

## SIMULACIONES COMPUTACIONALES: UN ANÁLISIS DE DOS CONCEPCIONES ANTAGÓNICAS

JUAN M. DURÁN

---

**Abstract.** In this article I propose an analysis of two opposing viewpoints on the notion of computer simulation. The first one, proposed by Stephan Hartmann, has been adopted by philosophers who believe that computer simulations can be understood by means of a familiar philosophy, such as a philosophy of scientific models or a philosophy of scientific experimentation. The second notion belongs to Paul Humphreys, who suggests that computer simulations are objects of genuine philosophical interest, and that understanding their nature allows us to understand their role in current scientific practice.

**Keywords:** Computer simulations; scientific models; computational templates.

---

### 1. Introducción

Una mirada rápida a la historia de la ciencia es suficiente para convencernos de que no podríamos hablar de progreso científico, en la definición favorita del filósofo, sin hablar también de instrumentos y métodos tecnológicos que guían dicho progreso. El siglo XX nos encuentra con lo que podría ser la forma más compleja de escudriñar el mundo que nos rodea: la computadora digital. De particular interés para este artículo serán las simulaciones computacionales, métodos —antes que instrumentos— que han abierto nuevas formas de entender el mundo y los fenómenos que hay en él.

A pesar de la ubicuidad de las simulaciones en la práctica científica, las posiciones filosóficas acerca de su novedad e importancia están divididas. El problema es que no está del todo claro si en verdad la discusión filosófica sobre simulaciones es importante, o si es simplemente un tema que se puede abordar desde una filosofía más familiar, como la de los modelos científicos —especialmente modelos matemáticos— (como por ejemplo Bailer-Jones (2009)) o desde la filosofía de la experimentación de laboratorio (como por ejemplo Radder (2003)).

En un trabajo del 2009, Roman Frigg y Julian Reiss analizaron la relevancia de las simulaciones computacionales para la práctica científica contemporánea. En este artículo se cuestionó fuertemente la postura de filósofos como Eric Winsberg (2001), Peter Galison (1996) y Fritz Rohrlich (1990), entre otros autores, que entienden las simulaciones computacionales como objetos de genuino estudio filosófico. Para estos últimos filósofos, el poder de cómputo asociado a las simulaciones excede nuestras capacidades cognitivas y por ello “proveen una metodología cualitativamente nueva y diferente para las ciencias naturales” (Rohrlich 1990, p.507), por lo que “tienen

una epistemología distintiva” (Winsberg 2001, p.447), una epistemología “que no es familiar para la mayoría de la filosofía de la ciencia” (Winsberg 1999, p.275).

Frigg y Reiss entienden que aquí hay un interés exagerado y hasta infundado sobre la necesidad de entender filosóficamente las simulaciones computacionales. Admiten, desde luego, que las simulaciones presentan problemas específicos de corte matemático, tecnológico y hasta psicológico, pero nada remotamente cercano a un problema filosófico. En particular, para Frigg y Reiss las citas anteriores se enmarcan en variantes de problemas que han sido discutidos en otros contextos anteriormente, como en la filosofía de los modelos científicos o la filosofía de la experimentación científica. “En suma”, concluyen estos autores, “estamos de acuerdo en que las simulaciones introducen algo nuevo y excitante para la ciencia, pero dudamos de que esto requiera reescribir la filosofía de la ciencia” (Frigg et al. 2009, p.595).

Las objeciones de Frigg y Reiss dan en el corazón del debate filosófico sobre simulaciones computacionales. Su posición, sin embargo, ignora aquello que las simulaciones computacionales pueden aportar en y por sí mismas. Así pues, interpretar las simulaciones desde la perspectiva de una filosofía familiar ilumina el problema solo desde un ángulo, a saber, el de la filosofía de los modelos matemáticos —tal y como son utilizados en las ciencias— o el de la experimentación científica. Se ignora, entonces, aquello que es propio de las simulaciones y que es excluyente de una filosofía más familiar.

En este trabajo analizo dos nociones antagónicas de simulación computacional con el fin de mostrar que existen problemas filosóficos genuinos en esta área de la ciencia, a diferencia de lo que Frigg y Reiss sostienen. En este sentido, primero discuto una noción bastante generalizada entre los filósofos, la cual supone que una simulación computacional es un proceso dinámico que imita otro proceso dinámico. Esta es la definición que elaboró Stephan Hartmann en (1996) y que se basa en reducir la complejidad metodológica —y la consiguiente evaluación epistémica— de las simulaciones computacionales. Por otro lado, y actuando como posición antagónica a la de Hartmann, encontramos la definición de Paul Humphreys. Este autor desarrolló la noción de simulación computacional como *plantilla computacional* (Humphreys 2004). Una plantilla computacional, como explicaré más adelante, es entendida como una definición que involucra conceptos de la filosofía de la ciencia con nociones de la ciencia de la computación. En este sentido, Humphreys propone una definición de simulación computacional filosóficamente más rica e interesante. En particular, considero que esta definición es la más adecuada que tenemos hoy para comprender la práctica contemporánea de simulaciones computacionales.

