

Especificación De Modelos de Simulación

Autor:

Basilio Puente Varela 



Especificación de Modelos de Simulación

Rev 0.0 – Noviembre 2023

INDICE

1. Introducción	5
2. Especificación del Modelo Conceptual.....	5
2.1. Descripción del Sistema	5
2.2. Propósito y Requisitos del Modelo	5
2.3. Características del Modelo	6
2.3.1. Dominios de la Física Modelados	6
2.3.2. Tipo de modelo	6
2.4. Fundamentos Teóricos del Sistema	6
2.4.1. Estado del arte y teorías seleccionadas	6
2.4.2. Hipótesis y simplificaciones usadas.....	7
2.4.3. Fuentes de datos usadas (data pedigree)	7
2.5. Variables de Interés.....	7
2.5.1. Entradas (inputs)	8
2.5.2. Salidas (outputs).....	9
2.5.3. Parámetros (parameters).....	10
2.5.4. Variables expuestas y valores iniciales (exposed variables)	11
2.6. Plan de Validación.....	12
2.7. Limitaciones del Modelo.....	12
2.7.1. Limitaciones Teóricas	12
2.7.2. Implementación de las limitaciones.....	12
2.7.3. Código de Funcionamiento	13
3. Especificación del Modelo de Simulación.....	14
3.1. Características técnicas	14
3.2. Documentación del modelo	14
3.2.1. Descripción del modelo.....	14
3.2.2. Experimento tipo.....	14
3.2.3. Documentación	14
3.3. Documentación específica del cliente.....	16
4. Control de Calidad.....	17
4.1. Control automático de calidad del FMU.....	17
4.2. Control de calidad específico del FMU	17
4.3. Control de calidad del código	17
4.4. Buenas prácticas en el modelo	18

5. Anexo 19

5.1. Versionado..... 19

5.2. Sistema de Unidades..... 20



INDICE DE FIGURAS

Figura 1: Esquema desarrollo de modelos (NASA-STD-7009) 5

Figura 2: Dominios de validación (Trucano, T.G. et al (2002))..... 12

INDICE DE TABLAS

Tabla 1: Matriz de requisitos del modelo 6

Tabla 2: Dominios de la Física Modelados 6

Tabla 3: Tabla de entradas del modelo (inputs)..... 8

Tabla 4: Tabla de salidas del modelo (outputs) 9

Tabla 5: Tabla de parámetros del modelo (parameters) 10

Tabla 6: Tabla de variables del modelo (variables) 11

Tabla 7: Descripción del modelo..... 14

Tabla 8: Checklist de calidad del FMU 17

Tabla 9: Checklist de calidad del código 17

Tabla 10: Checklist de buenas prácticas del modelo 18



1. Introducción

Esta especificación está destinada a definir modelos de simulación en formato FMU. Además de la especificación del modelo conceptual, se incluye una parte de la especificación del modelo de simulación, que se corresponde a la comunicación del mismo con el exterior y sus características principales deseadas.

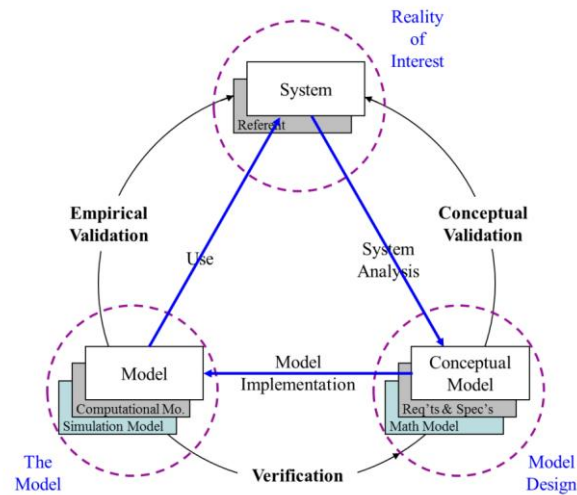


Figura 1: Esquema desarrollo de modelos (NASA-STD-7009)

Se incluye también la especificación de la documentación del modelo y un checklist básico de calidad que deberá ser comprobado previo a la emisión del mismo.

