

Capítulo 4.- TÉCNICAS DE MODELADO Y SIMULACIÓN

4.1. Importancia de la simulación

En la actualidad, la modelización y la simulación es una actividad indispensable cuando nos enfrentamos con el análisis y diseño de sistemas multi disciplinares de cierta complejidad.

El objetivo es ayudar o dar el soporte necesario al diseñador durante el proceso de diseño, análisis y diagnosis de sistemas ingenieriles. El software debe complementar el talento del diseñador para que éste pueda modelar y simular de forma lo mas eficientemente posible.

El software hace posible establecer una valoración final antes de que los sistemas sean construidos, 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 monta je hasta llegar a su funcionamiento.

4.1.1. Evolución

Hay una amplia literatura sobre la simulación. En el artículo de Cellier y otros [1995] se puede leer un excelente resumen sobre la evolución de las técnicas (métodos y lenguajes) de la modelización y la simulación.

Por otra parte, la SCS (Society for Computer Simulation) publica cada año una lista actualizada del software disponible.

En la tabla 4.1 se ha pretendido resumir los hitos más importantes. En la tabla 4.2 se hace referencia de algunos de los programas informáticos comercializados que tiene relación directa con la simulación de sistemas oleohidráulicos. En la tabla 4.1 se han incluido también algunos de los avances tecnológicos más significativos habida cuenta que en algunos periodos la tecnología ha sido un factor clave en el desarrollo de la simulación. Sin embargo, es interesante observar que las ideas cambian más lentamente que la tecnología.

Es solamente a partir de los 90 que se hace necesario un cambio de paradigma.

Este cambio ha sido motivado por los usuarios.

Los usuarios requieren:

1. la simulación de sistemas complejos multi disciplinares



- 2. la programación avanzada orientada al objeto,
- 3. software para la resolución de sistemas diferenciales algebraicos,
- 4. la computación simbólica y
- 5. métodos gráficos avanzados

Las metodologías modernas se construyen sobre la base de una modelizacion NO causal con ecuaciones matemáticas y el empleo de construcciones orientadas al objeto para facilitar la reutilización de conocimiento modelado.

4.1.2. Simulación Analógica

Los primeros simuladores fueron analógicos. La idea es modelar un sistema en términos de ecuaciones diferenciales ordinarias y después hacer un dispositivo físico que obedezca a las ecuaciones. El sistema físico se inicializa con valores iniciales apropiados y su desarrollo en un cierto plazo que simula la ecuación diferencial.

Inicialmente se desarrollaron analizadores diferenciales mecánicos como herramienta de propósitos generales para simular sistemas dinámicos los cuales fueron reemplazados por sistemas electrónicos.

La simulación analógica no puede tratar con ecuaciones diferenciales algebraicas (EDAs), sólo con ecuaciones diferenciales ordinarias (EDOs), lo cual no es muy grave, según Broenink J. F (1999), las EDAs se dan cuando se simplifica demasiado el modelo.

4.1.3. Simulación Numérica

La solución numérica de una ecuación diferencial es un esencial ingrediente de la simulación numérica.

Hay varias maneras de encontrar soluciones de aproximación numérica para las ecuaciones diferenciales. Los métodos son basados en la idea de reemplazar las ecuaciones diferenciales por una ecuación de diferencia. El método de Euler es basado en aproximación de la derivativa por una diferencia de primer orden. Hay técnicas más eficientes tales como Runge-Kutta y métodos de múltiple pasos. Estos métodos fueron muy conocidos cuando emergieron los simuladores digitales en el año de 1960. El campo de las matemáticas numéricas experimentó un renacimiento debido al impacto de las computadoras digitales.



La integración numérica de ODEs y DAEs son campos muy activos de la investigación que continúan teniendo impacto fuerte en el modelado y simulación, considera Hairer y Wanner (1991). Entre el desarrollo interesante está el algoritmo mejorado, una estructuración mejor del código en donde se separan los algoritmos y el error de control. Los algoritmos para ecuaciones algebraicas todavía no están muy desarrollados como los algoritmos para ecuaciones diferenciales ordinarias.