## 2. Hartmann y los modelos dinámicos

Stephan Hartmann ha popularizado la idea de que las simulaciones computacionales son procesos dinámicos que *imitan* otros procesos dinámicos. Filosófica y conceptualmente anodina, esta idea ha sido sorprendentemente bienvenida entre los filósofos de las simulaciones computacionales. Recientemente, Wendy Parker ha hecho explícita referencia a esta idea: “yo caracterizo una simulación como una secuencia de estados ordenada temporalmente que sirven como una representación de otra secuencia ordenada temporalmente de estados” (Parker 2009, p.486). A coro de esta, Francesco Guala (2002) sigue a Hartmann al distinguir entre modelos estáticos y dinámicos, evolución temporal de un sistema y en el uso de simulaciones para resolver matemáticamente el modelo implementado. Hartmann ofrece la siguiente definición:

Las simulaciones están muy relacionadas con los modelos dinámicos. Concretamente, una simulación resulta cuando las ecuaciones del modelo dinámico son resueltas. Este modelo es designado para imitar la evolución temporal de un sistema real. Para poner el mismo punto de otra manera, *una simulación imita un proceso mediante otro proceso*. En esta definición, el término ‘proceso’ se refiere solamente a algún objeto o sistema cuyo estado cambia en el tiempo. Si la simulación corre en una computadora, se denomina *simulación computacional*. (Hartmann 1996, p.83, énfasis en texto original)

Notemos primero que la noción de simulación computacional está asociada con la de modelo dinámico, más específicamente, cuando las ecuaciones de dicho *modelo dinámico* han sido resueltas en una computadora física. ¿Qué es para Hartmann un *modelo dinámico*? Hay buenas razones para suponer que esta noción de simulación computacional está íntimamente relacionada con la noción de modelo matemático, entendido este como el conjunto de supuestos teóricos sobre algún sistema empírico. Primero, porque el autor admite que el modelo dinámico hace uso de ecuaciones diferenciales. De hecho, en su artículo explicita que “el modelo dinámico correspondiente es convenientemente formulado en el lenguaje de las ecuaciones diferenciales” (Hartmann 1996, p.84), alineándose con la interpretación de Fritz Rohrlich (1990) sobre la relación modelo matemático-simulación computacional. La otra razón es que los casos que utiliza para ilustrar las simulaciones computacionales son todos dependientes de la física moderna. Así es que Hartmann (1996) entiende que la aproximación numérica de la densidad de espacio de fase continua constituye una simulación computacional. En este contexto, podemos reformular la definición y decir que una simulación computacional resulta cuando las ecuaciones de un modelo matemático —que imita la evolución temporal de un sistema físico— son resueltas en una computadora.

A fin de mostrar la relevancia filosófica de las simulaciones computacionales, veamos cómo Hartmann sostiene erróneamente al menos tres supuestos: primero, que una simulación computacional consiste en la *implementación directa* en una computadora de un modelo matemático; segundo, que una simulación computacional es un modelo matemático con la capacidad de obtener un conjunto de soluciones mediante el cálculo y tercero, que como caso estándar se implementa *un* modelo como simulación computacional que representa *un* sistema empírico. El primer supuesto ignora la metodología que hay por detrás de las simulaciones computacionales, clave para la crítica contra la sumisión de estas a una filosofía más familiar, como la de los modelos científicos. Por otra parte, el segundo supuesto es demasiado optimista sobre las posibilidades reales de cómputo de una simulación. El tercer supuesto directamente ignora gran parte de la práctica misma de simulaciones computacionales. Veamos esto con más detalle.

En cuanto a la metodología de las simulaciones computacionales (primer supuesto), un modelo matemático no es *directamente implementable* en una computadora digital, sino que sufre distintos tipos de transformaciones. Algunas de estas transformaciones son de tipo formal, como es el caso de utilizar el método de discretización de Euler para resolver ecuaciones diferenciales ordinarias integrando numéricamente a partir de un valor inicial dado. Otra forma de transformar el modelo matemático original consiste en interpretar algorítmicamente decisiones de diseño. Por ejemplo, si efectivamente se implementa Euler como método de discretización, entonces se deben tomar una serie de decisiones sobre los tipos de errores asociados a este método, por ejemplo errores de truncamiento local y global, errores de propagado, errores de redondeo, etc.