2. Especificación del Modelo Conceptual

2.1. Descripción del Sistema

En este apartado se describe en que consiste el sistema a modelar.

2.2. Propósito y Requisitos del Modelo

En este apartado se establece el propósito del modelo y los requisitos del mismo.

ID	Requisito	Etapas verificación

Tabla 1: Matriz de requisitos del modelo

2.3. Características del Modelo

2.3.1. Dominios de la Física Modelados

Dominio	Incluido	Opcional	Notas
Mecánica Clásica			
Térmico			
Fluidos			
Electricidad			
Máquinas Eléctricas			
Electrónica			
Electromagnetismo			
Química			
Control			
Acústica			
...			

Tabla 2: Dominios de la Física Modelados

2.3.2. Tipo de modelo

Indicar si el modelo es:

- Estático ó dinámico
- Físico / Físico-empírico / modelo de datos / IA / ROM
- Modelo tiempo real (tiempo ejecución << tiempo real)

2.4. Fundamentos Teóricos del Sistema

2.4.1. Estado del arte y teorías seleccionadas

Teoría física del comportamiento del modelo.

2.4.2. Hipótesis y simplificaciones usadas

Hipótesis y simplificación usadas al aplicar las teorías seleccionadas.

2.4.3. Fuentes de datos usadas (data pedigree)

Describir las fuentes de datos usadas en el desarrollo del modelo, su origen e incertidumbre asociada.

Envejecimiento del sistema y modelo

2.5. Variables de Interés

Las siguientes tablas muestran las variables de interés requeridas para el modelo de simulación. Están agrupadas en cuatro categorías:

- **Entradas (inputs):** Son las variables de entrada del FMU.
- **Salidas (outputs):** Son las variables de salida del FMU.
- **Parámetros (parameters):** Son los parámetros del modelo, a definir antes de comenzar la simulación.
- **Variables expuestas (exposed variables):** Son variables internas que están expuestas al usuario del FMU. Algunas de ellas necesitan inicializar su valor antes de la simulación para definir el estado inicial del sistema.

Los nombres de las variables serán en inglés y tendrán las siguientes características técnicas:

1. Los nombres de las variables serán de tipo jerárquico (ver FMI 2.0 standard, sección 2.2.9), por ejemplo: “Propeller.Diameter”. El FMU tendrá activado el atributo “*variableNamingConvention=structured*”.
2. Los nombres de las variables del modelo comienzan con un carácter en minúscula, por ejemplo “speed”, con la excepción que son habituales de representar con un carácter, por ejemplo “Q” para el calor.
3. Si el nombre de la variable consta de varias palabras, el primer carácter de cada palabra es escrito en mayúscula, con excepción de la primera palabra donde se aplicarán las reglas precedentes, por ejemplo “speedEngine”.
4. El guion bajo solo se emplea al final de los nombres o al final de cada palabra para indicar subíndices, por ejemplo “valve_1.position”.

Tabla de nombres típicos??

2.5.1. Entradas (inputs)

					Integer			Real			
Name	Description	Data Type	variability	start	min	max	quantity	unit	displayUnit	relativeQuantity	
		Real Integer Boolean String	discrete continuous							true?	
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											

Tabla 3: Tabla de entradas del modelo (inputs)

2.5.2. Salidas (outputs)

							Integer			Real				Tolerance = true
Name	Description	Data Type	variability	initial	start	min	max	quantity	unit	displayUnit	relativeQuantity	nominal	unbounded	
		Real Integer Boolean String	discrete continuous	exact approx calculated							true?		true?	
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														

Tabla 4: Tabla de salidas del modelo (outputs)

2.5.3. Parámetros (parameters)

						Integer			Real			
	Name	Description	Data Type	variability	initial	start	min	max	quantity	unit	displayUnit	relativeQuantity
			Real Integer Boolean String	discrete continuous	exact approx calculated							true?
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												

