

Análisis de Conformidad de Gemelos Digitales Multi-Fidelidad

Paula Muñoz, Javier Troya, Antonio Vallecillo

ITIS Software. Universidad de Málaga
{paulam, jtroya, av}@uma.es

Resumen Un gemelo digital es una réplica virtual de un sistema a un cierto nivel de fidelidad, sincronizados a una determinada frecuencia. Los gemelos digitales suelen emplearse para replicar sistemas físicos cuyas simulaciones son computacionalmente muy costosas. Una de las soluciones que la literatura propone a este problema es definir una jerarquía de gemelos digitales multi-fidelidad, donde dependiendo del propósito concreto, empleemos un gemelo u otro. Sin embargo, uno de los problemas que surgen es la necesidad de determinar si los distintos gemelos son equivalentes entre ellos y conforme al sistema físico. En este artículo exploramos distintos métodos para medir esta equivalencia analizando el estado y el comportamiento de los gemelos.

Keywords: Gemelos Digitales · simulación · fidelidad · multi-fidelidad

1. Introducción

Un gemelo digital (*DT* del inglés, *Digital Twin*) es una representación digital de un sistema, servicio o producto, el denominado gemelo físico (*PT* del inglés, *Physical Twin*), sincronizado a una cierta frecuencia y a un cierto nivel de fidelidad [2]. Los sistemas que incluyen gemelos digitales son complejos, puesto que necesitan no sólo de la interacción sincronizada entre sus distintos elementos, sino también de la simulación de los fenómenos físicos [4]. Esta interacción obliga a modelar y simular estructuras y comportamientos físicos que dan lugar a modelos con un alto nivel de complejidad y, con ello, a simulaciones muy sofisticadas. En algunos casos, el tiempo de simulación puede superar al real; por ejemplo, puede ocurrir que 1 minuto de simulación tarde 10 minutos en ejecutarse [1]. Estos costes computacionales impiden una sincronización con el sistema real, impidiendo la ejecución de algunas de las tareas más básicas para las que suelen emplearse los sistemas de gemelos digitales. Algunas de estas tareas son la predicción de cambios en el sistema para su calibración o el desarrollo de comportamientos auto adaptativos [6]. Si el DT no es capaz de sincronizarse con el sistema real en sus ejecuciones o desarrollar simulaciones de forma eficiente para las predicciones, podemos incurrir en que dicho DT no sirva para el propósito que se le ha establecido.

Para abordar estos problemas, algunos autores proponen el desarrollo de diferentes gemelos digitales con distintos niveles de fidelidad y con propósitos concretos en su simulación, dando lugar a una jerarquía de gemelos digitales *multi-fidelidad* [1]. En general, cuanto menor sea el nivel de fidelidad, menos

complejo y costoso de ejecutar será el gemelo digital. Sin embargo, los gemelos digitales deben ofrecer un nivel de fidelidad suficiente para cumplir con su propósito, y deben funcionar en conformidad con la especificación del sistema. Si su nivel de abstracción es demasiado alto o no tienen suficiente resolución, podrían dar lugar a resultados que no resulten útiles al no ser precisos.

Nuestra línea de investigación persigue el desarrollo de una metodología de pruebas para evaluar el nivel de fidelidad de los gemelos digitales, por lo que es necesario definir la conformidad con respecto a un oráculo. Para ello, proponemos el análisis de su estado y comportamiento mediante el análisis de trazas.

2. Ejemplo ilustrativo: El coche Lego Mindstorms NXJ

Para ilustrar nuestra propuesta, emplearemos una versión parcial del modelo de un gemelo digital de un coche Lego Mindstorms NXT, que desarrollamos en un trabajo anterior [7].

El modelo conceptual está definido empleando la herramienta USE [3], que permite definir la estructura del sistema empleando UML, sus condiciones e invariantes empleando OCL y su comportamiento empleando SOIL, un lenguaje imperativo de alto nivel incluido en USE. Este modelo conceptual está disponible en la figura 1.