4.1.4. Simulación Analógica Digital

Cuando las computadoras digitales aparecieron, era natural explorar si pudiesen ser usadas para la simulación. El desarrollo fue iniciado por Selfridge (1955) que demostró como una computadora digital puede emular un analizador diferencial. Este enfoque dio lugar a la aparición de numerosos lenguajes de programación. En este medio se desarrolló CSSL estándar [Strauss (ed) (1967)], fue un mayor hito, puesto que unificó los conceptos y estructuras de lenguaje de los programas disponibles de simulación. ACSL de Mitchell y Gauthier (1967), se basa en CSSL pero con ciertas modificaciones y mejoras. SIMNON fue desarrollado en la universidad de Lund iniciando en 1972 y distanciándose de CSSL estándar [Elmqvist (1975)].

El desarrollo en esta dirección sólo fue posible con la aparición de computadoras con buenas prestaciones en donde el PC con gráficos de tramas llegó a estar generalmente disponible.

Entre las aplicaciones de este tipo están VisSim [Darnell y Kolk (1990)]. Mitchell y Gauthier introdujeron el modelador gráfico ACSL en 1993 o SIMULINK.

BDSP (Block Diagram Simulation Program) es un programa de simulación usado en el entorno Windows desarrollado por la universidad de Gifu. Ofrece una librería que cubre un amplio rango de áreas físicas y aplicaciones que facilitan la construcción de los modelos. Este programa posee una rutina para identificar modelos a partir de la estimación experimental de su función de transferencia [Yamada y Muto (2001)].

SIMULINK (originalmente llamado SIMULAB) que se integra con MATLAB, apareció en 1991 [Grace (1991)]. Es especialmente diseñado para trabajar con diagramas de bloques usando MATLAB para el análisis dinámico del sistema. Los diagramas de bloques de SIMULINK pueden ser definidos como ecuación de estado.



A continuación se puede utilizar las funciones MATLAB para resolver numéricamente o procesarlas de diferentes maneras. De esta forma, es posible obtener la función de transferencia, el diagrama de Bode, margen de ganancia, etc. Este programa es muy útil en el campo de control automático.

4.1.5. Simuladores Específicos

Es posible diseñar el entorno del modelo, que son muy fáciles de usar, limitando el dominio del modelo. Inicialmente, estos programas solo abarcaban una rama de la ingeniería. Algunas herramientas de este tipo son: el sistema SPICE, el cual fue desarrollado para el modelo analógico de un circuito eléctrico, y DADS, desarrollado en la universidad de lowa (1984), para la simulación de sistemas mecánicos. Un gran número de herramientas de este tipo ha sido desarrollado en varias ramas de la ingeniería.

El programa Hysis, desarrollado por Mannesmann Rexroth, orientado al cálculo dinámico en el campo de la oleohidráulica, comprende cuatro subprogramas bien definidos para la simulación no lineal de circuitos electro-hidráulicos de regulación. Cada subprograma, disponibles para aplicaciones prácticas, permite simular un tipo diferente de sistema:

- 1) HYVOS: accionamientos que conciernen al actuador lineal controlado por una válvula
- 2) HYDRA: accionamientos que conciernen al actuador rotativo controlado por una válvula
- 3) HYSTA: accionamientos hidrostáticos
- 4) HYSEK: accionamientos con regulación secundaria

La ventaja del programa Hysis, según Murrenhoff (1998) es su notable facilidad de empleo siempre que el modelo desarrollado se pueda clasificar dentro de uno de los cuatros subprogramas.

4.1.6. Simuladores Multi-Disciplinares

El software para dominios específicos es muy útil de usar si el problema es adecuado a la herramienta directamente, siendo muy útiles en su campo de aplicación, pero hay muchos diseños que necesitan de programas de simulación que permitan la colaboración con otros programas.



