



Universidad Michoacana de San Nicolás de Hidalgo
Facultad de Ingeniería Eléctrica
Noviembre del 2013

**ANÁLISIS Y DESARROLLO DE PRUEBAS DEL
SIMULADOR DE SISTEMAS DINÁMICOS 20-SIM**

TESIS

**Para obtener el título de
INGENIERO EN COMPUTACIÓN**

**Presenta
JULIO EDGAR PÉREZ PAQUE**

**Asesor de Tesis
Dr. en Ingeniería Eléctrica
GILBERTO GONZÁLEZ AVALOS**

OCTUBRE 2013

Agradecimientos

Agradezco a los profesores de esta gran institución la “Facultad de Ingeniería Eléctrica” en especial a todos los profesores del área de computación por compartir sus conocimientos y consejos durante estos cinco años de grandes esfuerzos y sacrificios, los cuales sin duda los aplicaré en toda mi vida profesional y por supuesto a mi asesor el Dr. Gilberto González Avalos por sus atenciones y sus opiniones que me hizo a lo largo del presente trabajo; también quiero agradecer a mis padres Julio Pérez Ortiz y Gloria Paque Mendoza que fueron de gran apoyo moral y económico para continuar con mis estudios universitarios, a mi hermano Omar Pérez Paque y amigos de la generación 2005-2010 en particular a Roberto Rangel Heras, que juntos vivimos grandes experiencias en la universidad así como también quiero agradecer grandes personas, que son parte de mi vida María Guadalupe Torres, Víctor Madrigal Sosa, Ana Isela Rodríguez, Griselda Silva Melgarejo, Leonardo Daniel Silva Rodríguez, Fidel Isidro Ruiz, Rodrigo Ranyart Suarez Ponce de Leon, Yerezi Palma Marcelino, Yovvana Muñoz Constantino, Ludwing Roberto Valdez Aparicio, Dulce María Suarez Aparicio, Carmen Cortes Hernández y Janet Maricela Gonzales Constantino.

Dedicatoria

Dedico este proyecto a mi familia, amistades, las cuales me ayudaron con su apoyo incondicional a ampliar mis conocimientos y estar más cerca de mis metas profesionales. Gracias a los intercambios y exposiciones de ideas con mis compañeros y amigos de estudios durante el proceso de la licenciatura. También quiero dedicar esta tesis a Janet Maricela González Constantino que desde que la conozco ha sido mi apoyo incondicional en mi vida personal y profesional para salir adelante siempre en los buenos o malos momentos.

Resumen

En este trabajo se realiza una descripción detallada del software de simulación de sistemas dinámicos (motores eléctricos, manipuladores mecánicos, actuadores hidráulicos, circuitos de control por ordenador, etc.), 20 SIM, partiendo de una breve descripción del concepto de la simulación, su importancia y cuáles son sus ventajas en el ramo de la ingeniería. Una vez presentados estos puntos, se procede a describir sobre los antecedentes de la simulación, la evolución de los programas de simulación de sistemas dinámicos y en especial el de 20 SIM.

Una vez descrita la historia de la simulación y el software, se pasa a describirlo, las herramientas que se pueden utilizar, las formas en que podemos representar un sistema dinámico y los casos de estudio en que se puede utilizar.

Ya explicada la manera en que se maneja el software, se tomaran tres modelos físicos de los cuales se harán dos simulaciones diferentes de cada modelo para así, conocer la precisión que tiene una de la otra. Por último, se darán conclusiones de dichas pruebas y los posibles cambios que pudiera obtener dicho software, para mejorar el rendimiento del programa en el campo de la simulación de sistemas dinámicos.

Contenido

Agradecimientos.....	ii
Dedicatoria.....	iii
Resumen.....	iv
Contenido.....	v
Lista de Figuras.....	viii
Lista de Símbolos y Abreviaciones.....	x
 Capítulo 1. Introducción.....	 1
1.1. La importancia de la simulación.....	1
1.2. Objetivo.....	2
1.3. Justificación.....	2
1.4. Organigrama y estructura de la tesis.....	3
 Capítulo 2. Antecedentes de la simulación y la evolución de los programas de sistemas dinámicos.....	 5
2.1. Historia de la simulación.....	5
2.1.1. Historia de la simulación en la antigüedad.....	5
2.1.2. Periodo de formación (1945-1970).....	6
2.1.3. El arte de la simulación.....	6
2.1.4 Sistema de Simulación de Propósito General.....	7
2.1.5. Simula I y WSC.....	7
2.1.6. Periodo de Expansión de la simulación.....	7
2.1.7. Evolución de la Simulación.....	8
2.1.8. El concepto de inmersión.....	9
2.1.9. Modelos Matemáticos.....	9
2.2. Evolución de los programas de sistemas dinámicos.....	9

2.3. Evolución de 20-SIM.....	13
Capítulo 3.Descripción del software 20-SIM.....	16
3.1. Introducción.....	16
3.2. Modelado.....	17
3.2.1. Cuadro de controles.....	17
3.2.2. Herramientas de mecatrónica.....	19
3.2.3. Herramientas mecánicas en 3D.....	23
3.3. Simulación.....	24
3.3.1. Simulador.....	24
3.3.2. Animación 3D.....	26
3.4. Análisis.....	27
3.4.1. Herramientas en el dominio del tiempo.....	27
3.4.2. Herramientas en el dominio de la frecuencia.....	29
3.5. Exportar.....	31
3.5.1 Herramientas de generación de código.....	31
3.6 Demostración del estudio de 20 SIM.....	32
Capítulo 4. Casos de estudios para 20 SIM	40
4.1. Introducción.....	40
4.2. Primer caso de estudio.....	40
4.2.1. Diagrama de bloques para el primer caso de estudio.....	42
4.2.2. Diagrama de ecuaciones para el primer caso de estudio.....	43
4.2.3. Comparación de resultados del primer caso de estudio.....	44
4.3. Segundo caso de estudio.....	45
4.3.1. Diagrama de bloques para el segundo caso de estudio.....	46
4.3.2. Diagrama de ecuaciones para el segundo caso de estudio.....	48
4.3.3. Comparación de resultados del segundo sistema.....	49
4.4 Tercer caso de estudio.....	50
4.4.1. Diagrama de bloques para el tercer caso de estudio.....	51

4.4.2. Diagrama de ecuaciones para el tercer caso de estudio.....	52
4.4.3. Comparación de resultados del tercer caso.....	53
Capítulo 5. Conclusiones y recomendaciones.....	55
5.1. Conclusiones.....	55
5.2. Recomendaciones.....	56
Referencias.....	57

Lista de Figuras

Figura 3.1 Cuadro de controles.....	18
Figura 3.2 Adaptación del controlador.....	18
Figura 3.3 Gráfica de máximos.....	20
Figura 3.4 Representaciones 3D.....	23
Figura 3.5 Animación en 3D.....	26
Figura 3.6 Barrido de parámetros.....	28
Figura 3.7 Editor de sistema lineal.....	30
Figura 3.8 Área de trabajo 2D SIM.....	33
Figura 3.9 Listado de bloques.....	34
Figura 3.10 Orden de los bloques.....	34
Figura 3.11 Editor del código de bloque.....	35
Figura 3.12 Conexión de los bloques.....	35
Figura 3.13 Área de Gráficas.....	36
Figura 3.14 Gráfica del modelado.....	36
Figura 3.15 Área de trabajo para el modelado de ecuaciones.....	37
Figura 3.16 Plantilla para ecuaciones de Matrix.....	37
Figura 3.17 Plantilla para ecuaciones de Matrix modificada.....	38
Figura 3.18 Gráfica del modelado de ecuaciones.....	38
Figura 3.19 Comparación de la gráficas.....	39
Figura 3.20 Comparación del resultado de las gráficas.....	39
Figura 4.1 Primer sistema.....	40
Figura 4.2 Diagrama de bloques del primer caso de estudio.....	42
Figura 4.3 Gráfica del primer caso de estudio en base al diagrama de bloque.....	43
Figura 4.4 Diagrama de ecuaciones para el primer caso de estudio.....	43
Figura 4.5 Gráfica del primer caso de estudio en base al diagrama de ecuaciones.....	44
Figura 4.6 Comparación de gráficas de los modelados del primer caso de estudio.....	44
Figura 4.7 Comparación de gráficas de los modelados en base a datos numéricos.....	45

Figura 4.8 Segundo sistema	45
Figura 4.9 Diagrama de bloques del segundo caso de estudio.....	47
Figura 4.10 Gráfica del segundo caso de estudio en base al diagrama de bloques.....	47
Figura 4.11 Diagrama de ecuaciones del segundo caso de estudio.....	48
Figura 4.12 Gráfica del segundo caso de estudio en base al diagrama de ecuaciones...	48
Figura 4.13 Comparación de gráficas de los modelados del segundo sistema.....	49
Figura 4.14 Comparación de gráficas de los modelados del segundo caso en base a datos numéricos.....	49
Figura 4.15 Tercer sistema.....	50
Figura 4.16 Diagrama de bloques del tercer caso de estudio.....	51
Figura 4.17 Grafica del tercer caso de estudio en base al diagrama de bloques.....	52
Figura 4.18 Diagrama de ecuaciones para el tercer caso de estudio.....	52
Figura 4.19 Gráfica del tercer caso de estudio en base al diagrama de ecuaciones.....	53
Figura 4.20 Comparación de gráficas de los modelados del tercer sistema.....	53
Figura 4.21 Comparación de gráficas de los modelados del tercer caso en base a datos numéricos.....	54

Lista de Símbolos y Abreviaciones

ENIAC	Computador e Integrador Numérico Electrónico
IBM	International Business Machines, Sociedad Internacional de Material Electrónico
GPSS	General Purpose Simulation System
RAND	Research and Development
WSC	Winter Simulation Con-ference
MBS	Multibody Systems
ACSL	Advanced Continuos Simulation Language
CAMAS	Computer Aided Modeling, Analysis and Simulation
SIDOS	System Interdisciplinary Descrption OF System
FFT	Transformadas Rápidas de Fourier
LABSON	Laboratorio de Control de Calidad y Geotécnica
CAD	Diseño Asistido por Computadora
CAM	Fabricación Asistida por Computadora
LAB VIEW	Laboratorio Virtual
VHDL	Lenguaje para Descripción y Modelado de Circuitos
3D	3 Dimensiones
DAE	Ecuaciones Algebraico Diferenciales
SISO	Simulation Interoperability Standards Organization

Capítulo 1

Introducción

1.1 La importancia de la simulación.

Simulación es una técnica numérica para conducir experimentos en una computadora digital. Estos experimentos comprenden ciertos tipos de relaciones matemáticas y lógicas, las cuales son necesarias para describir el comportamiento y la estructura de sistemas complejos del mundo real a través de largos periodos de tiempo. [Thomas H. Taylor, *Técnicas de simulación en computadores*, México, DF: Limusa, 1996].

La definición anterior está hecha en un sentido digital, ya que puede incluir desde una plano arquitectónico hecho en la computadora, hasta un sofisticado sistema digital, este último, al ser simulado de manera digital tiene como ventajas los siguientes puntos:

- A través de un estudio de simulación, se puede estudiar el efecto de cambios internos y externos del sistema, al hacer alteraciones en el modelo del sistema y observando los efectos de esas transformaciones en su comportamiento.
- Una observación detallada del sistema que se está simulando puede conducir a un mejor entendimiento del mismo y por consiguiente a sugerir estrategias que mejoren su operación y eficiencia.
- La simulación de sistemas complejos puede ayudar a entender mejor su operación, a detectar las variables más importantes que interactúan y a entender mejor las interrelaciones entre estas variables.
- La técnica de simulación puede ser utilizada para experimentar con nuevas situaciones, sobre las cuales tiene poca o ninguna información. A través de esta

experimentación se puede anticipar mejor a posibles resultados no previstos.

- Cuando nuevos elementos son introducidos en un sistema, la simulación puede ser usada para anticipar cuellos de botella o algún otro problema que puede surgir en el comportamiento del mismo.

1.2 Objetivo

El objetivo de esta tesis radica en describir el software de modelado y simulación de sistemas dinámicos, 20-SIM versión 4.1, mediante un análisis del mismo y realizando aplicaciones para casos de estudio.

1.3 Justificación

Se propone a 20 SIM como software para predecir, modelar, simular y analizar el comportamiento de sistemas dinámicos (motores eléctricos, manipuladores mecánicos, actuadores hidráulicos, circuitos de control por ordenador, etc.), teniendo como herramienta para el amplio conocimiento del software, la realización de este trabajo, el cual sirva para estudiantes, maestros e ingenieros de la U.M.S.N.H. u otras universidades, debido a la poca información que se puede encontrar en internet, bibliotecas o libros. Dando como información, las herramientas con las que cuenta 20 SIM, los sistemas que se pueden simular, la manera en que se pueden simular, la manera cómo funciona, la exactitud de su simulación en comparación con las distintas maneras de simular que ofrece y en que campos de la ingeniería es aplicable. Las razones que indujeron a utilizar este programa en relación con otros simuladores dinámicos para dar a conocer a los alumnos, profesores e ingenieros fueron las siguientes:

- LABSON ofrece licencias gratuitas para el estudio del programa.
- 20 SIM está basado en Bond Graph, el cual es una herramienta que proporciona un modelo común para los diferentes campos de la técnica mediante la representación

gráfica de sus componentes y sus interrelaciones y en LABSON existe experiencia desde inicio de los 90.

- Con las técnicas de Bon Graph es muy sencillo añadir complejidad a un modelo mediante la incorporación de nuevos elementos o subsistemas. Una de las virtudes de Bon Graph es su versatilidad, porque permite modificar la estructura del sistema de forma rápida y certera, el cual es muy útil desde el punto de vista de diseño. Partiendo de esto, es posible tratar un sistema sencillo y refinarlo progresivamente hasta transformarlo en un modelo lo suficientemente exacto del sistema considerado.
- 20 SIM cuenta con una librería múltiple con un gran sistema de sub modelos de dominio orientado.

1.4 Organigrama y estructura de la tesis

Enseguida se presenta la estructura de la tesis, así como una breve descripción de los capítulos que la componen.

En el capítulo 1 se da una breve introducción a este trabajo, así como un panorama general de 20 SIM y su importancia en la simulación de sistemas dinámicos. También se presenta el objetivo, motivación y estructura de este trabajo.

En el capítulo 2 se dan los antecedentes de la simulación, de software de simulación para el modelado de sistemas dinámicos en especial del 20 SIM.

En el capítulo 3 se detallará la manera de usar el software, las características que ofrece para el desarrollo de la simulación de sistemas dinámicos y la manera para utilizarlas. Al igual se toma la ecuación de un sistema, para modelar con dos diagramas que ofrece el software para conocer el cómo se simula y la exactitud de simulación que se tiene entre los dos diagramas.

En el capítulo 4 se realiza el diagrama de bloques y el diagrama matemático de tres sistemas físicos complejos, que procederán a ser simulados utilizando el software 20-SIM, para obtener pruebas más contundentes en base a la realizada en el capítulo 3.

En el capítulo 5 se presentan conclusiones y recomendaciones de este trabajo de tesis, se observará si realmente convergen los desarrollos de modelado antes descrito con el proceso de simulación, es decir, si se observaron respuestas similares entre las distintas pruebas que se realizaron.

Capítulo 2

Antecedentes de la simulación y la evolución de los programas de sistemas dinámicos

2.1 Historia de la simulación

2.1.1 Historia de la simulación en la antigüedad

Algunos consideran que la simulación nace en 1777 con el planteamiento del problema “aguja de Buffon”, un método matemático sencillo para ir aproximando el valor del número PI a partir de sucesivos intentos.

Este modelo matemático se basa en una aguja de una longitud determinada lanzada sobre un plano segmentado por líneas paralelas por unidades.

