráfica de la comparación de función de costo del controlador LQR con la posición de	60
otros polos estables.....	
Figura 24 Código para determinar matrices de observabilidad del sistema.....	62
Figura 25 Código para el diseño del observador	62
Figura 26 Bloque Funcion S, Cart_Pendulum programable en lenguaje C++	63
Figura 27 Programación del bloque Cart_Pedulum en lenguaje C++.....	64
Figura 28 Bloque Funcion S, cartpen_sim programable en C++	64
Figura 29 Programación del bloque cartpend_sim en lenguaje C++.....	65
Figura 30 Diagrama del controlador LQR en Simulink	66
Figura 31 Parte izquierda del diagrama del controlador LQR en Simulink	67
Figura 32 Parte derecha del diagrama del controlador LQR en Simulink	67
Figura 33 Demultiplexor, con sus dos sumadores en línea correspondiente a X y sus bloques step	68
.....	
Figura 34 Programación del bloque Step	69
Figura 35 Programación del bloque del ruido	70
Figura 36 Programación del bloque del bias	70

Figura 37 Programación del bloque de “Gain”	71
Figura 38 Configuración del bloque VR signal Expander	72
Figura 39 Programación del bloque VR Sink	73
Figura 40 Animación del péndulo invertido lineal	74
Figura 41 Diagrama del controlador LQG en Simulink	75
Figura 42 Código inicializador para el modelo de Simulink del controlador LQG	76
Figura 43 Programación del bloque de “Gain”	77
Figura 44 Programación del bloque “State Sapce”	77
Figura 45 Código para solucionar el espacio de derivadas	81
Figura 46 Código para graficar el sistema dinámico	81
Figura 47 Gráfica del sistema dinámico, estados vs tiempo	82
Figura 48 Código para generar grafica de la posición del carro	82
Figura 49 Gráfica de la posición del carro vs tiempo	83
Figura 50 Código para generar gráfica de la velocidad del carro	84
Figura 51 Gráfica de la velocidad del carro vs tiempo	84
Figura 52 Código para generar gráfica de la posición del péndulo	85
Figura 53 Gráfica de la posición angular del péndulo vs tiempo	85
Figura 54 Código para generar gráfica de la velocidad angular del péndulo	86
Figura 55 Gráfica de la posición de la velocidad angular del péndulo vs tiempo	86
Figura 56 Código para generar gráfica comparativa de la posición del carro	88
Figura 57 Gráfica comparativa de la posición del carro entre LQR estimado completamente, LQR con medición de ruido y LQG (LQG estimado con filtro de Kalman)	89
Figura 58 Código y gráfica generada para comparar la posición angular entre el LQR y LQG	90
Figura 59 Código y gráfica generada para comparar las cuatro variables de estado del sistema entre LQR y LQG	91
Figura 60 Gráfica scope LQR	92
Figura 61 Gráfica scope de la posición angular	93
Figura 62 Gráfica scope de la estimación de ruido en el controlador LQR	94
Figura 63 Gráfica scope LQG	94
Figura 64 Gráfica scope LQG posición angular	95

Lista de Apéndices

Apéndice A. códigos de Matlab y Simulink

Apéndice B. Scripts en lenguaje C++ de los bloques Función S

Apéndice C. Videos Explicativos

Apéndice D. Instructivo para ejecutar correctamente los códigos de Matlab y Simulink

Los apéndices están adjuntos y pueden ser visualizados en la base de datos de la Biblioteca UIS.

Resumen

Título: Diseño y simulación de un controlador y regulador cuadrático lineal (LQR) con aplicación de filtros de Kalman (LQG), para complementar las materias de sistemas dinámicos y control en la Universidad Industrial de Santander.

Autores: Luis Felipe Carreño Sánchez, Leonardo Hermán Ramírez Plata.

Palabras Clave: Péndulo invertido, control óptimo, control moderno, regulador cuadrático lineal (LQR), control lineal cuadrático gaussiano (LQG), filtro de Kalman, espacio de estados.

Descripción:

La orientación que ha tomado la industria hoy en día va direccionada hacia el aumento de los niveles de automatización y digitalización en los procesos industriales. Por lo que resulta de gran importancia, dentro de este contexto, el estudio de disciplinas que vayan acorde con el desarrollo industrial actual, un ejemplo de esto es la ingeniería de control, una disciplina que aplica la teoría de control para obtener sistemas con comportamientos deseados.

Por consiguiente, resulta relevante manejar los principios relacionados a esta rama de la ingeniería, para esto, se plantea el péndulo invertido como un sistema de control correspondiente a un sistema dinámico no lineal e inestable, el cual constituye a uno de los ejemplos académicos más destacados en la teoría de control, debido a que permite la aplicación de diferentes técnicas de regulación y seguimiento de una o varias variables a controlar, sirviendo como base para el estudio de dichas técnicas en sistemas más complejos.

En este proyecto se modelará y simulará en señales de tiempo discreto en los programas de Matlab y Simulink, el comportamiento físico del péndulo invertido como un sistema de control moderno en espacio de estados, cuyas técnicas de control a aplicar para lograr la estabilización del péndulo corresponden al control óptimo del sistema. Estas técnicas son LQR (regulador cuadrático lineal) y LQG (control lineal cuadrático gaussiano) la cual implementa el filtro de Kalman a un controlador LQR, una vez realizados los controladores se comparan entre sí.

Estos códigos de simulación, la comparación entre las técnicas de control y una total de 6 videos acerca del proyecto como material audiovisual explicativo para los estudiantes, sirven como herramientas pedagógicas a la hora de estudiar los principios básicos de control moderno y óptimo en materias de la escuela de ingeniería mecánica de la Universidad Industrial de Santander, tales como sistemas dinámicos e ingeniería de control.

Abstract

Title: Design and simulation of linear-quadratic regulator (LQR) with the application of Kalman filter (LQG), to complement the subjects of dynamic systems and control at the Universidad Industrial de Santander.

Authors: Luis Felipe Carreño Sánchez, Leonardo Hermán Ramírez Plata.

Key words: Inverted pendulum, optimal control, modern control, linear quadratic regulator (LQR), linear quadratic gaussian (LQG), Kalman filter, state-space representation.

Description:

The orientation that the industry has taken today is directed towards increasing the levels of automatization and digitalization in industrial processes. Therefore, it is of great importance, within this context, the study of disciplines that are consistent with current industrial development, an example of this is control engineering, a discipline that applies control theory to obtain systems with desired behaviors.

Therefore, it is relevant to handle the principles related to this branch of engineering, for This matter, the inverted pendulum is proposed as a control system corresponding to a nonlinear and unstable dynamic system, which constitutes one of the most prominent academic examples in control theory, because it allows the application of different regulation and monitoring techniques of one or several variables to be controlled, serving as a basis for the study of said techniques in more complex systems.

In this project, the physical behavior of the inverted pendulum will be modeled and simulated in discrete-time signals, in Matlab and Simulink programs, as a modern control system in state space representation, whose control techniques to be applied to achieve the stabilization of the pendulum, correspond to the optimal control of the system, these techniques are LQR (linear-quadratic regulator) and LQG (linear-quadratic gaussian) which implements the Kalman filter to an LQR controller, once the controllers are made, they are compared with each other.

These simulation codes, the comparison between the control techniques and a total of 6 videos about the project as an explanatory audiovisual material for the students, serve as pedagogical tools, which allow the consolidation of the basic principles of optimal control and modern control theory in subjects from the School of Mechanical Engineering of the Industrial University of Santander, such as dynamic systems and control engineering.

Introducción

En la actualidad las ingenierías deberían presentar un enfoque de la denominada cuarta revolución industrial, que consiste en combinar las tecnologías operativas con las tecnologías de la información, para dar lugar a sistemas y máquinas automatizadas e interconectadas entre sí para lograr una optimización de recursos y procesos mediante la recolección y aprovechamiento de datos. Debido a lo anterior, se hace indispensable empezar a fomentar y fortalecer los conocimientos en las áreas de estudio que se encuentran a la vanguardia de la industria, tales como, la robótica, la programación, la simulación, la ingeniería de control, entre otras. Esto con el fin de formar ingenieros con las competencias necesarias para adaptarse a las necesidades actuales de las empresas e industrias.

El cambio completo hacia la industria 4.0 puede no realizarse en décadas. Sin embargo, para seguir siendo competitivas, las empresas deberán acelerar sus esfuerzos hacia la industria 4.0 (Sjøbakk, 2018). Colombia es un país inmerso en el proceso de transformación acorde a la revolución 4.0, muestra de esto, el 19 de abril del año 2019 se inauguró el Centro para la Cuarta Revolución industrial en la ciudad de Medellín, con el objetivo de desarrollar marcos legales y políticas públicas para el desarrollo industrial del país.

Teniendo en cuenta este contexto, el programa de ingeniería mecánica de la universidad Industrial de Santander maneja un plan de estudios en donde están incluidas materias como: “sistemas dinámicos” e “ingeniería de control”, las cuales se enfocan en conceptos relacionados a la teoría de control en donde se estudia el comportamiento de los sistemas dinámicos. Esta área de estudio a permitido dar forma al entorno tecnológico e industrial moderno, ya que hace parte de los procesos importantes actuales de fabricación y de las operaciones industriales en donde se necesita controlar diferentes variables tales como la presión, la temperatura, etc.

El presente trabajo de grado consiste en el estudio de un péndulo invertido como sistema de control, donde se planteó la representación de la dinámica del sistema en espacio de estados y se diseñó en Matlab y Simulink de un controlador cuadrático lineal (LQR) y un controlador cuadrático lineal Gaussiano (LQG), implementando el filtro de Kalman. Posteriormente, se creó un video explicativo de este sistema mecánico clásico y las técnicas de control empleadas para su estabilización; con el objetivo de generar un material didáctico para los estudiantes de ingeniería mecánica de la Universidad Industrial de Santander, que complemente los conocimientos acerca de la teoría de control moderno y control óptimo.

1. Objetivos

1.1 Objetivo general

Contribuir con la misión de la Universidad Industrial de Santander en la formación integral de ingenieros mecánicos de alta calidad técnica, científica e investigativa, al simular mediante el software Matlab y Simulink el modelo en tiempo discreto de control LQR con aplicación de filtros de Kalman para estabilizar un péndulo invertido, con el fin de que sirva como herramienta didáctica y pedagógica para los estudiantes de ingeniería de control y sistemas dinámicos de la escuela de ingeniería mecánica.

1.2 Objetivos específicos

- Obtener el modelo matemático lineal del péndulo invertido y simular su operación en ambiente Matlab.
- Diseñar, simular e implementar el modelo en tiempo discreto del control basado en LQR con uso de un filtro de Kalman para estimación de un estado del modelo, el cual se usará para la estabilización del péndulo invertido lineal.
- Analizar y comparar los resultados entre las dos técnicas de control discretas utilizadas (LQR vs LQR + filtro de Kalman).
- Desarrollar un material audiovisual de enseñanza pedagógico para los fundamentos de control óptimo.

2. Marco teórico

La ingeniería tuvo avances significativos antes, durante y después de la segunda guerra mundial, donde a muchos ingenieros y científicos tanto de Rusia como de Estados Unidos que venían investigando el espacio y las leyes del universo, tuvieron la necesidad de explorar el sistema solar, empezando por la luna, donde el principal problema a solucionar era el de llevar una nave espacial desde algún punto en la tierra a algún otro punto de la superficie lunar, haciéndolo en un tiempo mínimo y consumiendo la menor cantidad de combustible posible y de esta manera encontrar la trayectoria más óptima. Lo anterior no era posible de resolver aplicando las técnicas de optimización tradicionales, dejando así atrás a la teoría de control clásica para darle paso a la teoría de control moderna o también conocida como control óptimo. La teoría de control óptimo que comenzó su desarrollo en los años cincuenta por un equipo de matemáticos dirigidos por el ruso Lev Pontryagin con su principio del mínimo (máximo) en la antigua Unión Soviética y posteriormente complementada con los trabajos de Rudolf Kalman, en filtrado y análisis algebraico de problemas de control y Richard Bellman en programación dinámica en los Estados Unidos.

2.1 Ingeniería de control

2.1.1 Teoría de control lineal

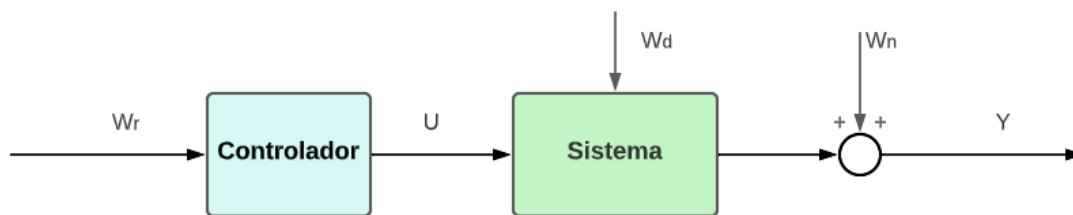
Se refiere al campo de estudio que busca manipular sistemas dinámicos y se encuentra fuertemente relacionado con la ciencia de datos puesto que para determinar el estado del sistema se emplean mediciones con sensores, las cuales se traducen en datos que deben ser procesados por el controlador para determinar la acción de los actuadores empleados por el diseñador y así lograr el comportamiento deseado del sistema.

Existen distintos métodos de control que pueden ser clasificados según sus características, si para controlar el sistema no se requiere suministrar energía se conoce como control pasivo, en caso contrario es llamado control activo. Cuando se emplean sensores que determinan el estado actual del sistema y actúan en consecuencia, se conoce como control retroalimentado o de lazo cerrado, en caso contrario donde se espera un comportamiento del sistema y el controlador asume que conoce el estado del sistema sin medirlo, se conoce como control prealimentado o de lazo abierto.

Los casos más estudiados en teoría de control se refieren a control activo del tipo lazo abierto y lazo cerrado, cuya representación en diagrama de bloques se muestra a continuación respectivamente (Figura 1 y Figura 2).

Figura 1.

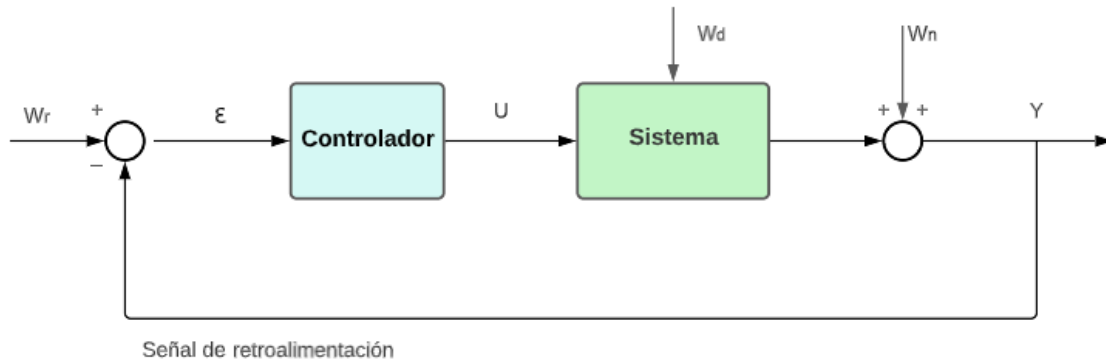
Diagrama de bloques de un sistema en lazo abierto



Nota. (Autores)

Figura 2.

Diagrama de bloques de un sistema en lazo cerrado



Nota. (Autores)

2.1.2 Control retroalimentado en lazo cerrado

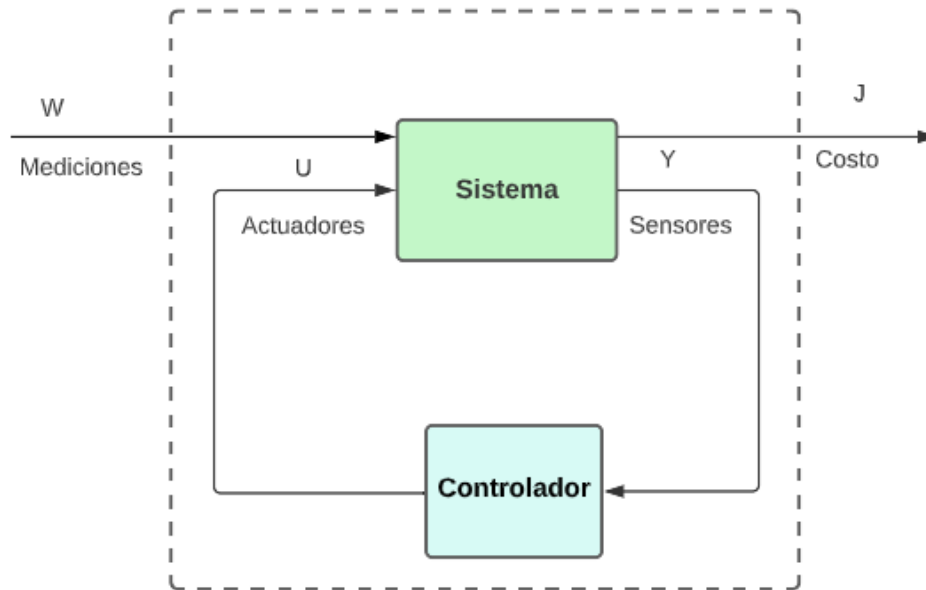
En este tipo de control se toman medidas del estado que son retroalimentadas al controlador constantemente, de esta manera se elige la señal de control adecuada para llegar a la referencia establecida, evitando de esta manera que perturbaciones lleven el sistema a un punto no deseado. Como se observa en la Ecuación 1, en las entradas del sistema se identifican tres componentes, perturbaciones provenientes del ambiente, perturbaciones originadas en los sensores y una trayectoria de referencia para el diseño.

$$w = [w_d^T w_n^T w_r^T]^T \quad (1)$$

- w : Entradas del sistema
- w_d : Perturbaciones provenientes del ambiente
- w_n : Perturbaciones provenientes de los sensores
- w_r : Referencia

Figura 3.

Diagrama de bloques de un sistema retroalimentado en lazo cerrado



Nota. (Autores)

Matemáticamente se suele describir el sistema mediante dos funciones, el espacio de derivadas de estado (Ecuación 2), y las mediciones (Ecuación 3-4-5), las cuales dependen de las perturbaciones provenientes de ambiente y de las originadas en los sensores respectivamente. El objetivo es obtener una ley de control que depende de las mediciones y la trayectoria de referencia, que minimice la función de costo asociada al sistema, la cual depende del estado del sistema, la señal de control y la trayectoria de referencia. En la Figura 3 se puede observar la representación en bloques.

$$\frac{d}{dt}x = f(x, u, w_d) \quad (2)$$

$$y = g(x, u, w_n) \quad (3)$$

$$u = k(y, w_r) \quad (4)$$

$$J \triangleq J(x, u, w_r) \quad (5)$$

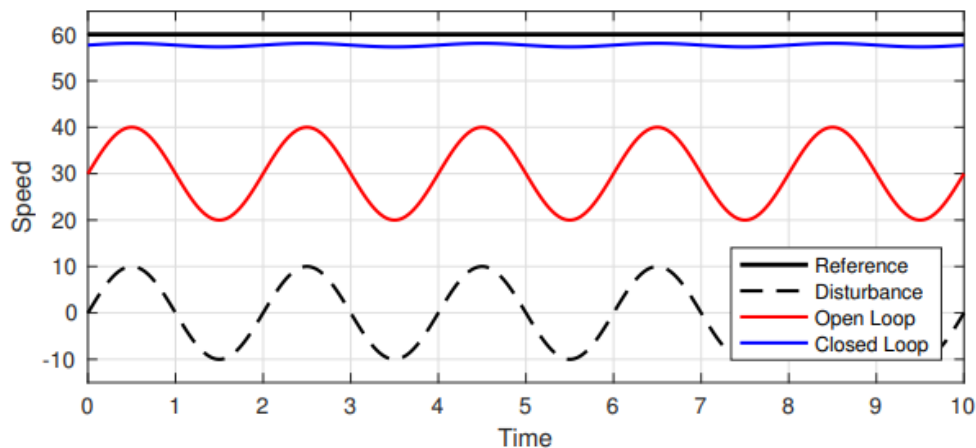
- x : Estado del sistema
- u : Señal de control
- y : Respuesta del sistema
- J : Función de costo asociada al sistema

La función de costo para un sistema dinámico de control se puede entender como una cuantificación de los intereses de control en el diseño, es decir la importancia que se le dan a las propiedades del sistema, tales como velocidad de respuesta y márgenes de tolerancia. El control moderno está basado en técnicas de optimización donde el controlador presenta la robustez suficiente para soportar afectaciones externas e incluso incertidumbres en el comportamiento del sistema dinámico a controlar, robustez asociada al control retroalimentado.

A continuación, en la Figura 4 se muestra un ejemplo de la respuesta para un sistema de control de velocidad en los casos donde se aplica control prealimentado y retroalimentado cuando se presentan perturbaciones externas que afectan su comportamiento.