Tabla 5: Tabla de parámetros del modelo (parameters)

2.5.4. Variables expuestas y valores iniciales (exposed variables)

	Name	Description	Data Type	variability	initial	start								
							Integer			Real			Tolerance = true	
							min	max	quantity	unit	displayUnit	relativeQuantity	nominal	unbounded
			Real Integer Boolean String	discrete continuous	exact approx calculated							true?		true?
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														

Tabla 6: Tabla de variables del modelo (variables)

2.6. Plan de Validación

Indicar el tipo de validación tanto del modelo conceptual como del modelo de simulación.

En el modelo conceptual es una validación teórica:

- Se está omitiendo un campo de la física que influye en los objetivos del modelo?
- Se está definiendo con el detalle óptimo el modelo acorde a sus objetivos (subparametrizar / sobreparametrizar)?
- El tipo de modelo y los fundamentos teóricos son compatibles?

En el modelo de simulación se debe especificar la validación empírica:

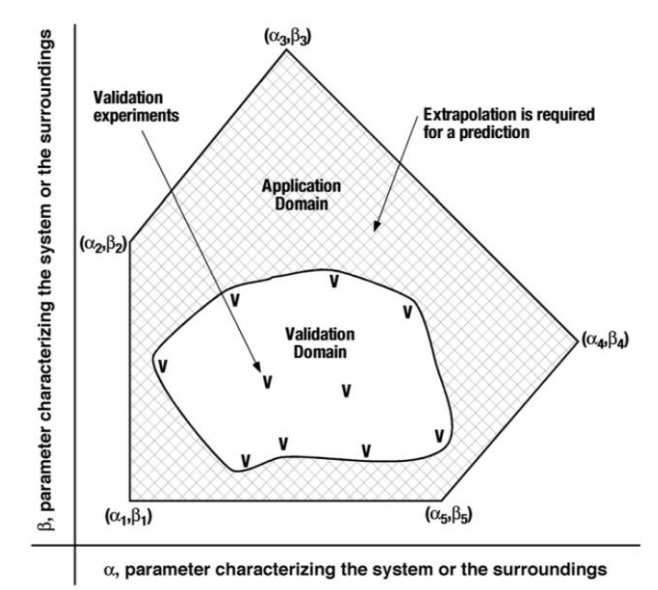


Figura 2: Dominios de validación (Trucano, T.G. et al (2002))

- Definir el tipo de validación:
 - Comparación con experimentos realizados ad-hoc para la validación del modelo
 - Comparación con experimentos previos obtenidos de la literatura
- Definir el dominio de validación
- Definir el muestreo de experimentos
- Simular y validar.

2.7. Limitaciones del Modelo

2.7.1. Limitaciones Teóricas

Indicar las limitaciones teóricas del modelo acorde al dominio de validación y a las teorías usadas.

2.7.2. Implementación de las limitaciones

Las limitaciones en el modelo se implementan mediante las siguientes técnicas:

- **Valores min/max de variables (FMI standard):**

Se deben establecer valores mínimos y/o máximos en variables cuyo exceso de dichos límites provoque un funcionamiento fuera de la zona de validación del modelo. En ese caso el FMU emitirá un mensaje de aviso pero continuará su funcionamiento.

NOTA: El uso de estos valores no previene un posible error numérico (p.ej. división entre cero), por lo cual no puede ser usado para este propósito y el código debe ser protegido adecuadamente.

- **Warnings (FMI standard):**

Su funcionamiento es similar a los valores min/max de variables, pero puede configurarse para que avise cuando se cumpla con una expresión cualquiera. Debe usarse cuando se cumpla una condición interna del modelo que provoque que trabaje fuera de la zona de validación (p.ej. el número de Reynolds exceda los límites de la validación).

Se establece el siguiente formato de mensaje:

“[W0001] Intensity above validation zone”

Donde “[W0001]” es el código de warning y a continuación el texto. En la documentación del modelo se listarán y explicarán los códigos disponibles.