Como ejemplo esté el programa ITI – SIM (ITI GMBH; Dresden Alemania), ver Klein y Grätz (2001). Es un software de simulación que cubre un amplio rango de áreas físicas y aplicaciones: hidráulica, neumática, mecánica, térmicas, transmisión electro – mecánica y bloque de señales. Este programa permite definir los modelos para facilitar su utilización. De esta forma la construcción de un modelo es adaptada a la descripción más común para cada disciplina de la ingeniería por medio de diagramas de circuitos hidráulicos, para dispositivos mecánicos y diagramas de bloques para la estructura de control. Incluye una librería con numerosos componentes, razón por la cual, la posibilidad de construir nuevos componentes definidos por el usuario, debe ser considerado como un rasgo importante. ITI – SIM proporciona un fichero DLL que permite modelar componentes especiales. El archivo DLL puede ser creado por cualquier compilador (Microsoft C++ ó MATLAB / SIMULINK).

4.1.7. Simulación Física

Un procedimiento físico para el modelado físico es dividir un sistema en subsistemas y explicar el comportamiento en los nodos de conexión. Cada subsistema puede tener numerosos niveles, en el cual, el nivel inferior representan elementos básicos que se agrupan para formar un subsistema de nivel superior. El modelo completo se obtiene agrupando los subsistemas.

Según Broenink (1999), se deben cumplir dos condiciones para garantizar que los submodelos sean acoplables:

- 1. Los puertos de conexión entre submodelos están definidos como pares de variables. El uso de pares de variables conjugadas en potencia hacen que las conexiones sean físicas, por ejemplo: par y velocidad, presión y caudal.
- Los submodelos han de estar escritos en estilo declarativo, es decir, estableciendo relaciones y no procedimientos para computar. Esta descripción también se llama causal.

Los submodelos 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 submodelos y de los puertos de conexión. Esto conduce, naturalmente, a las ecuaciones diferenciales algebraicas (EDAs).



Bond Graphs y Modelica estándar son ejemplos de programas que usan este tipo de modelo.

A) <u>Dymola</u>

Dynamic Modeling Languaje (Dymola) de Elmqvist (1978), fue un esfuerzo primitivo para ayudar al modelo físico. Actualmente utiliza el lenguaje Modelica y es deseado para el modelado de muchos dominios tales como circuitos eléctricos, sistemas termodinámicos, procesos químicos, sistemas de control, etc. Ofrece una librería para todos los dominios tecnológicos en lo que es aplicable. Para la mejor comprensión del sistema, los submodelos y sus conexiones están representados por iconos.

B) Bond Graphs

Bond graphs es dirigida a gráficas en donde los subsistemas son nodos y los flujos de energía en el sistema es presentados por ramas [Karnopp y Rosenberg (1968)]. Una de las ventajas de Bond Graphs es que puede pasar fácilmente de un dominio a otro a través de los elementos Transformer y Gyrator. Además se puede modificar fácilmente el modelo de un sistema añadiendo y/o quitando elementos. Algunos de los programas disponibles son los siguientes:

- Bond Graphs Simulation Program (BGSP) del laboratorio de Ingeniería Mecánica del Ministerio de Comercio e Industria de Japón. Es un programa de solución basado en Bond Graphs. Necesita de un programa externo para resolver numéricamente las ecuaciones de estado.
- 20-SIM fue desarrollado en la universidad de Twente, Holanda. Con este programa se puede simular el comportamiento de sistemas dinámicos tales como 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 Bond Graphs multipuerto, 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.



Los sistemas de modelado físico tienen el inconveniente, de que en principio, no se puede aplicar a sistemas con parámetros distribuidos. Sin embargo, en algunos casos, se puede obtener una solución aproximada dividiendo el componente continuo en numerosos elementos a los que se le asignan parámetros concentrados. Esta solución tiene varias limitaciones:

- 1. Exige mucha memoria y velocidad de proceso a la computadora
- 2. La solución obtenida solo es válida para bajas frecuencias. Esto implica que si aparecen modos propios en la simulación, solo son aceptables los que tienen la frecuencia más baja.