Figura 4.

Control de crucero de circuito abierto frente a circuito cerrado.



Nota. Tomado de Data driven science & engineering: Machine learning, dynamical systems and control (p.329), por S. L. Bruton & J. L. Kutz, University of Washington.

2.2 Sistemas lineales invariantes en el tiempo

La teoría de control se encuentra ampliamente desarrollada para sistemas lineales, debido a que la mayoría de los sistemas reales son no lineales, se suele emplear la linealización para obtener una representación lineal que describa adecuadamente el sistema en un rango más amplio, que aquel que se le da como tolerancia máxima al controlador, de aquí se obtiene una primera limitante para el rango operacional del controlador (Ecuación 6 y Ecuación 7).

$$\frac{d}{dt}x = f(x, u) \approx Ax + Bu \quad (6)$$

$$y = g(x, u, w_n) \approx Cx + Du \quad (7)$$

2.2.1 Linealización de sistemas no lineales

Las series de Taylor son una herramienta ampliamente usada y muy poderosa para la linealización puesto que permite describir una función no lineal en términos de su valor presente y sus derivadas, para ello se establece un punto de operación (\bar{x}, \bar{u}) . Asumiendo que las variaciones son pequeñas respecto al punto de operación, es posible linealizar el sistema y despreciar las derivadas de mayor orden de la expansión en series de Taylor, quedando únicamente las de primer orden, por tanto se puede describir un sistema no lineal alrededor de un punto de operación en términos de su estado actual (Ecuación 8) y su primera derivada (Ecuación 9).

$$f(x, u) \approx f(\bar{x} + \Delta x, \bar{u} + \Delta u) \approx f(\bar{x}, \bar{u}) + \left. \frac{df}{dx} \right|_{(\bar{x}, \bar{u})} \Delta x + \left. \frac{df}{du} \right|_{(\bar{x}, \bar{u})} \Delta u \quad (8)$$

$$g(x, u) \approx g(\bar{x} + \Delta x, \bar{u} + \Delta u) \approx g(\bar{x}, \bar{u}) + \left. \frac{dg}{dx} \right|_{(\bar{x}, \bar{u})} \Delta x + \left. \frac{dg}{du} \right|_{(\bar{x}, \bar{u})} \Delta u \quad (9)$$

Estableciendo el punto de operación como el origen $f(\bar{x}, \bar{u}) = 0$, podemos retirar el símbolo Δ , de tal manera que el sistema no lineal inicial, puede ser llevado a una expresión matricial compuesta del espacio de estados, el espacio de derivadas de estado y el vector de respuesta del sistema de la forma esperada (Ecuación 10 y Ecuación 11)

$$\frac{d}{dt}x \approx Ax + Bu \quad (10)$$

$$y \approx Cx + Du \quad (11)$$

2.2.2 Respuesta natural del sistema

Cuando un sistema lineal carece de control se encuentra completamente descrito únicamente por la matriz A , cuya solución está dada por las condiciones iniciales y la exponencial que puede definirse como una serie de potencias. Es posible disminuir la complejidad de cálculo, realizando la descomposición de la matriz en términos de la matriz diagonal de autovalores y la matriz de auto vectores, de tal manera que el cálculo de estados futuros del sistema dependa de las condiciones iniciales, la exponencial de la matriz diagonal de autovalores, la matriz de auto vectores y su matriz inversa. Este proceso se evidencia en las siguientes ecuaciones:

$$x(t) = e^{At}x(0) \quad (12)$$

$$e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \dots \quad (13)$$

$$AT = T\Lambda \quad (14)$$

$$A = T\Lambda T^{-1} \quad (15)$$

$$e^{\Lambda t} = \begin{bmatrix} e^{\lambda_1 t} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & e^{\lambda_n t} \end{bmatrix} \quad (16)$$

$$e^{At} = e^{T\Lambda T^{-1}t} = Te^{\Lambda t}T^{-1} \quad (17)$$

$$x(t) = Te^{\Lambda t}T^{-1}x(0) \quad (18)$$

- $x(t)$: Estado futuro del sistema

- $x(0)$: Condiciones iniciales del sistema
- I : Matriz identidad
- T : Matriz de auto vectores
- Λ : Matriz de auto valores
- λ_n : Auto valor n.

Descomponer la matriz A es particularmente útil para conocer las propiedades del sistema, puesto que el único término que varía en el tiempo es la exponencial de la matriz de autovalores, sabiendo que los autovalores son números complejos que representan la ubicación de los polos del sistema, se puede determinar la estabilidad del sistema según su parte real, recordando que el criterio de estabilidad para un sistema es que todos los polos se encuentren ubicados en el semiplano izquierdo del plano complejo.

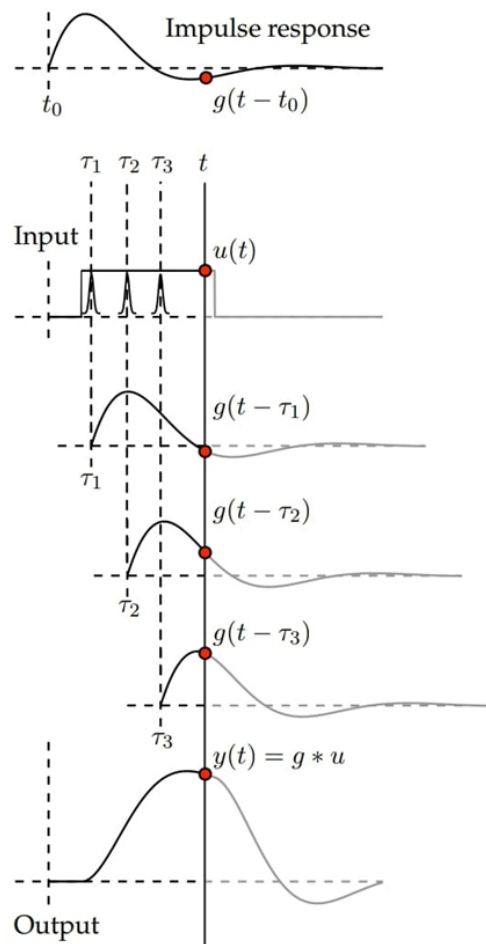
2.2.3 Respuesta forzada del sistema

Cuando se afecta el sistema mediante una señal de control, su estado futuro está determinado por la convolución de la señal de control con el kernel del sistema ($e^{At}B$), sabiendo que la respuesta está dada por $y = Cx$, se puede obtener entonces la respuesta del sistema mediante la expresión $y(t) = Ce^{At}B * u(t)$.

A continuación, en la Figura 5 se ilustra la convolución mediante la respuesta impulso de un sistema de única entrada y salida (SISO por sus siglas en inglés) afectado por 3 señales tipo impulso unitario en diferentes momentos τ .

Figura 5.

Convolución para un sistema de entrada única, salida única (SISO).



Nota. Tomado de Data driven science & engineering: Machine learning, dynamical systems and control (p.333), por S. L. Bruton & J. L. Kutz, University of Washington.

2.2.4 Sistema de tiempo discreto

Los sistemas de control modernos se caracterizan por emplear sensores, dichos sensores emplean señales digitales, las cuales son tomadas en tiempo discreto, por tanto cuentan con un tiempo de muestreo. En otras palabras, los sensores funcionan midiendo una variable cada cierto tiempo, se utilizan tiempos de muestreo pequeños para que la señal discreta sea lo más cercana posible a la señal continua.

Puesto que los sistemas de control funcionan en tiempo discreto y los modelos matemáticos con los que se describen los fenómenos físicos que se pretenden controlar lo hacen en tiempo continuo, es necesario realizar la discretización del sistema, lo cual resulta en una ecuación de diferencias, dicha ecuación modela el siguiente estado del sistema a partir de la medición actual y mediciones anteriores, resultando en un espacio de estados discretos, como se observa en la Ecuación 19 y 20.

$$x_{k+1} = A_d x_k + B_d u_k \quad (19)$$

$$y_k = C_d x_k + D_d u_k \quad (20)$$

Donde

$$x_k = x(k\Delta t)$$

$$A_d = e^{A\Delta t}$$

$$B_d = \int_0^{\Delta t} e^{A\tau} B d\tau$$

$$C_d = C$$

$$D_d = D$$

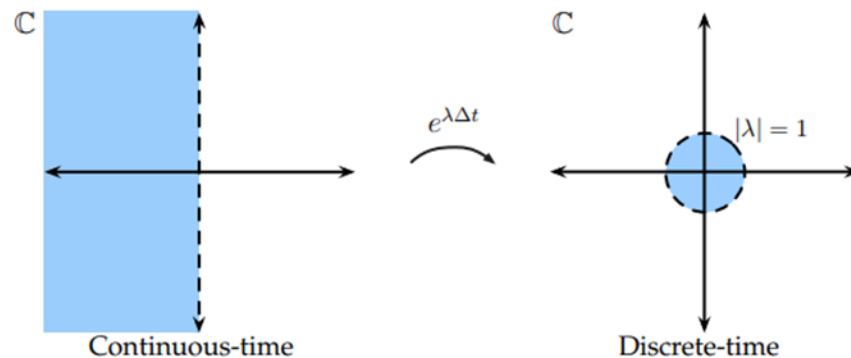
- x_k Estado k del sistema discretizado.
- A_d : Matriz A del sistema discretizado.
- B_d : Matriz B del sistema discretizado
- C_d : Matriz C del sistema discretizado

- D_d : Matriz D del sistema discretizado

En el caso de los sistemas de tiempo discreto se maneja un criterio de estabilidad similar al de tiempo continuo, pues está basado en los autovalores de la matriz A_d , con la diferencia de que ahora el criterio corresponde a que la ubicación de los polos en el plano complejo no debe superar la frontera de un círculo unitario ubicado en el origen $|\lambda| \leq 1$, esto se observa en la Figura 6.

Figura 6.

Ubicación de polos en tiempo continuo frente a tiempo discreto.



Nota. Tomado de Data driven science & engineering: Machine learning, dynamical systems and control (p.334), por S. L. Bruton & J. L. Kutz, University of Washington.

2.3 Dinámica de sistemas

2.3.1 Sistemas mecánicos con dos o más grados de libertad

Los sistemas físicos reales pueden presentar movimientos traslaciones y rotacionales con restricciones, para estudiarlos se emplea el concepto de grados de libertad, los cuales dependen de

la cantidad de ecuaciones de movimiento disponibles y las ecuaciones de restricción asociadas al comportamiento del sistema.

$$\# \text{ grados de libertad} = \# \text{ ecuaciones de movimiento} - \# \text{ ecuaciones de restricción}$$

Además, es importante resaltar que los grados de libertad están directamente relacionados con la cantidad de frecuencias naturales que pueden presentarse en el sistema, siendo que el número de grados de libertad es igual al número de frecuencias naturales posibles.

2.3.2 Ecuaciones de movimiento de Lagrange

La naturaleza compleja de algunos sistemas puede resultar en que la aplicación de las leyes de Newton sea poco conveniente para describir su comportamiento, las ecuaciones de movimiento de Lagrange permiten tener un acercamiento basado en consideraciones de energía que puede resultar conveniente en muchos casos, para plantearlas es necesario establecer el principio de Hamilton mediante las coordenadas generalizadas y el Lagrangiano.

2.3.3 Coordenadas generalizadas

Su número es igual al de grados de libertad y se refieren al número de coordenadas necesarias para describir completamente el estado de un sistema, el cuál para n grados de libertad debe ser descrito mediante n coordenadas (q_1, q_2, \dots, q_n) , este tipo de coordenadas no está directamente relacionado al sistema cartesiano o polar, pero puede basarse en estos para construir las coordenadas independientes que componen las coordenadas generalizadas.

2.3.4 Lagrangiano

Función escalar dependiente de las coordenadas generalizadas, sus derivadas y el tiempo, su magnitud está determinada por $L = T - V$, siendo T la energía cinética y V la energía potencial del sistema. Para un sistema de n grados de libertad, la magnitud escalar del Lagrangiano está determinada por la función $L = L(q_i, \dot{q}_i, t)$

2.3.5 Principio de Hamilton

Usado para determinar el movimiento de un punto del sistema en el espacio n-dimensional determinado por n coordenadas generalizadas entre dos instantes de tiempo, se basa en la optimización de la integral respecto al tiempo del Lagrangiano y establece que el movimiento estará orientado hacia el punto donde la integral del Lagrangiano sea un extremo de la trayectoria del movimiento. Esto se observa en la Ecuación 21.

$$I = \int_{t_1}^{t_2} L(q_i, \dot{q}_i, t) dt \quad (21)$$

2.3.6 Ecuaciones de movimiento de Lagrange para sistemas conservativos

En el caso donde no hay disipación de energía en el movimiento, si consideramos el caso anterior y asumimos la presencia de una función δq continua y con derivada continua entre t_1 y t_2 que además toma valores de 0 en esos instantes, podemos evaluar ΔI para pequeñas variaciones de q y \dot{q} , como se observa en la Ecuación 22 y 23.

$$\Delta I = \int_{t_1}^{t_2} L(q + \delta q, \dot{q} + \delta \dot{q}, t) dt - \int_{t_1}^{t_2} L(q, \dot{q}, t) dt \quad (22)$$

$$\Delta I = \int_{t_1}^{t_2} [L(q + \delta q, \dot{q} + \delta \dot{q}, t) - L(q, \dot{q}, t)] dt \quad (23)$$

Expandiendo en series de Taylor alrededor de (q, \dot{q}) , se obtiene la Ecuación 24.

$$\Delta I = \int_{t_1}^{t_2} \left(\frac{\delta L}{\delta q} \delta q + \frac{\delta L}{\delta \dot{q}} \delta \dot{q} \right) dt - \frac{1}{2} \int_{t_1}^{t_2} \left[\frac{\delta^2 L}{\delta q^2} (\delta q)^2 + 2 \frac{\delta^2 L}{\delta q \delta \dot{q}} \delta q \delta \dot{q} + \frac{\delta^2 L}{\delta \dot{q}^2} (\delta \dot{q})^2 \right] dt + \dots \quad (24)$$

Dónde $\frac{\delta L}{\delta q} \delta q + \frac{\delta L}{\delta \dot{q}} \delta \dot{q}$ se llama “primera variación de I”, la teoría de variaciones establece que para que I sea un extremo la primera variación de I o de δI debe ser 0, esto se muestra en la Ecuación 25.

$$\delta I = \int_{t_1}^{t_2} \left(\frac{\delta L}{\delta q} \delta q + \frac{\delta L}{\delta \dot{q}} \delta \dot{q} \right) dt = \int_{t_1}^{t_2} \frac{\delta L}{\delta q} \delta q dt + \int_{t_1}^{t_2} \frac{\delta L}{\delta \dot{q}} \delta \dot{q} dt = 0 \quad (25)$$

La Ecuación 25 puede reescribirse como la Ecuación 26.

$$\int_{t_1}^{t_2} \frac{\delta L}{\delta q} \delta q dt + \frac{\delta L}{\delta \dot{q}} \delta q \Big|_{t_1}^{t_2} - \int_{t_1}^{t_2} \frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}} \right) \delta q dt = 0 \quad (26)$$

Por tanto, se obtiene la Ecuación 27.

$$\frac{\delta L}{\delta \dot{q}} \delta q(t_2) - \frac{\delta L}{\delta \dot{q}} \delta q(t_1) + \int_{t_1}^{t_2} \left(\frac{\delta L}{\delta q} - \frac{d}{dt} \frac{\delta L}{\delta \dot{q}} \right) \delta q dt = 0 \quad (27)$$

Debido a que $\delta q(t_1) = \delta q(t_2) = 0$, se reemplaza en la Ecuación 27 y se obtiene la Ecuación 28.

$$\int_{t_1}^{t_2} \left(\frac{\delta L}{\delta q} - \frac{d}{dt} \frac{\delta L}{\delta \dot{q}} \right) \delta q dt = 0 \quad (28)$$

De acuerdo con la teoría de variaciones, el segundo integrando debe ser idénticamente cero, como se observa en la Ecuación 29.

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}} - \frac{\delta L}{\delta q} \right) = 0 \quad (29)$$

La Ecuación 29 es conocida como ecuación de movimiento de Lagrange para sistemas conservativos, la cual se reduce a la ecuación de movimiento del sistema, cuando el Lagrangiano es en función de coordenadas generalizadas de n variables, para $i=1, 2, \dots, n$ se obtiene la Ecuación 30.

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} - \frac{\delta L}{\delta q_i} \right) = 0 \quad (30)$$

2.4 Ingeniería de control moderna

Un sistema es controlable si se puede transferir desde un estado inicial $x(t_0)$ a la referencia deseada mediante una señal de control, además es observable si se puede determinar el estado a partir de la observación de la salida

2.4.1 Controlabilidad

Partiendo del hecho de que los fenómenos físicos suelen ser modelados en sistemas dinámicos no lineales que posteriormente linealizamos en un punto de operación para llegar a un

espacio lineal al que le podamos aplicar la teoría de control existente, podemos asumir que el objetivo de control será llevar el estado inicial $x(t_0)$ hasta el origen. Con el sistema descrito mediante la Ecuación 31.

$$\dot{x} = Ax + Bu \quad (31)$$

Tenemos que la solución está dada por la Ecuación 32

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \quad (32)$$

Aplicando la definición de controlabilidad planteada anteriormente se obtiene la Ecuación 33.

$$x(t_1) = 0 = e^{At_1}x(0) + \int_0^{t_1} e^{A(t_1-\tau)}Bu(\tau)d\tau \quad (33)$$

La cual es equivalente a la Ecuación 34

$$x(0) = - \int_0^{t_1} e^{-A\tau}Bu(\tau)d\tau \quad (34)$$

Basado en la interpolación de Sylvester se puede obtener la siguiente expresión para la exponencial (Ecuación 35).

$$e^{-A\tau} = \sum_{k=0}^{n-1} \alpha_k(\tau) A^k \quad (35)$$

Por tanto se obtiene la Ecuación 36

$$x(0) = - \sum_{k=0}^{n-1} A^k B \int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau \quad (36)$$

Definiendo la integral de la Ecuación 36 se obtiene la Ecuación 37

$$\int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau = \beta_k \quad (37)$$

Para llegar a la Ecuación 38

$$x(0) = - \sum_{k=0}^{n-1} A^k \beta_k \quad (37)$$

Si el sistema es completamente controlable, el rango de la matriz resultante debe ser n , dicha matriz es conocida como la matriz de controlabilidad.

2.4.2 Observabilidad

Para determinar la observabilidad, se parte del sistema no forzado que es descrito mediante la Ecuación 38 y 39.

$$\dot{x} = Ax \quad (38)$$

$$y = Cx \quad (39)$$

El sistema es observable en caso de que se pueda determinar el estado mediante la observación de la salida, para que esto ocurra, todas las transiciones de estado deben afectar la salida. La observabilidad es utilizada en los sistemas reales para estimar variables que no pueden ser medidas directamente y para disminuir la cantidad de sensores empleados, lo cual se traduce en una disminución del precio final del procesador.

El vector de salida está dado por la Ecuación 40

$$y(t) = Ce^{At}x(0) \quad (40)$$

Mediante la interpolación de Sylvester se puede llegar a la Ecuación 35. Por lo tanto para la observabilidad se llega a la Ecuación 41.

$$y(t) = \sum_{k=0}^{n-1} \alpha_k(t) CA^k x(0) \quad (41)$$

La Ecuación 41 es equivalente a la Ecuación 42

$$y(t) = \alpha_0(t)Cx(0) + \alpha_1(t)CAc(0) + \dots + \alpha_{n-1}(t)CA^{n-1}x(0) \quad (42)$$

Si el sistema es completamente observable, el estado inicial se puede determinar empleando únicamente la ecuación anterior, esto se cumple cuando el rango de la matriz es n .

2.4.3 Control optimo: LQR (Linear Quadratic Regulator)

Asumiendo que el sistema se encuentra completamente estimado, es decir que se tienen mediciones de todas las variables, si el sistema es controlable tenemos la opción de establecer una ley de control $u = -Kx$ que lleve el sistema a la estabilidad. Sabiendo que se tienen limitaciones en la señal de control y que una elección de polos demasiado alejada del eje imaginario puede causar que el sistema sea demasiado afectado por ruido, es necesario ubicar los polos correctamente para mantener la robustez del controlador, respetando las limitaciones del actuador.

Es importante elegir correctamente la ganancia K del controlador para estabilizar el sistema de la manera deseada sin requerir demasiado esfuerzo de control, a esto se conoce como control óptimo, el cual busca el equilibrio entre la estabilidad y agresividad de control. Para lograr esto se cuantifican las necesidades de diseño mediante una función de costo, la cual es optimizada para elegir adecuadamente la ubicación de los polos.

