

Verificación y validación en simulaciones computacionales: ideales epistémicos y prácticas metodológicas

Andrés A. Ilcic⁽¹⁾, Pío García⁽²⁾, Marisa Velasco⁽³⁾

(1) Centro de Investigaciones de la Facultad de Filosofía y Humanidades, Universidad Nacional de Córdoba (UNC-CONICET), Córdoba, Argentina, ailcic@ffyh.unc.edu.ar

(2) Centro de Investigaciones de la Facultad de Filosofía y Humanidades, Universidad Nacional de Córdoba (UNC), Córdoba, Argentina, piogarcia@ffyh.unc.edu.ar

(3) Centro de Investigaciones de la Facultad de Filosofía y Humanidades, Universidad Nacional de Córdoba (UNC), Córdoba, Argentina, marisa@ffyh.unc.edu.ar

Resumen

En este trabajo abordamos uno de los problemas filosóficos centrales respecto de las simulaciones computacionales en ciencia, el de sus garantías epistémicas. Tras un breve repaso de las caracterizaciones que se pueden encontrar en la literatura, nos concentramos en la idea reciente de utilizar criterios metodológicos para justificar el conocimiento que proviene de una simulación. Sugerimos que existen más tensiones y continuidades entre los ideales epistémicos y las prácticas metodológicas que las señaladas habitualmente, usando una lectura diferente de la distinción clásica en el ámbito de las simulaciones entre verificación y validación. Proponemos que la manera de resolver esta tensión es considerar el rol propio que tienen las simulaciones computacionales en un contexto ampliado de prácticas científicas, en el que la validez epistémica de los resultados no se encuentra garantizada por ideales universales sino por prácticas más bien locales, las que pueden variar de acuerdo al problema de interés particular que está siendo analizado.

Palabras Clave: prácticas científicas; metodología; valores epistémicos.

1. Introducción

Uno de los problemas filosóficos centrales respecto de las simulaciones computacionales en ciencia es el de sus garantías epistémicas. Esta problemática ha sido planteada de diversos modos y se ha intentado resolver con muy variadas estrategias. En este trabajo nos proponemos abordar el problema desde la perspectiva que utiliza una distinción clásica en el ámbito de las simulaciones, la de verificación y validación, pero que sin embargo intenta mostrar tensiones y continuidades entre ideales epistémicos y prácticas metodológicas, sobre las que se ha prestado muy poca atención en la literatura especializada. En particular, intentaremos mostrar que parte del fracaso de algunas caracterizaciones de la confiabilidad de las simulaciones se debe o bien a la “imposición forzada” de ideales epistémicos en prácticas metodológicas o a la simple sustitución de la epistemología por la descripción metodológica.

2. Epistemologías para las simulaciones computacionales

A grandes rasgos, podemos resumir las posturas disponibles en la literatura filosófica especializada para garantizar la credibilidad epistémica de las simulaciones computacionales de la siguiente manera:

- (1) Apoyándose en la ambigüedad de la expresión “experimento numérico”, se podría considerar que las simulaciones computacionales son experimentos. Así, la epistemología de las simulaciones no sería distinta de aquella que se construye a partir de los experimentos. Una subclase de esta categoría está representada por los trabajos de Wendy Parker, quien sostiene que sólo algunas de las estrategias—interventivas—empleadas en la experimentación son aplicables a las simulaciones (e.g. Parker (2008)).
- (2) Las simulaciones computacionales son un ejemplo más de modelos científicos por lo que no hay novedades específicas que deban ser tratadas por los epistemólogos. Un ejemplo de esta posición se puede encontrar en Frigg y Reiss (2009).
- (3) Las simulaciones incorporan algunos elementos que son comparables a los que están presentes en los experimentos y algunos que son parte de la práctica de modelado, por lo que una epistemología para las simulaciones computacionales requiere una estrategia híbrida. Una subclase de esta categoría pone el énfasis en el rol de los modelos en la experimentación, como Morrison (2009) y Winsberg (2009).
- (4) La metodología propia de las simulaciones computacionales en ciencia es garantía suficiente de su rol en la producción de conocimiento. Winsberg (2010) adscribiría parcialmente a esta tesis que ha sido defendida fundamentalmente por Morrison (2014; 2015).