Periodo	Concepto	Desarrollos significativos
Años 20-50	 Simulación con técnicas analógicas basadas en las ecuaciones diferenciales ordinarias y diagramas de bloques. 	
	Evolución tecnológica del simulador mecánico al simulador electrónico	Analizador diferencial mecánico Bush y otros, en el MIT (1931) Analizador diferencial electrónico Ragazzini y otros (1947)
Años 60	Simuladores digitales	Selfridge (1995) demostró que un computador digital podía emular a un simulador analógico
	 Optimización de los métodos numéricos para resolución de las ecuaciones diferenciales ordinarias (basados en sustituir las ec. Diferenciales por ecuaciones en diferencias finitas. 	
	 Estabilidad de las aproximaciones de diferencias finitas 	
	 Ajuste automático de la longitud del paso de integración 	Fehlberg (1964)
	 Eclosión de los programas comerciales 	MIMIC (Wright Patterson-1965) DYNASAR (General Electric- 1965) DSL/90 (IBM-1965) CSMP (IBM-1968)



	 Unificación de los conceptos y estructuras de los lenguajes de los programas de simulación comerciales 	CSSL report (Strauss ed1967)
Años 70	 Métodos numéricos para ecuaciones diferenciales algebraicas (EDA) 	Métodos numéricos de integración de EDA, p.e Gear (1971)
	Nota: Los algoritmos de las ecuaciones diferenciales algebraicas aun no están tan bien desarrollados como las ecuaciones diferenciales ordinarias. Sigue siendo un campo de frenética investigación	Hairer and Wanner (1991)
	 Eclosión de programas comerciales basados en el CSSL Nota: en ellos es posible trabajar con la modelización de sistemas discretos y sistemas continuos en el tiempo 	Ejemplos: ACSL (Mitchell y Gauthier- 1976) SIMNON (Elmqvist-1975)
	 Desarrollo de entornos gráficos con el usuario. Empleo de tubos de rayos catódicos (CRT) y lápices luminosos para dibujar los diagramas de bloques. Nota: limitado por tecnología. 	Van der Bosch and Bruijin (1977) EASY5 (1976)
	 Manipulación simbólica de formulas Se utiliza para la convertir las ecuaciones EDA en EDO. Aplicación de las teorías de grafos 	Tarjan (1972) Wiberg (1977)
	 Primeras ideas sobre lenguajes orientados al objeto y descomposición por niveles jerárquicos 	Simula (Birtwistle y otros-1973)
Años 80	Desarrollo de entornos matriciales	SystemBuild integrado con MATRIX (1985) SIMULINK integrado con MATLAB (1991) VisSim (1990) ACSL Graphics (1993)
	 Primeros pasos hacia la modelización física (se requiere un nuevo paradigma para la modelización) 	Tutsim (Bond Graph) 20Sim (version actualizada del Tutsim, Broenink-1997) Dymola (Elmqvist –1978, primera versión comercial 1992)



	 Desarrollo de lenguajes orientados al objeto 	Omola (Andersson-1988)
	 Disponibilidad de máquinas con altas prestaciones: WorkStation y PC con potentes entornos gráficos 	
	 Disponibilidad de algoritmos numéricos para resolver EDA 	Brenan y otros (1989) Hairer y otros (1989)
Años 90	 Aparición de una amplia variedad de lenguajes con ideas similares a Dymola Nota: situación similar a los años 60 cuando se acordó definir el CSSL 	OmSim (Mattsson-1993) ASCEND (Piela y otros-1991) GPROMS (Sahlin y otros-1996) OjectMath (Fritzson y otros- 1995) SIDOPS+ (Broenink-1997) Smile (Kloas y otros-1995) ULM (Jeandel y otros-1996)
	 Unificación de técnicas e ideas de los diferentes programas de simulación con el objetivo de alcanzar un alto grado de conocimiento sobre la modelización física lenguajes orientados al objeto aplicación a diferentes dominios (multidisciplinar) empleo de una amplia variedad de formalismos: ODE, DAE, bond graph, redes de Petri, etc. 	Modelica(1997) Nota: con el Modelica se pretendió disponer de un formato estandar, tal que los modelos desarrollados en diferentes dominios puedan ser intercambiados entre herramientas y usuarios