Además, las simulaciones computacionales no hacen uso de un lenguaje matemático, como supone Hartmann, sino que más bien el modelo simulacional debe ser escrito en un lenguaje computacional apropiado.<sup>1</sup> Así, de representar procesos continuos, se pasa a representar procesos discretos; de tener errores asociados a simplificaciones en el sistema de ecuaciones, se pasa a tener errores de limitación en la capacidad de cálculo y de poseer un lenguaje con un alto grado de expresividad, se pasa a tener que evaluar el poder de expresividad de una multiplicidad de lenguajes, dependiendo de las características de la simulación que se quieran enfatizar.

Un caso que ilustra lo que se ha dicho hasta ahora es el modelo de oscilador armónico simple. Este modelo es tratado por matemáticos como un sistema de ecuaciones lineales, cuando en realidad la verdadera ecuación para el péndulo no lo es. El problema son nuestras limitadas capacidades de cálculo. Así, para poder encontrar soluciones al modelo, el procedimiento clásico 'linealiza' lo que no es lineal, eliminando todos los términos incómodos de la ecuación. Este método efectivamente proporciona una buena aproximación a las ecuaciones del péndulo, una en la que se supone que las oscilaciones son muy pequeñas. Si se implementa eventualmente el modelo

de oscilador armónico simple en una computadora, lo primero que hay que hacer es discretizar el modelo, ya que así se garantiza su tratabilidad en la computadora. En particular, las ecuaciones que se discretizan y calculan son efectivamente las del péndulo, esto es, ecuaciones no lineales. Los errores que surjan provienen del tipo de método de discretización utilizado y de los límites en la capacidad de la computadora —ya sea en forma de errores de truncados o de redondeo. La elección del lenguaje de programación es también importante, ya que dependerá de cómo este compute diferentes funciones —como la raíz cuadrada y la multiplicación—, de la precisión de constantes numéricas —como  $\pi$ — y el tratamiento de infinito —como números muy grandes, pero finitos—, entre otros temas. En definitiva, solo ignorando estos elementos de una metodología exclusiva de las simulaciones computacionales es posible considerar que el modelo matemático original del oscilador armónico simple se implementa directamente en una computadora.

Sostengo, como segundo supuesto, que Hartmann es demasiado optimista en suponer que todo modelo matemático es implementable. Esto, desde la perspectiva de la teoría de la computación, no es correcto. Como ya se indicó, hay una metodología asociada a las simulaciones computacionales que establece los límites de lo que es implementable y de lo que no lo es. Los métodos de discretización empleados deben hacer un uso cuidadoso en el tratamiento de los valores nominales del sistema de ecuaciones. Discretizar atributos continuos para un conjunto de ecuaciones comienza con la elección de la representación discreta de las variables, escalares, vectores y demás del sistema de ecuaciones. Por ejemplo, en electromagnetismo es común elegir como variables principales las proyecciones normales de la densidad del flujo magnético para las celdas computacionales y la proyección normal de la intensidad del cuerpo eléctrico para los bordes de dichas celdas. Esto es así porque los campos magnéticos y eléctricos son continuos en la interacción entre diferentes materiales. Pero no siempre es posible formular operadores discretos que satisfagan todas las propiedades deseadas. Por ejemplo, en dinámica multidimensional de gases Lagrangianos no es posible construir una discretización que simultáneamente conserve energía y preserve entropía en flujos isentrópicos (Knoll et al. 2005, p.190). Estos ejemplos muestran que presuponer que es posible obtener un conjunto de resultados —correctos— de *cualquier* modelo matemático es ir más allá de lo que la metodología y la ontología de las simulaciones computacionales nos permiten.

No voy a adentrarme más en los detalles de estos dos puntos. Entiendo que una mirada desde la metodología de las simulaciones computacionales —y los cambios ontológicos subyacentes— es suficiente para desestimar la definición de Hartmann como apropiada. Nótese que hasta aquí hemos aceptado que un modelo matemático es implementado, una vez que se han hecho previamente los pasos metodológicos apropiados, como una simulación computacional. Pero debemos preguntarnos, ¿es posible una simulación computacional sin contar previamente con *un* modelo sub-

yacente completo (tercer supuesto)? La respuesta es que es perfectamente posible simular diversos sistemas empíricos sobre los cuales solo tenemos modelos parciales. Esto significa que la simulación está compuesta de *múltiples* modelos que, individualmente, solo representan una porción del sistema objeto. Un buen ejemplo de esto es Marco Ajelli y su grupo (2010), quienes utilizan dos tipos diferentes de simulaciones computacionales para comparar posibles escenarios de una epidemia.