Otra forma de analizar las posturas anteriores, que no necesariamente hace los mismos recortes o clasifica de la misma forma, es a partir de la consideración de las dimensiones ontológicas, metodológicas y epistémicas. La distinción entre la confiabilidad de los experimentos y de las simulaciones a veces es vista como un problema cuya raíz es ontológica. En la literatura filosófica la dimensión ontológica de las simulaciones habitualmente se discute en términos tan generales como el de “materialidad”. Desde otro punto de vista, se ha destacado la importancia de la dimensión metodológica como forma de evaluar la credibilidad de las simulaciones. Pero este aspecto a veces se entiende como íntimamente vinculado con los aspectos ontológicos. La insistencia en encontrar prácticas metodológicas interventivas en las simulaciones es un ejemplo de esta perspectiva (Parker 2008). Más prometedora, para la forma en la cual se trabaja efectivamente con simulaciones, parece la estrategia de elucidar la manera en la cual las prácticas metodológicas permiten dar cuenta de la producción de conocimiento. Sin embargo, un problema para esta estrategia es el de dónde buscar las “recetas” metodológicas. Los manuales canónicos de metodología para las simulaciones computacionales son, evidentemente, un contexto privilegiado. Pero, aquí es importante distinguir, porque a veces la literatura filosófica no lo hace, entre los contextos de recomendaciones metodológicas para la ingeniería de las “recetas” para las ciencias naturales. Nociones centrales en estos ámbitos, como las de verificación y validación, adquieren un sentido distinto en cada uno de estos contextos de diseño y puesta a prueba.

3. Verificación y validación

El objetivo de esta sección es señalar el complejo entramado de prácticas que se puede observar en la práctica científica e ingenieril contemporánea, proponiendo una lectura alternativa de la bibliografía técnica sobre la metodología empleada en dichos contextos para confiar en los resultados de las simulaciones. En nuestra lectura ponemos el énfasis en cómo necesariamente existen ciertos momentos de ajuste y bucles en el proceso, que responden a criterios que provienen de valores epistémicos, incluso cuando se trata de los elementos “más técnicos” del proceso.

De manera tradicional, por verificación de una simulación se entiende el proceso mediante el cual el desarrollador y el usuario de dicha simulación pueden asegurarse de que el algoritmo que discretiza el modelo matemático que se está implementando lo hace sin introducir distorsiones. Esto implica dos etapas, como sugirió Roache (1998). Por un lado, la verificación del código, es decir, verificar que el código resuelve la versión discreta de las ecuaciones que gobiernan al modelo. Por otro lado, es necesario llevar a cabo una verificación del cálculo o de las soluciones, por lo que hay que atender al proceso por el cual se puede demostrar que las soluciones que se obtienen con dicho código

son numéricamente exactas, dentro de los errores racionalmente aceptados para el método que se está usando y los recursos disponibles. Hasta aquí no hay física involucrada, que es lo que corresponde evaluar en la etapa de validación. Por ésta se entiende el proceso por el que se contrastan los resultados del modelo verificado con resultados provenientes de medios externos a la simulación que se pretende validar. Validación es uno de los conceptos más discutidos en la literatura técnica y el que más se presta a confusiones conceptuales. Creemos que hay que distinguir entre al menos dos concepciones del proceso, que hemos dado a llamar validación débil y validación fuerte. En la versión débil, las fuentes de datos con los que se puede contrastar un modelo son variadas, incluyendo comparaciones con otros modelos, tanto analíticos como simulaciones ya validadas, datos históricos y experimentales disponibles acerca del fenómeno o sistema de interés, tests de Turings (en los que se le pide a un experto que discrimine entre resultados del modelo y resultados empíricos), etc. y responden a los lineamientos trabajados por Sargent (2013) y gran parte de las referencias incluidas allí. La versión fuerte sólo permite realizar la validación contrastando los resultados del modelo verificado con las mediciones realizadas sobre un experimento de validación, diseñado y llevado a cabo siguiendo los mismos supuestos y especificaciones que se consideraron al plantear el modelo matemático. Esta línea es la que prevalece dentro de la comunidad de ingeniería mecánica y de dinámica computacional de fluidos. Es importante señalar que ambas concepciones reconocen que los procesos de verificación y validación son continuos y no se debe hablar de modelos completamente verificados y validados, sino de si cumplen los requisitos previstos para los mismos por parte de sus desarrolladores y usuarios.