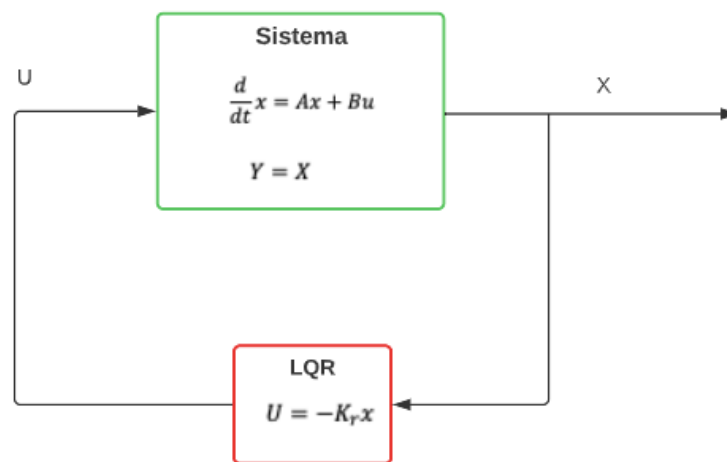
Para construir la función de costo se asignan las matrices Q y R que representan el costo del error en cada variable de estado y el costo de la señal de control, juntas se emplean para formar la función de costo, dada por la ecuación 43.

$$J(t) = \int_0^t x(\tau) * Qx(\tau) + u(\tau) * Ru(\tau)d\tau \quad (43)$$

Esta selección de polos es utilizada para un controlador como el mostrado en la Figura 7.

Figura 7.

Diagrama de bloques de un controlador optimo (LQR).



Nota. (Autores)

El algoritmo toma su nombre puesto que se refiere a una ley de control lineal, diseñada para un sistema lineal, que minimiza una función de costo cuadrática, con la Ecuación 44 es posible obtener una solución analítica para determinar la ganancia K_r del controlador.

$$K_r = R^{-1}B * X \quad (44)$$

Siendo X la solución de una ecuación de Riccati, se obtiene la Ecuación 45

$$A * X + XA - XBR^{-1}B * X + Q = 0 \quad (45)$$

Para hallar la ecuación de Riccati se apoya en la similitud entre el control optimo y la formulación del Lagrangiano y el Hamiltoniano en mecánica clásica.

2.4.4 Estimador optimo: LQR (Filtro de Kalman)

El controlador LQR requiere estimación total del sistema, por tanto es necesario un sensor para cada variable o en su defecto, realizar estimaciones a través de las mediciones disponibles, pues algunas de ellas pueden significar costos muy elevados o simplemente no ser posibles de medir, para solucionar este inconveniente se puede emplear un observador, elemento que tiene la capacidad de determinar el estado del sistema mediante mediciones de la salida.

Es posible estimar completamente el sistema si se determina que la matriz de observabilidad tiene rango completo, la efectividad de la estimación depende del grado de observabilidad, cuantificado por el Gramiano de observabilidad, El filtro de Kalman es el observador más usado dado que cuenta con la capacidad de compensar el ruido, perturbaciones e incertidumbre del modelo.

Para elegir el estimador es necesario incluir los parámetros de perturbación externa y de medición a las ecuaciones constitutivas del espacio de estados, recordando que el filtro de Kalman asume que su media es 0, por tanto se puede caracterizar el ruido a partir de su covarianza.

Se puede estimar el sistema mediante la Ecuación 46 y 47.

$$\frac{d}{dt} \hat{x} = A\hat{x} + Bu + K_f(y - \hat{y}) \quad (46)$$

$$\hat{y} = C\hat{x} + Du \quad (47)$$

Donde K_f está dada por la Ecuación 48

$$K_f = YC * V_n \quad (48)$$

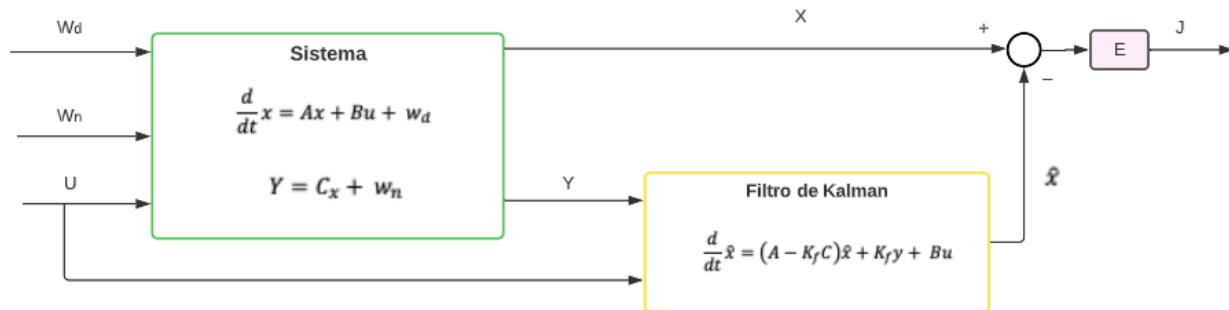
Donde Y es la solución de otra ecuación de Riccati en forma de la Ecuación 49

$$YA * +AY - YC * V_n^{-1}CY + V_d = 0 \quad (49)$$

A continuación, en la Figura 8 se puede ver un diagrama donde se muestra el funcionamiento esquemático del filtro de Kalman.

Figura 8.

Diagrama de bloques del funcionamiento del filtro de Kalman (LQR).



Nota. (Autores)

2.4.5 Control óptimo basado en sensores: LQG (Linear Quadratic Gaussian)

Es común emplear la estimación obtenida mediante el filtro de Kalman en conjunto con la ley de control óptimo para sistemas completamente estimados LQR, resultando en el LQG. A pesar de que se construye el estimador y el controlador por aparte, sus características óptimas siguen siendo válidas y son llevadas al controlador final, el sistema dinámico final cuenta con entrada, salida y estimaciones internas de estado \hat{x} , mostradas en la Ecuación 50 y 51.

$$\frac{d}{dt} \hat{x} = (A - K_f C - B K_r) \hat{x} + K_f y \quad (50)$$

$$u = -K_r \hat{x} \quad (51)$$

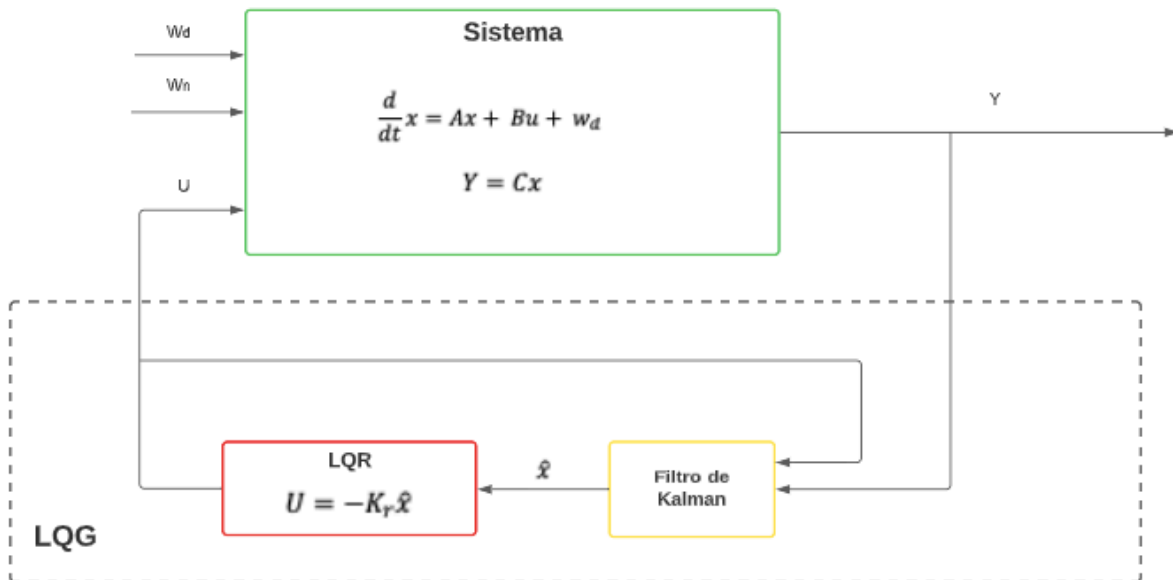
El controlador LQG es óptimo con respecto a una función de costo conjunta, adicionalmente el controlador incluye las perturbaciones, de tal manera que el sistema final resulta descrito por la Ecuación 52.

$$\frac{d}{dt}x = Ax - BK_r x + BK_r \epsilon + w_d \quad (52)$$

A continuación, en la Figura 9 se muestra el diagrama de bloques correspondiente al controlador óptimo basado en sensores con ruido.

Figura 9.

Diagrama de bloques de un controlador óptimo (LQR) con aplicación del filtro de Kalman (LQG).



Nota. (Autores)

Por tanto, las matrices que definen el comportamiento del sistema controlado están dadas por la Ecuación 53.

$$\frac{d}{dt} \begin{bmatrix} x \\ \epsilon \end{bmatrix} = \begin{bmatrix} A - BK_r & BK_r \\ 0 & A - K_f C \end{bmatrix} \begin{bmatrix} x \\ \epsilon \end{bmatrix} + \begin{bmatrix} I & 0 \\ I & -K_f \end{bmatrix} \begin{bmatrix} w_d \\ w_n \end{bmatrix} \quad (53)$$

De tal manera se puede ver que los autovalores para el sistema controlado mediante LQG son proporcionados por los autovalores escogidos mediante LQR y LQE. Es importante tener en cuenta que el controlador se basa en un adecuado conocimiento del modelo y de las perturbaciones, además de asumir que estas últimas presentan distribución Gaussiana, en caso de que alguna de estas afirmaciones resulte no ser verdadera se puede perder la robustez del controlador, resultando en inestabilidad del sistema.

3. Metodología

En este capítulo se describirán los métodos y herramientas utilizados para la creación del material didáctico con énfasis en la teoría de control a través del estudio de uno de los sistemas mecánicos más clásicos y típicos en la investigación de nuevas ideas de control, el péndulo invertido lineal.

3.1 Modelo Matemático del Péndulo Invertido Lineal

Para determinar un modelo matemático del péndulo invertido lineal, es necesario, determinar las ecuaciones matemáticas que representan el comportamiento físico de este sistema dinámico no lineal e inestable.

En este caso de estudio, las variables con las que se logró definir todo el sistema son la posición del carro y el ángulo del péndulo respecto a la vertical, se encontraron las fórmulas que contemplan estas variables en función de una fuerza, la cual fue la variable que se controló, ya que

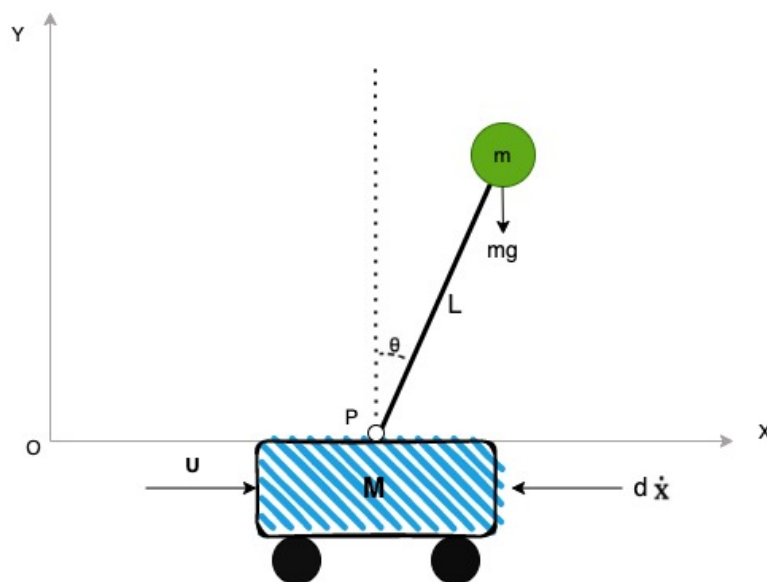
sin dicha fuerza de control, el péndulo será siempre inestable por su libertad de girar en cualquier momento y dirección.

3.1.1 DCL del Péndulo Invertido Lineal

En la Figura 10 se muestra el diagrama de cuerpo libre del péndulo invertido lineal, donde se observa que el sistema mecánico está conformado por un péndulo pivotado sobre un carro. En el péndulo se puede observar, que la masa se ubica en el extremo de la barra, siendo este entonces su centro de gravedad. También se tiene en cuenta una fuerza disipativa de amortiguación por fricción, la cual será proporcional a la velocidad del carro en cada instante. Otras referencias como lo son el origen del sistema coordenado, el ángulo que forma el péndulo respecto a la vertical y el punto donde pivota el péndulo con el carro, aparecen representados en el DCL.

Figura 10

DCL del Péndulo Invertido Lineal



Donde:

$M =$ Masa del carro

$m =$ masa del péndulo

$L =$ Longitud del péndulo

$g =$ Gravedad

$d =$ Factor de amortiguamiento

$U =$ Fuerza de control (fuerza aplicada al carro)

$\theta =$ Angulo de posición del péndulo respecto a la vertical

$P =$ Punto pivote entre el carro y péndulo

$O =$ Origen del sistema de coordenada X, Y

$\dot{X} =$ Velocidad del carro

3.1.2 Ecuaciones de Movimiento

Para describir el comportamiento físico en términos de las variables del sistema, se utilizó la mecánica lagrangiana. El péndulo invertido lineal es un sistema dinámico que cuenta con dos grados de libertad, uno rotacional y el otro traslacional, por lo tanto, se consideraron dos coordenadas generalizadas y se definieron dos ecuaciones de movimiento en forma de ecuaciones diferenciales.

Las ecuaciones de movimiento se derivan a partir de la ecuación de Euler – Lagrange (Ecuación 54).

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i \quad (i = 1,2) = Q_i \quad (i = X, \theta) \quad (54)$$

Donde

$q_i =$ *Coordenada generalizada*

$\dot{q}_i =$ *velocidad generalizada (derivación respecto al tiempo de la coordenada generalizada)*

$Q_i =$ *Fuerzas generalizadas aplicadas*

$L =$ *Lagrangiano del sistema*

$i =$ *Subíndice que indica el número de grados de libertad, en este caso son dos, el primero (1) será respecto a la posición del carro en el eje x, y el segundo (2) corresponderá a la posición angular del péndulo respecto a la vertical (θ).*

Para determinar la primera ecuación de movimiento se toma la primera coordenada generalizada $q_1 = X$, por lo tanto $\dot{q}_1 = \dot{X}$, $Q_1 = U - d\dot{X}$, ya que son las fuerzas inmersas en la energía cinética que dictan el movimiento en el eje X del sistema.

Reemplazando estos términos en la Ecuación 54 de Euler – Lagrange se obtiene la Ecuación 55

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{X}} \right) - \frac{\partial L}{\partial X} = U - d\dot{X} \quad (55)$$

La Ecuación 56 consiste en la segunda ecuación de movimiento que se obtiene teniendo en cuenta la segunda coordenada generalizada $q_2 = \theta$, en donde $\dot{q}_2 = \dot{\theta}$ y $Q_2 = 0$, debido a que en este caso no se consideran fuerzas generalizadas con esta coordenada; se procede a reemplazar estos términos en la Ecuación 54 de Euler – Lagrange.

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = 0 \quad (56)$$

Para determinar entonces las ecuaciones de movimiento 55 y 56, se necesita resolver la ecuación 57 correspondiente al lagrangiano $L = L(q_i, \dot{q}_i, t)$, el cual está en función de q_i , \dot{q}_i y el tiempo (t).

$$L = T - V \quad (57)$$

Donde

$T = \text{Energía cinética del sistema}$

$V = \text{Energía potencial del sistema}$

La energía cinética del sistema, correspondiente a la Ecuación 58, se expresa como la suma de la energía cinética del carro (T_c) definida en la Ecuación 59 más la energía cinética del péndulo (T_p) definida en la Ecuación 60 y se encuentra en función de $T(q_i, \dot{q}_i)$.

$$T = T_c + T_p \quad (58)$$

$$T_c = \frac{1}{2} M \dot{X}^2 \quad (59)$$

$$T_p = \frac{1}{2} m v_p^2 \quad (60)$$

En la Ecuación 60, v_p representa la velocidad del péndulo, la cual está conformada por dos componentes: la componente en el eje X (v_{xp}) y la componente en el eje Y (v_{yp}). Esto se expresa en la ecuación 61.

$$v_p = \sqrt{v_{xp}^2 + v_{yp}^2} \quad (61)$$

Para poder determinar la velocidad del péndulo (v_p), se debe establecer la posición del centro de gravedad con respecto al origen del sistema de coordenadas (O) en la Figura 10, dicha posición en el plano XY de la Figura 10 viene dado por la Ecuación 62 correspondiente a la coordenada en el eje X y la Ecuación 63 correspondiente a la coordenada en el eje Y.

$$X_p = X + L \sin \theta \quad (62)$$

$$Y_p = L \cos \theta \quad (63)$$

Derivando la Ecuación 62 y 63, se obtiene las dos componentes que conforman la velocidad del péndulo (v_p). El producto de estas derivadas corresponde a la Ecuación 64 y 65.

$$v_{xp} = \dot{X}_p = \dot{X} + L\dot{\theta} \cos \theta \quad (64)$$

$$v_{yp} = \dot{Y}_p = -L\dot{\theta} \sin \theta \quad (65)$$

Reemplazando las ecuaciones 64 y 65 en la Ecuación 61, se logra definir la velocidad del péndulo (v_p) con la Ecuación 66.

$$v_p^2 = (\dot{X} + L\dot{\theta} \cos \theta)^2 + (-L\dot{\theta} \sin \theta)^2 \quad (66)$$

Una vez definida la velocidad del péndulo se tendrán todos los términos necesarios para reemplazar en la Ecuación 60 y determinar la Ecuación 67 correspondiente a la energía cinética del péndulo en términos de las variables del sistema.

$$T_p = \frac{1}{2} m \left[(\dot{X} + L\dot{\theta} \cos \theta)^2 + (-L\dot{\theta} \sin \theta)^2 \right] \quad (67)$$

Desarrollando la Ecuación 67 y simplificando términos se obtiene la Ecuación 68.

$$T_p = \frac{1}{2} m \dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2} mL^2 \dot{\theta}^2 \quad (68)$$

Habiendo definido la energía cinética del péndulo con la ecuación 68 y la energía cinética del carro con la ecuación 59, se procede a reemplazar dichos términos en la Ecuación 58, para determinar la ecuación 69.

$$T = \frac{1}{2}(M + m)\dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2}mL^2\dot{\theta}^2 \quad (69)$$

La energía potencial del sistema (V) se encuentra en función únicamente de la coordenada generalizada $V(q_i)$ y se define como la suma de la energía potencial del péndulo (V_p) más la energía potencial de carro (V_c), es decir $V = V_c + V_p$.

Según el sistema coordinado ubicado en la Figura 10, el punto de referencia para determinar la altura, con la cual se estima la energía potencial, se ubica sobre la parte superior del carro, más específicamente en el punto P, donde pivota el péndulo con el carro, por lo tanto la energía potencial del carro será nula. $V_c = 0$ y la energía potencial del péndulo (V_p) será dada por $V_p = mgL \cos \theta$, en donde la altura corresponde a la distancia vertical que hay desde el punto P hasta el centro de gravedad del péndulo ($L \cos \theta$). Teniendo en cuenta esto la Ecuación 70 será la representación de la energía potencial del sistema.

$$V = mgL \cos \theta \quad (70)$$

Una vez definida la energía cinética con la Ecuación 69 y la energía potencial del sistema con la Ecuación 70, se tienen las variables a reemplazar en la Ecuación 57 para obtener la Ecuación 71 que representa el Lagrangiano del sistema.

$$L = \frac{1}{2}(M + m)\dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2}mL^2\dot{\theta}^2 - mgL \cos \theta \quad (71)$$

La Ecuación 72 hace referencia a la primera ecuación de movimiento con respecto a la coordenada $q_1 = X$ y se obtiene reemplazando la Ecuación 71 en la Ecuación 55.

$$\begin{aligned} & \frac{d}{dt} \left[\frac{\partial}{\partial \dot{X}} \left(\frac{1}{2} (M + m) \dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2} mL^2 \dot{\theta}^2 - mgL \cos \theta \right) \right] \\ & - \left[\frac{\partial}{\partial X} \left(\frac{1}{2} (M + m) \dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2} mL^2 \dot{\theta}^2 - mgL \cos \theta \right) \right] \quad (72) \\ & = U - d\dot{X} \end{aligned}$$

Resolviendo las derivadas en la Ecuación 72 se obtiene la Ecuación 73.

$$(M + m)\ddot{X} - mL\dot{\theta}^2 \sin \theta + mL\ddot{\theta} \cos \theta + d\dot{X} = U \quad (73)$$

La Ecuación 74 será entonces la segunda ecuación de movimiento con respecto a la coordenada $q_2 = \theta$ y se obtiene reemplazando la Ecuación 71 en la Ecuación 56.

$$\begin{aligned} & \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\theta}} \left(\frac{1}{2} (M + m) \dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2} mL^2 \dot{\theta}^2 - mgL \cos \theta \right) \right] \\ & - \left[\frac{\partial}{\partial \theta} \left(\frac{1}{2} (M + m) \dot{X}^2 + mL\dot{X}\dot{\theta} \cos \theta + \frac{1}{2} mL^2 \dot{\theta}^2 - mgL \cos \theta \right) \right] \quad (74) \\ & = 0 \end{aligned}$$

Resolviendo las derivadas de la Ecuación 74 y simplificando términos se obtiene la Ecuación 75.

$$mL^2\ddot{\theta} + mL\dot{X} \cos \theta - mgL \sin \theta = 0 \quad (75)$$

Las Ecuaciones 73 y 75 consisten entonces en ecuaciones diferenciales de segundo orden, que definen el movimiento del péndulo invertido lineal.