El caso es muy interesante como muestra de las nuevas prácticas científicas. Ajelli et al. proveen una comparación de los resultados obtenidos con una simulación estocástica basada en agente y con una simulación basada en ecuaciones —conocido como GLocal Epidemic and Mobility o GLEaM. La primera simulación incluye una representación de la población Italiana a través de datos de la estructura socio-demográfica. A su vez, y para determinar la probabilidad que tiene un agente de conmutar de una municipalidad a otra, los autores utilizan un modelo obtenido de teorías de transporte. La dinámica de la epidemia se basa en modelos estocásticos que integran infecciones susceptibles y latentes. Los autores definen esta simulación basado en agentes como “una simulación estocástica, explícita espacialmente, y discreta en tiempo, donde los agentes representan individuos humanos [...] Una de las características principales de la simulación es la caracterización de una red de contactos entre individuos basada en un modelo realista de la estructura socio-demográfica de Italia” Ajelli et al. (2010).

Por otro lado, la simulación GLEaM es una red multiescala de movilidad basada en datos de la población. Un GLEaM típico consiste en al menos tres niveles de modelos. Un primer nivel, en el cual la población y la movilidad permiten la partición del mundo en regiones geográficas. Un segundo nivel, denominado red de subpoblación, en el que las interconexiones representan los flujos de individuos a través de la infraestructura de transporte y los patrones generales de movimiento. Finalmente, un tercer nivel que define dentro de cada subpoblación la dinámica de las infecciones.

Notemos que los criterios de validación de estas simulaciones no provienen enteramente del modelo matemático ni de datos empíricos, como es generalmente el caso, sino más bien de las distintas estrategias de integración de resultados y de los supuestos que hay detrás de cada simulación (Ajelli et al. 2010, p.2). Además, como la simulación basada en agentes y la GLEaM representan el mismo fenómeno a distintos niveles, es posible utilizar una para validar los resultados de la otra.

Con estos ejemplos a mano, la única manera para entender autores como Hartmann es suponiendo que las simulaciones computacionales implementan *un* modelo matemático que represente *un* sistema empírico. Esto, como hemos mostrado más arriba, contradice la práctica científica misma.

### 3. Humphreys y las plantillas computacionales

Hasta aquí podemos decir que es común entre los filósofos —ya se trate de Hartmann, Parker, Guala, o algún otro— dar por sentado que las simulaciones computacionales son posibles *gracias a* uno o varios modelos subyacentes. Aceptado este supuesto, la discusión discurre entonces entre aceptar las simulaciones como un nuevo tipo de modelo o diferenciarlas tratándolas como algo distinto, usualmente un experimento —pero al que en definitiva le subyace un modelo (Morgan 2003; 2005; Morrison 2009).

Paul Humphreys es de una opinión diferente. Para este autor la relación entre simulaciones computacionales y modelos científicos debe ser reconsiderada, puesto que el concepto de modelo con el que tradicionalmente se ha trabajado la ciencia y la filosofía de la ciencia no puede dar cuenta de esto que llamamos ‘simulaciones computacionales’. Esta lectura, correcta o incorrecta, abre nuevos caminos y nuevos interrogantes para el estudio que estoy llevando a cabo.<sup>2</sup>

Humphreys afirma que la filosofía de la ciencia ha tomado diferentes objetos como unidades básicas de análisis: teorías, leyes, programas de investigación, paradigmas y modelos. Sin embargo, ninguna de estas denominadas unidades básicas contribuye a los propósitos de la investigación y caracterización de las simulaciones computacionales. Será necesario, entonces, un cambio conceptual que abra el juego desde las bases. En este sentido, pues, introduce la noción de *plantilla* como término teórico que funciona, en principio, de sustituto de las viejas unidades básicas, ahora desgastadas y sin fuerzas. Como mostraré al final de la sección 3.1, no son las plantillas computacionales las que se desempeñan como unidades de análisis de las simulaciones computacionales, sino una 6-upla mucho más compleja denominada por el autor como *modelo computacional*.

Como indiqué anteriormente, Humphreys introduce un nuevo término teórico que no ha sido previamente utilizado por otros autores. Tomo el término *plantilla* como la mejor traducción del inglés “template” (Humphreys 2004, p.60).