La verificación del código incluye, a su vez, dos clases de actividades. Por un lado están aquellas que involucran todas las pruebas que se puedan realizar para determinar el grado de precisión numérica del algoritmo, incluyendo las soluciones analíticas para las partes que resuelven modelos físicos simples, soluciones de alto grado de confianza [o *benchmark*] para las ecuaciones diferenciales parciales y ordinales, pruebas de conservación, pruebas de simetría, pruebas de convergencia iterada y el método de las soluciones manufacturadas. Se dice que estas operaciones son netamente empíricas ya que responden a procesos de testeo, observación y comparación, en tanto no puede probarse formalmente la equivalencia entre el código y las soluciones de contraste (Oberkampff & Roy 2010, 33). El otro aspecto involucrado en la verificación del código consiste en las prácticas usadas para asegurar la calidad del software, como son desarrolladas e implementadas por las comunidades de la ciencia de la computación y la ingeniería del software, incluyendo las etapas de diseño, desarrollo, operación y mantenimiento, así como también las relativas a la administración y documentación. Las tres formas clásicas de llevar a cabo el análisis de la calidad de software son el análisis estático (en el que se inspecciona el código en busca de errores), el análisis dinámico (en el que se verifica el comportamiento del programa en un conjunto finito de casos contra el resultado esperado para los mismos) y el análisis o verificación formal.

Si bien posee una dinámica epistémica muy fértil, la verificación formal no suele ser utilizada en contextos de ciencia computacional e ingenieriles debido a razones que van desde que normalmente la mayoría de los ingenieros de software no reciben entrenamiento en ella hasta los problemas de escalabilidad del método cuando se intentan simular sistemas muy grandes. Se suele aplicar en sistemas críticos (como el diseño de una planta nuclear) en el que el impacto económico y temporal de llevarlos a cabo es mínimo con respecto al riesgo involucrado (Heitmeyer, 2004). Si bien Oberkampff & Roy (2010, 159) recomiendan no utilizar métodos formales para el software de computación científica señalando las razones ya citadas, probablemente también se deba a que ven en dicha práctica un “proyecto filosófico” distinto para V&V, centrado en la deducción desde primeros principios mientras que, según su mirada, estas prácticas deberían verse más bien como “alineada con los procesos de razonamiento inductivo, es decir procesos que presentan la corrección de las piezas individuales de evidencia como soporte de la conclusión de corrección de la generalización” (Oberkampff & Roy 2010, 32). Estos autores (2010, 153) también señalan que las especificidades de la práctica de la computación científica obligan a los simuladores a generar un complejo entramado de procesos para lograr la validación del código. En particular porque el cambio de campo de aplicación impide una importación directa de lo que se consideran buenas prácticas de validación en los contextos de ingeniería de

software. En dichos contextos normalmente se conoce de antemano el comportamiento esperado y que se pretende obtener con el código que se está implementando, lo que incluso raramente sucede en casos aplicados como en ingeniería, mucho menos en las simulaciones computacionales en ciencia.

El segundo aspecto es la verificación de las soluciones, que intenta identificar y cuantificar los errores que éstas pueden contener, provenientes de errores de los usuarios al manipular los valores de entrada y en el procesamiento de los valores generados por la simulación, además de los errores numéricos que pueden presentarse por la acción misma del algoritmo que ejecutó el modelo discretizado. Las fuentes usuales de la última clase de errores yacen en la discretización temporal y espacial que se lleva a cabo en las soluciones numéricas a las ecuaciones del modelo original y en los errores de soluciones iterativas que dependen del método elegido para linealizar y resolver un conjunto de ecuaciones no lineales. Además, hay que considerar posibles errores de redondeo y de muestreo estadístico. Si bien algunas de las estrategias que se emplean son similares a las utilizadas en la verificación del código, la gran diferencia radica en que en esta parte del proceso los errores numéricos no pueden ser simplemente evaluados, sino que deben ser estimados debido a que no se conocen las soluciones exactas del modelo. Si el modelo con el que se trabaja tiene singularidades o discontinuidades, los errores numéricos son muy difíciles de estimar ya que los métodos para lidiar con ellos requieren la continuidad y la diferenciabilidad de las soluciones. Algo similar ocurre con los modelos multiescala, en los que se resuelve un fenómeno físico con submodelos de distintas escalas temporales o espaciales, lo que puede atar los errores físicos con los errores numéricos, especialmente cuando el cambio de escala obliga a realizar un cambio de alguno de los submodelos involucrados.