En este caso de estudio relacionando las dos ecuaciones de movimiento entre sí, se podrá expresar toda la dinámica no lineal del sistema con las siguientes ecuaciones:

$$\dot{X} = v \quad (76)$$

$$\ddot{X} = \dot{v} = \frac{-m^2L^2g \cos \theta \sin \theta + mL^2(mLw^2 \sin \theta - dv) + mL^2U}{mL^2[M + m(1 - \cos^2\theta)]} \quad (77)$$

$$\dot{\theta} = w \quad (78)$$

$$\ddot{\theta} = \dot{w} = \frac{(M + m)mgL \sin \theta - mL \cos \theta (mLw^2 \sin \theta - dv) - mL \cos \theta U}{mL^2[M + m(1 - \cos^2\theta)]} \quad (79)$$

Donde

$v =$ velocidad lineal del sistema

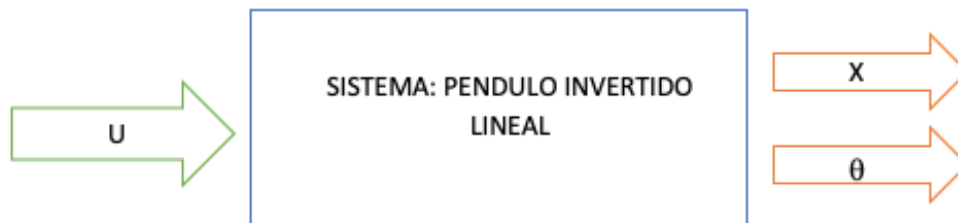
$w =$ velocidad angular del carro

3.2 Espacios de estados

El espacio de estados hace parte de la teoría de control moderna ya que permite modelar un sistema dinámico con múltiples entradas y/o salidas en contraste con el control clásico el cual solo es aplicable a sistemas lineales de una entrada y una sola salida invariantes en el tiempo, por ende el espacio de estados representa la mejor manera para controlar el sistema no lineal de una entrada correspondiente a la fuerza de control (U) y dos salidas, las cuales son el ángulo del péndulo respecto a la vertical (θ) y la posición del carro (X). En la Figura 11 se muestra el diagrama de entrada y salidas del péndulo invertido.

Figura 11

Diagrama de entrada y salidas del sistema



Nota. El péndulo invertido corresponde a un sistema SIMO (single input y multiple-output) (Autores)

3.2.1 Variables de Estado

Una vez definida la entrada y salidas del péndulo invertido, se procede a definir las variables de estado del sistema.

$$X_1 = X$$

$$X_2 = \dot{X}$$

$$X_3 = \theta$$

$$X_4 = \dot{\theta}$$

Derivando se obtiene

$$\dot{X}_1 = X_2$$

$$\dot{X}_2 = \ddot{X}$$

$$\dot{X}_3 = X_4$$

$$\dot{X}_4 = \ddot{\theta}$$

3.2.2 Linealización de las Ecuaciones de Movimiento

El objetivo de aplicar el control moderno al sistema es mantener el péndulo invertido en posición vertical, por lo tanto, se tienen dos puntos fijos, el primero es cuando el péndulo se encuentre hacia abajo es decir $\theta = 0$, y el segundo corresponde a cuando el péndulo se ubique hacia arriba $\theta = \pi$, en ambos casos la velocidad angular será $w = 0$, y la posición del carro es una variable libre.

En este caso de estudio, el estado de operación del péndulo, es en posición vertical hacia arriba, por lo tanto, se procede a linealizar la Ecuación 73 y 75 alrededor de $\theta = \pi$, por lo que se toma entonces $\theta = \pi - \varphi$, donde φ es un ángulo muy pequeño respecto a la vertical. Linealizar el modelo matemático facilita el análisis del sistema y el diseño de controladores.

Teniendo en cuenta la configuración del péndulo escogida, $\cos \theta = \cos(\pi - \varphi) = -1$, $\sin \theta = \sin(\pi - \varphi) = -\varphi$, y $\dot{\theta}^2 = 0$.

Aplicando estas condiciones y reemplazando las variables de estado en la Ecuación 77 y 79, con la Ecuación 80 y 81 se obtiene el modelo lineal del sistema.

$$\ddot{X} = \dot{v} = \frac{mg}{M}X_3 - \frac{d}{M}X_2 + \frac{1}{M}U \quad (80)$$

$$\ddot{\theta} = \dot{w} = \frac{-(M+m)g}{ML}X_3 - \frac{d}{ML}X_2 + \frac{1}{ML}U \quad (81)$$

3.2.3 Ecuación de Estado

Una vez linealizado el sistema, se plantea la Ecuación 82 de primer orden y de forma matricial, en la matriz de estado se agrega el parámetro b el cual considera los dos puntos fijos establecidos para el sistema dinámico, el primero b=1 cuando el péndulo se encuentra hacia arriba y el segundo b= -1 cuando el péndulo está hacia abajo.

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \end{bmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{d}{M} & \frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{d}{ML} & -\frac{(M+m)g}{ML} & 0 \end{pmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{ML} \end{bmatrix} U \quad (82)$$

La ecuación de estado incluye las matrices de estado y de entrada, las cuales son las matrices A y B respectivamente.

3.2.4 Ecuación de Salida

La Ecuación 83 corresponde a la ecuación en forma matricial de salida del sistema. La matriz C corresponde a la matriz de Salida.

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} \quad (83)$$

3.3 Sistema del Péndulo Invertido en Ambiente Matlab

Una vez determinada la representación en espacio de estados del péndulo invertido, se procede a crear el código que determina la simulación del péndulo invertido lineal como sistema de control, (este código recibe el nombre de Metodología.mlx). Para esto se indican las variables, que son las mismas utilizadas en las ecuaciones de movimiento y se les asigna un valor para determinar numéricamente las matrices de estado y de entrada (indicadas en la ecuación de estado), las cuales en el código aparecerán como matriz A y B respectivamente. (Todos los códigos de Matlab se encuentran en el apéndice A).

La tabla 1, muestra las variables a utilizar junto con sus valores y unidades de medición.

Tabla 1

Variables del sistema

Variable	Valor
M	5 Kg
m	1 Kg
L	2 m

g	-10 m/s ²
b	1 ó -1
d	1

Nota. La variable b es el parámetro que indica la posición del péndulo. La descripción de las otras variables se encuentra en la Figura 10. (Autores)

Figura 12

Código con asignación de valores a variables y matrices de estado (A) y entrada (B).

```
m=1; M=5; L=2; g=-10; d=1;
b=1; %Péndulo arriba (b=1), péndulo abajo (b=-1)

A=[0 1 0 0;0 -d/M b*m*g/M 0;0 0 0 1;0 -b*d/(M*L) -b*(m+M)*g/(M*L) 0];
B=[0; 1/M; 0; b/(M*L)];
```

Nota. Livescript (Metodologia.mlx).

3.3.1 Comportamiento no Lineal del Péndulo Invertido

En la Figura 13, se muestra un código con la función llamada Pendcart en términos de las variables de estado del sistema, la masa del carro (M), masa del péndulo (m), longitud del péndulo (L), gravedad (g), coeficiente de amortiguamiento (d) y fuerza aplicada al carro o de control (U). Esta función determina el comportamiento no lineal del sistema dinámico expresado en las ecuaciones 76, 77, 78 y 79. Todo con el fin de determinar el espacio de derivadas.

Figura 13

Código de la función Pendcart

```

1 function dx = pendcart(x,m,M,L,g,d,u)
2
3   Sx = sin(x(3));
4   Cx = cos(x(3));
5   D = m*L*(M+m*(1-Cx^2));
6
7   dx(1,1) = x(2);
8   dx(2,1) = (1/D)*(-m^2*L^2*g*Cx*Sx + m*L^2*(m*L*x(4)^2*Sx - d*x(2))) + m*L*(1/D)*u;
9   dx(3,1) = x(4);
10  dx(4,1) = (1/D)*((m+M)*m*g*L*Sx - m*L*Cx*(m*L*x(4)^2*Sx - d*x(2))) - m*L*Cx*(1/D)*u;

```

Nota. Esta función se crea en un script complementario y el código se encuentra en el apéndice A (Pendcart.m)

3.3.2 Propiedades del Sistema de Control

3.3.2.1 Estabilidad. Se puede comprobar la estabilidad de la planta sin control mediante la ubicación de sus polos en el plano complejo, teniendo en cuenta que el criterio de estabilidad enuncia que para que un sistema sea estable, todos sus polos deben ubicarse en el semiplano izquierdo del plano complejo, mediante la parte real de los polos podemos determinar la estabilidad del sistema. En la Figura 14, se muestra el comando aplicado a la matriz de estado (A) para determinar la estabilidad del sistema.

Figura 14

Comando para determinar la estabilidad del sistema

```
lambda=eig(A)
```

```

lambda = 4x1
         0
        -2.4311
        -0.2336
         2.4648

```

Nota. Livescript (Metodologia.mlx).

Se puede observar que el sistema cuenta con 4 polos donde el último polo cuenta con parte real positiva, es decir que se encuentra ubicado en el semiplano derecho, por lo tanto se puede concluir que el sistema es inestable en lazo abierto. Puesto que el objetivo es mantener la posición del péndulo arriba, surge la necesidad de un controlador que permita desplazar los polos del sistema con control al semiplano izquierdo.

3.3.2.2 Controlabilidad. Antes de diseñar el controlador se debe saber si es posible estabilizar el sistema mediante una señal de control, por lo tanto se debe controlar la controlabilidad del sistema con base en la matriz de controlabilidad, formada a partir de las matrices A y B, cuyo rango debe ser igual al número de variables de estado en el sistema, es decir que tenga rango completo. En la Figura 15 se muestra la línea de código con el comando para determinar esta propiedad del sistema.

Figura 15

Comando para determinar la controlabilidad del sistema

```
rank(ctrb(A,B))  
ans = 4
```

Nota. El rango de la matriz de controlabilidad es igual a 4, lo cual corresponde al mismo número de variables de estado del sistema, por lo tanto el péndulo invertido es un sistema controlable.

Livescript (Metodologia.mlx).

3.4 Diseño del Controlador

3.4.1 Diseño del Controlador LQR

El controlador LQR corresponde a un sistema de control completamente estimado, por lo tanto para diseñar el controlador se parte del caso ideal, donde se cuenta con mediciones precisas de todas las variables de estado en cada momento, situación que difiere de la realidad puesto que las mediciones tomadas suelen presentar ruido asociado a los sensores, además de las perturbaciones del entorno que afectan el comportamiento esperado del sistema y su respuesta al control aplicado.

3.4.1.1 Matrices de ponderación. El método de control LQR es ampliamente utilizado para el control mediante variables de estado, gracias a la practicidad que presenta a la hora de optimizar el desempeño del controlador, ya que realiza una ponderación a la importancia de respuesta para las variables de estado según los valores de la matriz Q y el impacto de las variables de control sobre el sistema mediante los valores de la matriz R . Las líneas de código utilizadas para determinar estas dos matrices Q y R se muestran en la Figura 16.

Figura 16

Código para determinar las matrices de ponderación Q y R .

```
Q=eye(4);  
R=0.0001;
```

Nota. La matriz Q está dada por el comando `eye(4)` el cual devuelve una matriz identidad de 4×4 . *Livescript* (Metodologia.mlx).

3.4.1.2 Matriz de ganancia. Teniendo en cuenta que para este método de control se emplea una ley de control $U = -K(X - W_r)$, donde U es la señal de control y K es la matriz de ganancia, la cual puede ser determinada empleando el comando nativo de Matlab `lqr` (Figura 17) el cual da como resultado representa ganancia que tendrá la señal de control en función del error.

Figura 17

Comando LQR

```
K=lqr(A,B,Q,R);
```

Nota. El comando LQR calcula la ganancia optima. *Livescript* (Metodologia.mlx).

3.4.1.3 Referencias y condiciones iniciales. Para una referencia $W_r = [1 \ 0 \ \pi \ 0]^T$ Y condiciones iniciales $X(0) = [-1 \ 0 \ \pi + 0.1 \ 0]^T$, donde lo anterior indica que el sistema inicia con una posición del carrito a dos metros de la referencia y una posición angular del péndulo desviada 0.1 radianes de la referencia, se puede obtener la respuesta del sistema ante la ley de control planteada. Para obtener los resultados se crea un vector de tiempo discreto desde 0 hasta 10 segundos con pasos de 0.001 denominado "tspan", a este vector se le asocian los estados resultantes de la solución de las ecuaciones diferenciales no lineales que describen el sistema, por tanto, se obtuvo muestras discretas del comportamiento del modelo completamente estimado. Estas referencias y condiciones iniciales se muestran en la Figura 18.

Figura 18

Código de referencias, condiciones iniciales y vector de tiempo discreto

```
tspan=0:0.001:10;
x0=[-1;0;pi+0.1;0];
wr=[1;0;pi;0];
```

Nota. Livescript (Metodologia.mlx).

3.4.1.4 Funcionamiento del controlador. Implementando la ley de control para el actuador en la variable "U", correspondiente a la señal de control y la cual depende de los estados "x", dependencia representada en el código mediante @, y cuyo valor está definido por $U = -K(X - W_r)$. La Figura 19 muestra la línea de código correspondiente al funcionamiento del controlador.

Figura 19

Comando para implementar la ley de control

```
u=@(x)-K*(x-wr);
```

Nota. Livescript (Metodologia.mlx).

3.4.1.5 Comparación con distintos controladores. Se puede comprobar que el controlador diseñado es la opción óptima, comparando el comportamiento del controlador escogido mediante el algoritmo LQR con otras ubicaciones estables de polos para evidenciar las diferencias de desempeño del controlador seleccionado respecto a los demás, para ello almacenamos los datos del controlador en la variable "xLQR" y creamos un bucle "for" que nos crea las demás gráficas mediante semillas aleatorias para cada uno los polos que serán ubicados mediante la función de Matlab "place" dentro de la región estable del semiplano complejo izquierdo, a las gráficas obtenidas mediante este proceso se les cambia ligeramente el color para

poder diferenciarlas de la obtenida mediante el algoritmo LQR. En la Figura 20 se muestra el código para generar la gráfica comparativa.

Figura 20

Código para la comparación grafica de la respuesta del controlador LQR.

```

for k=1:length(t)
    JLQR(k) = (xLQR(k,:) - wr') * Q * (xLQR(k,:) - wr) + u(xLQR(k,:))' ^ 2 * R;
end

for count = 1:100
    p = [-.5-3*rand; -.5-3*rand; -.5-3*rand; -.5-3*rand];
    K = place(A,B,p);
    u=@(x)-K*(x - wr);
    [t,x] = ode45(@(t,x)pendcart(x,m,M,L,g,d,u(x)),tspan,x0);

    for k=1:length(t)
        J(k) = (x(k,:) - wr') * Q * (x(k,:) - wr) + u(x(k,:))' ^ 2 * R;
    end

    Jz = cumtrapz(t,J);
    plot(t(1:50:end),Jz(1:50:end),'Color',[.5 .5 .5]), hold on;
end

plot(t,cumtrapz(t,JLQR),'k','LineWidth',2)

set(gcf,'Position',[100 100 500 300])

```

```

set(gcf,'PaperPositionMode','auto')
xlabel('Tiempo [s]')
ylabel('Costo')
hold off
grid on

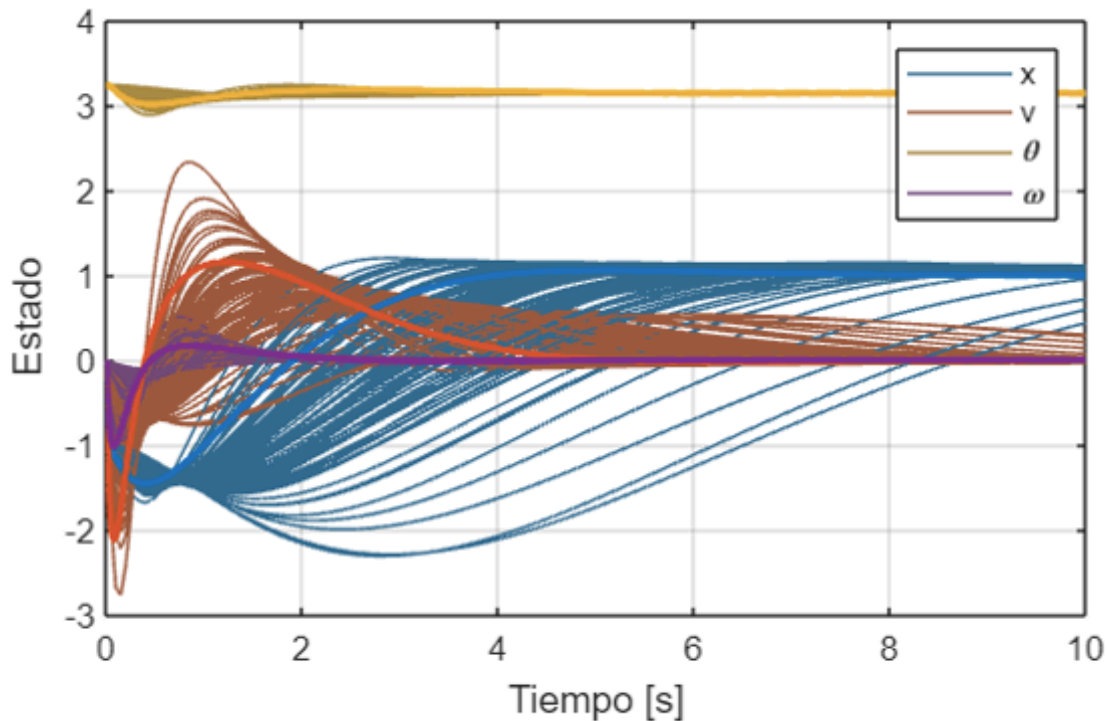
```

Nota. *Livescript* (Metodologia.mlx).

La Figura 21 corresponde a la gráfica generada por el código de la Figura 20.

Figura 21

Gráfica de la comparación de respuestas del controlador LQR con la posición de otros polos estables



Nota. Livescript (Metodologia.mlx).

Se dice que el algoritmo LQR selecciona la ubicación óptima para los polos, debido a que se emplea una función de costo (J) que toma en cuenta la ponderación planteada en las matrices Q y R , para calcular un valor numérico que caracteriza a cada selección de polos, dicho valor es minimizado mediante el algoritmo LQR. Podemos comparar el valor acumulado de la función de costo para distintas selecciones estables de polos, comparada con aquellos seleccionados mediante el algoritmo LQR, para ello hallamos el costo en cada instante de tiempo en cada caso, para luego mediante la función de Matlab "cumtrapz" realizar la integración trapezoidal que permite obtener

el acumulado, que al final del tiempo de simulación, se espera sea menor para los polos seleccionados mediante el algoritmo LQR. En la Figura 22 se muestra el código creado para generar la comparación gráfica.