Veamos ahora cómo el autor da forma a las plantillas mediante un ejemplo. La segunda ley de Newton, tal y como la conocemos, se describe para cualquier fuerza, masa y aceleración. Sin embargo para hacer uso de ella aplicándola en un caso particular es necesario primero especificar una función de fuerza particular, ya sea la fuerza gravitacional, electrostática, magnética o cualquier otra; algo similar ocurre con los restantes elementos de los que habla la ley: masa, aceleración, etc. (Humphreys 2004, p.60). A partir esto es posible interpretar la segunda ley de Newton como una “plantilla teórica”, esto es, como restricciones muy generales que no son aplicables a ningún fenómeno particular sin la previa especificación de algunos de sus componentes. Variando de especificación, entonces, será posible dar cuenta de los distintos fenómenos que puede ‘contener’ la plantilla. Otro ejemplo es la teoría de

la probabilidad, donde una función de distribución, una función de densidad y otras variables son completadas para que la plantilla adquiera una funcionalidad específica. Una vez completada con esta información, pierde su generalidad característica particularizándose en un fenómeno.

En términos más generales, una *plantilla teórica* se constituye como una construcción sintáctica de alto grado de generalidad y abstracción, que se aplica a diferentes fenómenos al aumentar los grados de especificidad de las entidades de las que habla. Es en este sentido que Humphreys especula que estas plantillas se constituyen como un objeto de estudio propio e independiente de lo que tradicionalmente la filosofía y la ciencia han tomado como unidades de análisis.

Estas plantillas son comunes en distintos ámbitos científicos y no deben identificarse estrictamente con leyes o teorías, modelos o proyectos de investigación, como la mente filosófica podría verse tentada a pensar. Por otro lado, tampoco son construcciones del todo inertes, pues el autor les adscribe al menos dos funciones principales: una de ellas consiste en representar el mundo; la otra, en facilitar el cálculo. Así pues Humphreys es de la opinión que, para lograr una buena caracterización de las simulaciones computacionales, una unidad básica de análisis debe poder dar cuenta de estas dos características, algo que ni la concepción sintáctica ni la concepción semántica de las teorías logran. De este modo las plantillas no solo suplirían la vacante dejada por los modelos, sino, por extensión, se convertirían en las nuevas unidades básicas de análisis filosófico de la ciencia. Veamos pues, los motivos por los cuales los modelos han sido destituidos.

La concepción sintáctica de las teorías, tal y como se la conoce en su versión tradicional, ve a las teorías como organizadas deductivamente en conjuntos de sentencias especificadas en algún lenguaje, usualmente en una lógica de primer orden. A los términos del lenguaje se le suele atribuir semántica mediante reglas de correspondencia que vinculan los objetos sintácticos con propiedades en el mundo. Así pues, la concepción sintáctica de las teorías puede reconciliar fines representacionales y fines de cálculo pero solo “a un nivel tan alto de abstracción, que las realidades del cálculo científico son dejadas atrás” (Humphreys 2004, p.96).

Su rival tradicional, la concepción semántica de las teorías, hace abstracción de representaciones particulares de las teorías mediante la identificación de estas con una clase o familia de modelos matemáticos. Una de las razones primarias de este paso abstractivo es remover las peculiaridades sintácticas de las diferentes formulaciones de la misma teoría, permitiendo así que uno se centre en la estructura subyacente. Lo importante es que se trata de una poderosa perspectiva, en especial porque desde esta concepción un modelo exitoso puede fácilmente ser extendido a otros fenómenos o sistemas similares.

Lo que ambas concepciones ha olvidado, de acuerdo con Humphreys, es el hecho de que representar con un alto grado de abstracción conduciría a descuidar lo



que se hace diariamente en la práctica científica. Así pues, representar importa y en gran medida (Humphreys 2004, p.57). En el caso de la concepción sintáctica de las teorías, la lógica dominante enfatiza las representaciones proposicionales; en el caso de la concepción semántica, las representaciones se dan mediante estructuras matemáticas. Y si bien para ambos casos cualquier sistema puede, en principio, ser representado, se ignora la posibilidad epistémica de asimilar fácilmente la información contenida en las diversas representaciones. Humphreys, en todo caso, considera más importante abogar por mecanismos de representación comprensibles desde la perspectiva cognitiva humana.