- **Error (FMI standard):**

Se configura mediante una expresión que emite un mensaje de error y termina la simulación cuando suceda. Debe usarse para proteger (o indicar al usuario) al modelo de un error fatal o una operación fuera del dominio de aplicación del modelo.

Se establece el siguiente formato de mensaje:

“[E0001] Intensity above application zone”

Donde “[E0001]” es el código de error y a continuación el texto. En la documentación del modelo se listarán y explicarán los códigos disponibles.

2.7.3. Código de Funcionamiento

El código de funcionamiento es una variable expuesta del modelo, de tipo “integer”, situada en el primer nivel del FMU y llamada “WorkingStatus” que tendrá los siguientes valores:

- 1 = Modelo funcionando dentro de los límites de validación
- 2 = Modelo funcionando fuera de los límites de validación y dentro de los límites de aplicabilidad
- 3 = Modelo funcionando dentro de los límites de validación, tras funcionar fuera de los mismos (sólo para modelos dinámicos)

3. Especificación del Modelo de Simulación

3.1. Características técnicas

Las características del modelo entregable en formato FMU son las siguientes:

- FMI Standard 2.0 (con características hasta v2.0.2)
- FMU Model Exchange y Co-Simulation compilado como mínimo para la plataforma “win64”.
- Código fuente incluido en el FMU, siendo posible una compilación posterior a otras plataformas (como “linux32”, “linux64”, “arm64”, etc...)
- Sin licencias de ejecución (no runtime licenses). Debe indicarse en el FMU como “needsExecutionTool=false” (elemento “CoSimulation” del XML).
- Nombres de variables estructuradas (“variableNamingConvention=structured”).
- Indicar el solver usado en CS dentro de la documentación: Euler explicit, CVODE, etc...
- Indicar que sistema de índices de arrays es usado (basados en 0- o en 1-).

3.2. Documentación del modelo

3.2.1. Descripción del modelo

En el XML de *fmiModelDescription*, se detallarán los siguientes campos:

Atributo	Descripción
<i>modelName</i>	
<i>description</i>	
<i>author</i>	(nombre y empresa desarrolladora)
<i>version</i>	(ver “versionado” en el Anexo)
<i>copyright</i>	(© empresa)
<i>license</i>	see /documentation/licenses/license.txt
<i>variableNamingConvention</i>	structured

Tabla 7: Descripción del modelo

3.2.2. Experimento tipo

No es necesario establecer un tiempo inicial y final determinado del experimento tipo “DefaultExperiment” en el *fmiModelDescription*.

Sin embargo, es necesario establecer la tolerancia de cálculo (“tolerance”) que es de especial interés cuando se usan solvers de paso variable. Además, se debe indicar el paso de tiempo típico del modelo (“stepSize”) que indica en la resolución nominal en la que fue diseñado el modelo. No es recomendable utilizar el modelo con pasos de tiempo superiores a este valor.

3.2.3. Documentación

La documentación del modelo se incluirá acorde al estándar FMI, como un archivo HTML en la ruta “/documentation/index.html” del FMU.



Este archivo de documentación contendrá los siguientes apartados:

- **Información:**

En este apartado se resume la descripción del modelo conceptual, citando los campos de la física modelados y las teorías e hipótesis usadas.

Se debe indicar al final la herramienta de simulación y su versión utilizada para generar el FMU, así como las opciones principales seleccionadas. Se debe indicar si los arrays son basados en 0- o en 1-.

- **Funcionamiento:**

En este apartado se describe el funcionamiento haciendo uso del nombre de los parámetros, variables internas expuestas y su inicialización, variables de entrada (inputs) y variables de salida (outputs). Se pueden describir casos de ejemplo en este apartado.

Además, se incluirá la definición de cada uno de los estados del código de funcionamiento.

- **Solver:**

En este apartado se cita el solver usado en el desarrollo y validación del modelo, así como la tolerancia usada y el paso de tiempo típico.