Figura 22

Código para la comparación grafica de la función de costo del controlador LQR

```

for k=1:length(t)
    JLQR(k) = (xLQR(k,:)-wr')*Q*(xLQR(k,:)-wr) + u(xLQR(k,:))^2*R;
end

for count = 1:100
    p = [-.5-3*rand; -.5-3*rand; -.5-3*rand; -.5-3*rand];
    K = place(A,B,p);
    u=@(x)-K*(x - wr);
    [t,x] = ode45(@(t,x)pendcart(x,m,M,L,g,d,u(x)),tspan,x0);

    for k=1:length(t)
        J(k) = (x(k,:)-wr')*Q*(x(k,:)-wr) + u(x(k,:))^2*R;
    end

    Jz = cumtrapz(t,J);
    plot(t(1:50:end),Jz(1:50:end),'Color',[.5 .5 .5]), hold on;
end

plot(t,cumtrapz(t,JLQR),'k','LineWidth',2)

set(gcf,'Position',[100 100 500 300])

```

```

set(gcf,'PaperPositionMode','auto')
xlabel('Tiempo [s]')
ylabel('Costo')
hold off
grid on

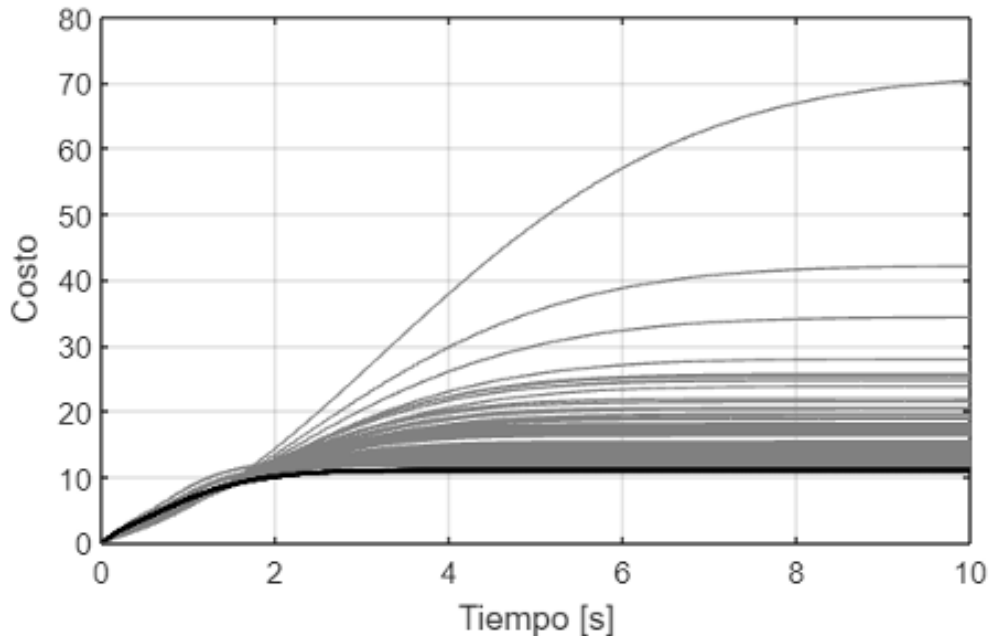
```

Nota. Livescript (Metodologia.mlx).

En la Figura 23 se muestra la gráfica obtenida mediante el código de la figura 22.

Figura 23

Grafica de la comparación de función de costo del controlador LQR con la posición de otros polos estables



Nota. La grafica en negrilla corresponde a la función de costo del controlador diseñado mediante la función “lqr”. *Livescript* (Metodologia.mlx).

Con la Figura 21 y 23 se confirma que el controlador diseñado mediante la función de Matlab “lqr” representa la opción óptima del sistema dinámico, con la ubicación adecuada de polos.

3.5 Filtro de Kalman

Para reducir la cantidad de sensores a implementar en un sistema real, se emplea un observador basado en el filtro de Kalman, el cual permite estimar el estado actual a partir de la medición de las salidas del sistema, teniendo en cuenta que estas mediciones pueden contener ruido originado en los sensores.

3.5.1 Observabilidad

Teniendo en cuenta que en la matriz de salida (representada en la ecuación 30), se indican las variables de estado de las cuales obtenemos las mediciones para nuestro sistema, se procede a verificar la observabilidad del sistema en función de la variable de estado que se pueda medir, para conocer la observabilidad del sistema es necesario que este sea estable, por tanto definimos la posición naturalmente estable del péndulo (abajo) mediante nuestra variable "b", el criterio de observabilidad es análogo al de controlabilidad, con la diferencia de que la matriz de observabilidad se construye a partir de las matrices de estado (A) y de salida (C). En la Figura 24 se muestra el código correspondiente a la determinación de matrices de observabilidad, con el fin de determinar la posición de la variable de estado en la matriz de salida que cumpla con el rango del mismo número de variables de estado, es decir cuatro.

Figura 24

Código para determinar matrices observabilidad del sistema

```
b=-1;
A=[0 1 0 0;0 -d/M b*m*g/M 0;0 0 0 1;0 -b*d/(M*L) -b*(m+M)*g/(M*L) 0];
B=[0; 1/M; 0; b/(M*L)];
C=[0 0 0 1];
rank(observ(A,C))
```

ans = 3

```
C=[0 0 1 0];
rank(observ(A,C))
```

ans = 3

```
C=[0 1 0 0];  
rank(observ(A,C))
```

```
ans = 3
```

```
C=[1 0 0 0];  
rank(observ(A,C))
```

Nota. Livescript (Metodologia.mlx).

3.5.2 Diseño del Observador

A partir de lo anterior podemos decir que el sistema es completamente observable si se incluye un sensor para la posición del carro, pues es el único caso donde la matriz de observabilidad tiene rango completo, siendo así se puede obtener la ganancia K_f del filtro de Kalman mediante el algoritmo LQR, junto a las estimaciones de covarianza de ruido y magnitud de perturbaciones, a partir de las cuales podremos establecer las matrices extendidas a usar junto al controlador LQR. Para este diseño del observador se emplea el código mostrado en la Figura 25.

Figura 25

Código para el diseño del observador

```
Vdmag = .04;  
Vd = Vdmag*eye(4);  
Vn = 0.02;  
Kf=(lqr(A',C',Vd,Vn))';  
B_aug=[B eye(4) 0*B];  
D_aug=[0 0 0 0 0 1];
```

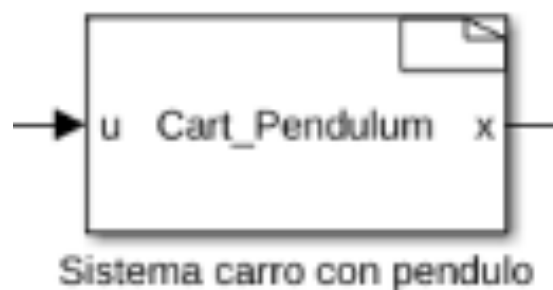
Nota. v_{dmag} corresponde a la magnitud de las perturbaciones que se le están aplicando al sistema, v_n corresponde a la varianza del ruido de los sensores, v_d matriz de la perturbaciones del sistema sobre las variables de estado. *Livescript* (Metodologia.mlx).

3.6 Sistema del péndulo invertido en ambiente Simulink

Mediante el software Simulink se realiza la simulación del comportamiento no lineal del péndulo invertido lineal y se da la aplicación de las técnicas de control LQR y LQG. Para dicha simulación se utiliza el bloque de “función S” de Simulink, al cual se le dió el nombre de `Cart_Pendulum`, como se muestra en la Figura 26. Este bloque se programó en lenguaje C++ para describir el modelo dinámico del sistema. La Figura 27 muestra la programación del bloque “funcion S”, dicha programación se utiliza en el diseño del controlador LQR. Para el controlador LQG también se utilizó un bloque “función S” el cual se llamó `cartpend_sim` (Figura 28) y su programación es C++ se observa en la Figura 29. (Los archivos de Simulink se encuentran en el apéndice A).

Figura 26

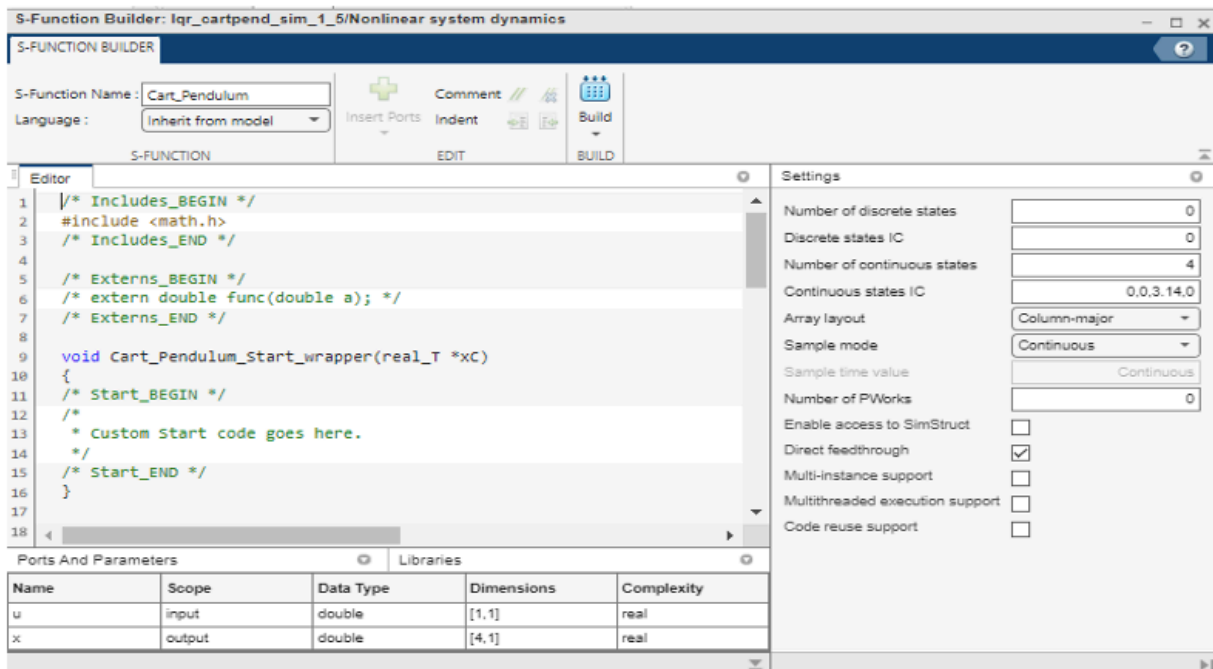
Bloque Funcion S, cart_pendulum programable en lenguaje C++



Nota.(LQR.slx)

Figura 27

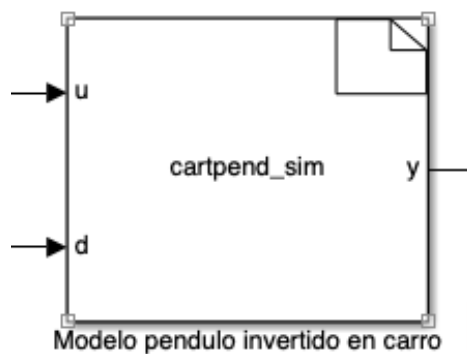
Programación del bloque `cart_pendulum` en lenguaje C++



Nota. En este bloque se ingresa el código C++ (ver apéndice B) el cual tiene en cuenta la figura 13 con la función `pendcart.` (LQR.slx)

Figura 28

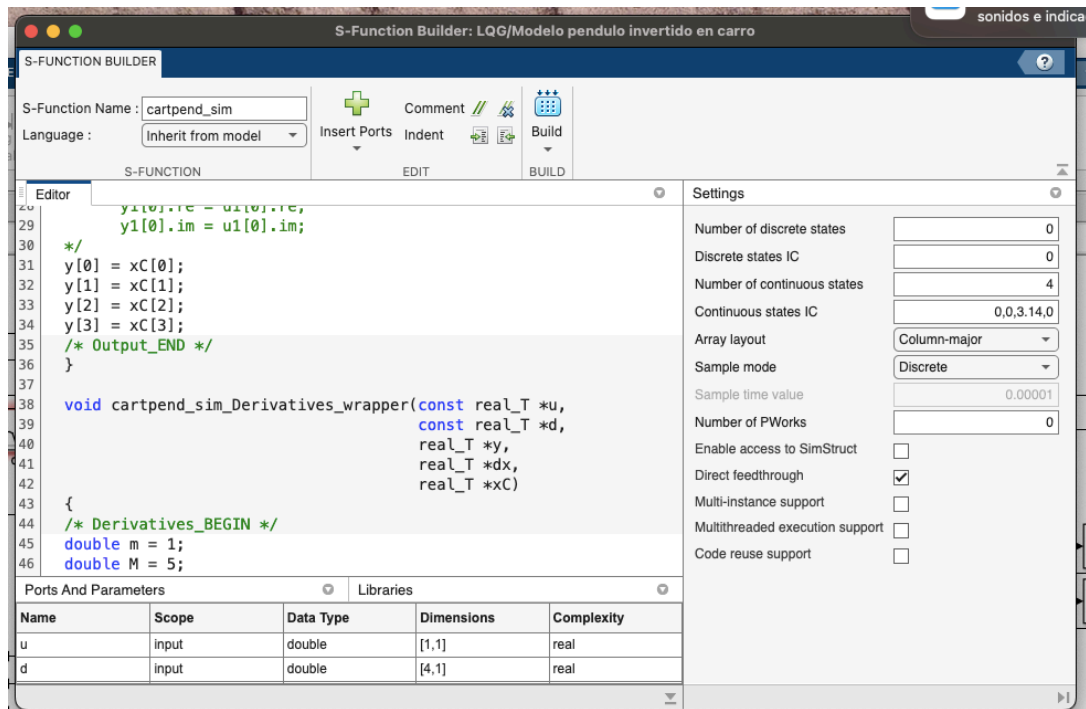
Bloque Función S, `cartpend_sim` programable en lenguaje C++



Nota.(LQG.slx)

Figura 29

Programacion del bloque cartpend_sim en lenguaje C++



Nota. En este bloque se ingresa el código C++ (ver apéndice B) el cual tiene en cuenta la figura 13 con la función pendcart. (LQR.slx)

Tanto en el bloque Cart_Pendulum (Figura 26), como en el bloque cartpend_sim (Figura 28) se modela la respuesta del sistema por medio de las ecuaciones diferenciales estipuladas, en la Figura 13 con la función pendcart. En el bloque Cart_Pendulum se tiene una entrada de control u y el vector de salida y (expresado como x en el diagrama de la Figura 30). Este bloque permite tener una simulación del sistema completo, ya que devuelve la posición del carro, su velocidad, el ángulo del péndulo y su velocidad angular en cualquier instante de tiempo. Cabe destacar que en el caso del LQR se asume que se puede medir exactamente todas estas variables y que sobre esta

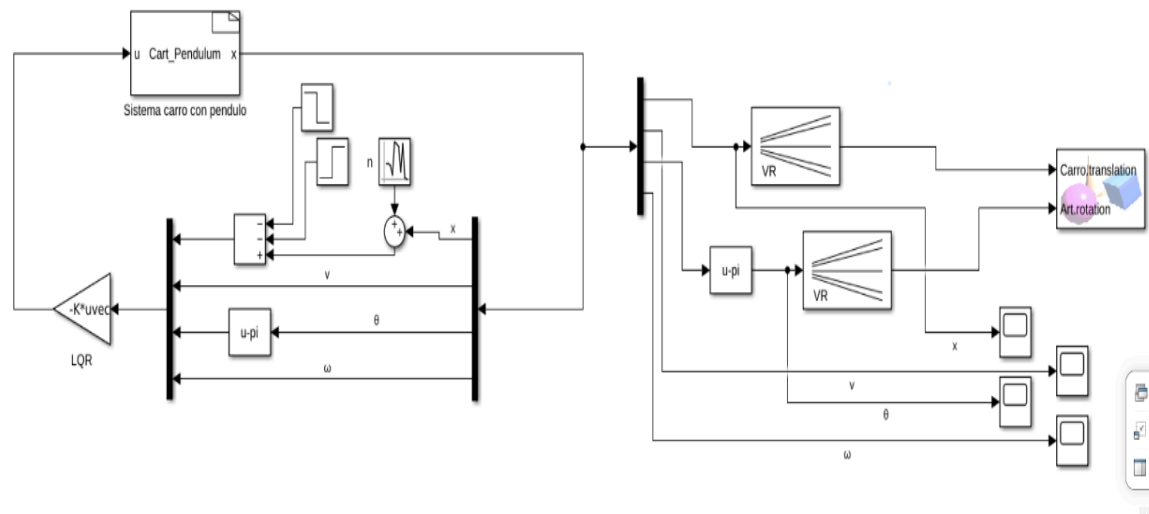
medición ideal se hace el control. La única perturbación que se tiene es un bloque de ruido que será explicado posteriormente. Por otra parte en el bloque carpend_sim correspondiente al controlador LQG, se tiene una entrada de control u y una entrada d para el ruido o perturbaciones en las cuatro variables de estado.

3.6.1 Controlador LQR

En la Figura 30 mediante la implementación del sistema no lineal en Simulink se puede encontrar el comportamiento del sistema cuando se tiene un sensor para cada variable de estado, para ello se emplea el siguiente modelo para el controlador LQR.

Figura 30

Diagrama del controlador LQR en Simulink

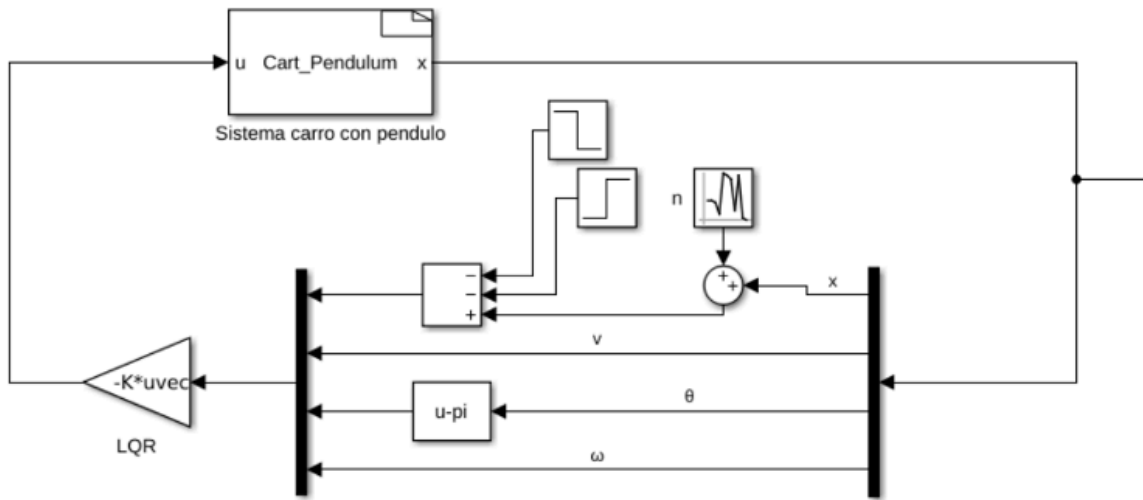


Nota. (LQR.slx)

La Figura 31 es la parte izquierda del diagrama LQR (Figura 30) y corresponde al sistema dinámico y el control del péndulo invertido.

Figura 31

Parte izquierda del diagrama del controlador LQR en Simulink

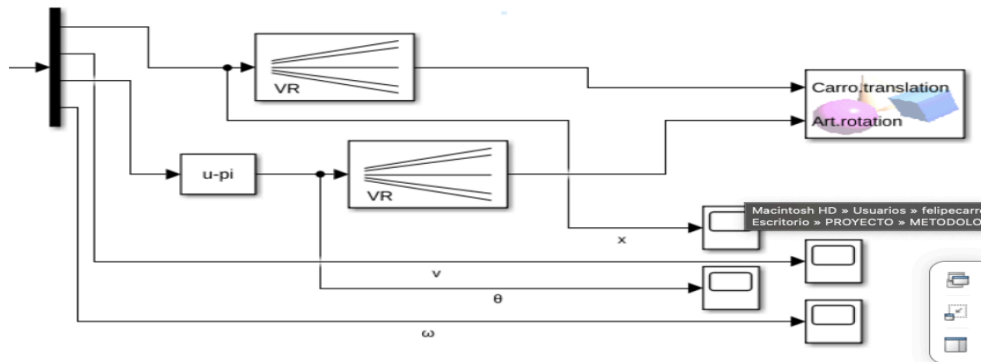


Nota. (LQR.slx).

La Figura 32 es la parte derecha del diagrama LQR (Figura 30) y corresponde a la animación del péndulo invertido al sistema .

Figura 32

Parte derecha del diagrama del controlador LQR en Simulink

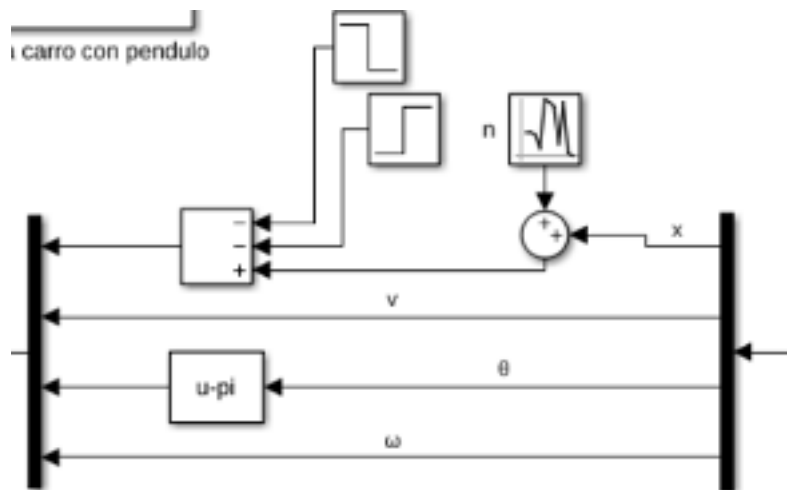


Nota. (LQR.slx)

La parte izquierda del diagrama (Figura 31) inicia con el bloque Cart_Pendulum (Figura 26), a continuación, como se observa en la Figura 33, se ubica un demultiplexor, que permite manipular cada uno de los componentes individualmente. En esta sección se tienen dos sumadores para la línea correspondiente a la posición del carro (x). Uno de los sumadores contiene dos bloques “step” que controlan la posición del vehículo en el tiempo.