La validación es el otro pilar para generar confianza en los resultados de las simulaciones y es la que puede comparar las soluciones obtenidas del modelo computacional con resultados de otras fuentes. Nos concentramos aquí en lo que hemos dado a llamar “validación fuerte”, esto es, la que requiere que la comparación de dichos resultados se realice contra las mediciones llevadas a cabo sobre un experimento diseñado de tal manera que instancie lo más fielmente posible la física que subyace al modelo simulado. Concentrarnos en la variante fuerte de la validación nos permite mostrar cómo, incluso en la versión “más técnica” posible de esta práctica, deben ser incorporados criterios epistémicos para proceder en la evaluación del modelo, además de notar lo intrincada que se vuelven las prácticas experimentales con las simulaciones.

Lo primero que hay que remarcar es que para validar una simulación en sentido fuerte, es necesario dejar de lado la capacidad predictiva que pueda tener el modelo que está bajo evaluación. Esto quiere decir que los resultados de simulación con los que se trabajará no son los que se esperan del modelo sino los resultados que puedan ser medidos bajo las condiciones de control disponibles en un experimento de validación. Como señalamos antes, dicho experimento debe encarnar los supuestos físicos que se consideraron en el modelo y la magnitud física que se mida en dicho experimento debe ser exactamente la misma que se obtenga de la simulación, a efectos de poder generar un valor métrico que señale cuantitativamente la diferencia entre los mismos. Esto elimina la posibilidad de utilizar datos que provengan de otra fuente, incluso si son de resultados empíricos de la misma magnitud. La principal razón es que más allá de la calidad de los datos disponibles, es muy difícil que en los reportes de experimentos se incluya el nivel de detalle necesario para determinar si todos los supuestos de dichos experimentos coinciden con los supuestos del modelo a validar. Esto tiene dos consecuencias importantes. Por un lado, los experimentos de validación tienen una clara diferencia con respecto a otras clases de experimentos, especialmente los denominados experimentos de calibración, que se llevan a cabo para ajustar el valor de un parámetro en una simulación debido a que su valor no puede calcularse desde las ecuaciones de la teoría o no corresponde a una magnitud física. Un modelo que incluya parámetros a definir por esta vía no puede ser fuertemente validado. De hecho hay que considerar que los experimentos de calibración son una subclase de las actividades de estimación de parámetros, que muchas veces también se realiza apelando a datos generados por simulaciones. Por otro lado, debe entrar en juego una suerte de sinergia entre los experimentadores y los modeladores ya que deben cerciorarse de estar implementando la misma configuración física en el experimento como en la simulación. A su vez, si bien el experimento debe ser desarrollado de manera conjunta, es importante que los analistas que obtengan los resultados de la simulación no tengan acceso a los

valores que provengan del experimento, para poder asegurar que su conocimiento de dicho valor no les llevó a modificar algún aspecto del modelo. Una vez realizado un número suficiente de mediciones experimentales y de simulaciones, se comparan los resultados para obtener una métrica de validación, un valor que indica la diferencia entre los resultados. Si la diferencia está dentro de los márgenes establecidos en las especificaciones, se puede proceder a realizar las predicciones. Si no, se evalúan las causas posibles de la diferencia y se intenta modificar el experimento o la simulación volviendo a repetir todos los pasos anteriores (especialmente los de verificación, si es que se modifica algo del código) hasta que la métrica de validación corresponda a los valores necesarios.

La principal dificultad de este procedimiento radica en que los modelos que pueden ser fuertemente validados son muy pocos, especialmente porque realizar experimentos de validación para sistemas altamente complicados puede ser extremadamente costoso, si es que no simplemente imposible. Para lidiar con este problema, se recurre a una jerarquía de validación, en la que se descompone al sistema de interés en distintos niveles, identificando así sistemas y subsistemas hasta llegar a los problemas unitarios [*unit problems*]. Para cada uno de ellos se deben llevar a cabo experimentos de validación. Esto deja entrever que el proceso de validación fuerte es claramente constructivista en el sentido en que reconoce que la calidad y la precisión de los datos experimentales disminuyen a medida que se asciende en la jerarquía, obligando a considerar en cada nivel el entramado de resultados y submodelos que están en juego. Así, la descomposición en niveles implica poder conocer de qué manera están atados los diferentes niveles a efectos de lograr un nivel de validación adecuado para el sistema general. Claramente en esta estrategia opera una comprensión lineal del enlace entre los niveles, lo que puede causar una disminución de la confianza en los resultados del proceso cuando se aplica a sistemas complejos. Una vez que se ha obtenido un grado adecuado según las métricas de validación, se puede proceder a realizar predicciones con dicho modelo.¹