La clase principal es la denominada **Car**, que contiene todos los atributos de estado (posición en coordenadas cartesianas medidas en centímetros, ángulo hacia el está posicionado, acción que está ejecutando, largo, ancho, nivel de batería restante, intensidad de la señal de Bluetooth, e incluso el color del chasis). Por otro lado, la clase **Motor** es la que permite al coche realizar movimientos rectilíneos y giros.

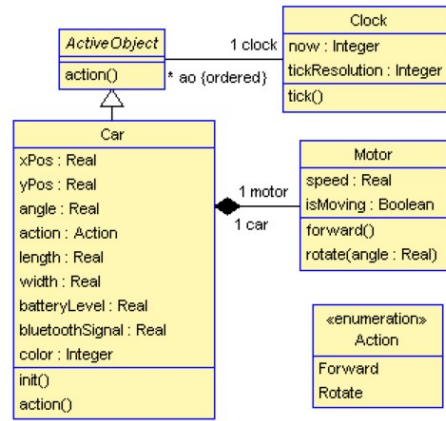


Figura 1. El modelo conceptual del coche Lego Mindstorms.

3. Medidas de fidelidad

La fidelidad se define como el grado en el que un modelo reproduce el *estado* y el *comportamiento* de un sistema de forma medible [5]. En esta definición, se asume el hecho de que un sistema no está definido únicamente por sus estados o comportamiento, sino que requiere ambos aspectos. Por ello, proponemos métodos para medir el nivel de fidelidad conforme a los dos.

Estos métodos tienen el objetivo de poder aplicarse a gemelos digitales definidos a cualquier nivel de fidelidad, de forma que los resultados sean comparables entre todas las instancias de dicho gemelo. Para que esto sea posible, trataremos a todos los gemelos como cajas negras, y procederemos al análisis de las trazas de ejecución de los sistemas.

Medidas de la fidelidad de los estados del sistema. Para definir el estado del sistema en un instante concreto, se emplean los llamados *snapshots* [3], que especifican los objetos, las relaciones entre ellos y los valores específicos de estos en ese preciso instante. Estos *snapshots* se generan extrayendo los valores del sistema periódicamente y almacenándolos en una base de datos. Para definirlos, deberemos establecer una función de abstracción con respecto al gemelo que extraiga los parámetros que nos interesan en la validación del estado del sistema. Esta función de abstracción es esencial, puesto que si tenemos sistemas a distintos niveles de fidelidad, su nivel de abstracción también puede ser distinto.

Por ejemplo, en la figura 2 podemos ver cuatro *snapshots* de la clase `Car`, que representan los estados del gemelo digital (DT) y el físico (PT) en dos instantes de su ejecución. Para determinar el nivel de fidelidad del gemelo digital, deberemos comparar los dos *snapshots*. Estas comparaciones deben hacerse considerando un cierto grado de precisión. Por ejemplo, si el nivel de precisión que requerimos es de 0.5 cm, los dos *snapshots* de la izquierda serían equivalentes, mientras que los dos de la derecha no lo son. Para determinar el nivel de fidelidad, habría que repetir esta operación con el resto de parámetros.

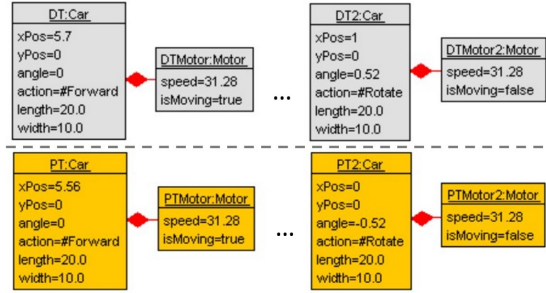


Figura 2. Dos *snapshots* de la clase `Car`.

Para obtener un resultado global una vez se han comparado todos los atributos, podríamos usar distintas medidas estadísticas. Por ejemplo, la desviación estándar de los valores de los atributos lineales o la máxima diferencia con respecto a la mediana de los valores. Además, también se podrían usar medidas similares a las que se emplean para la incertidumbre de la medida.