Figura 33

Demultiplexor, con sus dos sumadores en la línea correspondiente a X y sus bloques step.

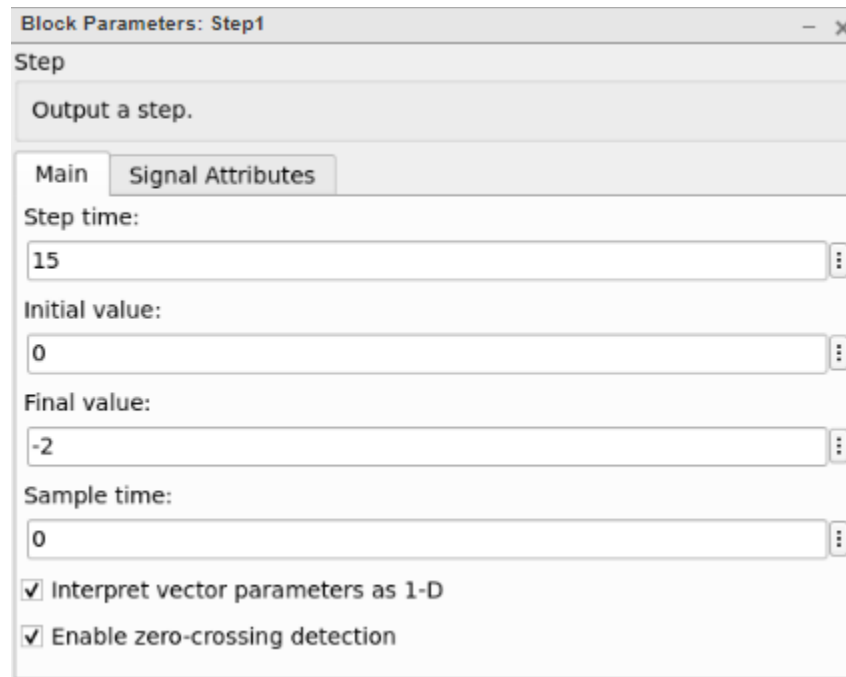


Nota. (LQR.slx)

En la Figura 34 se muestra la configuración del bloque “Step”, que son los parámetros para estos bloques, que se encuentran en la librería por defecto de simulink, se tiene el tiempo al que se da el step (Step time), que en el caso del ejemplo mostrado es a los 15 segundos, y el valor del step (Final Value), que en este caso es de -2, lo cual implica un movimiento de 2 unidades a la izquierda desde la posición en la que se encuentra el carro en ese instante.

Figura 34

Programación del bloque Step

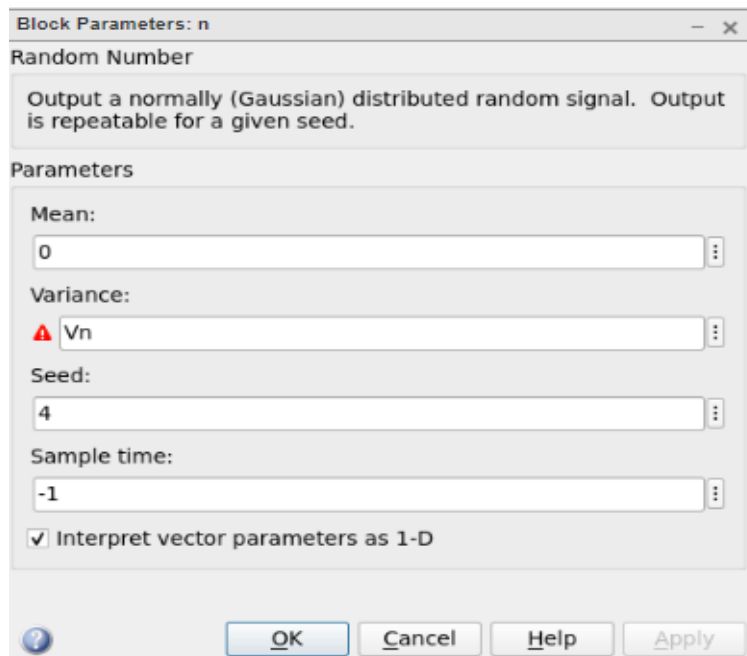


Nota. (LQR.slx)

La Figura 35 corresponde al otro sumador el cual permite añadir ruido al sistema, simulando un error en la lectura de la posición del vehículo.

Figura 35

Programación del bloque de ruido

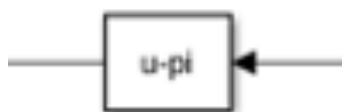


Nota. (LQR.slx)

El ruido deseado fue generado por medio de un bloque que genera números aleatorios (random number), disponible en la librería por defecto de simulink. En general para los bloques de ruido se ajustó el parámetro de varianza. Para los bloques de ruido en la medición se asignó la variable V_n y para los del error en el control del vehículo (señal de entrada al modelo) se asignó la variable V_d . Adicionalmente, se tiene un bloque de bias (Figura 36) cuya función es restar π a la línea correspondiente al ángulo del péndulo, para ajustar el punto de referencia del mismo.

Figura 36

Programación del bloque de bias

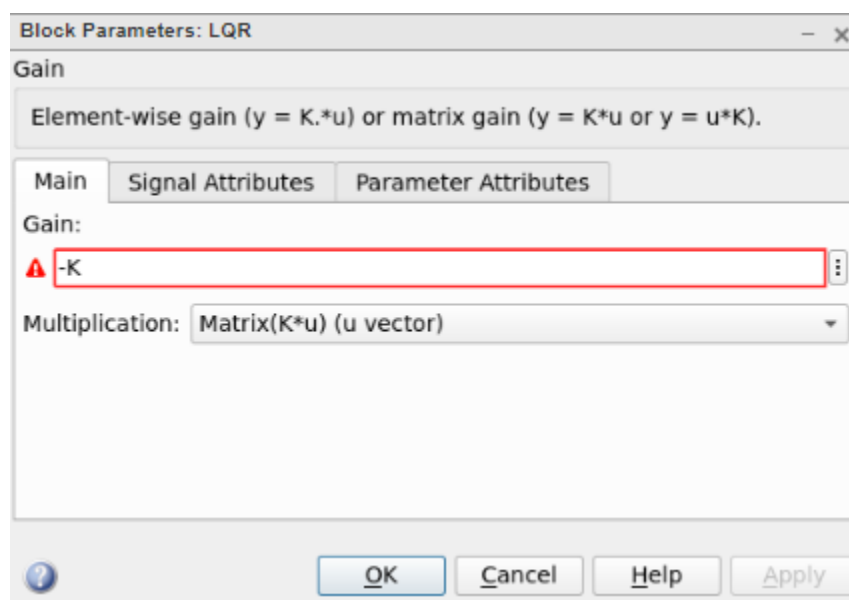


Nota.(LQR.slx)

Por último, en esta sección izquierda (Figura 31) se tiene un bloque multiplexor, que arma de nuevo el vector y otro bloque de la librería estándar de simulink (Gain) para aplicar una ganancia, que multiplica la señal por la ganancia K del controlador LQR diseñado, en la Figura 37 se encuentra la programación de dicho bloque.

Figura 37

Programación del bloque de "Gain"



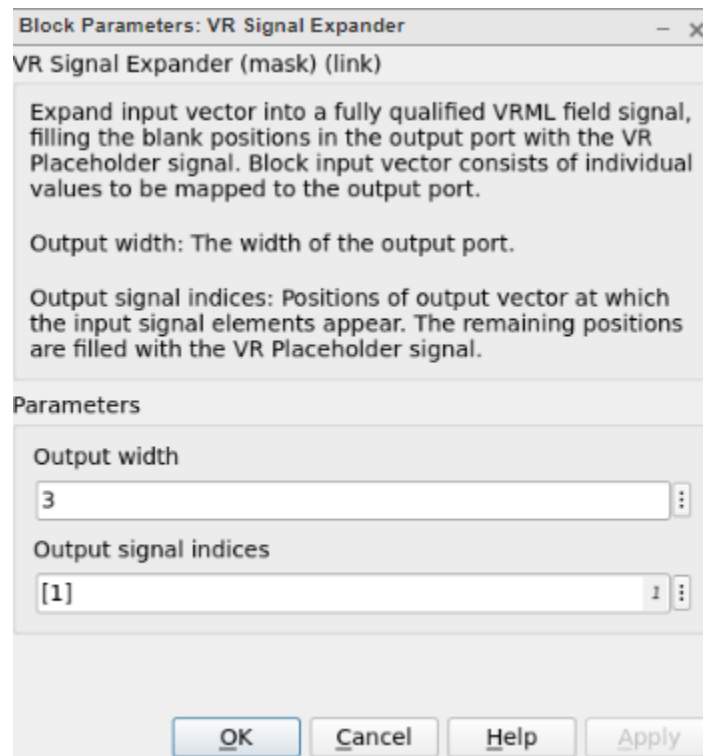
Nota. (LQR.slx)

En la parte derecha (Figura 32) se toma la salida del modelo y se procesa para su visualización. Para esto se realizó una demultiplexación de la señal y se mandó las salidas correspondientes a la posición del carro y ángulo del péndulo al visualizador. Para esto se tienen dos bloques: el primero VR signal Expander, correspondiente a la librería Simulink 3D Animation, etiquetados como VR, que cumple la tarea de ajustar las entradas a forma vectorial, de manera que

el modelo 3D diseñado pueda procesar dicha información de manera adecuada. La figura 38 corresponde a la configuración del bloque “VR signal Expander”.

Figura 38

Configuración del bloque de VR signal Expander

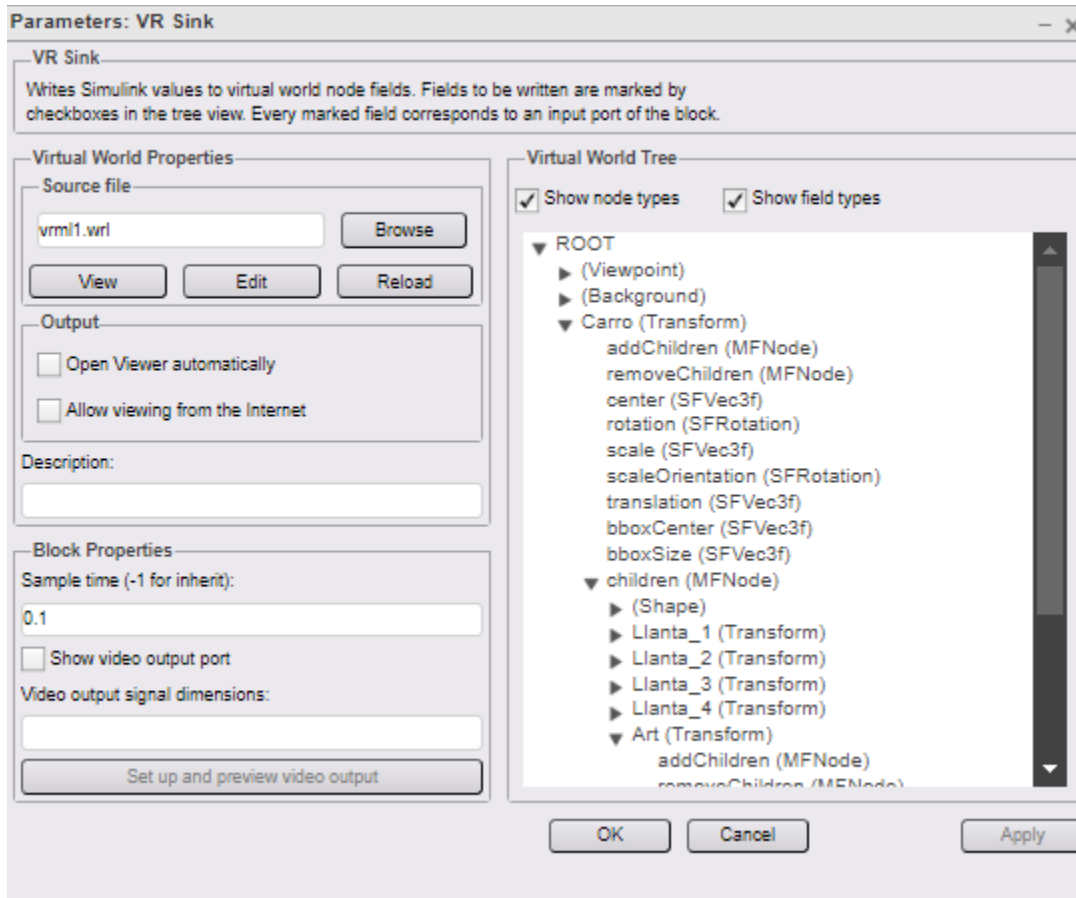


Nota. (LQR.slx)

El segundo es el bloque de realidad virtual VR Sink, tomado de la misma librería, donde se visualiza el modelo 3D. Este modelo fue creado por medio de la herramienta V-Realm, que permite crear un modelo en un entorno de realidad virtual por medio de objetos representados por nodos en determinada jerarquía. Una vez creado dicho modelo, se deben seleccionar los nodos que se quieren controlar en el modelo, lo cual se logra por medio del bloque VR Sink mencionado. La configuración de este último se muestra en la Figura 39.

Figura 39

Programación del bloque de VR Sink

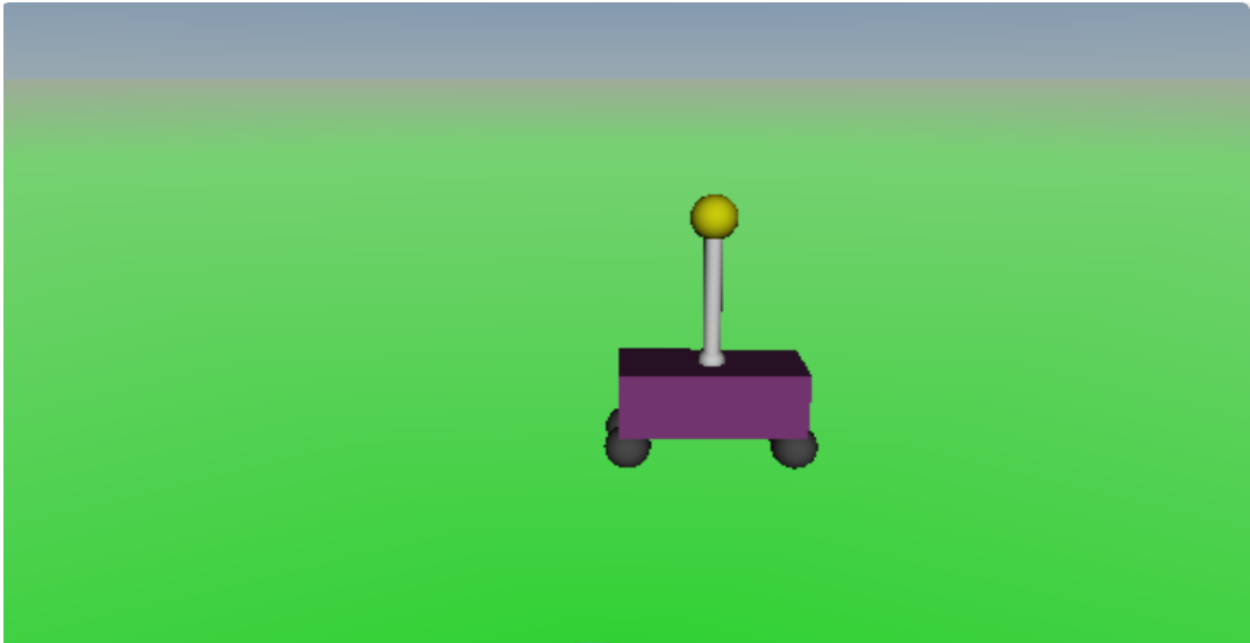


Nota. (LQR.slx)

A continuación, en la Figura 40 se muestra una imagen de la animación final conseguida por medio de esta herramienta.

Figura 40

Animación del péndulo invertido lineal



Nota. (LQR.slx)

3.7 Controlador LQG

La figura 41 muestra el diagrama diseñado en simulink para el controlador LQG aplicado al péndulo invertido.

Figura 41

Diagrama del controlador LQG en Simulink


```

1  clear all, close all, clc
2  m = 1;
3  M = 5;
4  L = 2;
5  g = -10;
6  d = 1;
7  s = 1; % pendulum up (s=1)
8  A = [0 1 0 0; 0 -d/M -(s*(m*g))/M 0; 0 0 0 1; 0 -s*d/(M*L) -(s*(m+M)*g)/(M*L) 0];
9  B = [0; 1/M; 0; s*1/(M*L)];
10 C = [1 0 0 0];
11 D = zeros(size(C,1),size(B,2));
12 Q = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1];
13 R = 0.0001;
14 K = lqr(A,B,Q,R);
15 Vdmag = .04;
16 Vd = Vdmag*eye(4);
17 Vn = .0002;
18 Kf = (lqr(A',C',Vd,Vn))';
19 sysKF = ss(A-Kf*C,[B Kf],eye(4),0*[B Kf]);

```

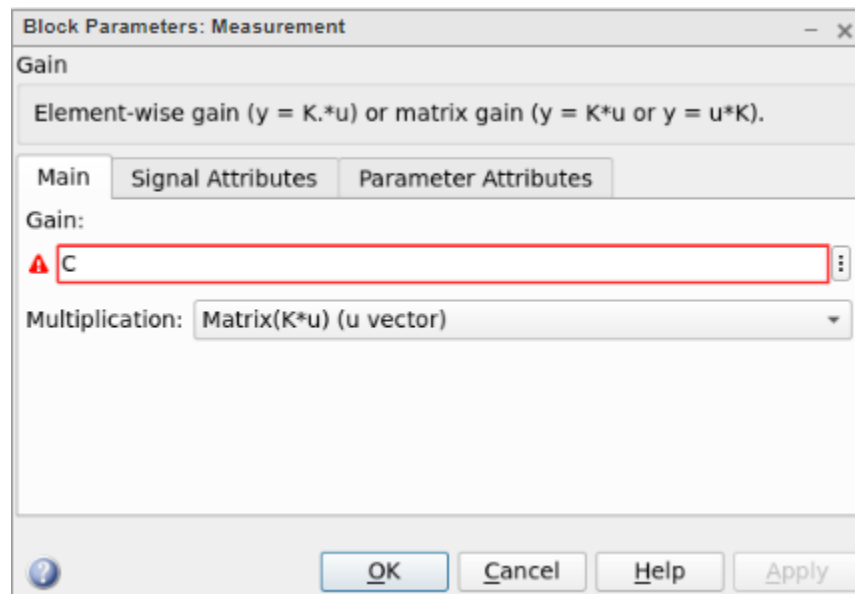
Nota. Ver apéndice A para ver el código del Script complementario (Inicializador.m)

En este modelo se agregan unos pocos elementos adicionales. En primer lugar, se tiene un modelo del péndulo con el carro que tiene una entrada adicional para el ruido, etiquetada como d. este ruido es generado por 4 bloques iguales al ya expuesto en el LQR en la Figura 35, con la precisión de que la variable utilizada para la varianza fue Vd.

Posteriormente, en la Figura 43 se tiene un bloque de ganancia (Gain) cuya función es filtrar la salida del modelo, de modo que solo se tengan los componentes de la señal que son observables.