- **Limitaciones del modelo:**

En este apartado se describe la zona de validación y la zona de aplicación. De la misma forma se citan los warning y errors preprogramados y su significado.

- **Problemas conocidos:**

Cualquier problema conocido del modelo debe ser indicado aquí. Es habitual que dichos problemas no interfieran en el funcionamiento normal del mismo y sólo sucedan ante unas determinadas situaciones.

- **Referencias:**

Las referencias bibliográficas usadas tanto en el desarrollo conceptual del modelo como en su validación deben ser indicadas.

De la misma forma, se deberán listar todas las librerías de terceros cuyos componentes fueron usados en el modelo, indicando el nombre de la librería y la versión usada.

- **Copyright:**

En este apartado se indicará el desarrollador/desarrolladores principales del modelo y la empresa desarrolladora, además del año de emisión del modelo. Se establecerá una dirección de contacto con la misma en forma de email y teléfono. También se reproducirá el texto de licencia que se encuentra en la ruta “/documentation/licenses/license.txt”.

- **Revisión:**

Se indicará la versión actual del modelo de la forma:

“Rev. 1.3.0 (13-06-2021) [B.Puente]: Modelo actualizado para aceptar diferentes tipos de líquidos.”

Donde:



Rev. 1.3.0 : Es la revisión mayor (1) junto con la revisión menor (3) y el parche (0).

(13-06-2021) : Es la fecha de la emisión de la revisión va entre paréntesis en formato dd-mm-aaaa.

[B.Puente] : Es la identificación del autor de la revisión.

A continuación, se añade una breve descripción del motivo de revisión.

3.3. Documentación específica del cliente

En la carpeta “/extra/com.cliente” del FMU (FMI v2.0.2) se incluirá la documentación específica del cliente, por ejemplo:

- meta.xml: Metadatos específicos del cliente sobre el modelo.
- “.../sourcefiles/” : Carpeta donde se incluyen los archivos y/o código fuente que generan el FMU. En caso de necesitar alguna explicación, se incluirá un archivo “instructions.pdf” con las instrucciones. El archivo será del tipo PDF/A.
- “.../validation/” : Carpeta donde se incluyen los archivos y justificación técnica de la validación del modelo.

Cualquier documentación incorporada en PDF deberá cumplir con el estándar PDF/A.

4. Control de Calidad

4.1. Control automático de calidad del FMU

El FMU deberá pasar el chequeo indicado en <https://fmi-standard.org/validation/>.

Importante: la herramienta “FMUComplianceChecker” está descatalogada y el chequeo se debe realizar mediante FMPy con el comando `fmpy.validation -> validate_fmu`.

4.2. Control de calidad específico del FMU

Item	Check
¿Las características técnicas del FMU se corresponden a las indicadas?	
¿El nombre, descripción, autor, versión, copyright y licencia están incluidos en el XML?	
¿La tolerancia y el paso de tiempo están incluidos en el experimento tipo?	
¿El FMU contiene el texto de la licencia en <code>"/documentation/licenses/license.txt"</code> ?	
¿El archivo de documentación <code>"/documentation/index.html"</code> está completo y funciona una vez descomprimido el FMU?	
¿La documentación específica del cliente está incluida en <code>"/extra/..."</code> ?	

Tabla 8: Checklist de calidad del FMU

4.3. Control de calidad del código

Item	Check
¿Los atributos de las variables se corresponden a los indicados en este documento?	
¿El código fuente usa el indentado correctamente?	
¿El código fuente está correctamente organizado y comentado?	
¿Están protegidos los límites de aplicabilidad del modelo?	
¿Están protegidas las posibles divisiones entre cero?	
¿Están protegidas las posibles raíces de números negativos?	
¿Están protegidas las funciones trigonométricas?	
¿Están protegidas otras funciones que puedan llevar a un error en el modelo?	
¿Están protegidas las comparaciones entre números reales? ¿Están protegidos los eventos basados en dichas comparaciones?	
¿El paso de tiempo es adecuado para el funcionamiento adecuado del modelo?	