Un aspecto estándar del método científico requiere que se recolecten datos y se los compare con las predicciones de nuestras teorías o modelos. En muchas áreas de la ciencia es absolutamente imposible realizar este trabajo ‘a mano’ ya que los cálculos son astronómicos y los resultados se presentan, muchas veces, en grandes cantidades de números a los que es imposible darle una interpretación sin algún tipo de ayuda. Para estos conjuntos de datos, la imagen de un hombre sentado frente a un instrumento mientras recolecta concienzudamente información sobre las observaciones que finalmente serán comparadas con predicciones teóricas es poco menos que realista. Por el contrario, recurrir a ayuda tecnológica que aumente o mejore nuestras habilidades cognitivas naturales es un paso necesario, si pretendemos procesar grandes cantidades de información.<sup>3</sup> De aquí que para Humphreys la ayuda tecnológica se presenta como *extensión de nosotros mismos*. Así pues, para los propósitos de la ciencia, la representación de los resultados de una simulación es más que un dispositivo heurístico, ya que se convierte en parte de la simulación misma. No es solo la extensión de las habilidades computacionales humanas lo que produce un nuevo tipo de método científico, sino la combinación de la ‘extensión de nuestras capacidades cognitivas’ y la ayuda en el modo de presentación de los resultados de la simulación.

Como hemos afirmado al comienzo, para el autor ninguna de las unidades básicas de análisis tradicionales cumple un papel predominante y decisivo en la caracterización de las simulaciones computacionales. Será la inclusión de la noción de plantilla la que procure la doble posibilidad de ‘eliminar’ los tradicionales modelos de este dominio y dé una novedosa caracterización de las simulaciones computacionales. En lo que sigue discutiré la concepción de Humphreys acerca de las simulaciones computacionales en el marco de lo expuesto hasta el momento.

### 3.1. Plantillas computacionales

Hemos visto que una plantilla se describe como un conjunto de restricciones muy generales y abstractas. Si, al especificar la plantilla, obtenemos un resultado computacionalmente tratable,<sup>4</sup> entonces hemos obtenido una *plantilla computacional*.

Una plantilla computacional, entonces, puede ser representada mediante la

5-upla denominada *supuestos de construcción* [construction assumptions] (Humphreys 2004, p.78).

⟨Ontología, Idealizaciones, Abstracciones, Restricciones,<sup>5</sup> Aproximaciones⟩

El primer elemento de la 5-upla, la *ontología*, especifica tanto el tipo de objetos representados en la plantilla como la naturaleza de las propiedades y relaciones que ocurren dentro de la plantilla. Esta especificación incluye, usualmente, mecanismos que operan a nivel intra-sistémico e inter-sistémico. A diferencia de los otros componentes, la ontología posibilita la conversión de un modelo general a un modelo de fenómeno específico, facilitando, asimismo, las bases de construcción para los otros elementos de la 5-upla. Así, un modelo podría ser inestable, si la ontología no es especificada correctamente, puesto que no se sabría, por ejemplo, si las propiedades imputadas al sistema se encuentran presentes en el modelo.

El segundo componente importante debería entenderse técnicamente como una sub-upla de los supuestos de construcción que contiene los últimos cuatro elementos. Esta sub-upla usualmente es formulada durante el proceso de construcción y se denomina el *conjunto de corrección* [correction set]. El propósito principal de este conjunto de corrección es indicar, por adelantado, los modos en los que las plantillas computacionales necesitarán ser ajustadas cuando los resultados no se ajusten a los datos empíricos. Asimismo, este conjunto de corrección usualmente no es utilizado en la construcción inicial de la plantilla por una serie de razones. Una de ellas es que aún son necesarios ajustes entre las predicciones de la plantilla y el comportamiento del sistema; otra razón de peso es que la capacidad de cómputo requerida para usar una plantilla más refinada y precisa no suele estar disponible.

Los componentes del conjunto de corrección varían, dependiendo de los supuestos de construcción que han sido utilizados para la plantilla computacional. Los componentes más importantes son, de acuerdo con el autor (Humphreys 2004, p.79):

- Relajación de las Idealizaciones (por ejemplo, pasar de considerar al sol como una esfera perfecta a tomarlo como un esferoide achatado).
- Relajación de las Abstracciones (por ejemplo, agregar variables previamente omitidas al modelo, tales como (...) incrementar el número de individuos en una simulación ecológica).
- Relajación de las Restricciones (por ejemplo, pasar de considerar un conductor dentro del cual el calor se conserva a uno donde el calor fluye de un extremo a otro).
- Refinamiento de las Aproximaciones (por ejemplo, reduciendo el tamaño de la grilla espacial en aproximaciones de diferencia finitas a modelos continuos o reemplazando una variable agregada con valores precisos).