Figura 43

Programación del bloque de “Gain”

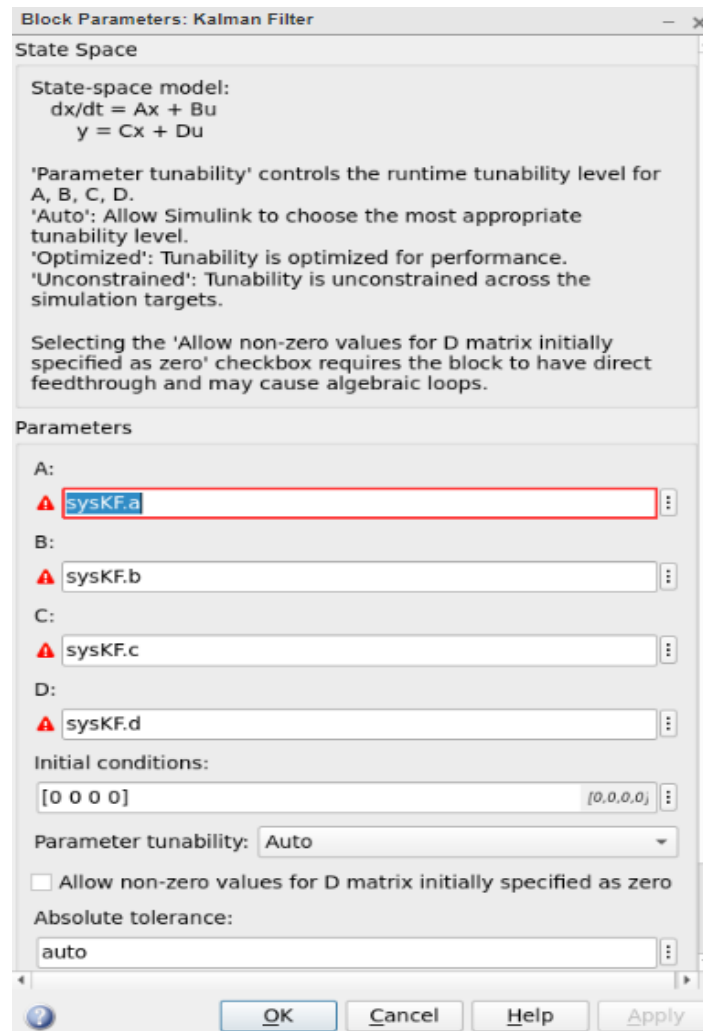


Nota. (LQG.slx).

Después tenemos la señal de ruido en la medición, que ya fue expuesto en el LQR (Figura 35). A continuación, tenemos el bloque del filtro de Kalman, que multiplica cada una de las componentes de la señal por la ganancia obtenida a partir del sistema sysKf obtenido a partir del diseño de filtro. Esto se logra por medio de un bloque de espacio de estados (State Space), mostrado en la Figura 44, cuyos parámetros ABCD corresponden a las componentes del ya mencionado sistema sysKf.

Figura 44

Programación del bloque de “State Space”



Nota. (LQG.slx).

El resto de bloques son iguales a los utilizados y explicados en el LQR.

cabe precisar, que en los bloques steps destinados a cambiar la posición, se modificó el tiempo en el que se dan los pasos. Esto se hizo con el fin de simplificar el proceso de simulación y así acortar el tiempo necesario para estar manteniendo el proceso fundamental que se busca recrear.

3.8 Elaboración de Videos explicativos

Se elaboraron 6 videos con el editor de video ClipChamp, en donde se estudió el péndulo

invertido como un sistema dinámico inestable, representado mediante el espacio de estados y en donde se aplicó técnicas de control óptimo tales como LQR y LQG. Esta herramienta audiovisual busca fortalecer en los estudiantes, los conceptos relacionados al control moderno y óptimo aplicados a un sistema dinámico. (Los videos se encuentran en el apéndice C).

3.8.1 Video 1

- Teoría de control
 - Control retroalimentado en caso cerrado
 - Sistemas lineales invariantes en el tiempo
 - Linealización de sistemas no lineales
 - Respuesta natural del sistema
 - Respuesta forzada del sistema
 - Sistemas de tiempo discreto

3.8.2 Video 2

- Ecuaciones de movimiento
 - Sistemas mecánicos con dos o más grados de libertad
 - Coordenadas generalizadas
 - Ecuaciones de movimiento de Lagrange
 - Lagrangiano
 - Ecuaciones de movimiento no lineal pendulo invertido

3.8.4 Video 3

- Espacio de estados
 - Variables de estado
 - Linealización

- Ecuaciones de movimiento linealizadas
- Ecuación de espacio de estados

3.8.4 Video 4

- Diseño de controlador
 - Control moderno
 - Controlabilidad
 - Controlado óptimo: LQR

4.8.5 Video 5

- Filtro de Kalman
 - Estimador óptimo: LQE – Filtro de Kalman
 - Observabilidad

4.8.6 Video 6

- Controlador Optimo Gaussiano (LQG)

4. Análisis de resultados

En este capítulo se analizan los resultados obtenidos de las simulaciones en Matlab y Simulink.

4.1 Matlab

4.1.1 Resultados LQR

Para realizar el análisis de resultados, se crea el código mostrado en la Figura 45 el cual es necesario para solucionar el sistema de ecuaciones diferenciales para el vector "tspan" , tomando las condiciones iniciales almacenadas en "x0" mediante el comando "ode45", comando empleado para solucionar sistemas de ecuaciones diferenciales de complejidad estándar en Matlab.

Figura 45

Código para solucionar el espacio de derivadas

```
[t,x]=ode45(@(t,x)pendcart(x,m,M,L,g,d,u(x)),tspan,x0);
```

Nota. El termino tspan está indicado en la Figura 18 y la función pendcart en la Figura 13.

(Metodologia.mlx)

En la Figura 46 se utiliza el comando "plot" para graficar la respuesta del sistema, al cual le definió unas propiedades estéticas para la gráfica, tales como leyenda, posición, nombre de los ejes y cuadrícula, logrando la visualización mostrada a continuación.

Figura 46

Código para graficar el sistema dinámico

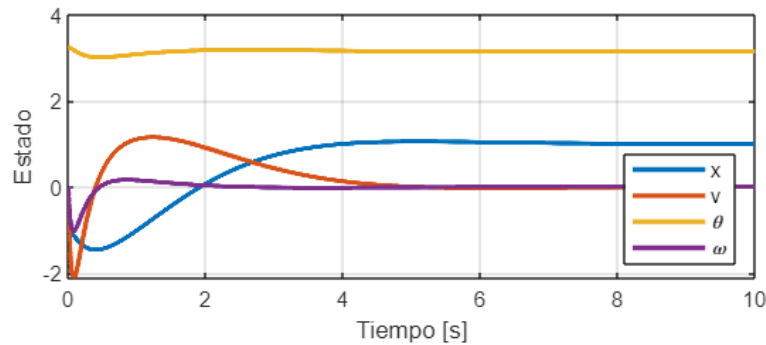
```
plot(t,x,'LineWidth',2);  
l1 = legend('x','v','\theta','\omega');  
set(l1,'Location','SouthEast')  
set(gcf,'Position',[100 100 500 200])  
set(gcf,'PaperPositionMode','auto')  
xlabel('Tiempo [s]')  
ylabel('Estado')  
grid on
```

Nota. (Metodología.mlx)

La Figura 47 corresponde a la gráfica generada por los códigos de la figura 45 y 46.

Figura 47

Grafica del sistema dinámico, estados vs tiempo.



Nota. (Metodologia.mlx)

Aprovechando el mismo comando es posible obtener gráficas separadas para cada uno de los estados almacenados en el vector de estados "x", lo cual resulta útil para puesto que las magnitudes individuales pueden apreciarse de forma mucho más clara, nos apoyamos en una matriz de color en código RGB llamada "CC", para permitir que coincidan los colores de la gráfica conjunta y las gráficas individuales. En la Figura 48 se muestra el código creado para generar la primera grafica separada (Figura 49) correspondiente a la posición del carro.

Figura 48

Código para generar grafica de la posición del carro

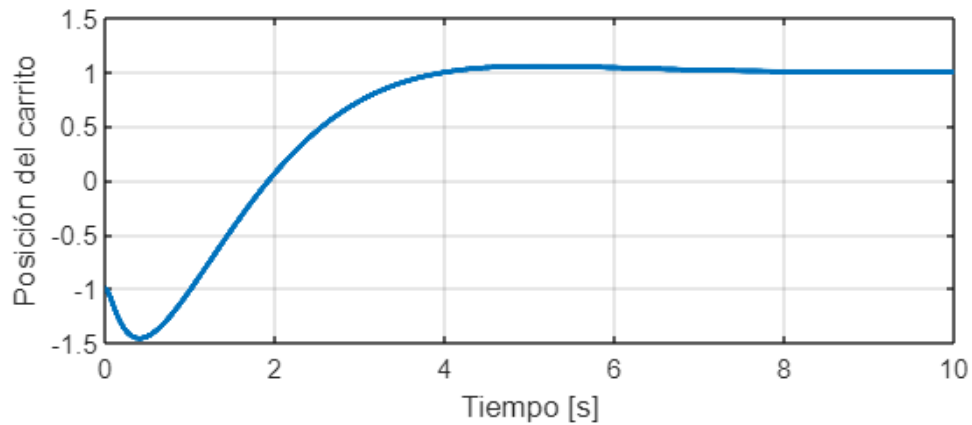
```
CC = [0    0.4470    0.7410
      0.8500    0.3250    0.0980
      0.9290    0.6940    0.1250
      0.4940    0.1840    0.5560];

plot(t,x(:,1),'LineWidth',2,'Color',CC(1,:));
set(gcf,'Position',[100 100 500 200])
set(gcf,'PaperPositionMode','auto')
xlabel('Tiempo [s]')
ylabel('Posición del carrito')
grid on
```

Nota. (Metodologia.mlx)

Figura 49

Grafica de la posición del carro vs tiempo



Nota. (Metodologia.mlx)

Para la primera variable de estado, la posición del carrito, se observa un subpico de 0.5 metros los cuales se atribuyen al intento del controlador por mantener la posición angular del péndulo, puesto que las condiciones iniciales indican que el sistema arranca en -1 y la referencia está en 1, se puede ver como luego del subpico se acerca rápidamente a la referencia con un tiempo de establecimiento de 4 segundos.

En la Figura 50 se muestra el código creado para generar la primera grafica separada (Figura 51) correspondiente a la velocidad del carro.

Figura 50

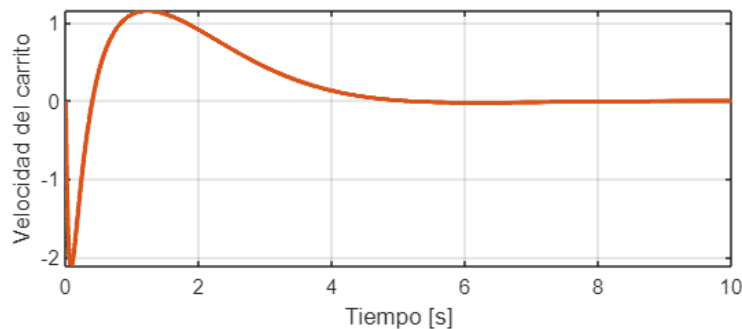
Código para generar grafica de la velocidad del carro

```
plot(t,x(:,2),'LineWidth',2,'Color',CC(2,:));  
set(gcf,'Position',[100 100 500 200])  
set(gcf,'PaperPositionMode','auto')  
xlabel('Tiempo [s]')  
ylabel('Velocidad del carrito')  
grid on
```

Nota. (Metodologia.mlx)

Figura 51

Grafica de la velocidad del carro vs tiempo



Nota. (Metodologia.mlx)

La segunda variable de estado está directamente relacionada con la primera pues es su derivada, la velocidad del carrito inicialmente es negativa pues el desplazamiento es contrario al eje positivo en el sistema de referencia establecido, la magnitud de la velocidad es negativa durante el sub pico en la posición del carrito, una vez el carrito vuelve a las condiciones iniciales se observa que la magnitud de la velocidad pasa a ser positiva con un pico máximo durante la pendiente más pronunciada de la posición del carrito, luego se acerca suavemente al 0 debido a que la posición ha alcanzado la referencia.

En la Figura 52 se muestra el código creado para generar la primera grafica separada (Figura 53) correspondiente a la posición angular del péndulo.

Figura 52

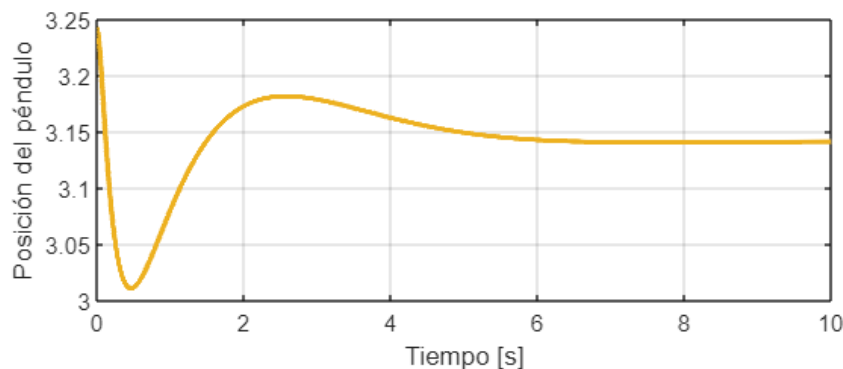
Código para generar grafica de la posición del péndulo

```
plot(t,x(:,3),'LineWidth',2,'Color',CC(3,:));  
set(gcf,'Position',[100 100 500 200])  
set(gcf,'PaperPositionMode','auto')  
xlabel('Tiempo [s]')  
ylabel('Posición del péndulo')  
grid on
```

Nota. (Metodologia.mlx)

Figura 53

Grafica de la posición angular del péndulo vs tiempo



Nota. (Metodologia.mlx)

La tercera variable de estado es la posición angular del péndulo, la referencia del sistema se encuentra en π radianes (180°) y la condición inicial difiere en 0.1 radianes (5.73°), se observa como el sistema procura corregir el error durante un sub pico con decrecimiento que deja de crecer

en el momento que la dirección del carrito cambia, luego supera brevemente la referencia y se acerca suavemente, comportamiento similar a la posición lineal del carrito.

En la Figura 54 se muestra el código creado para generar la primera grafica separada (Figura 55) correspondiente a la velocidad angular del péndulo.

Figura 54

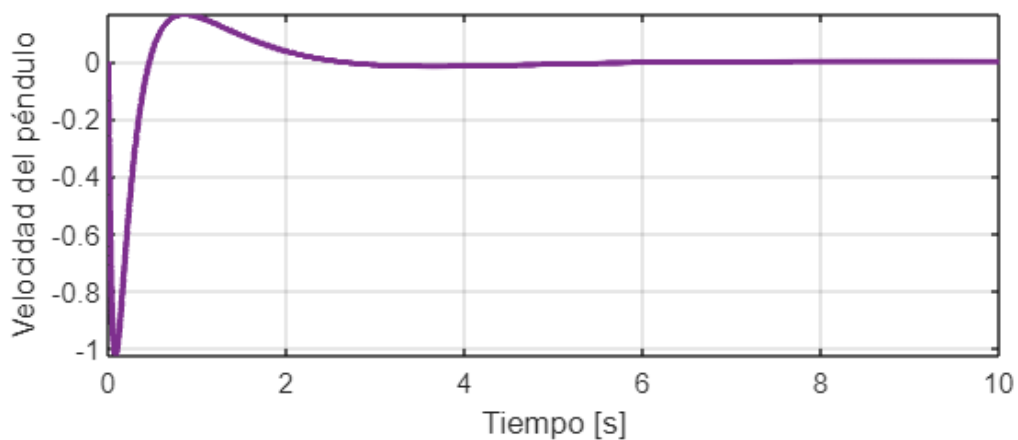
Código para generar grafica de la velocidad angular del péndulo

```
plot(t,x(:,4),'LineWidth',2,'Color',CC(4,:));  
set(gcf,'Position',[100 100 500 200])  
set(gcf,'PaperPositionMode','auto')  
xlabel('Tiempo [s]')  
ylabel('Velocidad del péndulo')  
grid on
```

Nota. (Metodologia.mlx)

Figura 55

Grafica de la posición de la velocidad angular del péndulo vs tiempo



Nota. (Metodologia.mlx)

Las dos primeras variables de estado hacen referencia al carrito mientras que las dos últimas al péndulo, cada par de variables de estado presentan un comportamiento similar pues tienen una estrecha relación entre sí, así como sucede con la segunda variable de estado, la cuarta está directamente relacionada a la tercera mediante su derivada, se puede ver como inicialmente la velocidad angular del péndulo es negativa con un decrecimiento de corta duración, el tiempo que la cuarta variable de estado tiene magnitud negativa coincide con el del punto mínimo del sub pico para la tercera, de igual manera vemos como la duración de magnitud positiva de la cuarta variable de estado, coincide con la transición entre el sub pico y el sobre pico de la tercera variable de estado hasta hacerse muy pequeña y cercana al cero luego de este.

4.1.2 Comparación de las técnicas de control utilizadas

Los resultados obtenidos en Matlab serán analizados mediante la comparación de la respuesta para un impulso positivo y negativo aplicado en $t = 1$ y $t = 15$ respectivamente para los 3 sistemas a estudiar, LQR completamente estimado, LQR con mediciones ruidosas y LQG. Puesto que la intención del filtro de Kalman es corregir las inconsistencias del algoritmo LQR asociadas al ruido con la menor cantidad de sensores, se espera que el comportamiento de los sistemas LQG empleando un sensor para la posición del carro y el LQR completamente estimado sea similar.

Se crean los modelos en variables de estado para los 3 casos y un vector de tiempo discreto que establece los puntos donde se toman datos discretos para nuestra simulación, la cual emplea un tiempo de muestreo de 0.01 segundos durante 50 segundos y es obtenida mediante el comando "lsim", se introducen perturbaciones con magnitud de 2 y se grafica la posición del carro para los 3 sistemas. En la Figura 56 se muestra el código creado para realizar la comparación gráfica, la cual se observa en la Figura 57.

Figura 56

Código para generar grafica comparativa de la posición del carro

```
D = zeros(size(C,1),size(B,2));

sysC=ss(A,B_aug,C,D_aug);
sysTruth=ss(A,B_aug,eye(4),zeros(4,size(B_aug,2)));
sysKF=ss(A-Kf*C,[B Kf],eye(4),0*[B Kf]);

dt=0.01;
```

```
t=0:dt:50;

uDIST=sqrt(Vd)*randn(4,size(t,2));
uNOISE=sqrt(Vn)*randn(size(t));
u=0*t;
u(1/dt)=2/dt;
u(15/dt)=-2/dt;

u_aug=[u; uDIST; uNOISE];

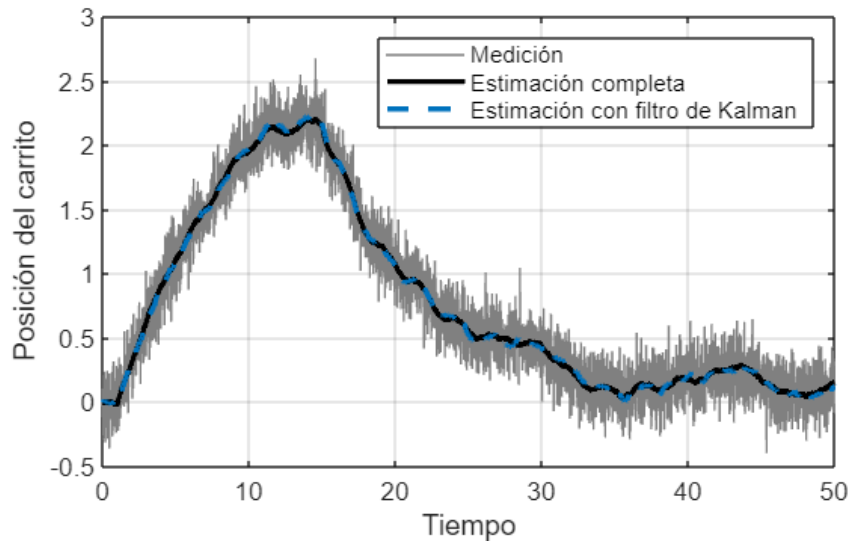
[y,t]=lsim(sysC,u_aug,t);
[xtrue,t]=lsim(sysTruth,u_aug,t);
[xhat,t]=lsim(sysKF,[u; y'],t);
```

```
plot(t,y,'Color',[.5 .5 .5]), hold on
plot(t,xtrue(:,1),'Color',[0 0 0],'LineWidth',2)
plot(t,xhat(:,1),'--','Color',CC(1,:), 'LineWidth',2)
set(gcf,'Position',[100 100 500 300])
xlabel('Tiempo')
ylabel('Posición del carrito')
l1 = legend('Medición','Estimación completa','Estimación con filtro de
Kalman');
set(l1,'Location','NorthEast')
grid on
set(gcf,'PaperPositionMode','auto')
hold off
```

Nota. (Metodologia.mlx)

Figura 57

Grafica comparativa de la posición del carro entre LQR estimado completamente, LQR con medición de ruido y LQG (LQR estimado con filtro de Kalman)



Nota. (Metodologia.mlx)

De la Figura 57 podemos confirmar que para la primer variable de estado, el comportamiento del sistema controlado mediante el algoritmo LQR completamente estimado y aquel controlado mediante el algoritmo LQR complementado con un observador tipo filtro de Kalman, presentan un comportamiento similar, permitiendo afirmar que su implementación, elimina la problemática del ruido en los sensores, por tanto procedemos a realizar la comparación entre estos dos sistemas para la tercera variable de estado, la posición del péndulo invertido, la cual resulta ser nuestra variable de mayor interés. Esta comparación se muestra en la Figura. 58.