En 1812 Laplace mejoró y corrigió la solución de Buffon y desde entonces se conoce como solución Buffon-Laplace. Posteriormente, el estadístico William Sealy Gosset, que trabajaba en la destilería de Arthur Guinness, ya aplicaba sus conocimientos estadísticos en la destilería y en su propia explotación agrícola. El especial interés de Gosset en el cultivo de la cebada le llevo a especular que el diseño de experimentos debería dirigirse no sólo a mejorar la producción media, sino también a desarrollar variedades de cebada cuya mayor robustez permitiese que la producción no se viese afectada por las variaciones en el suelo y el clima.

Para evitar futuras filtraciones de información confidencial, Guinness prohibió a sus

empleados la publicación de cualquier tipo de artículo independiente de su contenido, de ahí el estudio que hizo Gosset en sus publicaciones del seudónimo “Student”, para evitar que su empleador lo detectara. Es por esta razón que su logro más famoso se conoce como la “distribución t de Student”, que de otra manera hubiera sido conocida como la “distribución t de Gosset”.

Este hito histórico abrió las puertas a la aplicación de la simulación en el campo del proceso de control industrial así como a las sinergias que generaba esta simulación basada en la experimentación y técnicas de análisis para descubrir soluciones exactas a problemas clásicos de la industria y la ingeniería.

2.1.2 Periodo de formación (1945-1970)

A mediados de los años 40 dos hechos sentaron las bases para la rápida evolución del campo de la simulación:

- La construcción de los primeros computadores de propósito general como el ENIAC (Computador e Integrador Numérico Electrónico).
- El trabajo de Stanislaw Ulam, John Von Neuman y otros científicos para usar el método de Montecarlo en computadores modernos y solucionar problemas de difusión de neutrones en el diseño y desarrollo de la bomba de hidrogeno. Ulam y Von Neumann estuvieron presentes en el proyecto Manhattan.

2.1.3 El arte de la simulación

En 1960, Keith Douglas Tocher desarrolló un programa de simulación general cuya principal tarea era la de simular el funcionamiento de una planta de producción donde las máquinas ciclaban por estados: Ocupando, Esperando, No disponible y Fallo; de manera que las simulaciones en los cambios de estado de las máquinas marcarán el estado definitivo de la producción de la planta. Este trabajo produjo además el primer libro sobre

simulación. The Art of Simulation (1963).

2.1.4 Sistema de Simulación de Propósito General

Para aquel entonces, IBM (International Business Machines, Sociedad Internacional de Material Electrónico) desarrolló entre 1960 y 1961 el sistema de Simulación de propósito general o GPSS (General Purpose Simulation System). El GPSS se diseñó para realizar simulaciones de teleprocesos involucrados, por ejemplo: control de tráfico urbano, gestión de llamadas telefónicas, reservas de billetes de avión, etc. La sencillez de estudio de este sistema lo popularizó como el lenguaje de simulación más usado de la época.

Por otro lado, en 1963 se desarrolló SIMSCRIPT, otra tecnología alternativa al GPSS basada en FORTRAN, más enfocada a usuarios que no tenían porqué ser obligatoriamente expertos informáticos en RAND CORPORATION (Research and Development).

2.1.5 Simula I y WSC

Complementariamente a los desarrollos llevados a cabo por RAND e IBM, el Royal Norwegian Computing Center inició en 1961 el desarrollo del programa SIMULA con ayuda de Univac. El resultado fue SIMULA I, probablemente el lenguaje de programación más importante de toda la historia.

En 1967 se fundó el WSC (Winter Simulation Conference), lugar donde desde entonces y hasta ahora se archivan los lenguajes de simulación y aplicaciones derivadas, siendo en la actualidad el referente en lo que a avances en el campo de los sistemas de simulación se refiere.

2.1.6 Periodo de Expansión de la simulación

Durante el periodo de 1970-1981 se desarrollaron avanzadas herramientas de

modelado y de análisis de resultados. Gracias también a los desarrollos obtenidos en la generación de datos y a las técnicas de optimización y representación de datos, la simulación llega a su fase de expansión donde comienza a aplicarse en múltiples campos. Anteriormente, los datos de salida obtenidos de una simulación por computadora se presentaban en una tabla o matriz, de manera que se mostraba el efecto que los múltiples cambios en los parámetros tenían sobre los datos. El empleo del formato de matriz se debía al estudio tradicional que se hacía de la matriz en los modelos matemáticos. Sin embargo, los psicólogos advirtieron que los seres humanos percibían mejor los cambios en el desarrollo de las situaciones si miraban gráficos o incluso imágenes en movimiento o animaciones generadas a partir de dichos datos, como las que se ejecutan en las animaciones de imágenes generadas por computadora.

2.1.7 Evolución de la Simulación

El galopante desarrollo tecnológico de las últimas décadas (1970—1981) hace que los términos que hasta hace poco tiempo eran exclusivos de las novelas de ficción formen parte de nuestro día a día. Con el aumento de la capacidad de cálculo de las computadoras, y el planteamiento de las técnicas visuales, surgen nuevas ideas sobre desarrollos de herramientas de entrenamiento basadas en tecnologías innovadoras. El empleo de la simulación se presenta como uno de los métodos más efectivos a la hora de transmitir los conocimientos y en el análisis en determinadas materias, pero por su elevado coste solamente es alcanzable para las grandes empresas o entidades con importante financiación pública. Gracias a los avances tecnológicos la simulación ha evolucionado enormemente, permitiendo alcanzar excelentes cotas de fiabilidad. Las capacidades de cálculo actuales permiten ejecutar una serie de modelos matemáticos que conjuntamente con las técnicas visuales reflejan fielmente la realidad en toda su complejidad. Con ellos un amplio sector de la sociedad es capaz de adquirir un simulador, mejorar la calidad y el contenido de la formación. El progreso continúa y con él cada avance en el plano tecnológico deberá estar traducido en beneficio de las personas.

2.1.8 El concepto de inmersión

El objetivo final de toda herramienta de simulación consiste en reproducir unas determinadas condiciones de entorno para generar en el usuario una serie de sensaciones que resulten tan parecidas a la realidad como sea posible. Si existiera un simulador capaz de generar sensaciones tan realistas que el usuario no supiera encontrar diferencias entre la realidad y la experiencia simulada, se diría que dicho simulador es totalmente inmersivo, o que proporciona una inmersión del 100%.

2.1.9 Modelos Matemáticos

Proporcionar el funcionamiento satisfactorio de un simulador solamente es posible si actúan una serie de modelos matemáticos capaces de simular el comportamiento de la máquina real con un nivel de precisión y sofisticación lo suficientemente alto como para generar todos los efectos y matices que permitan identificarla de manera inequívoca. Sólo así las experiencias acumuladas en el simulador pueden proporcionar una experiencia aplicable a la vida real.

2.2 Evolución de los programas de sistemas dinámicos

El objetivo principal de la modelización de un sistema y su simulación es por un lado la generación de un conjunto de ecuaciones matemáticas que describan su comportamiento, y por otro obtener la evolución de los valores de las distintas variables del sistema en el dominio del tiempo o de la frecuencia.

Debido a la complejidad de las ecuaciones dinámicas correspondientes a un sistema y la aparición de computadoras con buenas prestaciones y generalmente disponibles a nivel de usuario, el estudio de las computadoras se fue introduciendo poco a poco en la década de los años 70. El contexto teórico para la formulación de las ecuaciones dinámicas está disponible desde hace dos décadas en un gran número de libros.

Una primera generación de herramientas que surgieron para la simulación de sistemas dinámicos fue construida en base a la aplicación de códigos específicos solo para resolver problemas concretos.

En el campo de la simulación dinámica de sistemas mecánicos, fueron desarrollados un gran número de herramientas de análisis que permitían la formulación de sistemas multicuerpo, MBS (Multibody Systems) y la resolución de sus ecuaciones de movimiento. Existían distintitos tipos de software disponibles que permitían la generación de las ecuaciones de movimiento de forma numérica o simbólica y en algunos casos, contenían integradores numéricos.

La simulación en ingeniería mecánica fue introducida por programas como NEWEUL, MESA VERDE, ADAMS o DADS, creados a principios de los 80. Estos programas formulaban las ecuaciones de movimiento de Newton-Euler o Lagrange y las integraba para unas condiciones iniciales dadas. Con el tiempo, alguno de los software anteriores incluyeron entrada de datos gráfica, animaciones, conectividad con programas con modelos CAD (Diseño Asistido por Computadora) y análisis de señales, siendo ADAMS y DADS los que introdujeron la simulación en sistemas dinámicos mediante CAD/CAM (Fabricación Asistida por Computadora).

Este tipo de programas constituyen un primer grupo de programas comerciales que están destinados a la generación de sistemas multicuerpo. Adicionalmente a estos, se crearon pocos años más tarde otros programas similares como SIMPACK o COMPAMM.

Normalmente, una herramienta de diseño está relacionada con librerías que incluyen distintos componentes de modelado que son reutilizables y hacen más fácil y más rápida la descripción del modelo y su posterior validación. Rupperl y Wysor analizaron y compararon algunas de estas herramientas de modelado hace casi una década.

Para ese entonces existían algunas herramientas de simulación muy específica, todavía existente hoy en día, como LAB VIEW (Laboratorio Virtual) para instrumentación

y VHDL (Lenguaje para Descripción y Modelado de Circuitos) para componentes electrónicos. Entre las herramientas de carácter general que existían en ese momento destacaban ACSL (Advanced Continuous Simulation Language) que fue la primera herramienta en ser introducida en el mercado, MATRIX x, compuesta por Xmath, System Build y Real Sim, que habían sido adquiridas por The Math Works y tendían a desaparecer al ser absorbidas por Matlab la primera de ellas y por Simulink el resto, SCILAB-SCICOS, que eran y siguen siendo software de libre distribución, EASY5, que permitía implementar modelos mecánicos, neumáticos, hidráulicos y digitales, entre otros, y 20-Sim, que se basaba en el modelado mediante bloques y bond graph y que hoy en día permite interactuar con Matlab/Simulink. MATRIX x volvió a ser relanzada años más tarde por la misma empresa propietaria de LAB VIEW, bajo el nombre de NI MATRIX x.

Otra herramienta no muy extendida en ese momento era MODELICA, un lenguaje de modelado muy centrado en la reutilización de componentes. Estaba pensado como un nuevo estándar para intercambio de modelos y librerías, por lo que se trataba de un lenguaje de simulación, sin entornos tan amigables como podía tener Matlab, ACSL, 20-Sim, o SCILAB-SCICOS.

Por otra parte, existían programas más especializados como ADAMS/CAR (un módulo de ADAMS), SOPHIA, AUTOLEV, GENSYS, DYMOLA y Math Modelica. SOPHIA y AUTOLEV estaban basados en la formulación de Kane, mientras que DYMOLA y Math Modelica empleaban MODELICA. Estas herramientas tenían la ventaja de ser altamente flexibles para el usuario, requiriendo posibilidades de modelado más avanzadas o el control directo sobre las ecuaciones.

Por último, había aplicaciones más complejas dirigidas a mercados más específicos, como por ejemplo Auto Sim, en el que se generaban sistemas multicuerpo de forma simbólica y se generaba código mediante Fortran o C para realizar su simulación numérica.

Las aplicaciones informáticas de carácter general, conforman un segundo grupo de programas comerciales cuya característica principal es que ayudan a realizar de una forma

más cómoda los modelos de simulación, pudiendo identificar tres enfoques principales:

- Mediante diagramas de bloque;
- Modelado orientado a objetos; y
- Bond-graph.

Los sistemas pertenecientes a la primera categoría incluyen herramientas como ACSL, Matlab/Simulink, SCILAB-SCICOS etc. Todos estos programas están basados en un entorno interactivo de modelado donde el usuario define el sistema por medio de una red de bloques interconectados entre sí. Conforman lo que se denomina la segunda generación de herramientas.

En la segunda categoría se pueden incluir EASY, DYMOLA, Math Modelica etc. estos últimos por medio del estudio del lenguaje orientado a objetos MODELICA. La ventaja que tiene este tipo de modelado es la de poder emplear bibliotecas de objetos, señalándose a menudo estas herramientas como de tercera generación u orientados directamente hacia las ecuaciones.

Finalmente, el programa de interés 20-Sim, ENPORT o CAMP-G, modelan los sistemas basándose en la metodología del bond-graph. En el capítulo 3 se explicará la técnica que emplean dichos programas.

Según broenink [Broenink J.F., *20-sim software for hierarchical bond-graph/bolck diagram models*, Simulation Practice and Theory, volume 7, pp.481-492, 1999], se deben cumplir dos condiciones para garantizar que los submodelos sean acoplables:

- Los puertos de conexión entre submodelos están definidos como pares de variables. El estudio de pares de variables conjugadas en potencia hacen que las conexiones sean físicas, por ejemplo: par y velocidad, presión y caudal.

- Los sub modelos han de estar escritos mediante declaraciones, es decir, estableciendo relaciones entre cada uno de los componentes. Esta descripción también se llama casual. [Broenink J.F., *20-sim software for hierarchical bond-graph/bolck diagram models*, Simulation Practice and Theory, volume 7, pp.481-492, 1999].

Los sub modelos se han de describir aplicando ecuaciones de conservación, aunque también puede ser necesario describir las propiedades de los elementos usados. El sistema de ecuaciones se obtiene a través de la combinación de las ecuaciones de los sub modelos y de los puertos de conexión, conduciendo finalmente a las ecuaciones algebraico diferenciales (DAEs) del modelo.

En el caso de esta tesis, el software Bond Graph y MODELICA son ejemplos que usan este tipo de modelado, ya que describen la dinámica de un modelo sin tener en cuenta la causa y esto proporciona una mayor flexibilidad al no forzar al modelador a usar relaciones entre las distintas entradas y salidas.

Por último, estaría todo aquel software que se basan en análisis mediante elementos finitos, como por ejemplo ANSYS, y aquellos propios de dominios específicos como por ejemplo SPICE para circuitos eléctricos o MANIP para problemas térmicos, que necesitan en general la colaboración con otros programas.

2.3 Evolución de 20-SIM

CAMAS (Computer Aided Modeling, Analysis and Simulation) fue desarrollado por el mismo grupo de investigación que THTSIM-TUTSIM a principio de los años 90.

Inicialmente trabajaba bajo sistema operativo SUN pero el año 95 se presentó una versión para PC. Es necesario describir los modelos usando sub modelos interconectados

entre sí mediante multibonds direccionados, los ‘bonds’ representan el flujo de potencia del sistema.

Está compuesto de dos partes, cada una de las cuales realiza la función de modelización y simulación respectivamente.

En la primera de ellas, se realiza la edición del modelo a través de un intérprete, bien sea una entrada interactiva de Bond Graph, o bien un texto generado en SIDOPS, la grabación de la descripción del modelo en una base de datos y la preparación de los modelos para su simulación. Esta última parte remplace las llamadas a sub modelos por su respectiva descripción, transformándolo a uno único y analiza la causalidad del modelo mediante un algoritmo automático.

SIDOPS (System Interdisciplinary Description OF System) es un lenguaje orientado a objetos, mediante el cual se describe un modelo correspondiente a un sistema físico de forma estructurada mediante la teoría de Bond Graph. Un modelo creado así consta de una descripción y un conjunto de parámetros.

En la segunda parte, se procede a la simulación y al procesamiento opcional de sus resultados. Admite ecuaciones de restricción y determina las coordenadas dependientes en función de la asignación casual.

20-SIM es una evolución del programa CAMAS, sigue utilizando SIDOPS como lenguaje de simulación y funciona bajo Windows, estando actualmente en la versión 4.1. Permite la simulación de sistemas dinámicos pertenecientes a múltiples dominios de la física mediante el empleo de iconos, diagramas de bloques, grafos y ecuaciones, siendo posible la conexión directa con Matlab. Para poder simular un modelo es preciso compilarlo previamente, lo cual se realiza mediante una herramienta que lleva incorporada.