Como se sugirió anteriormente, cambios en la ontología del sistema son posibles y significan movimientos radicales que ocurren cuando la plantilla computacional es aplicada a un sistema diferente —como por ejemplo cuando utilizamos un modelo de contagio de una epidemia a un modelo de la propagación de ideas revolucionarias en una sociedad. Así, cambios en la ontología requerirán, usualmente, cambios en otros elementos del conjunto de corrección. Naturalmente, un cambio en la ontología está muy próximo a comenzar directamente un nuevo proceso de construcción de una plantilla computacional.

La relación (plantilla computacional, conjunto de corrección) puede ser analizada desde distintas perspectivas. Por ejemplo, una decisión práctica usual, cuando se presentan diferencias entre las predicciones de la plantilla y los datos empíricos, es que, antes de abandonar la plantilla por otra más precisa, es preferible primero poner la atención en el conjunto de corrección y desde allí ajustar la plantilla a fin de obtener predicciones más precisas. Otro aspecto interesante adjudicado a las plantillas es que los individuos que las construyen usualmente “tienen un refinado sentido acerca de qué componentes incluidos en la construcción de la plantilla están justificados y cuáles no. Usualmente tienen, antes del testeo, ideas claras acerca de las partes de la construcción que serán las primeras en ser revisadas cuando la plantilla falle en dar predicciones precisas” (Humphreys 2004, p.81). Diferentes procesos de construcción serán elegidos sobre la base de la ontología del sistema y la capacidad de corrección de la misma, aun cuando las ecuaciones involucradas sean, formalmente hablando, las mismas.

Pese a que las plantillas computacionales resultan más específicas que las plantillas teóricas, una característica de estas es que no son construidas a partir de leyes o teorías. Este es uno de los pocos asuntos en el que Humphreys no dejó espacio para dudar: “las teorías no juegan ningún papel en la construcción de plantillas” (Humphreys 2004, p.89). Partamos de un ejemplo claro como es un proceso de Poisson. Este proceso puede consistir en una plantilla de gran generalidad que puede ser utilizada para representar fenómenos extremadamente variados tales como el número de llamadas telefónicas pasadas por una central de teléfonos, el número de clientes que llegan a un banco, el número de desintegraciones radioactivas por unidad de tiempo, etc. Estos ejemplos muestran que una plantilla puede ser construida con la ausencia de algo que se parezca a una teoría sistemática tangencial a un dominio de objetos. Es difícil pensar que el número de llamadas por teléfono, el número de desintegraciones radiactivas y el número de clientes que llegan a un banco tienen todos en común una misma teoría que justifique el uso de los mismos supuestos. Por el contrario, lo único en común entre estos casos es que son factibles de ser representados y computados por medio de un proceso de Poisson.

Algo similar sucede con las leyes. En muchos casos en los que se construye una plantilla, las leyes intervienen en el proceso de construcción, pero en el caso del

ejemplo del proceso de Poisson no es plausible que ninguno de los supuestos dados cuente, al final, como una ley. Se podría pensar que los valores de los resultados pueden ser explicados en virtud de una distribución y, como tal, respondiendo a alguna ley. Pero si consideramos la distribución que cubre todo el proceso como una ley, esa ley no es en sí misma entendida en términos de leyes futuras, sino en términos de hechos estructurales acerca del sistema en cuestión y, por extensión, modelables con una plantilla. Para Humphreys estas ideas son suficientes para indicar que el entendimiento científico puede tomar un lugar aun en la ausencia de leyes.

Hasta aquí la noción de plantilla computacional tal y como la entiende Humphreys. Recordemos además que Humphreys sostiene que las unidades de análisis típicas de la filosofía de la ciencia no son adecuadas para dar cuenta de simulaciones computacionales. Es de esperar, entonces, que sean las plantillas computacionales las nuevas y más adecuadas unidades de análisis. Sin embargo lo que Humphreys tiene en mente es un poco más complejo.

Una vez desvinculadas las unidades básicas de análisis tradicionales de las simulaciones computacionales e instauradas en las ‘plantillas computacionales’ como las nuevas unidades básicas, se pasa a aceptar que la ciencia de la computación consiste en el desarrollo, exploración e implementación de *modelos computacionales* en computadoras físicas concretas. Estos modelos computacionales son identificados como una 6–upla que consiste en (Humphreys 2004, pp.102–103):

(Plantilla, Supuestos de Construcción, Conjunto de Corrección, Interpretación, Justificación Inicial, Representación)

donde

- Una plantilla computacional consiste en ecuaciones diferenciales junto con las cotas y las condiciones iniciales. Ecuaciones integrales, ecuaciones de diferencia, ecuaciones simultáneas, ecuaciones iterativas y cualquier otro aparato formal puede ser utilizado. Estos objetos sintácticos proveen la forma básica del modelo computacional.
- Los supuestos de construcción.
- El conjunto de corrección.
- Una interpretación.
- La justificación inicial de la plantilla.
- Una representación que puede ser un arreglo de datos, un dibujo gráfico, una solución general en forma de función, un número o cualquier otra forma. La representación juega un rol clave en las simulaciones computacionales.