Figura 58

Código y grafica generada para comparar la posición angular entre el LQR y LQG

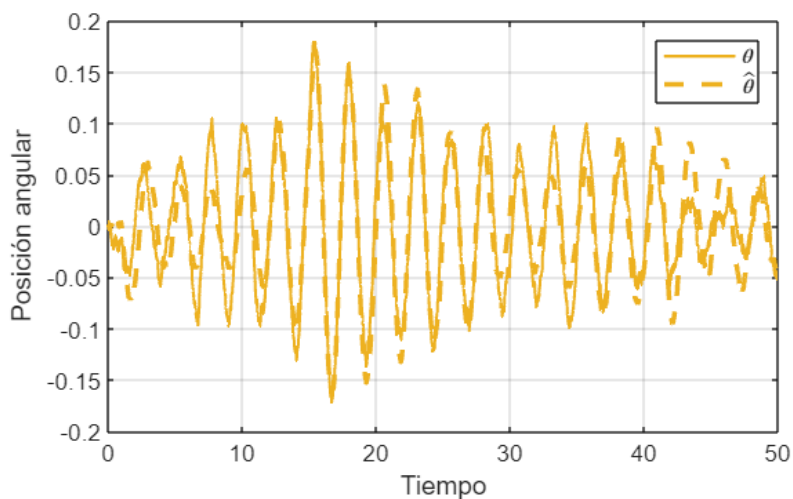
```

plot(t,xtrue(:,3),'-','LineWidth',1.2,'Color',CC(3,:));
hold on
plot(t,xhat(:,3),'--','LineWidth',2,'Color',CC(3,:))

set(gcf,'Position',[100 100 500 300])
xlabel('Tiempo')
ylabel('Posición angular')
l1 = legend('$\theta$', '$\hat{\theta}$');

set(l1,'Location','NorthEast','interpreter','latex')
grid on
set(gcf,'PaperPositionMode','auto')
hold off

```



Nota. (Metodologia.mlx)

La gráfica anterior nos permite interpretar el comportamiento de los sistemas, el cuál aunque es bastante similar, presenta ligeras diferencias en algunos puntos, como por ejemplo alrededor del segundo 10 y el segundo 43, donde el comportamiento del controlador LQR y LQG llega a diferir en valores cercanos a 0.05 radianes (2.86°), pero aun así el desempeño de ambos controladores mantiene esta variable naturalmente inestable controlada con un error menor a 0.2 radianes (11.45°). Al visualizar el comportamiento de las 4 variables de estado para los sistemas LQR y LQG obtenemos el código y grafica mostrado en la Figura 59.

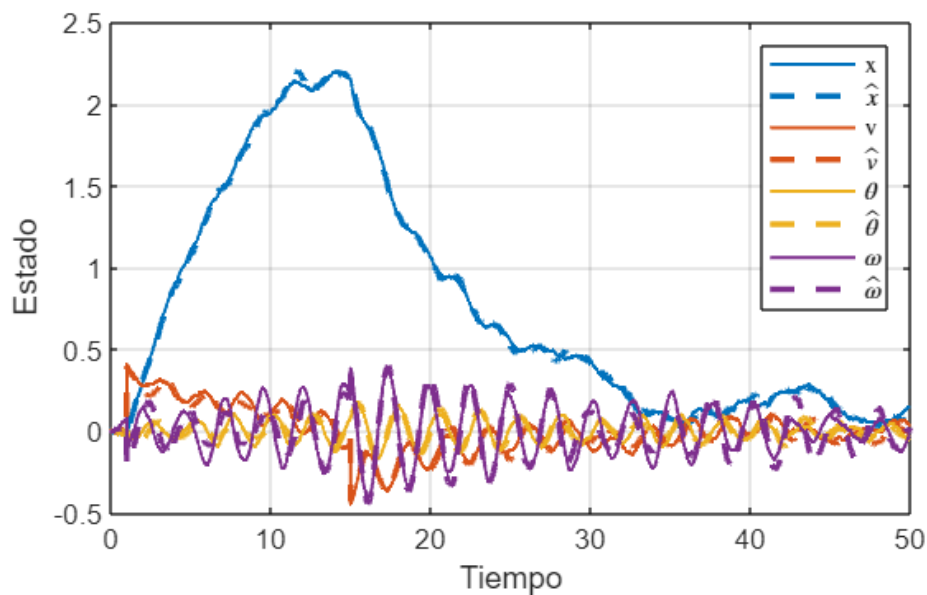
Figura 59

Código y grafica generada para comparar las cuatro variables de estado del sistema entre el LQR y LQG

```
for k=1:4
plot(t,xtrue(:,k),'-','LineWidth',1.2,'Color',CC(k,:));
hold on
plot(t,xhat(:,k),'--','LineWidth',2,'Color',CC(k,:))
end
set(gcf,'Position',[100 100 500 300])
xlabel('Tiempo')
```

```
ylabel('Estado')
l1 =
legend('x',' $\hat{x}$ ','v',' $\hat{v}$ ',' $\theta$ ',' $\hat{\theta}$ ',' $\omega$ ',' $\hat{\omega}$ ');

set(l1,'Location','NorthEast','interpreter','latex')
grid on
set(gcf,'PaperPositionMode','auto')
hold off
```



Nota. (Metodologia.mlx)

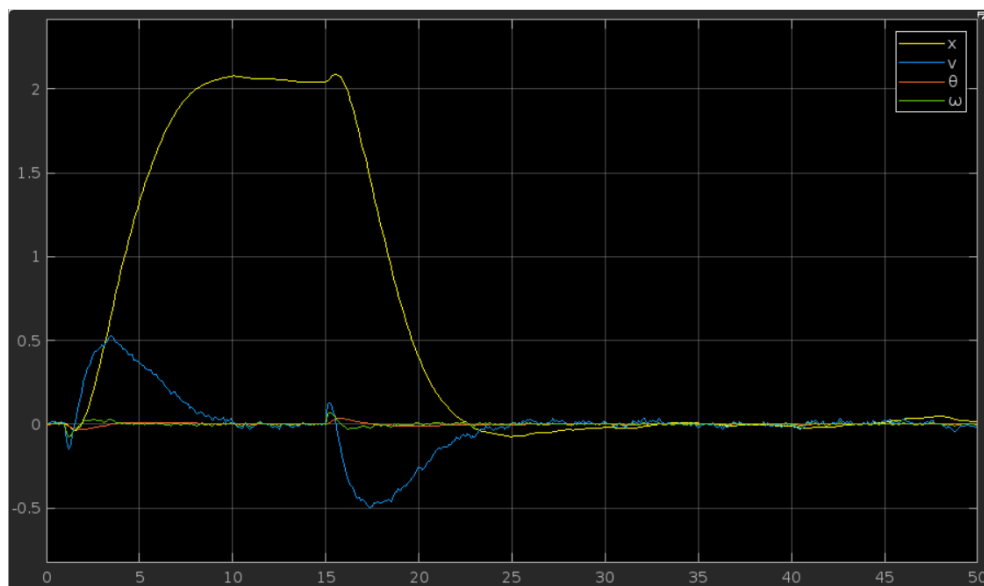
4.2 Simulink

4.2.1 Resultados LQR

Para el análisis de resultado en los diagramas creados en simulink, se utiliza el bloque Scope que permite visualizar las señales que entran en él. Alimentándolo con las señales de salida, para el caso LQR se obtiene una respuesta en el scope similar a la obtenida en Matlab pero se puede notar que la magnitud de las perturbaciones se vuelve menos significativa, teniendo en cuenta que se emplean 4 sensores. En la Figura 60 se observa la gráfica que aparece en el scope del diagrama LQR (Figura 30).

Figura 60

Grafica scope LQR



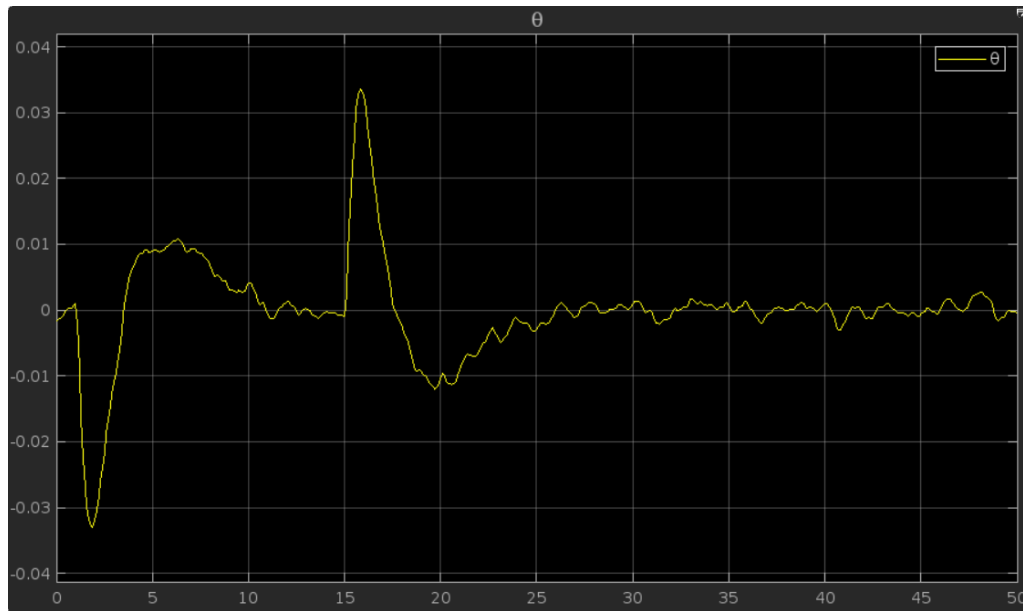
Nota (LQR.slx)

Analizando únicamente la tercera variable de estado, en la Figura 60, se evidencia que el comportamiento del sistema resulta más preciso que el encontrado en Matlab, en el caso que se

emplean 4 sensores, cada uno asignado a su respectiva variable de estado. Este análisis de la tercera variable de estado se observa en la Figura 61.

Figura 61

Grafica scope de la posición angular

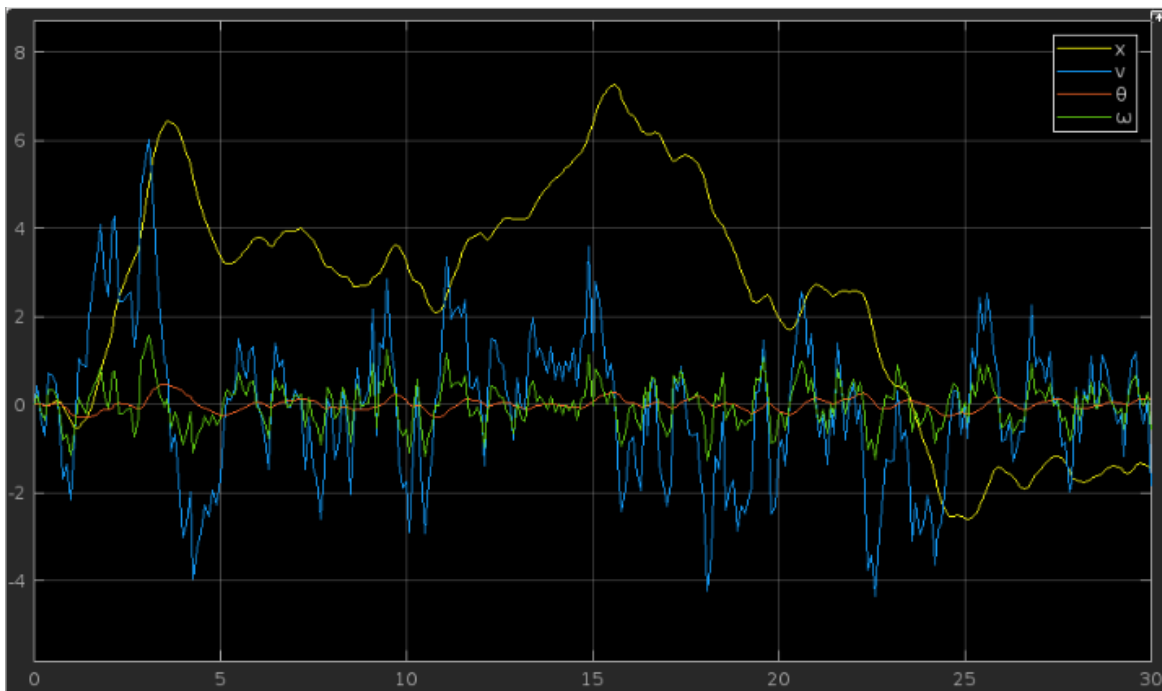


Nota (LQR.slx)

El modelo muestra ser bastante preciso en caso de que se esté tratando con un sistema completamente estimado sin ruido, en el caso donde tenemos en cuenta un ruido significativo, vemos que el control puede salirse de su margen operacional llegando a la inestabilidad como se ve a continuación en la Figura 62.

Figura 62

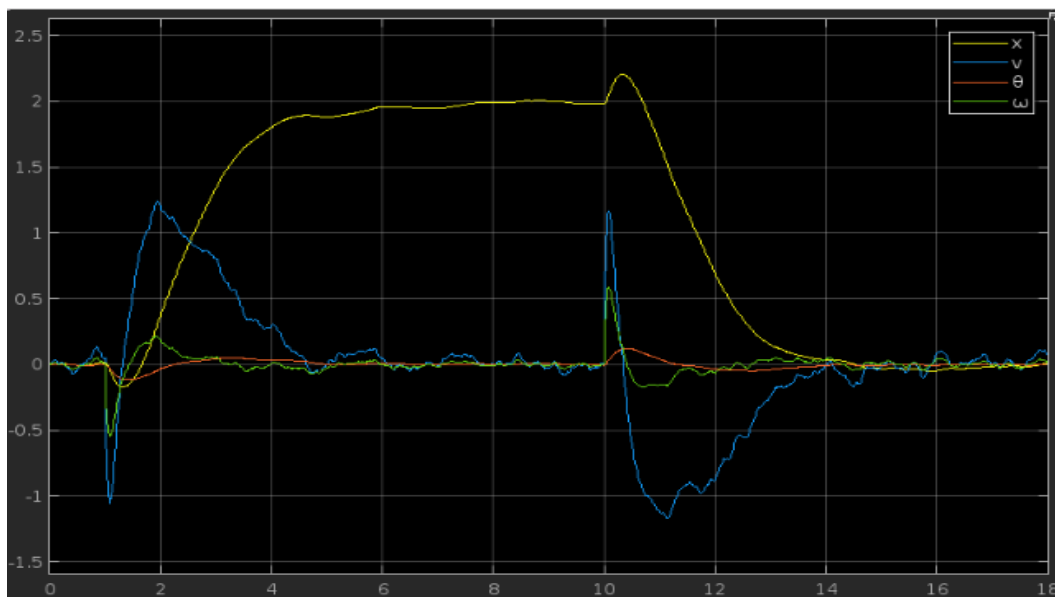
Grafica scope de la estimación de ruido en el controlador LQR



5.2.2 Resultados LQG

Figura 63

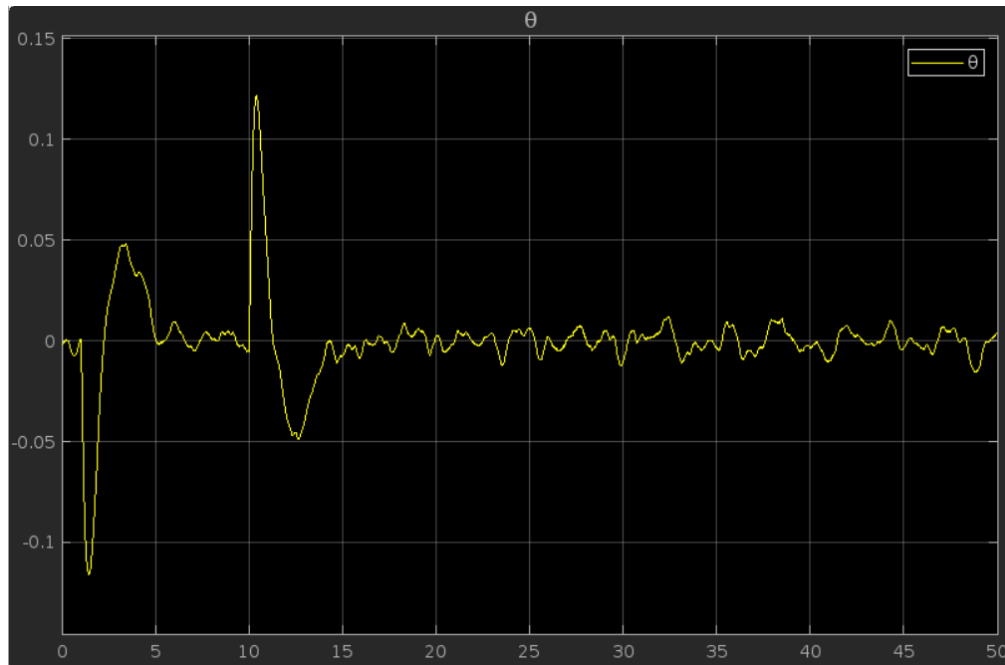
Grafica scope LQG



Si se alimenta únicamente la señal del ángulo del péndulo en el scope se obtiene la siguiente gráfica mostrada en la Figura 64.

Figura 64

Gráfica scope LQG posición angular



Se puede notar entonces que si bien la respuesta tiene mayor margen de error, siendo el error obtenido menor a 0.15 radianes (8.59°), el filtro de Kalman permitió ahorrar 3 sensores, por tanto ahora se logró diseñar un controlador que logra controlar el sistema en su posición inestable a partir de la implementación de un único sensor para la posición del carrito, ventaja que permite ahorrar costos de fabricación e implementación para aplicaciones reales.

5. Conclusiones

- Se generó un material académico que contribuirá con el desarrollo de las clases de sistemas dinámicos e ingeniería de control, ya que se trata un ejercicio clásico en teoría de control, el péndulo invertido, en donde se aplica la teoría de control moderno y además consiste en un sistema al cual se le aplican diversas técnicas de control para comprobar si dicha técnica es aplicable a sistemas más complejos, por lo tanto el tratamiento de este sistema dinámico sirve como una herramienta valiosa a la hora de aprender los conceptos teóricos de dichas materias.
- Si bien se presume que el controlador LQR es robusto, este es débil ante la incertidumbre del modelo, por tanto, una inadecuada identificación de la planta a controlar puede terminar en un pobre desempeño del controlador.
- Al ser el péndulo invertido un sistema de control de una sola entrada y dos salidas, la mejor forma de representar este sistema dinámico es el espacio de estados, ya que como modelo matemático contempla múltiples salidas y múltiples entradas.
- Mediante la comparación realizada entre los controladores LQR y LQG (Figura 57), se observa cómo en la gráfica se comportan de forma similar, lo cual permite concluir que la implementación del filtro de Kalman como observador, soluciona la presencia de ruido en los sensores del sistema, además de optimizar costos al reducir la cantidad de variables de estado que deben ser medidas directamente, es decir el número de sensores necesarios para estabilizar el péndulo.

Referencias

- Castaños (2003) “*Levantamiento y estabilización del péndulo invertido*” de la Universidad Nacional autónoma
- Hernandez, Legaspi, Pelaez (2012) “*Control inteligente del péndulo invertido*” de la Universidad Complutense de Madrid.
- <https://www.semana.com/tecnologia/articulo/cuales-son-los-retos-que-debe-superar-colombia-ante-la-cuarta-revolucion-industrial-1001/280911/>
- Montoya, Valenzuela, Giraldo (2013) "Balanceo y estabilización del péndulo invertido empleando redes neuronales artificiales y un regulador lineal óptimo con criterio cuadrático (LQR)" publicado por la revista Scientia et Technica.
- RAMÍREZ-GONZÁLEZ, Luis^{1†}, GARCÍA-MARTÍNEZ, José², SIORDIA-VÁSQUEZ y Xóchitl¹, GARCIA-RAMOS (Diciembre 2017) “Método de Euler-Lagrange en el modelado y control de un péndulo invertido sobre un carro”, Roman¹ Revista de Ingeniería Eléctrica Vol.1 No.3, 1-8
- Romero, Marin, Jimenez (2013) "Modelado, simulación e implementación de controladores LQR y RLVE al sistema péndulo invertido rotacional usando la plataforma NI ELVIS II" publicado por la Revista Científica Guillermo de Ockham.
- Steve L Brunton – J Nathan Kutz (2019, 15 de Febrero) Data Driven Science & Engineering *Machine Learning, Dynamical Systems, and Control*