A diferencia de las versiones anteriores, permite mostrar de forma similar a Simulink y consta de un módulo, mediante el cual se puede construir y simular mecanismos

de forma tridimensional.

Soporta modelos que contienen eventos que varían de estado en función del tiempo, como por ejemplo problemas de colisión o sistemas continuos controlados de forma digital. También se han implementado formulaciones de sólido rígido, mejores modelos de fricción modelos de muelles y juntas tridimensionales.

Capítulo 3

Descripción del software 20-SIM

3.1 Introducción

La siguiente sección consta de una descripción general del entorno con el que cuenta el software *20 SIM 4.1*, así como cada una de las herramientas con las que cuenta y de qué manera pueden ser aplicadas en el campo de la ingeniería. El cual puede ser una herramienta de apoyo para estudiantes, maestros, ingenieros etc. para el conocimiento amplio sobre el mismo software.

Como ya se mencionó en la justificación, 20 SIM, puede simular el comportamiento de sistemas eléctricos mecánicos e hidráulicos o cualquier combinación de estos sistemas. Se puede entrar al modelo en una forma gráfica por medio de un editor gráfico. Ofrece un modelado, simulación y entorno de análisis para sistemas de ingeniería que apoya a la jerarquía del modelo que utiliza diferentes lenguajes en cada elemento. Puede trabajar con Bon Graph multicuerpo, con diagrama de bloques, diagrama iconos y con ecuaciones escritas en SIDOPS+. El programa se divide en dos partes, con la primera se introduce el modelo y se compila y con la segunda se realizan experimentos sobre el modelo. En el último caso, el usuario tiene la posibilidad de escoger el algoritmo de integración, todo esto se explica a brevedad en los siguientes capítulos.

De igual manera como se mencionó en el apartado 2.2 “Evolución de 20-SIM”, el programa 20-SIM utiliza la técnica del Bond Graph la cual consiste en realizar la descripción de un modelo basándose en la energía del sistema, por medio de la cual las estructuras de los distintos elementos están conectadas de tal forma que mantiene la energía. “Bond Graph define un conjunto mínimo de elementos generalizados que pueden ser usados para modelar el comportamiento de un sistema a través de diferentes dominios de energía”.

[Karnopp y Rosenberg, *System dynamics: a unified approach*, USA: John wiley & Sons, 1968]. Las conexiones se hacen a través de bonds.

3.2 Modelado

Los sistemas ya modelados se construyen en el editor de 20 SIM. Este editor contiene una gran biblioteca de bloques que representan magnitudes físicas, que ayudan a construir modelos de sistemas usando diagramas de bloques, diagramas icónicos, gráficos de ligaduras o una combinación de estos. Varias herramientas ayudan a crear modelos especiales como lo son el cuadro de controles, mecatrónica y 3D (3 Dimensiones).

3.2.1 Cuadro de controles

Cuadro de controles es la opción de herramientas de control de 20 SIM la cual contiene varias herramientas que pueden ayudar en el desarrollo de controladores para máquinas de modelado como lo son, el editor de diseño del controlador, el editor de filtros y los editores de red neuronal como se explican a continuación.

- El editor de diseño del controlador es una herramienta especializada para el diseño de sistemas de control de retroalimentación. Una estructura de retroalimentación de subsistemas se presenta con una planta lineal, controlador, medición y pre-filtro. También las ganancias de lazo abierto y lazo cerrado y las sensibilidades están disponibles como se muestra en la Figura 3.1.

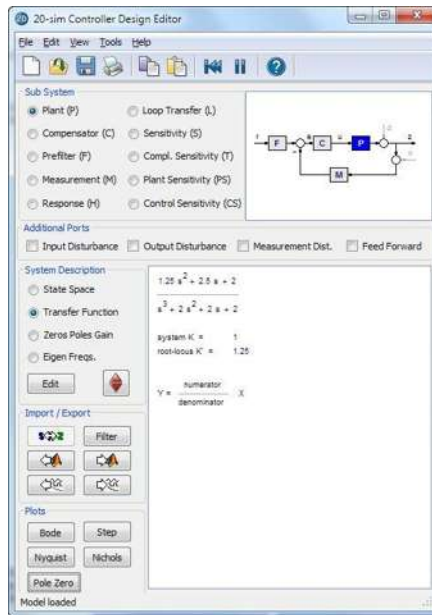


Figura 3.1 Cuadro de controles.

Se puede editar un controlador como un sistema Espacio de Estado ABCD, una función de transferencia o en una forma Polo Ganancia Cero. Los cambios en uno de los subsistemas se pueden actualizar directamente en cada uno de los gráficos y cuadros de diálogo abiertos. Por ejemplo, la adaptación de la ganancia del controlador cambia inmediatamente de los polos y ceros del sistema a un bucle cerrado y la respuesta de paso general, como se muestra en la Figura 3.2. La integración dentro de 20-sim y sistema lineal de intercambio con MATLAB hace que este editor sea una poderosa herramienta para el diseño de sistemas de captación de control.

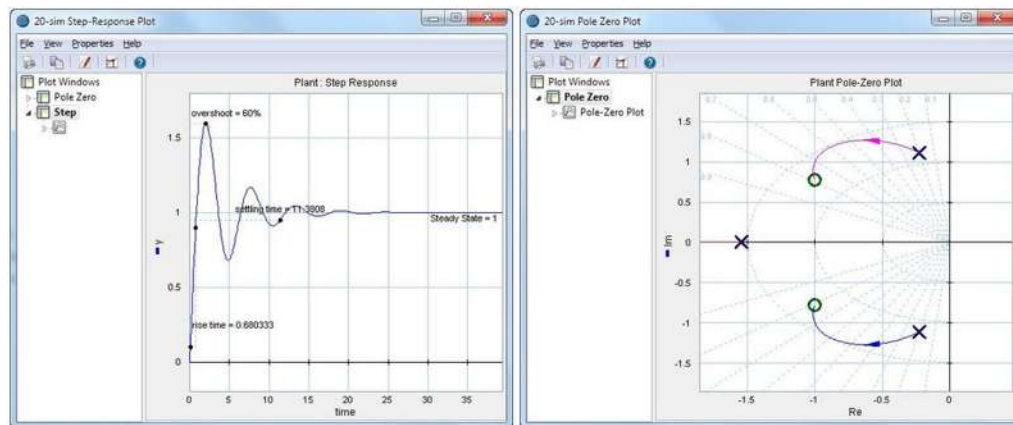


Figura 3.2 Adaptación del controlador.

- Otra opción con la que cuenta 20 SIM es el editor de filtros en el cual se pueden crear filtros lineales de acuerdo con sus especificaciones. Los filtros disponibles son Bessel, Butterworth y filtros Chebychev, donde se especifica el orden y frecuencias características. Una opción que también se puede hacer es el PID, de avance, retardo, o filtros de muesca.
- Las herramientas de control de 20-SIM soporta dos conocidas redes adaptativas: B-Spline Redes y Multi-Layer Redes Perceptrón. Para esto el programa cuenta con un ejemplo de cada red. Cada ejemplo incluye dos entradas y salidas deseadas. Basado en el error entre las salidas deseadas y la salida real de la red, la red neuronal se adapta al peso de cada neurona de acuerdo con la tasa de aprendizaje por el usuario. Si la respuesta es lo suficientemente precisa, puede guardar los pesos de las neuronas, para utilizar la red neuronal en el controlador. Un número ilimitado de neuronas puede ser utilizado.

3.2.2 Herramientas de mecatrónica

Las herramientas de mecatrónica incluye el Asistente de perfil Motion, el Asistente para CAM y el Editor de Servo Motor. La Caja de herramientas mecatrónica sólo está disponible en la versión profesional de 20-sim.

- **El Editor de Servo Motor 20 SIM** es un programa que ayuda a los ingenieros a seleccionar el servo motor adecuado para cualquier sistema electromecánico:
 - 1) Brush DC (Plancha para ropa Armature Motor, Motor Rotor Hollow, Disc Armature de vehículos automóviles)
 - 2) Sin escobillas DC
 - 3) De CA sincrónico

4) AC síncrono lineal

En colaboración con los fabricantes de motores, se han creado tablas de datos para el editor de servo motor. El desempeño de cada motor se puede mostrar por la curva de velocidad de par de torsión como se muestra en la Figura 3.3.

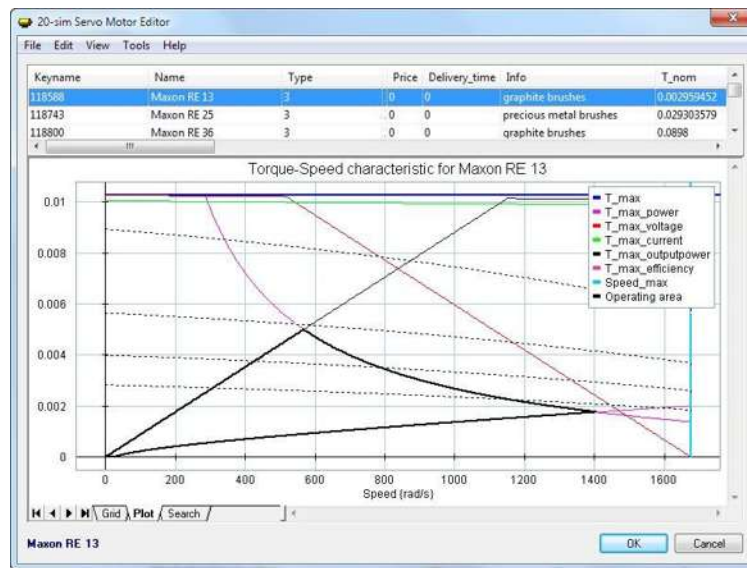


Figura 3.3 Gráfica de máximos

El editor de servo motor puede generar modelos dinámicos para el programa de simulación 20-sim. En este programa se puede simular el comportamiento térmico y dinámico del servo motor en combinación con circuitos de control y las cargas dinámicas.

Cualquier ingeniero involucrado en el diseño de máquinas electromecánicas se puede beneficiar a partir del editor de servo motor. Se ahorra tiempo y dinero mediante la búsqueda del servo motor óptima, en tan solo unos minutos, sin correr el riesgo sobrecalentamiento o under powering.

Características

- 1) Añadir los motores deseados a la mesa de datos.

- 2) Búsqueda rápida por varios parámetros.
- 3) Con la opción de Torque se puede obtener el máximo de: corriente de par, tensión, velocidad, la eficiencia y la salida de potencia.
- 4) Área de operación segura, Zona de funcionamiento deseado.
- 5) Generar modelos dinámicos para el programa de simulación de 20-sim. Estos modelos incluyen los efectos térmicos de las bobinas, las pérdidas eléctricas a través de la disipación, pérdidas magnéticas a través de histéresis, corrientes parásitas y dentado.
- 6) Mostrar curva de esfuerzo de torsión velocidad con la curva de carga del dinámico.
- 7) Modelos de plantillas, que cubren los esquemas de control más comerciales y muchas cargas mecánicas.

Áreas de aplicación:

- 1) Diseño de Control
 - 2) Equipo Industrial y Maquinaria
 - 3) Ingeniería de precisión
 - 4) Simulación
 - 5) Energía
- **El Asistente para perfiles de movimiento** ayuda a crear perfiles de movimiento de una manera fácil de usar. Las siguientes mociones son compatibles:

- 1) Plano
- 2) Rampa
- 3) Trapezoidal
- 4) Parcial trapezoidal
- 5) Seno
- 6) Cúbico
- 7) Parcial Cubico
- 8) Estándar para modificar el seno
- 9) Modificado Trapezoidal
- 10) Seno modificado con velocidad constante
- 11) General seno modificado con velocidad constante
- 12) 3-4-5 Poli nominal
- 13) 1-3-5-7-9 Poli nominal

- **CAM Wizard.** 20-sim palanca asistente, ayuda a generar palancas y modelarlas. Al igual que el Asistente para perfiles de movimiento, puede utilizar varios tipos de perfiles de movimiento que son continuas en la velocidad, la aceleración o jerk.

3.2.3 Herramientas mecánicas en 3D

En 20-sim la opción de herramientas mecánicas en 3D proporciona la herramienta que hace más fácil el modelado dinámico 3D, el editor Mechanics 3D.

- **Cuerpos.** Se pueden crear modelos 3D arrastrando cuerpos al espacio de trabajo 3D. Las representaciones de cada cuerpo se puede cambiar a una esfera, bloque, cilindro etc. Además, los colores se pueden cambiar y pueden ser añadidas descripciones. El tamaño y la forma de un cuerpo son meramente para la representación, un cuerpo está completamente caracterizado por sus coeficientes de inercia y masa, algo de esto se muestra la Figura 3.4.

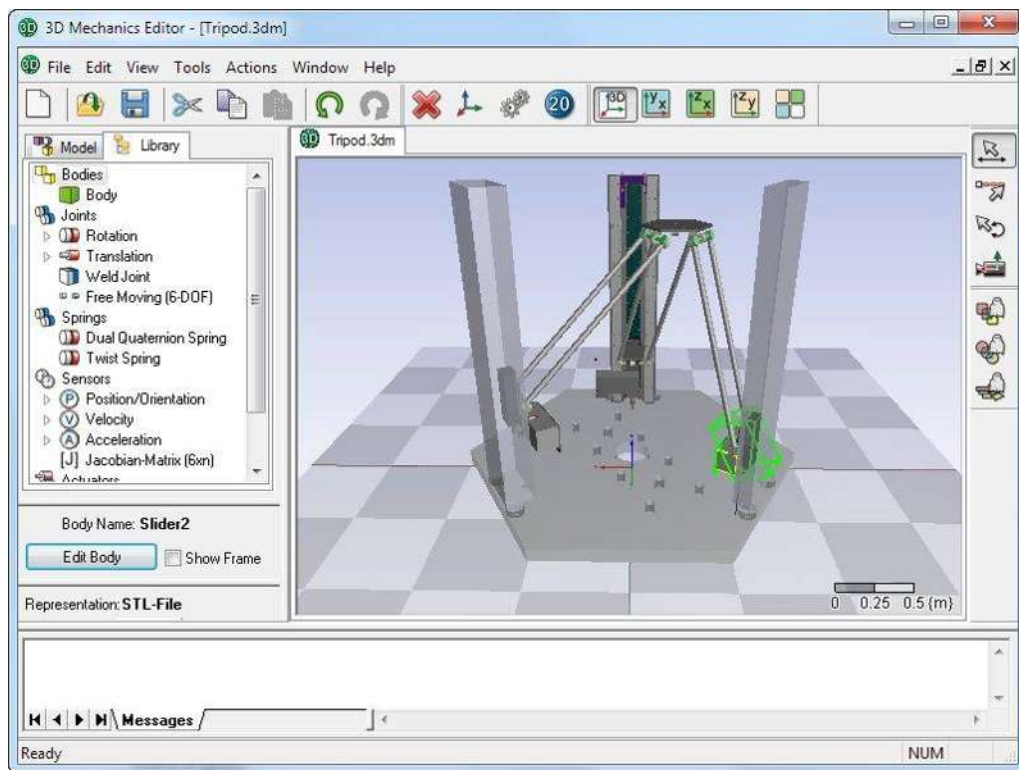


Figura 3.4 Representaciones 3D

- **Articulaciones.** Los cuerpos están interconectados por el estudio de las articulaciones. Varias articulaciones están presentes en la biblioteca, divididos en dos grupos, las articulaciones de rotación y traslación. Estas articulaciones también

pueden ser arrastradas y ser soltadas en el área de trabajo. Las restricciones se pueden agregar para crear sistemas de bucle cerrado como mecanismos de cuatro barras o plataformas Stewart.