Tabla 9: Checklist de calidad del código

4.4. Buenas prácticas en el modelo

Item	Check
<i>¿Se usan ayudas a la inicialización en sistemas grandes de ecuaciones no lineales?</i>	
<i>¿Existe la opción de linearizar ecuaciones para aumentar la velocidad de ejecución?</i>	
<i>¿Se indica la derivabilidad de funciones al compilador?</i>	
<i>¿Se evitan los eventos (discontinuidades) en todo lo posible?</i>	
<i>¿Las variables necesarias están reducidas al mínimo y ocultas las no relevantes?</i>	

Tabla 10: Checklist de buenas prácticas del modelo

5. Anexo

5.1. Versionado

La codificación de la versión seguirá el siguiente criterio:

MAYOR.MENOR.PARCHE

Donde:

MAYOR: Revisión mayor, cuando los parámetros, conectores ó funcionamiento interno no es compatible con una revisión mayor anterior. Las revisiones menor y parche deben ser reiniciadas a cero al emitir una revisión mayor.

MENOR: Revisión menor, cuando se añade funcionalidad que es compatible con versiones anteriores dentro de la misma revisión mayor.

PARCHE: Corrección de errores compatible con las revisiones anteriores. Debe ser reiniciado a cero cuando una nueva revisión menor se emita. En algunos casos se omite la revisión PARCHE, acumulando todas las correcciones para emitirlas en una nueva revisión menor.

Las siguientes reglas aplican a las revisiones:

- Los números usados para MAYOR, MENOR y PARCHE deben ser enteros no negativos sin ceros iniciales. Los números serán incrementados con las revisiones, es decir la revisión “1.11.0” es posterior a la “1.9.3”.
- Tras un lanzamiento de una versión, cualquier modificación debe ser en una revisión nueva.
- La revisión MAYOR igual a zero (0.y.z) es reservada para el desarrollo inicial, donde todo puede cambiar en cualquier momento. Estas revisiones pueden no ser compatibles entre sí.
- Una revisión en pruebas (pre-release) se marca con un guion al final y una indicación de la forma “1.3.1-alpha”, donde:
 - *-nightly*: Versión diaria en continuo desarrollo.
 - *-alpha*: Versión con nuevas características añadidas.
 - *-beta*: Versión completa en fase de depuración de errores.
 - *-release-candidate*: Versión con una distribución ampliada a otros usuarios para su prueba final antes de emitir la versión oficial.

Para mas información, véase *Semantic Versioning 2.0.0* (<https://semver.org>).

5.2. Sistema de Unidades

Los conceptos clave al trabajar con unidades son los siguientes:

- **Unidades (*unit*):** Son las unidades de medida del dato en cuestión. Serán en el sistema internacional de acuerdo a la ISO 31-0 (1992). Por ejemplo “N.m” como Newton-metro.
- **Dimensiones:** Son los exponentes expresados en las unidades básicas de medida (BaseUnit) siguientes: kg, m, s, A, K, mol, cd y rad (unidad derivada). Por ejemplo, la presión tiene dimensiones de (kg=1, m=-1, s=-2), es decir $\text{kg} \cdot \text{m}^{-1} \cdot \text{s}^{-2}$.
- **Cantidades (*quantity*):** Son las cantidades físicas que miden cada unidad. Por ejemplo, un par es una fuerza multiplicada por una distancia ($\text{N} \cdot \text{m}$), al igual que la energía ($\text{N} \cdot \text{m} = \text{J}$). Por lo tanto, aunque el momento y la energía tengan las mismas dimensiones (Fuerza * distancia) se tratan de dos cantidades diferentes.

Reglas a seguir:

1. Se utilizarán unidades del sistema internacional de unidades, de acuerdo a la ISO 31-0 (1992).
2. Se establecerán en el FMU los exponentes, el factor y el offset que permita comprobar su conexión con otros FMU's.

Pendiente: Lista de unidades y cantidades con nombre estandarizado