Tabla 4.1 – Evolución del desarrollo de los programas de simulación



Programas	Año	Observaciones
HydCal		Nacido en el entorno científico es un software utilizado para dibujar basado en simbología normalizada ISO. Permite calcular las pérdidas, caudales, pérdidas de energía en las resistencias hidráulicas.
HydrauSim		Permite diseñar esquemas hidráulicos, comprobar su funcionamiento, pero no permite el cálculo en régimen transitorio
Hysys		Es un sistema de programas para simulación no lineal de circuitos electro – hidráulicos de regulación. Para el desarrollo se emplearon las experiencias obtenidas por Mannesmann Rexroth en materia de técnica hidráulica. Es un software limitado al catálogo de la marca Rexroth. Los accionamientos pueden simularse tanto como cadena abierta como en circuito cerrado de regulación. Es un programa configurado por cuatro subprogramas para aplicaciones específicas: HYVOS, HYDRA, HYSTA HYSEV
Bathfp	1990	Es una herramienta de simulación del dominio del tiempo interactivo. Las herramientas de desarrollo del modelo completo permiten la definición de un nuevo modelo y su incorporación dentro de la librería estimula una aproximación estructurada del modelado asegurando una futura aplicabilidad de cada modelo. El paquete puede ser usado en la etapa de diseño para evaluar las posibles configuraciones del circuito y optimizar el tamaño de los componentes y seleccionar los parámetros de control. También se usa para optimizar sistemas existentes y sirve como una herramienta de diagnóstico para identificar la causa del problema. La última versión se estableció el 17 de Octubre del 2001 en la Universidad de Bath.
DSH - plus	1995	Creado por la sociedad Fluidon GmbH y tiene como objetivo principal el cálculo dinámico de sistemas complejos hidráulicos y neumáticos. El modelo matemático utilizado es el de sistema de ecuaciones diferenciales de primeros órdenes lineales y no lineales. El esquema hidráulico se construye conectando elementos y líneas cuya descripción está basada en las leyes de Kirchoff, aplicadas sobre un volumen de aceite considerado como nodo. Utiliza el método de integración de Runge Kutta de 4º orden.



		Universistat Politècnica de Catalunya
HOPSAN	1991	Es un programa de simulación general, creado especialmente para simular sistemas óleo — hidráulicos y neumáticos. La compleja carga dinámica puede ser estudiada así como la propagación de las ondas en grandes líneas. Originalmente se usaron diferentes modelos matemáticos 1) Un sistema de ecuaciones algebraicas no lineales para el modelado en régimen permanente 2) Un sistema de ecuaciones diferenciales ordinarias y algebraicas para la simulación del dominio del tiempo 3) Un sistema de ecuaciones diferenciales linealizadas para las respuestas de las investigaciones de la frecuencia. Posteriormente se incorporó un algoritmo FFT la que permitió el análisis frecuenciales sobre simulaciones en el dominio del tiempo. El régimen permanente puede obtenerse a partir de la simulación en el dominio del tiempo. El modelo se complementa mediante la incorporación de un sistema nodal.
AME - Sim	1995	Fue desarrollado por IMAGINE (Michel Lebrun 1987) el cual ha sido capaz de capitalizar los conocimientos mecánicos e hidráulicos y en el modelado y diseño de sistemas complejos. IMAGINE tomó parte en la introducción de Bond Graphs en Francia. Previo al desarrollo de AME — Sim, IMAGINE usó para modelar y simular sus estudios de consulta usando ACSL (Advanced Continuous Simulation Language), una potente pero no muy interactiva herramienta dedicado al modelado de sistemas dinámicos. En 1995, IMAGINE rompe la barrera entre el usuario de software y sus promotores creando AME — Sim. IMAGINE ha sido desde entonces el que proporciona AME — Sim, y desde entonces, la simulación se apega a las medidas experimentales. Se puede comunicar interactivamente con otros programas (ADAMS, MATLAB, SIMULINK, SIMPACK).
ITI - Sim	2001	La forma de construcción del modelo ha sido adaptada a la descripción más común de cada disciplina de la ingeniería; así se dispone: 1) Diagramas de circuitos para los componentes y sistemas oleohidráulicos 2) Diagramas funcionales para los sistemas mecánicos 3) Diagramas de bloques para los sistemas de control El software se complementa con una aplicación que permite al usuario definir su propio submodelo y otra aplicación que permite simular partes del sistema por una herramienta de simulación externa, comunicándose entre sí a ciertos intervalos de tiempo para intercambiar datos, ejemplos ADAMS para la modelización y simulación de sistemas mecánicos y MATLAB/SIMULINK para la modelización y simulación de los elementos de control.