- **Interface.** La interface de usuario tiene 4 diferentes modos en los que se puede seleccionar, conectar, traducir y girar los cuerpos y las articulaciones. Mucho esfuerzo se hace para mantener la interfaz gráfica de usuario lo más natural posible. Múltiples puntos de vista son compatibles. Además del entorno 3D, se puede ver en la intersección 2D xz , xy , y el plano yz .
- **Modelos.** El editor de Mecánica 3D puede generar un modelo de 20-sim de un modelo 3D. Este modelo de 20-sim comprende toda la dinámica y la cinemática del modelo. Las fuerzas pueden aplicarse a las articulaciones o cada cuerpo directamente. También pueden ser resortes y amortiguadores, parte de la biblioteca de la mecánica en 20-sim, a las articulaciones, ya que todo el modelo está basado en puertos. La gravedad se puede establecer como una fuerza externa. Con el tiempo, la respuesta dinámica del modelo completo se puede mostrar en el cuadro de herramientas de animación 3D.

3.3 Simulación

En el simulador se puede ejecutar modelos y mostrar los resultados en gráficos y animaciones 3D.

3.3.1 Simulador

Después de introducir un modelo en el editor se puede comprobar y compilarlo. Esto se realiza de forma automática en segundo plano al abrir el simulador. El simulador se utiliza para la simulación del modelo y análisis.

- **Espacio de Windows.** Los resultados de la simulación se pueden mostrar en gráficos y la animación de Windows. La ventana principal del simulador se utiliza para especificar la configuración de la simulación. Durante el funcionamiento las ventanas de animación se actualizan simultáneamente. Los espacios son totalmente configurables. Vistas logarítmicas, fuentes true-type, estilos de línea, estilos y procedencias de marcadores son compatibles. Los espacios de animaciones se hacen fácilmente listo para ser publicadas (copiar al portapapeles y se pega en cualquier documento).
- **Modo de depuración y modo rápido.** 20 SIM puede funcionar en dos modos: modo de depuración y modo rápido. Esto se indica mediante el botón de modo en la barra de herramientas. Se puede cambiar rápidamente entre estos modos pulsando el botón de modo. En el modo de depuración hace todas las comprobaciones posibles a realizar y advertencias que se generan para posibles errores del modelo. En el modo rápido incorpora un compilador en tiempo de ejecución el cual traduce el modelo de simulación en la plataforma de código específico de máquina 32-bit. El resultado es un dramático incremento de velocidad en la simulación. A menudo, el código máquina 20 SIM funciona más rápido que el equivalente compilado en código C. Compilar el código de la máquina, incluso con modelos de gran tamaño, se hace mientras se arranca el simulador. El compilador es una parte interna del software 20 SIM, así que no hay compilador externo o software que se necesita instalar para tal requerimiento.
- **Los algoritmos de simulación.** 20 SIM contiene potentes algoritmos de simulación para la resolución de ecuaciones diferenciales ordinarias (ODE) y ecuaciones diferenciales algebraicas (DAE). Se dispone de una variedad de métodos de integración numérica: de un sólo paso, multi-etapa y de orden variado.
- **Tiempo discreto para modelos.** 20 SIM detectará automáticamente los bucles de tiempo discreto en un modelo y asignará a cada circuito independiente la velocidad de muestreo por separado. Las señales discretas se muestran en el Editor en verde.

El tiempo discreto es activado en partes, por eventos de tiempo, para que los modelos mixtos de tiempo continuo y tiempo discreto se manejen correctamente.

- **Eventos.** 20 SIM también puede controlar los eventos estáticos basados en algoritmos de cruce por cero. Esto resulta en una detección de eventos rápida y detección precisa de la localización

3.3.2 Animación 3D

La animación es un poderoso método para inspeccionar y verificar los resultados de la simulación, especialmente cuando se trabaja en tres dimensiones. Resultados de la simulación de 20-sim puede ser mostrado como animaciones utilizando un editor de animación 3D. Las animaciones se componen de objetos predefinidos, como cubos, esferas, líneas, cuadrados, cámaras y luces. Los objetos complejos se pueden importar de paquetes de CAD utilizando formatos comunes de intercambio. Un ejemplo de simulación en 3D se muestra en la Figura 3.5.

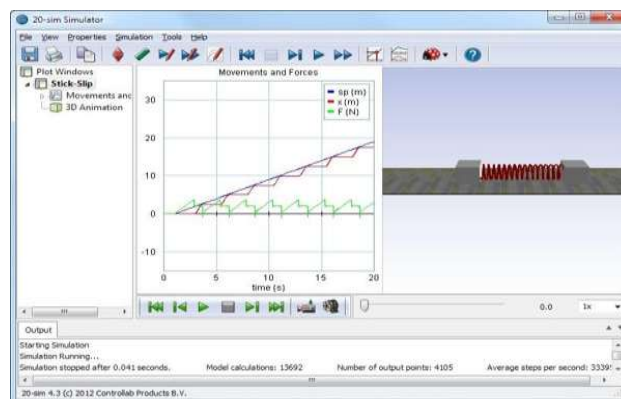


Figura 3.5 Animación en 3D

- **Objetos.** Cualquier variable de un modelo de 20-sim puede ser conectado a un objeto. Diversos atributos del objeto se puede controlar de esta manera: la posición, orientación, tamaño, etcétera. Gráficas térmicas pueden ser creados mediante el control del color de los objetos. Los objetos de referencia de marco se pueden utilizar para objetos de animación de grupos y heredar atributos del objeto. Los

objetos pueden ser duplicados, lo que resulta en animaciones complejas con unos clics del ratón.

- **Conectando a la simulación.** Las animaciones son completamente vinculadas al simulador en 20-sim. Mientras que una parcela se está elaborando, a la vez que la animación se muestra. Este enlace se mantiene después de una simulación ha terminado. Al inspeccionar los valores numéricos, se puede apreciar los cambios de animación en 3D al mismo tiempo.
- **Cine.** Cada animación se puede exportar como una película. 20-sim soporta todos los formatos estándar, incluyendo You Tube, AVI y WMV.

3.4 Análisis

Se puede analizar los modelos usando la opción de herramientas de dominio del tiempo y de la opción de herramientas de dominio de la frecuencia.

3.4.1 Herramientas en el dominio del tiempo

Durante la simulación, el comportamiento en el dominio del tiempo de un modelo es calculada. Sobre la base de este comportamiento, el modelo puede ser analizado. Un conjunto de potentes métodos para el análisis del dominio del tiempo está disponible en 20 SIM los cuales se mencionan a continuación.

- **Barridos de parámetros.** Es una opción de 20-sim la cual permite, múltiples simulaciones con una variación en el valor de cada parámetro. Es un método rápido para ver su comportamiento a través de los distintitos lapsos del tiempo como se muestra en la Figura 3.6.

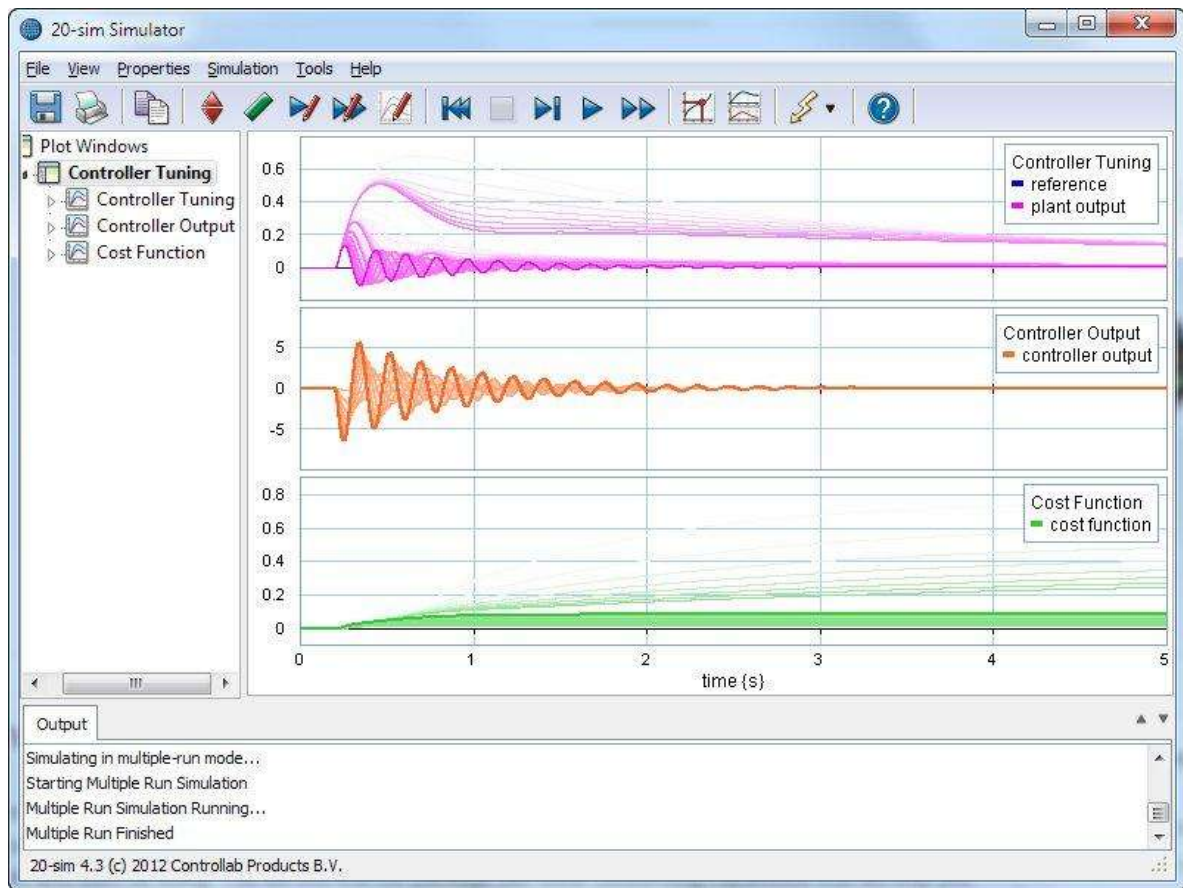


Figura 3.6 Barrido de parámetros

- **Optimización.** Se puede maximizar el rendimiento de un modelo mediante la variación de los parámetros especificados en el mismo. Funciones de coste predefinidos o definidos por el usuario puede ser utilizado como una medida del rendimiento del modelo. Un número de métodos de optimización bien conocidos se puede utilizar para minimizar o maximizar estas funciones de costes. Con optimización se puede encontrar los parámetros del controlador de rechazo de perturbaciones óptima.
- **Ajuste de curvas.** Con ajuste de curvas se puede encajar el rendimiento del modelo a un resultado determinado por la variación de parámetros. Se trata de un método muy útil para optimizar los parámetros del modelo cuando los datos de medición están disponibles.

- **Análisis de sensibilidad.** Se utiliza para investigar el efecto de la variación de los parámetros de funcionamiento del modelo. El cambio del rendimiento dividido por el cambio de parámetro se traza en una tabla. Análisis de sensibilidad es un método bien conocido en la ingeniería de producción para encontrar tolerancias de sistemas críticos en las primeras etapas del diseño.
- **Análisis Monte-Carlo.** Puede realizar cuantas simulaciones se ejecuta con la variación estadística de los valores de los parámetros. Los resultados pueden ser mostrados como histogramas o inspeccionado numéricamente. El análisis Monte Carlo es un poderoso método para averiguar el peor comportamiento del modelo caso.
- **Análisis de Variación.** Con el Análisis de Variación se puede encontrar el rango estadístico de los valores de los parámetros que producen un modelo con un nivel de rendimiento. Análisis de variación es un poderoso método para encontrar una medida para tolerancias del sistema en la etapa inicial del diseño.

3.4.2 Herramientas en el dominio de la frecuencia

Las herramientas en el dominio de la frecuencia de 20 SIM consiste en el editor del sistema lineal, herramientas de análisis de FFT (Transformada Rápida de Fourier) y funciones de linealización del modelo, las cuales se explican a continuación:

- **El editor del sistema lineal.** Es una herramienta especializada para el diseño y análisis de sistemas lineales. El editor soporta sistemas SISO (Simulation Interoperability Standards Organization), tiempo continuo y tiempo discreto, usando varias representaciones. La interfaz gráfica permite editar un sistema lineal en cualquier forma deseada: espacio de estado ABCD, la función de transferencia o Ganancia Polo Zero, análisis de la respuesta al escalón, Bode, Nyquist diagrama, carta de Nichols y parcelas polo-cero lo que permite evaluar rápidamente el comportamiento del sistema como se muestra en la Figura 3.7.

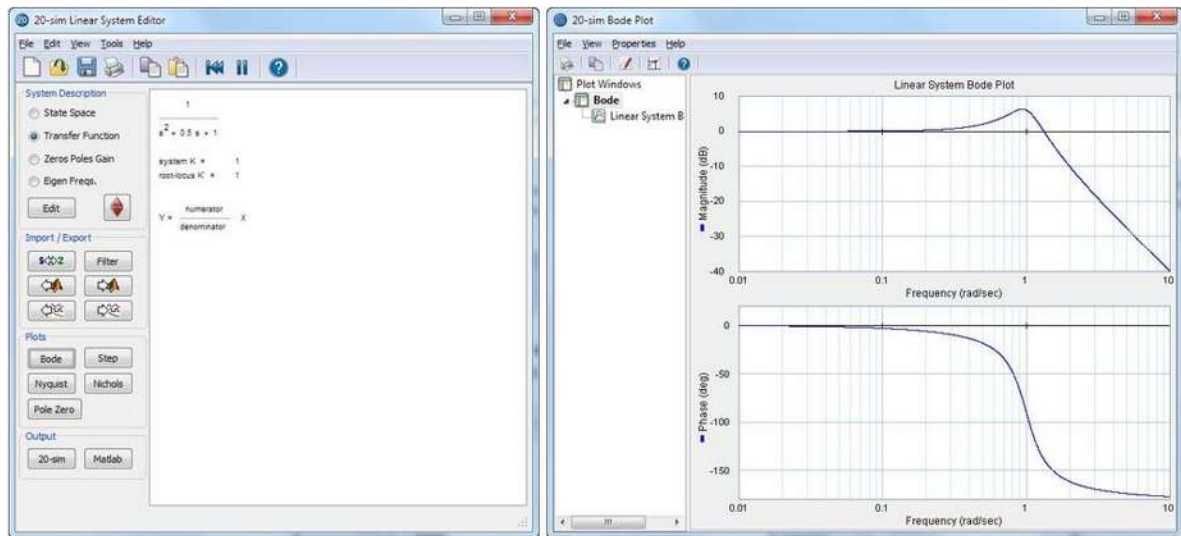


Figura 3.7 Editor de sistema lineal

Como se mostró en la Figura 3.7 márgenes de fase, y el módulo de ganancia se calcula, así como tiempo de subida, sobre impulso y valor de estado estacionario. La entrada puede originarse a partir de un modelo de sistema de 20-sim lineal, 20-sim editor de filtro o control, espacio de trabajo de MATLAB, o la entrada del usuario. Se puede generar una salida para todos los editores de 20-sim, el portapapeles y el espacio de trabajo de MATLAB.

- **Fast Fourier Transform.** Transformadas rápidas de Fourier (FFT) puede ser aplicado a cualquier trama de dominio de tiempo en 20-sim. O los resultados de la simulación o la medición de datos que se va hacer. Cuando los datos no están espaciados por igual, la interpolación lineal se aplica por primera vez después de que la transformada rápida de Fourier se utiliza para calcular el contenido de frecuencia. Tres representaciones son compatibles: parcela de amplitud y fase, la trama y el argumento Frecuencia densidad espectral de potencia.
- **Linealización.** Cualquier modelo de 20 SIM se puede linealizar a la forma de espacio de estado. Si es posible, la linealización se realiza simbólicamente. De lo contrario la linealización se realiza numéricamente. El modelo de espacio de estado resultante se muestra en el Editor del sistema de 20 SIM lineal. El Editor del

sistema lineal es una herramienta especializada para el diseño y análisis de sistemas lineales. El editor soporta SISO tiempo continuo y tiempo discreto usando varias representaciones. Parcelas estandarizadas permiten evaluar rápidamente el comportamiento del sistema. Si es posible cualquier modelo se puede linealizar.