Medidas de la fidelidad del comportamiento. A pesar de que hay distintos métodos para definir la semántica del comportamiento en los modelos, en nuestro trabajo usaremos la semántica de trazas. Estas trazas de ejecución están compuestas de una secuencia finita de ocurrencias de eventos. Para mantener la consistencia con la medida de fidelidad en estados, estas trazas estarán compuestas por *snapshots*, tales como los que se presentaron en el ejemplo anterior.

Dos comportamientos se consideran equivalentes si el conjunto de sus trazas son iguales. También pueden definirse otros tipos de relaciones como la simulación o la bisimulación, siendo esta última la más fuerte y la que implica

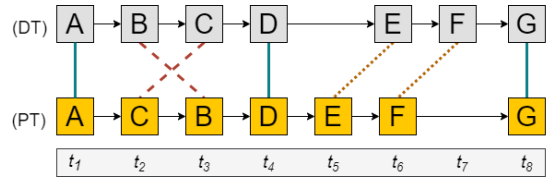


Figura 3. Dos trazas de ejecución de la clase `Car`.

que dos sistemas no se pueden distinguir, es decir, que un sistema puede simular al otro y viceversa.

En la figura 3, nos encontramos con las trazas de ambos gemelos del coche. Cada una de las trazas está compuesta por siete *snapshots*, nombrados de la *A* a la *G*, y que han sucedido en instantes de tiempo t_1, \dots, t_8 . A la vista de estas trazas podríamos detectar diversas discrepancias, como el cambio de orden entre B y C, o un ligero retraso en la ejecución de E en el gemelo digital.

El primer paso para ejecutar un análisis de trazas es averiguar cuáles de ellas son equivalentes, para posteriormente analizar la relación entre los eventos, como hemos hecho con el ejemplo. Para ello, estamos estudiando la aplicación de técnicas de alineamiento de trazas. Una vez hayamos estimado esta relación, seremos capaces de establecer un nivel de fidelidad. Para obtener una medida objetiva de este análisis, podemos emplear distintas medidas estadísticas como el porcentaje de estados fieles, detectando el nivel de equivalencia entre *snapshots*.

4. Conclusiones y trabajos futuros

En este trabajo hemos presentado un primer acercamiento al análisis de la fidelidad en gemelos digitales, con el objetivo de emplear gemelos definidos a distintos niveles de fidelidad para optimizar las tareas que los involucren.

Los siguientes pasos que planeamos se centran en la definiciones de medidas para evaluar la distancia entre trazas y el desarrollo de una herramienta para determinar la equivalencia entre trazas de ejecución de gemelos digitales a distinto nivel de fidelidad. De esta forma podremos establecer relaciones de semántica de trazas y medir la precisión entre los atributos de estado.

Financiación. Para la realización de este trabajo se ha contado con financiación del II Plan Propio de Investigación, Transferencia y Divulgación Científica de la UMA y del proyecto COSCA (PGC2018-094905-B-I00) del Ministerio de Ciencia e Innovación - Agencia Estatal de Investigación.

Referencias

1. Arrieta, A.: Multi-fidelity digital twins: a means for better cyber-physical systems testing? CoRR **abs/2101.05697** (2021)
2. Digital Twin Consortium: Glossary of digital twins. <https://www.digitaltwinconsortium.org/glossary/index.htm> (2021)
3. Gogolla, M., Bohling, J., Richters, M.: Validating UML and OCL Models in USE by Automatic Snapshot Generation. *Sosym* **4**(4), 386–398 (2005)
4. Grieves, M., Vickers, J.: Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, pp. 85–113. Springer (2017). https://doi.org/10.1007/978-3-319-38756-7_4
5. Gross, D.C.: Report from the fidelity implementation study group. In: Proc. of the Fall Simulation Interoperability Workshops (1999)
6. Madni, A.M., Madni, C.C., Lucero, S.D.: Leveraging digital twin technology in model-based systems engineering. *Systems* **7**(1), 7 (2019). <https://doi.org/10.3390/systems7010007>
7. Muñoz, P., Troya, J., Vallecillo, A.: Using UML and OCL Models to Realize High-Level Digital Twins. In: Proc. of ModDiT2021@MODELS'21. pp. 212–220. IEEE (2021). <https://doi.org/10.1109/MODELS-C53483.2021.00037>