ITI - TOOL		Es un software desarrollado para la optimización de accionamientos oleohidráulicos para grúas y está basado en el ODHAMN (Optimal Design of Hydraulically Actuated Manipulators) que permite: 1) Análisis del régimen permanente durante la elevación y bajada 2) Análisis dinámico en la operación de elevación y bajada 3) Trazado del diagrama de Bode de las funciones de transferencia de los lazos abierto y cerrados para la posición más elevada y las condiciones de carga máxima y mínima 4) Animación del sistema mecánico y del movimiento de la corredera de la válvula "Over – Center"
VisSim	1990	Es un programa usado para el modelado y simulación y en la aplicación del diseño de los sistemas de control. Con la interfase del diagrama de bloques se puede construir, y modificar el modelo. Proporciona soluciones para los sistemas lineales, no lineales, tiempo continuo, tiempo discreto. El usuario construye su propio modelo seleccionando un bloque predefinido en la librería y realizando la conexión gráfica de los bloques dentro del diagrama. Cada diagrama de bloque realiza una función matemática o función entrada/salida. Estos bloques pueden representar algoritmos complejos, entradas de variables o salidas de las gráficas similares, esquemas, trazados o archivos de datos. El usuario puede crear un bloque en C, FORTRAN o Pascal y añadirlo a la librería de VisSim. Después el modelo es configurado, se pone en marcha la simulación y los resultados son desplegados.
20 - SIM	1995	Desarrollado en el laboratorio de control de la Universidad de Twente, como sucesor del paquete TUTSIM. Después de un extensivo examen, en Agosto de 1995, la versión 1.0 de CAMAS fue comercialmente liberado bajo el nombre de 20 – SIM. La última versión tiene origen en varios prototipos (MAX, CAMAS y TUTSIM). Usa varios algoritmos de integración avanzada. Permite crear submodelos y combinarlos para formar un modelo complejo. Utiliza el sistema de modelado a través de diagramas de iconos, Bond Graphs y ecuaciones. Permite una interacción cerrada con MATLAB y SIMULINK. Comprende cajas de herramientas para la optimización de parámetros, generación de código ANSI – C, linealización, animación gráfica y animación 3D.



OHC - SIM	2001	Desarrollado en Japón con el apoyo de JFPS (The Japan Fluid Power Systems Society ó mejor conocido como The Japan Hydraulics and Pneumatics Society) y mejorado en Sakurai (1999 – 2000). Es dedicado al diseño y mejoramiento de circuitos óleo – hidráulicos en la base de los resultados simulados y el análisis de sus características dinámicas. La función "User – Customized" ejecuta la simulación para la cual era necesario conocer de Bond Graphs, pero que en las próximas versiones se proporcionará un entorno en donde un nuevo componente óleo – hidráulico pueda ser registrado sin conocimiento de Bond Graphs.
-----------	------	---

Tabla 4.2 - Programas disponibles o comerciales

Temiendo en cuenta los aspectos antes mencionados de los programas citados, se ha considerado utilizar 20-SIM para simular el comportamiento dinámico de sistemas, el cual es un programa que permite separar los diferentes dominios tecnológicos en submodelos independientes. Las razones que indujeron a utilizar este programa fueron las siguientes:

- En LABSON hay licencia de uso del programa
- 20-SIM está basado en Bond Graphs, 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 Bond Graphs es muy sencillo añadir complejidad a un modelo mediante la incorporación de nuevos elementos o subsistemas. Una de las virtudes de Bond Graphs 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 submodelos de dominio orientado.