3.5 Exportar

Con la opción de herramientas de generación de código se puede exportar modelos, como en código C para su estudio en 20-sim 4C, Matlab, Simulink y otros paquetes.

3.5.1 Herramientas de generación de código

20-sim es capaz de generar código C para todos los modelos de 20-sim que se han creado. Para ser usadas como plantillas o modelos base, el código C puede ser generado por diversos objetivos tales como:

- **20-sim 4C.** El paquete de 20-sim 4C, ayuda a ejecutar código C en el hardware para el control de máquinas e instalaciones. 20-sim hace importaciones en 4C modelos (código C) de 20-sim y los ejecuta en hardware como tarjetas de brazos empotrados, 104 sistemas de PC y mucho más.
- **Matlab Simulink.** Generación de código C para su estudio en MATLAB Simulink también incluye un bloque sub modelo con terminales de entrada y de salida. 20-sim utiliza el MEX-compilador para compilar el código directamente en un S-función. Estas funciones S también se puede utilizar en el taller en tiempo real con el fin de generar código para una plataforma específica, por ejemplo Target xPC.
- **Código C.** 20-sim puede generar código C independiente para su estudio en C++ y C. La generación del código C se suministra con varios algoritmos fijos el cual se pasa por la simulación para asegurar que va a ejecutar en tiempo real. Los métodos de Euler y Runge Kutta están soportados por defecto.

- **Plantillas.** Todas las plantillas de código C-están abiertas y pueden ser adaptados por el usuario para asignar compiladores, ejecutar FTP de sesiones y automatizar casi todo, entre la generación de código de 20-sim y el funcionamiento real del código en una máquina (remota). Durante la generación de algún programa en código C de soporte se proporciona en 20-sim, operadores para cálculos con matrices y funciones trigonométricas.

3.6 Demostración del estudio de 20 SIM

Como ya se mencionó en el apartado, Evolución de 20 SIM, el mismo programa permite la simulación de sistemas dinámicos pertenecientes a múltiples dominios de la física mediante el empleo de íconos, diagramas de bloques, grafos y ecuaciones. Para los casos de estudio de esta tesis utilizaremos diagrama de bloques y ecuaciones, de los cuales compararemos la respuesta de salida y de esta manera comprobaremos la exactitud de simulación.

Para un pequeño caso de estudio y sólo para una introducción de cómo usar el modelado de diagramas de bloques y ecuaciones, se trabajará con la siguiente ecuación.

$$G(s) = \frac{y(s)}{U(s)} = \frac{s+2}{s^2+8s+4} \quad (3.1)$$

Para poder modelar la ecuación (3.1) es necesario pasar a su forma matricial quedando de la siguiente manera.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u \quad (3.2)$$

Donde los valores para el caso de estudio de la ecuación (3.2) son los siguientes parámetros.

$$a_{11} = -10$$

$$a_{22} = -4$$

$$a_{12} = -1$$

$$a_{21} = 2$$

$$b_1 = 2$$

$$b_2 = 1$$

u = escalón

Con la ecuación (3.2) y los parámetros anteriormente mencionados se pasa a elaborar el diagrama de bloques en 20 SIM, en la Figura 3.8, encontraremos el área de trabajo y la librería, donde se encuentran las herramientas de trabajo que permitirán realizar el diagrama.

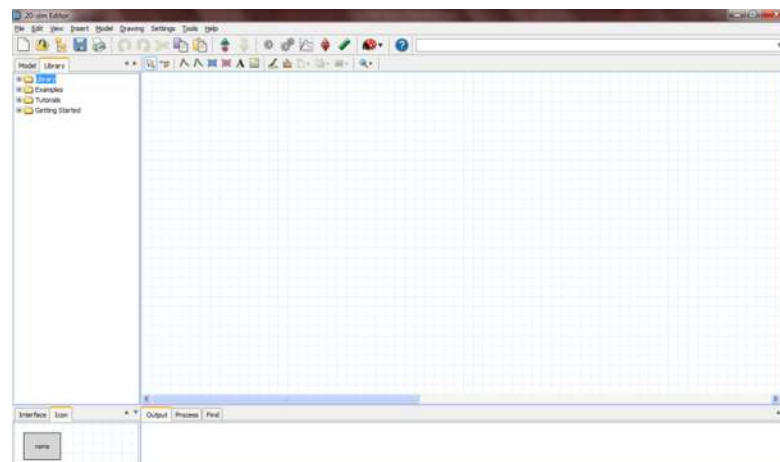


Figura 3.8 Área de trabajo 20 SIM

Para encontrar las herramientas o bloques para este modelado nos dirigimos a *Library/Signal/BlockDiagram*, de esta manera como se muestra en la Figura 3.9 aparecerán en orden alfabético.

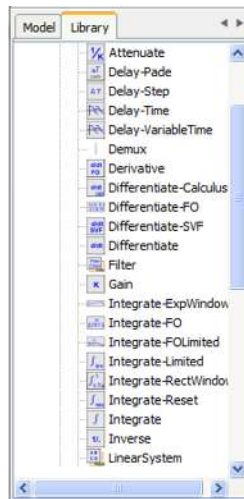


Figura 3.9 Listado de bloques

Para poder ser usados los bloques basta con seleccionar el bloque deseado y arrastrarlo al área de trabajo. Para modelar la ecuación (3.2), en diagrama de bloques es necesario un *Step* (fuente escalón), dos *Signal Monitor* (display), dos *Integrate*, seis *gain*, dos *Plusminus* y cinco *splitter* y ordenarlos de tal manera que sea fácil la conexión de los elementos como se muestra en la Figura 3.10.

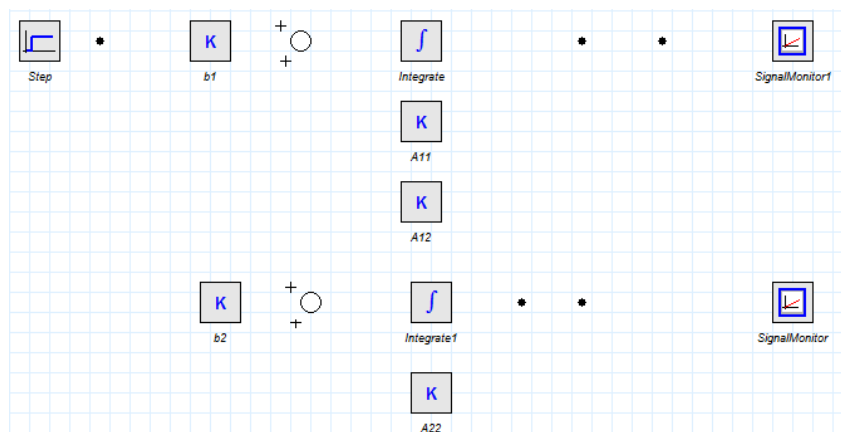


Figura 3.10 Orden de los bloques

Para que las ganancias tengan el mismo nombre de los parámetros ya mencionados, basta con seleccionar cada elemento (k), dar clic derecho, seleccionar propiedades y en nombre, cambiarlo por el deseado. De igual forma para que cada ganancia tenga el valor de

los parámetros antes mencionados, es necesario dar doble clic en la ganancia y aparecerá el editor de programación, siendo el código del mismo, como se muestra el Figura 3.11, el valor es cambiado con solo editarlo, al deseado.

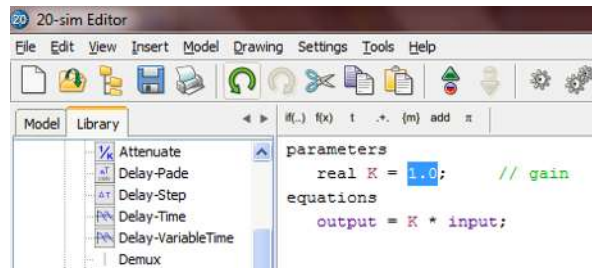


Figura 3.11 Editor del código de bloque

Una vez que las ganancias tengan sus nombres y valores definidos en base a los parámetros ya suscitados, pasamos a las conexiones de los elementos los cuales debe de ser como se muestra en la Figura 3.12, para ello seleccionamos *connect* (spacebartoggles), que se encuentra en la barra de herramientas de 20 SIM, ya seleccionada esta opción basta con dar un clic en el bloque origen y después otro clic en el bloque destino, quedando lista la conexión entre los elementos. En el caso de que la conexión, sea el bloque destino PlusMinus, el software preguntara si desea que la operación sea una suma o resta, en el caso de este ejemplo toda operación será suma.

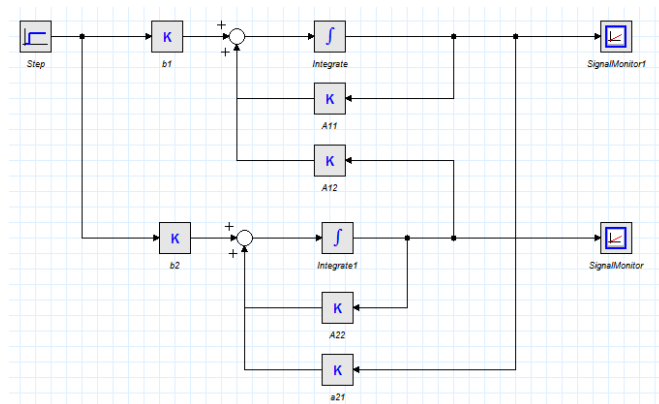


Figura 3.12 Conexión de los bloques

Una vez lograda la conexión, pasamos a compilar el modelado con la opción *Start Simulator* que se encuentra en las opciones de 20 SIM. En primera instancia el programa

deberá ser ejecutado sin ningún error de lo contrario el programa lo señalará ya sea en la conexión o en la programación de los bloques. Ya corregidos los errores como se muestra en la Figura 3.13 aparecerá el entorno donde se ejecutan las graficas, para correr la misma se da clic en *Run Simulation*.

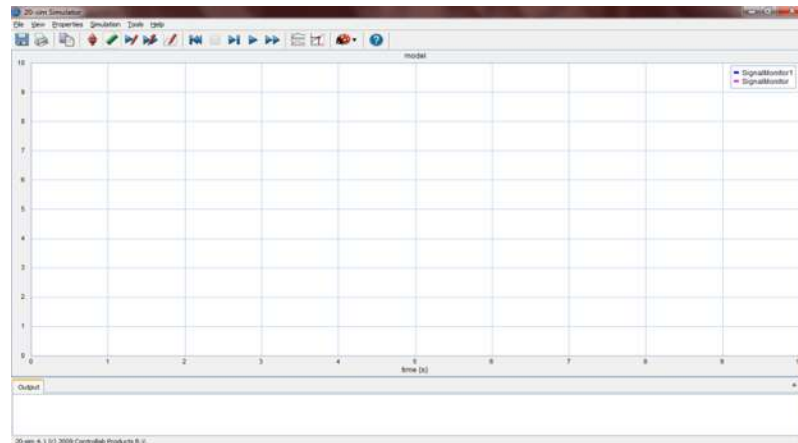


Figura 3.13 Área de Gráficas

En este ejemplo se ejecuta pero sin mostrar las gráficas de las dos variables de estado, debido a que en la fuente (step) no se ha señalado la amplitud, para esto se da doble clic en el mismo y se configura a uno, se guarda, se vuelve a compilar, y muestra la gráfica de la Figura 3.14.

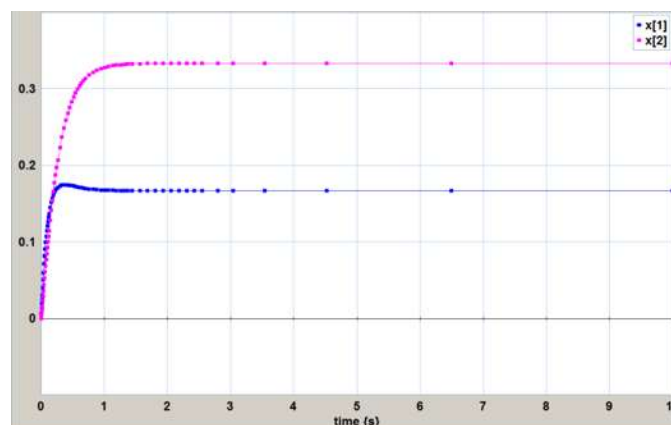


Figura 3.14 Gráfica del modelado

La gráfica de la figura 3.14 debe coincidir con el resultado de la gráfica del modelado de ecuaciones para comprobar la exactitud que se tiene entre los distintos diagramas que ofrece 20 SIM.

Para abrir el área de trabajo del modelado de ecuaciones como se muestra en la Figura 3.15, nos dirigimos a esta ruta: *File/New/EquationModel/*.

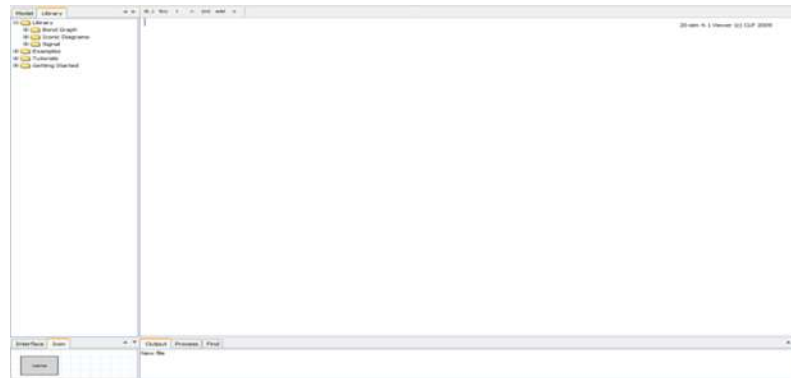


Figura 3.15 Área de trabajo para el modelado de ecuaciones.

Encontrando en el área de trabajo la opción “add”, en donde a través de la ruta *add/EquationExamples/Matrix*, permite establecer ecuaciones ya diseñadas, en donde se encuentra la necesaria para el modelado, como muestra la Figura 3.16.

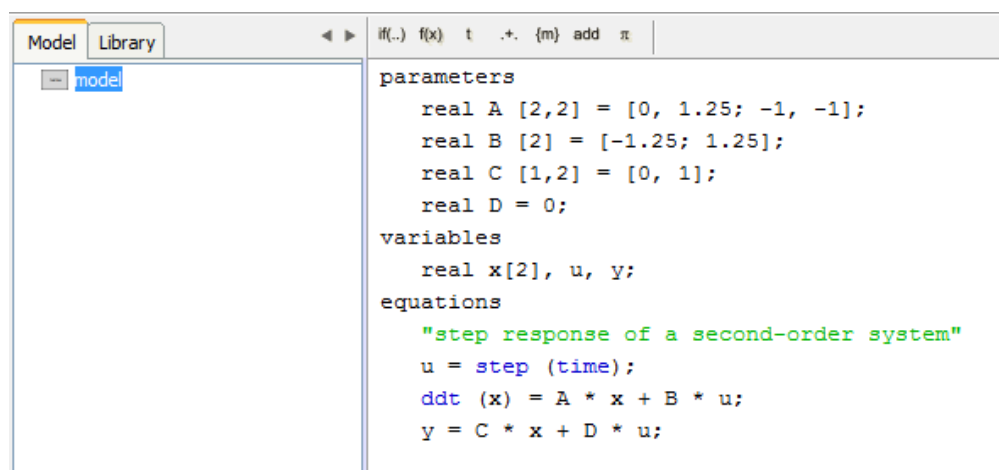


Figura 3.16 Plantilla para ecuaciones de Matrix.