4. Conclusiones

Una vez apreciada la compleja dinámica de la práctica de verificación y validación (V&V), la pregunta que hay que responder es qué tanto puede dicha metodología funcionar como garantía epistémica para los resultados de las simulaciones. Creemos que una correcta interpretación de dicha práctica nos lleva a una posición intermedia entre la defendida por Winsberg en un extremo y Morrison en el otro. Esto es, acordamos en la importancia de rescatar la aproximación desde la metodología, ya que permite dar cuenta de las prácticas de construcción y de la confiabilidad de los resultados de las simulaciones computacionales, así como de los límites de la clasificación original para la epistemología de las simulaciones (ni las aproximaciones que ven a las simulaciones como modelos, ni las que las ven como casos de experimentación, ni aun las perspectivas híbridas pueden hacerlo). Ahora bien, resulta importante que la metodología en general—y V&V en particular—no se conviertan en un ideal metodológico que deba cumplirse paso a paso y de modo secuencial ya que como se puede observar en la reconstrucción de dicha práctica de la sección anterior, esto es casi imposible de lograr, incluso en sistemas muy sencillos. En general, la literatura filosófica ha tomado la estrategia de separar ideales metodológicos de ideales epistémicos sin atender a sus vínculos. La estrategia de autores como Morrison consiste en convertir a los ideales metodológicos en ideales epistémicos. Sin embargo, tanto en la primera alternativa como en la segunda no se puede dar cuenta de (a) los procesos de construcción de los modelos computacionales, (b) del carácter heurístico, reconocido por los propios metodólogos, del proceso de V&V, y (c) la dinámica de retroalimentación entre verificación y validación que hace que la distinción entre ambas fases resulte menos nítida. En consecuencia, no se debería considerar a la práctica de V&V como una mera estrategia evaluada pragmáticamente, en la que las

¹ Dichas predicciones por definición están fuera del rango de validación, por lo que es necesario introducir en todo el análisis otras herramientas que permitan establecer el grado de confianza de dichos resultados. El aspecto clave aquí—y que se vuelve un elemento crucial para entender la posibilidad de garantizar a las simulaciones mediante V&V—es el de las incertidumbres, que no podemos abordar aquí por razones de espacio.

simulaciones meramente se ajustan para que los resultados coincidan con los valores empíricos disponibles, ni como una mera receta metodológica para garantizar resultados. Se trata más bien de un componente clave de las prácticas científicas e ingenieriles en el que se puede observar la rica dinámica existente entre los ideales epistémicos y metodológicos y los valores epistémicos que entran en juego al reconocer que dichos ideales nunca pueden ser alcanzados fehacientemente. Por lo tanto, las simulaciones computacionales deben considerarse sólo en un contexto ampliado de prácticas científicas, en el que la validez epistémica de los resultados no se encuentra garantizada por ideales universales sino por prácticas más bien locales que pueden variar de acuerdo al problema de interés particular que está siendo analizado.

5. Referencias

- Frigg, Roman, Reiss, Julian (2009), "The philosophy of simulation: hot new issues or same old stew?", *Synthese* 169 (3), 593-613.
- Heitmeyer, Constance (2004), "Managing complexity in software development with formally based tools". *Electronic Notes in Theoretical Computer Science* 108, 11-19.
- Morrison, Margaret (2009), "Models, measurement and computer simulation: The changing face of experimentation", *Philosophical Studies* 143 (1):33-57.
- Morrison, Margaret (2014), "Values and Uncertainty in Simulation Models", *Erkenntnis* 79 (5), 939-959.
- Morrison, Margaret (2015), *Reconstructing reality: Models, mathematics, and simulations*. Oxford University Press.
- Oberkampf, William L., Roy, Christopher J. (2010), *Verification and validation in scientific computing*, Cambridge University Press.
- Parker, Wendy S. (2008), "Franklin, Holmes, and the epistemology of computer simulation", *International Studies in the Philosophy of Science* 22 (2), 165-183.
- Roache, Patrick J. (1998), "Verification of Codes and Calculations". *AIAA Journal* 36 (5), 696-702.
- Sargent, Robert G. (2013) "Verification and validation of simulation models", *Journal of simulation* 7 (1), 12-24.
- Winsberg, Eric (2009), "A tale of two methods", *Synthese* 169 (3), 575-592.
- Winsberg, Eric (2010). *Science in the Age of Computer Simulation*. The University of Chicago Press.