En la plantilla de la Figura 3.16, sólo serán modificados los parámetros A, B, C y D, en base a los parámetros ya mencionados anteriormente, como se muestra en la Figura 3.17.

```

parameters
  real A [2,2] = [-10, -1; 2, -4];
  real B [2] = [2; 1];
  real C [1,2] = [1; 1];
  real D = 0;
variables
  real x[2], u, y;
equations
  "step response of a second-order system"
  u = step (time);
  ddt (x) = A * x + B * u;
  y = C * x + D * u;

```

Figura 3.17 Plantilla para ecuaciones de Matrix modificada.

Ya obtenido el modelado de ecuaciones, se pasa a compilarlo, previamente es necesario en este modelado ver cuáles son las variables de salida que se desean graficar en contra del tiempo, para ello estando en el área grafica, se selecciona *Plot/Add/X Vector(2)*, mostrando la grafica de la Figura 3.18.

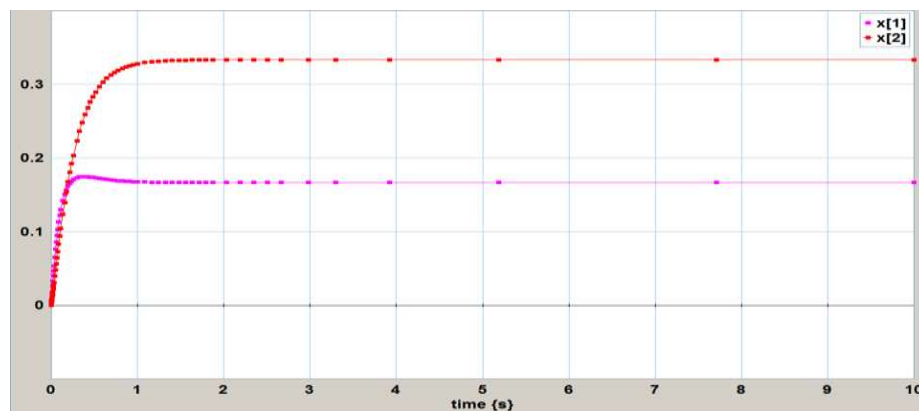


Figura 3.18 Gráfica del modelado de ecuaciones.

De esta manera obtenidas las gráficas de los dos modelados pasamos a comparar, como se puede observar en la Figura 3.19, a simple vista se denota una gran similitud, en la

gráfica del lado derecho corresponde al diagrama de ecuaciones y la del lado izquierdo al diagrama de bloques.

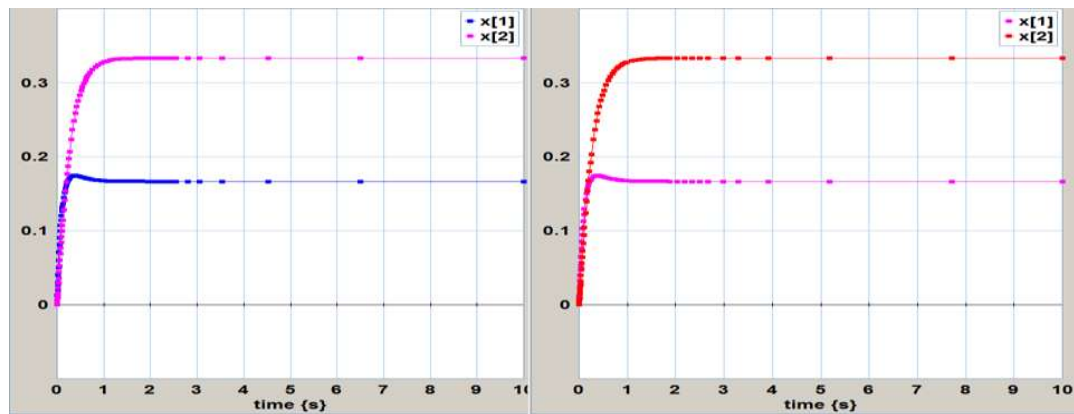


Figura 3.19 Comparación de las gráficas.

Para una comparación más exacta, en el área donde se dan las gráficas, seleccionamos *Variables*. Para el modelado de bloque aparecerán todas las variables definidas, donde se debe buscar las variables de salida y en el modelado de ecuaciones basta con seleccionar las variables *x*, mostrando los siguientes resultados.

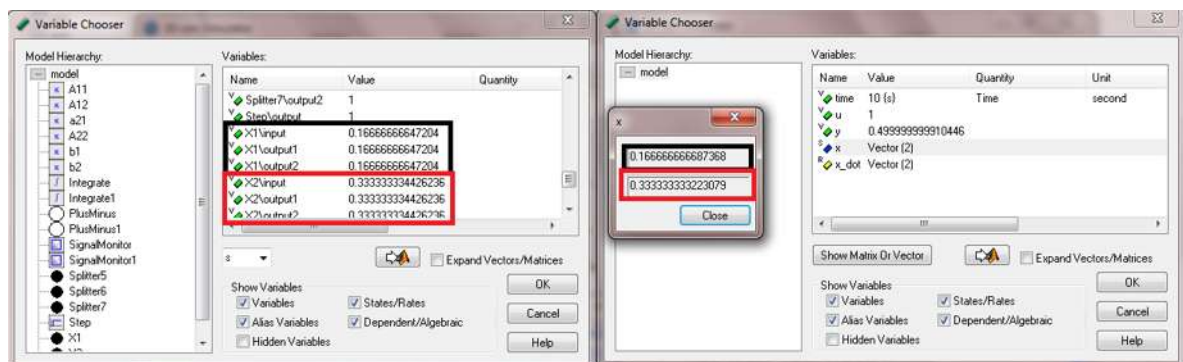


Figura 3.20 Comparación del resultado de las gráficas.

De esta manera como se puede apreciar en la figura 3.20, los valores de las variables *x1* y *x2* del diagrama de ecuaciones y de bloques, coinciden en los primeros 8 dígitos del punto hacia la derecha, mostrando la precisión matemática que se tienen.

Capítulo 4

Casos de estudio para 20 SIM

4.1 Introducción

Después de haber realizado una descripción detallada de 20 SIM y de haber realizado pruebas con un caso de estudio simple, en este capítulo se pasarán hacer más pruebas con sistemas físicos complejos que permitirán entender el funcionamiento completo de 20 SIM, en base al diagrama de bloques y de ecuaciones, comparar las respuestas de salida, para así saber que tan exacta es la simulación en comparación una de la otra.

Cabe señalar que no se tendrá mucho énfasis en la explicación del desarrollo de las ecuaciones debido a que no es el objetivo de esta tesis y al igual no se detallará el cómo se desarrollaron los modelos en 20 SIM y sus resultados, debido a que ya fue explicado a detalle en el apartado 3.6 “Demostración del estudio de 20 SIM”.

4.2 Primer caso de estudio

Para el primer caso de estudio tomaremos el sistema mostrado en la Figura 4.1.

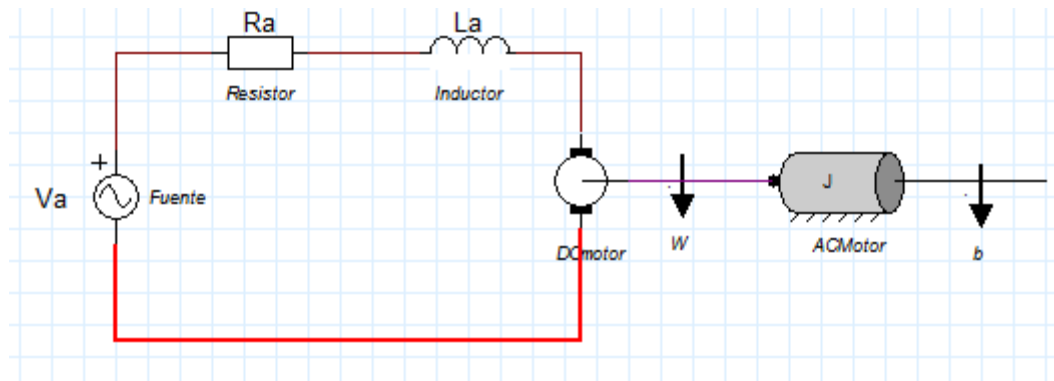


Figura 4.1 Primer sistema

La ecuación matemática del sistema es:

$$\begin{bmatrix} \dot{\lambda a} \\ \dot{h j} \end{bmatrix} = \begin{bmatrix} -\frac{Ra}{La} & -\frac{n}{j} \\ \frac{n}{La} & -\frac{b}{j} \end{bmatrix} \begin{bmatrix} \lambda a \\ h s \end{bmatrix} + \begin{bmatrix} i \\ o \end{bmatrix} V a \quad (4.1)$$

$$x = \begin{bmatrix} \lambda a \\ h s \end{bmatrix} \quad (4.2)$$

$$z = \begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} \quad (4.3)$$

$$\begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \frac{1}{La} & 0 \\ 0 & \frac{1}{j} \end{bmatrix} \begin{bmatrix} \lambda a \\ h s \end{bmatrix} \quad (4.4)$$

$$z = Fx \quad (4.5)$$

$$\dot{z} = F\dot{x} \quad (4.6)$$

Para poder modelar el sistema en base a las plantillas con las que cuenta el software, se debe simplificar la ecuación, de la siguiente manera:

$$\begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \frac{1}{La} & 0 \\ 0 & \frac{1}{j} \end{bmatrix} \begin{bmatrix} -\frac{Ra}{La} & -\frac{n}{j} \\ \frac{n}{La} & -\frac{b}{j} \end{bmatrix} \begin{bmatrix} La & 0 \\ 0 & j \end{bmatrix} \begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} \frac{1}{La} & 0 \\ 0 & \frac{1}{j} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} V a \quad (4.7)$$

$$\begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \frac{1}{La} & 0 \\ 0 & \frac{1}{j} \end{bmatrix} \begin{bmatrix} -Ra & -n \\ n & -b \end{bmatrix} \begin{bmatrix} i \ddot{a} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} \frac{1}{La} \\ 0 \end{bmatrix} V a \quad (4.8)$$

$$\begin{bmatrix} \dot{ia} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \frac{1}{La} & 0 \\ 0 & \frac{1}{J} \end{bmatrix} \begin{bmatrix} -Ra & -n \\ n & -b \end{bmatrix} \begin{bmatrix} ia \\ w \end{bmatrix} + \begin{bmatrix} \frac{1}{La} \\ 0 \end{bmatrix} Va \quad (4.9)$$

Sirviendo de referencia la ecuación (4.9) para modelar el diagrama de bloque y de ecuación, se utilizaron los siguientes parámetros:

$$La = .1 H$$

$$B = .1 \frac{N.s}{m}$$

$$Ra = .5 \Omega$$

$$J = 1.5 N.m.s^2$$

$$n = 3$$

$$Va = 10v$$

4.2.1 Diagrama de bloques para el primer caso de estudio

La ecuación (4.9) del primer sistema queda representada en base al diagrama de bloques en 20 SIM, con los parámetros ya mencionados, como se muestra a continuación:

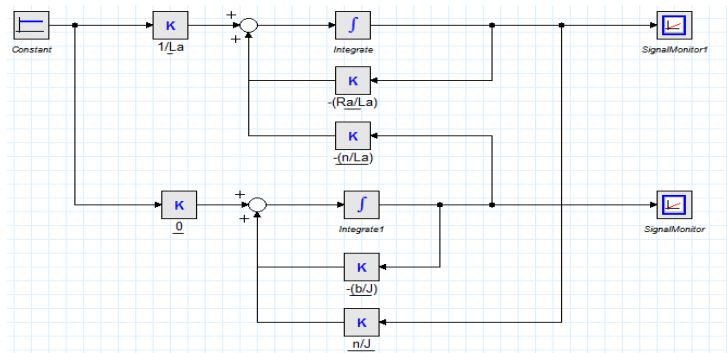


Figura 4.2 Diagrama de bloques del primer caso de estudio

Siendo la gráfica de la Figura 4.3 el resultado de compilar correctamente el diagrama de bloques mostrado anteriormente, correspondiendo las variables de estado $x[2]$ a i_a y $x[1]$ a ω , como se mostro en la ecuación 4.9.

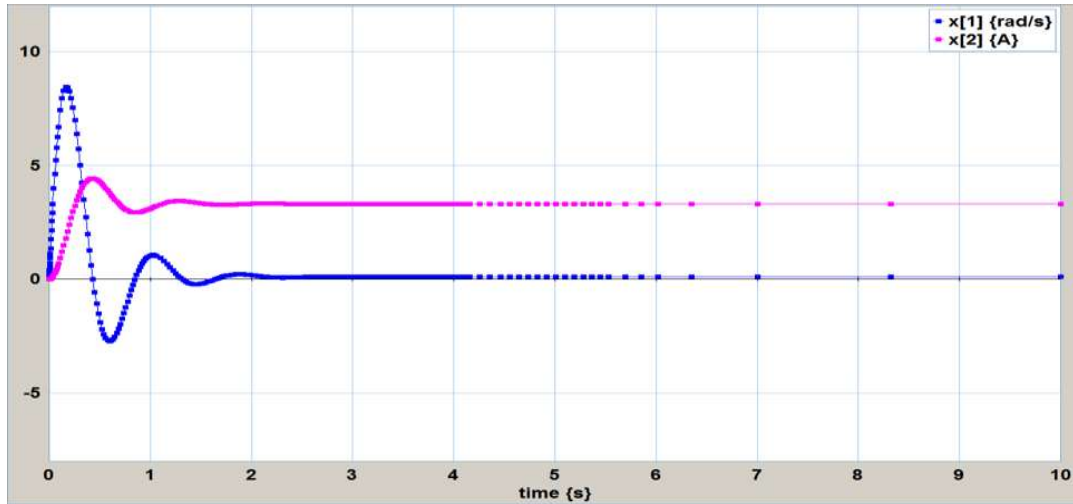


Figura 4.3 Gráfica del primer caso de estudio en base al diagrama de bloques

4.2.2 Diagrama de ecuaciones para el primer caso de estudio

Ahora la ecuación (4.9) del primer sistema queda representada en base al diagrama de ecuaciones en 20 SIM, con los parámetros ya mencionados, como se muestra a continuación:

```
parameters
  real A [2,2] = [-5.0, -30.0;2.0, -0.0666666666666667];
  real B [2] = [10.0;0.0];
  real C [1,2] = [1; 1];
  real D = 0;
variables
  real x[2], u, y;
equations
  "step response of a second-order system"
  u = 10;
  ddt (x) = A * x + B * u;
  y = C * x + D * u;
```

Figura 4.4 Diagrama de ecuaciones para el primer caso de estudio

Siendo la gráfica de la Figura 4.5 el resultado de copilar correctamente el diagrama de ecuaciones mostrado anteriormente.

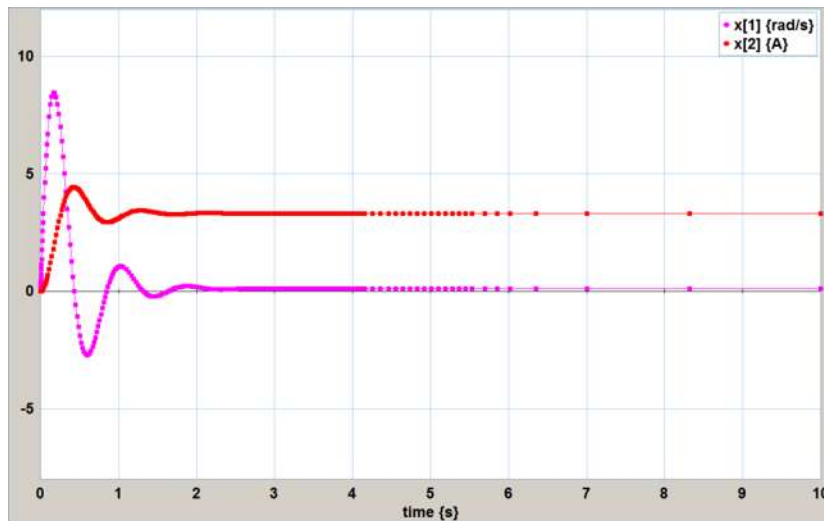


Figura 4.5 Gráfica del primer caso de estudio en base al diagrama de ecuaciones

4.2.3 Comparación de resultados del primer caso de estudio

Después de haber obtenido la respuesta del sistema en base a los dos diagramas ya mencionados, se pasa a compararlos. La primera comparación se hace de manera simple mostrando las dos gráficas en una misma imagen donde se puede apreciar la similitud de éstas.

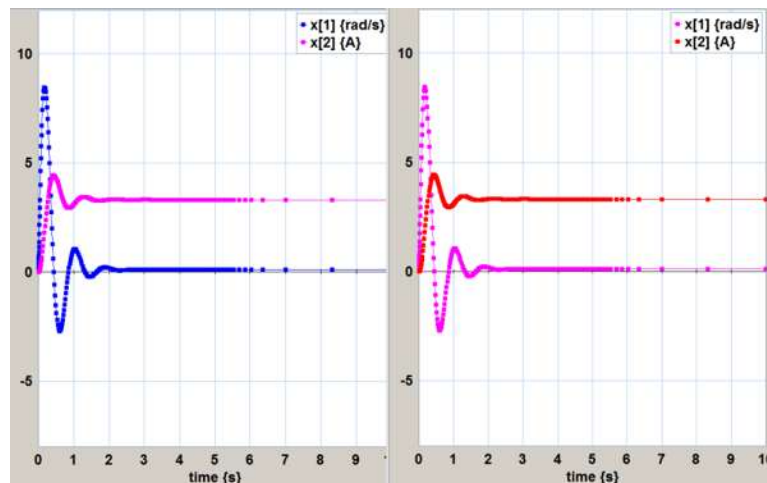


Figura 4.6 Comparación de gráficas de los modelados del primer caso de estudio

Para una comparación más precisa se muestran los datos numéricos de las gráficas donde se puede notar la precisión de esta con un margen de error muy pequeño, debido a como se puede apreciar en la figura 4.7, los valores de las variables x_1 y x_2 del diagrama de ecuaciones y de bloques, coinciden en los primeros 8 dígitos del punto hacia la derecha, mostrando la precisión matemática que se tienen.

Name	Value	Quantity
Step\output	1	
X1\input	0.110497237810664	
X1\output1	0.110497237810664	0.110497231600962
X1\output2	0.110497237810664	
X2\input	3.31491712728384	3.31491712412558
X2\output1	3.31491712728384	
X2\output2	3.31491712728384	
_input	10	

Figura 4.7 Comparación de graficas de los modelados en base a datos numéricos

4.3 Segundo caso de estudio

Para el segundo caso de estudio tomaremos el sistema mostrado en la Figura 4.8.

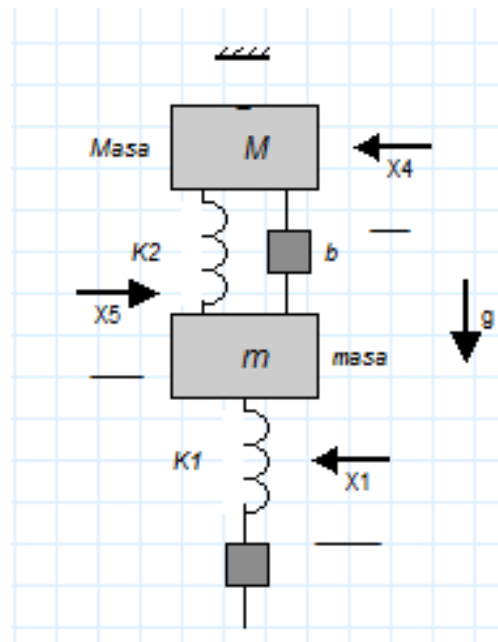


Figura 4.8 Segundo sistema

La ecuación matemática del segundo sistema es:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{m} & 0 & 0 \\ k_1 & \frac{-b}{m} & -k_2 & \frac{b}{M} \\ 0 & \frac{1}{m} & 0 & \frac{-b}{M} \\ 0 & \frac{b}{m} & k_2 & \frac{-b}{M} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (4.10)$$

Sirviendo de referencia la ecuación (4.10) para modelar el diagrama de bloque y de ecuación, se utilizaron los siguientes parámetros:

$$m = .5 \text{ kg}$$

$$k_1 = 1.0 \frac{N}{m}$$

$$b = 1.8 \frac{N \cdot s}{m}$$

$$k_2 = 1.4 \frac{N}{m}$$

$$M = 1.1 \text{ kg}$$

4.3.1 Diagrama de bloques para el segundo caso de estudio

La ecuación (4.10) del segundo sistema queda representada en base al diagrama de bloques en 20 SIM, con los parámetros ya mencionados, como se muestra a continuación:

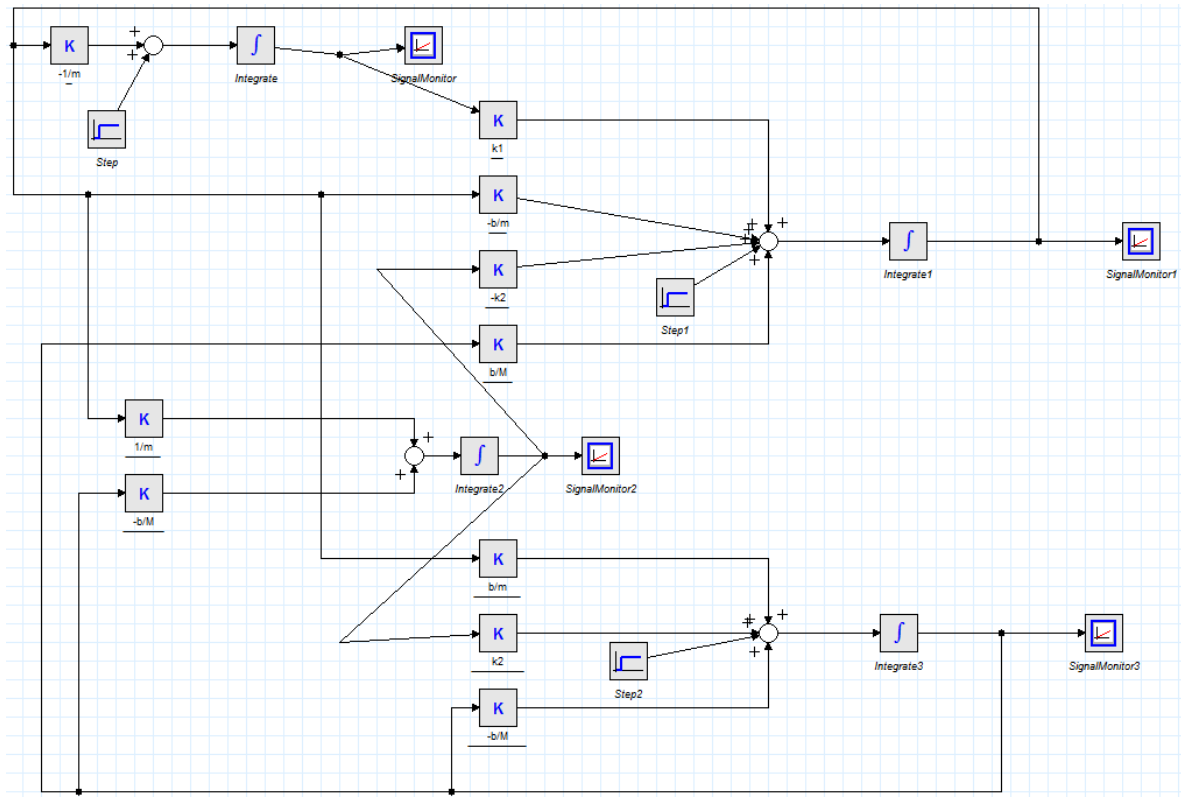


Figura 4.9 Diagrama de bloques del segundo caso de estudio

Siendo la gráfica de la Figura 4.10 el resultado de compilar correctamente el diagrama de bloques mostrado anteriormente.

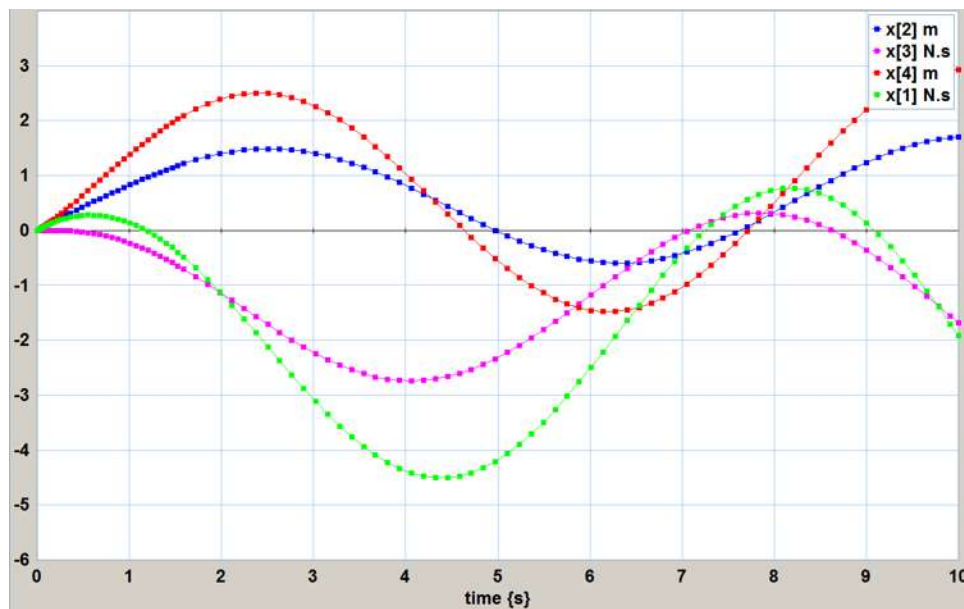


Figura 4.10 Gráfica del segundo caso de estudio en base al diagrama de bloques

4.3.2 Diagrama de ecuaciones para el segundo caso de estudio

Ahora la ecuación (4.13) del segundo sistema queda representada en base al diagrama de ecuaciones en 20 SIM, con los parámetros ya mencionados, como se muestra a continuación:

```
parameters
  real A [4,4] = [0.0, -2.0, 0.0, 0.0;1.0, -3.6, -1.4, 1.6363;0.0, 2.0, 0.0, -1.6363;0.0, 3.6, 1.4, -1.6363];
  real B [4,3] = [1,0,0;0,1,0;0,0,0;0,0,1];
  real C [1,4] = [1,1,1,1];
  real D = 0;
variables
  real x[4], u1,u2,u3, y; "Variables de enetrada U y Y salida"
equations
  "step response of a second-order system"
  u1 = step (time);
  u2 = step (time);
  u3 = step (time);

  ddt (x) = A * x + B * [u1;u2;u3];
  y = C * x ;
```

Figura 4.11 Diagrama de ecuaciones del segundo caso de estudio

Siendo la gráfica de la Figura 4.12 el resultado de compilar correctamente el diagrama de ecuaciones mostrado anteriormente.

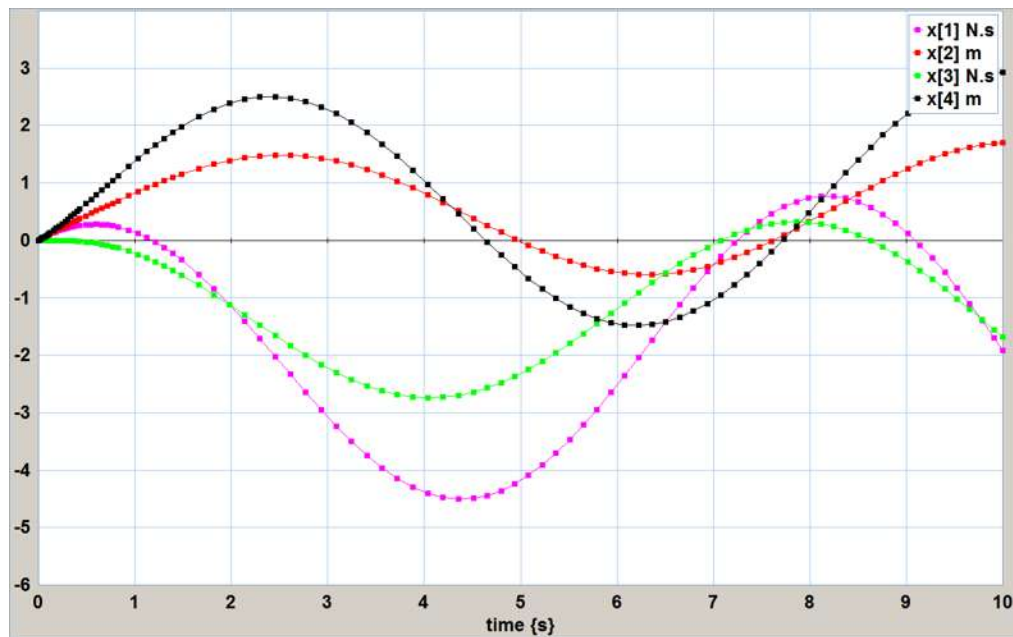


Figura 4.12 Gráfica del segundo caso de estudio en base al diagrama de ecuaciones

4.3.3 Comparación de resultados del segundo sistema

Después de haber obtenido la respuesta del sistema en base a los dos modelados ya mencionados, se pasa a compararlos. La primera comparación se hace de manera simple mostrando las dos gráficas en una misma imagen donde se puede apreciar la similitud de estas, donde del lado derecho se muestra el diagrama de ecuaciones y el izquierdo de bloques.

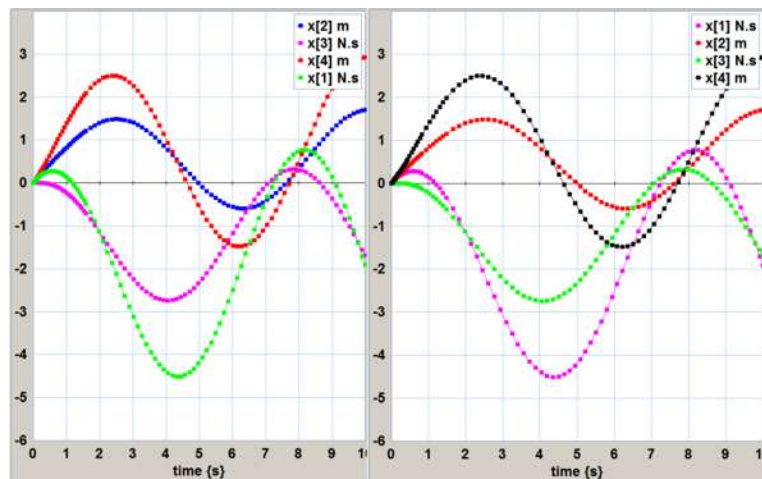


Figura 4.13 Comparación de gráficas de los modelados del segundo sistema

Para una comparación más precisa se muestran los datos numéricos de las gráficas donde se puede notar la precisión de esta con un margen de error muy pequeño, debido a como se puede apreciar en la figura 4.14, los valores de las variables x_1 , x_2 , x_3 y x_4 del diagrama de ecuaciones y de bloques, coinciden en los primeros 5 dígitos del punto hacia la derecha, mostrando la precisión matemática que se tienen.

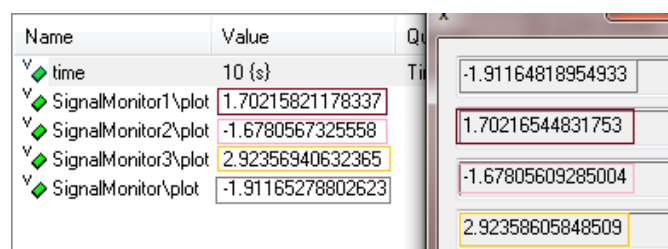


Figura 4.14 Comparación de gráficas de los modelados del segundo caso en base a datos numéricos

4.4 Tercer caso de estudio

Para el tercer caso de estudio tomaremos el sistema mostrado en la Figura 4.15.

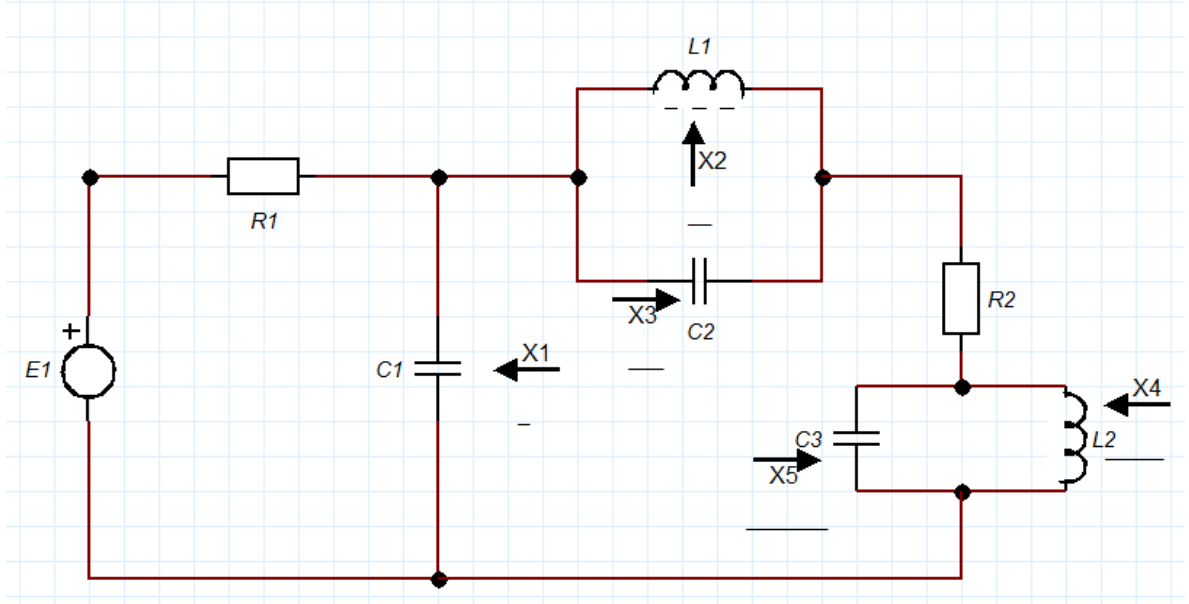


Figura 4.15 Tercer sistema

La ecuación matemática del tercer sistema es:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} \frac{-1}{c_1} \left(\frac{1}{R_1} + \frac{1}{R_2} \right) & 0 & \frac{1}{C_2 R_2} & 0 & \frac{1}{C_3 R_2} \\ 0 & 0 & \frac{1}{C_2} & 0 & 0 \\ \frac{1}{R_2 C_1} & \frac{-1}{L_1} & \frac{-1}{R_2 C_2} & 0 & \frac{-1}{R_2 C_3} \\ 0 & 0 & 0 & 0 & \frac{1}{C_3} \\ \frac{1}{R_2 C_1} & 0 & \frac{-1}{R_2 C_2} & \frac{-1}{L_2} & \frac{-1}{R_2 C_3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} \frac{1}{R_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.11)$$

Sirviendo de referencia la ecuación (4.11) para modelar el diagrama de bloque y de ecuación, se utilizaron los siguientes parámetros:

$$R_1 = .5 \, \Omega$$

$$R_2 = .2 \, \Omega$$

Siendo la gráfica de la Figura 4.17 el resultado de compilar correctamente el diagrama de bloques mostrado anteriormente.

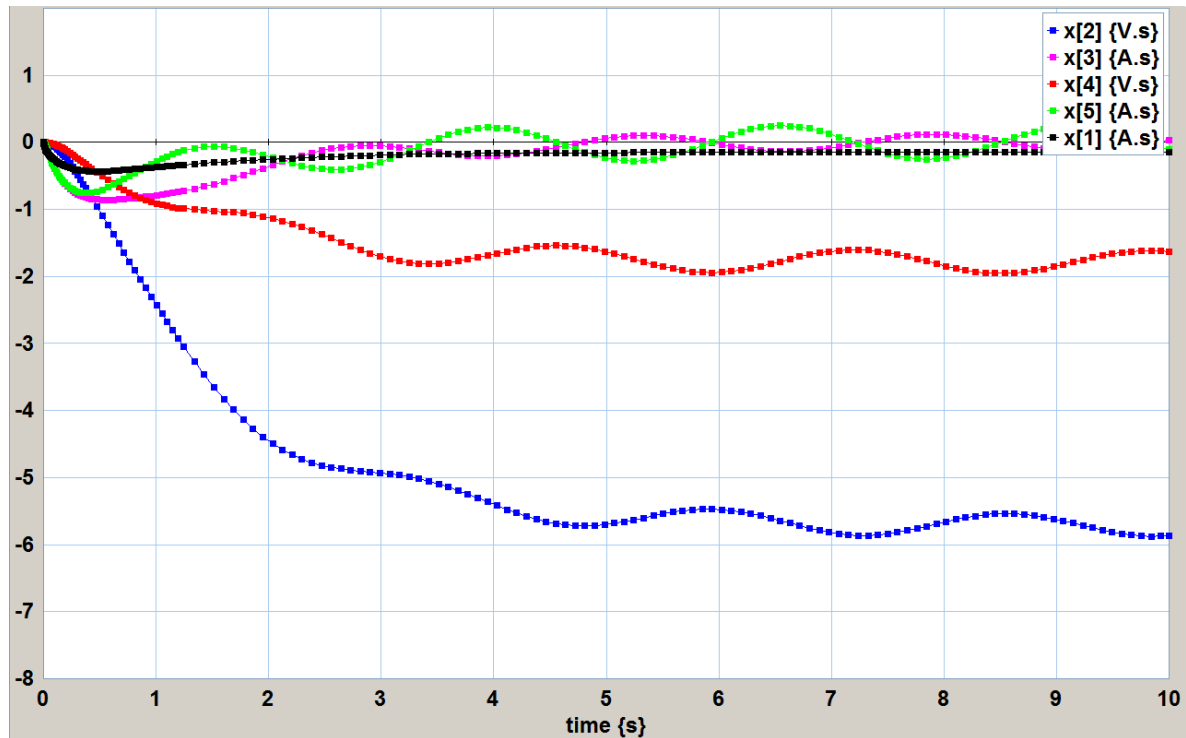


Figura 4.17 Gráfica del tercer caso de estudio en base al diagrama de bloques

4.4.2 Diagrama de ecuaciones para el tercer caso de estudio

Ahora la ecuación (4.11) del tercer sistema queda representada en base al diagrama de ecuaciones en 20 SIM, con los parámetros ya mencionados, como se muestra a continuación:

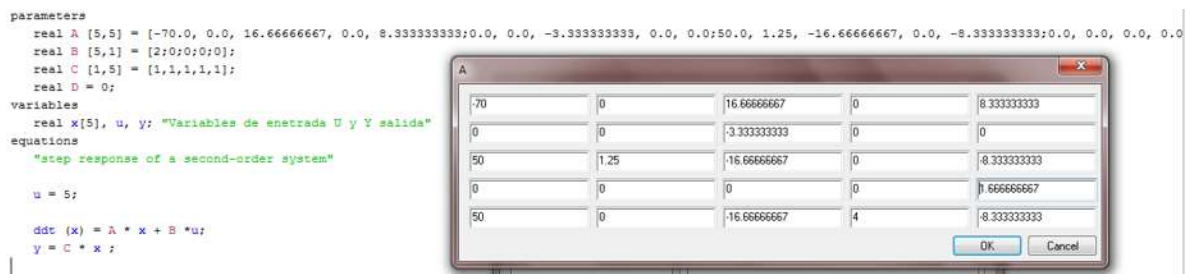


Figura 4.18 Diagrama de ecuaciones para el tercer caso de estudio

Siendo la gráfica de la Figura (4.19) el resultado de compilar correctamente el diagrama de ecuaciones mostrado anteriormente.

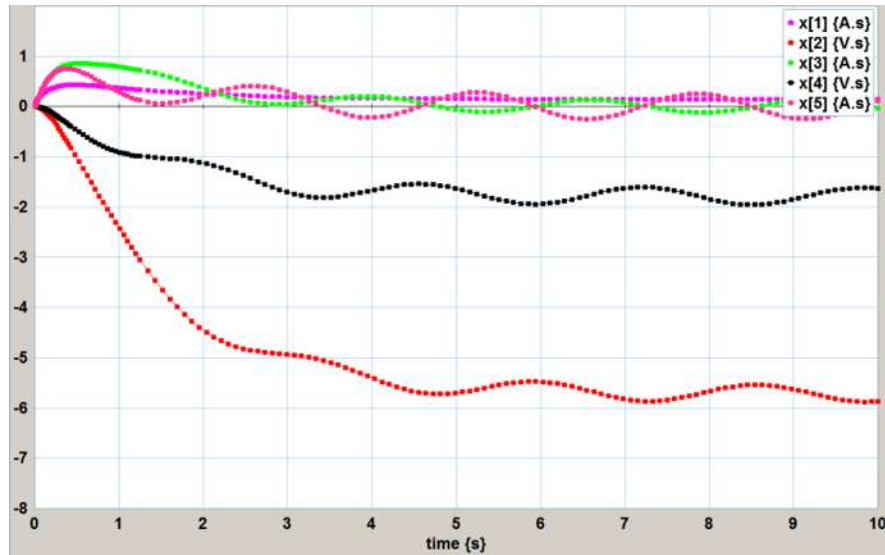


Figura 4.19 Gráfica del tercer caso de estudio en base al diagrama de ecuaciones

4.3.3 Comparación de resultados del tercer caso de estudio

Después de haber obtenido la respuesta del sistema en base a los dos modelados ya mencionados, se pasa a compararlos. La primera comparación se hace de manera simple mostrando las dos graficas en una misma imagen donde se puede apreciar la similitud de estas, siendo la del lado derecha la gráfica del diagrama de ecuaciones y la izquierda al de bloques.

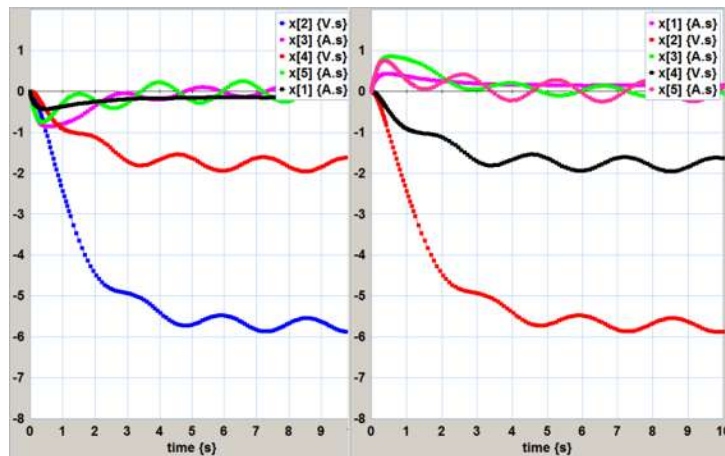


Figura 4.20 Comparación de gráficas de los modelados del tercer sistema

Para una comparación más precisa se muestran los datos numéricos de las graficas donde se puede notar la precisión de esta con un margen de error muy pequeño, debido a como se puede apreciar en la figura 4.21, los valores de las variables x1, x2, x3, x4, x5 del diagrama de ecuaciones y de bloques, coinciden en los primeros 10 dígitos del punto hacia la derecha, mostrando la precisión matemática que se tienen.

Variables:			
Name	Value	Quantity	L
time	10 {s}	Time	s
SignalMonitor1\plot	-5.86734040324065		
SignalMonitor2\plot	0.0377836047792374		
SignalMonitor3\plot	-1.63528596330956		
SignalMonitor4\plot	0.110581022035544		
SignalMonitor\plot	0.147063732056841		

0.147063732056946
-5.8673404032446
0.0377836047729887
-1.63528596330679
0.110581022023872

Figura 4.21 Comparación de gráficas de los modelados del tercer caso en base a datos numéricos

Capítulo 5

Conclusiones y recomendaciones

5.1 Conclusiones

Después de que se ha fundamentado el marco teórico se puede concluir, que se cumple con el objetivo y justificación de este trabajo porque se da el conocimiento de la historia de la simulación, la evolución de 20 SIM, todas las características con las que cuenta, cual es su campo de aplicación y por último se propuso el modelado y la simulación de los sistemas equivalentes, así como un procedimiento definido para realizar el funcionamiento de los mismos. La simulación de estos sistemas permite verificar su exactitud y el estudio de las herramientas con las que cuenta el software 20 SIM en el cual, se pudo observar que es un paquete computacional muy versátil debido a que se pueden simular sistemas eléctricos, mecánicos, hidráulicos y térmicos; así como sistemas dinámicos o físicos con la técnica de Bond Graph. La representación de los elementos de los sistemas físicos en 20 SIM, depende de su ecuación característica, pero también es posible crear elementos con la causalidad deseada, tanto su ecuación, tipo sistema, puertos de salida y entrada; así como su icono que lo simboliza.

Tras haber realizado tres casos de estudio para tres sistemas diferentes, simulados mediante el modelo de diagramas de bloques y ecuaciones, las cuales al momento de comparar el resultado de un modelo al otro, fue una simulación casi exacta, con un margen de error muy pequeño, con la cual concluimos que el usuario puede tener la seguridad de obtener resultados muy precisos sin importar que modelo desee trabajar o más se acople, dependiendo de las necesidades del usuario.

Dada toda esta información referente a 20 SIM, el estudiante, ingeniero o profesor podrá tener este trabajo como un marco de apoyo para las actividades o conocimientos que se deseen tener sobre el software. Debido a los pocos manuales, libros, tesis o foros web en

español que existen sobre el funcionamiento o descripción de 20 SIM, esto se debe a que el programa cuenta con licencia privada, sólo en LABSON ofrece licencias gratuitas para el estudio del programa. Tener un paquete que nos permita modelar y simular es una actividad indispensable cuando nos enfrentamos con el análisis y diseño de sistemas multidisciplinares de cierta complejidad con lo observamos en esta tesis.

Uno de los objetivos de esta tesis es tener un material de apoyo para dar el soporte necesario al diseñador durante el proceso de diseño, análisis y diagnosis de sistemas ingenieriles.

5.2 Recomendaciones

El software sólo complementa el talento del diseñador para que pueda modelar y simular de la forma más eficientemente posible.

El software sólo hace posible establecer una valoración final antes de que los sistemas sean contruidos, y pueden aliviar la necesidad de experimentos caros y dar soporte a todas las etapas de un proyecto desde el diseño conceptual, pasando por el montaje hasta llegar a su funcionamiento

Referencias

Libro:

1. [Tomas H.1982]

Tomas, Técnicas de simulación en computadores, México: Limusa, 1982.

2. [D'Angelo 1970]

H. D'Angelo, Linear Time-Varying Systems: Analysis and Synthesis, Boston: Allyn and Bacon, 1970.

3. [Broenink J.F. 1999]

Broenink J.F, 20-sim software for hierarchical bond-graph/bolck diagram models, 1999.

4. [Karnopp y Rosenberg. 1968]

Karnopp y Rosenberg, System dynamics /a unified approach, 1999.

Internet:

5. [20-SIM 2006]

20-SIM. Página principal. Estados Unidos. 17 de mayo de 2006. <http://www.20-SIM.org>