Esta 6–upla constituye, en última instancia, la unidad de análisis de las simulaciones computacionales. Nótese que, a pesar de que cada elemento de la 6–upla puede

ser vinculado independientemente con una unidad de análisis filosófico, Humphreys entiende que es en el conjunto en donde reside aquello que hace a los modelos computacionales unidades de análisis apropiadas y únicas.

### 3.2. Hacia una (nueva) definición

Humphreys retoma sus viejas ideas presentadas en (1991) acerca de las simulaciones computacionales como mecanismos de cálculo al tiempo que hace uso de las críticas que recibió en su momento de parte de Hartmann. Recordemos que este último autor entendió que una simulación posee vínculos muy estrechos con un modelo dinámico subyacente, diseñado para imitar la evolución temporal de un sistema real.

Así, es la idea de un proceso dinámico temporal la que le interesa a Humphreys y que resulta ser la clave de todo el asunto. Para este, es el paso de representaciones abstractas de inferencias lógicas y matemáticas a procesos temporales de cálculo (más representación) lo que permite y explica que ninguna de las unidades básicas tradicionales pueda capturar lo que las simulaciones hacen. En este sentido, Humphreys indica que “las simulaciones descansan en un modelo computacional subyacente o son ellas mismas modelos de un sistema, y por lo tanto involucran o son ellas mismas representaciones” (Humphreys 2004, p.107).

Veamos esto con mayor detenimiento. Humphreys llama la parte temporal del proceso computacional el *núcleo de la simulación* [core simulation]. Este consiste en un proceso computacional temporal que produce soluciones a un modelo computacional —en el sentido dado en 3.1— que correctamente representa un sistema objeto (Humphreys 2004, p.110). De este modo queda claro que lo que se entiende por simulación computacional consiste en dos procesos interconectados: el proceso temporal de *computar* las soluciones del modelo computacional y el proceso de *representar* los resultados del cálculo en un modo comprensible.

Sobre esto último, y a diferencia de Hartmann y otros autores, Humphreys indica que es en los procesos de representación de los resultados de una simulación donde emergen peculiaridades metodológicas. Considérese como ejemplo tres modos diferentes de representar el resultado del núcleo de la simulación. El primer modo consiste en representar los resultados en una tabla numérica; en el segundo modo, el resultado es representado gráficamente mediante un dibujo estático; y el tercer modo es mediante la visualización de un gráfico en movimiento. Si los resultados del núcleo de la simulación son visualizados en un arreglo de datos numéricos, este arreglo de datos será una *representación estática*, así como lo será el caso del dibujo que presentamos como segundo ejemplo. Si los resultados son presentados como un gráfico en movimiento, entonces se presenta como una *representación dinámica*. Lo que se quiere remarcar aquí es que el modo en que se presentan los datos en la simulación será metodológicamente relevante. De este modo los resultados de una

simulación afectarán a la interpretación del modelo computacional y viceversa.

Dado que uno de los objetivos de la ciencia es alcanzar cierto entendimiento humano, el modo en que es representado el resultado de una simulación es de gran importancia epistemológica cuando se proveen mecanismos de visualización. Los valores de respuesta que genera un instrumento es incorporado como uno de los objetivos principales de la ciencia contemporánea. Desde luego, el crecimiento en el entendimiento humano no es siempre facilitado por representaciones visuales, aunque la forma que tome la representación puede afectar profundamente nuestro entendimiento del problema.

#### **4. Conclusión**

Preguntarnos si un tema cualquiera de la ciencia —sea un método científico, un instrumento, una práctica, etc.— es filosóficamente relevante, parecería que nos compromete con un ideal de lo que es hacer filosofía. ¿Qué hizo que los estudios sobre la experimentación fuera filosóficamente menos relevante para los empiristas lógicos que para sus sucesores? Una respuesta posible es que se comprendió que la práctica experimental aporta elementos filosóficos relevantes que permiten comprender mejor una filosofía (positivista) de las teorías —previo a que la experimentación de laboratorio se convirtiera en un tema de interés filosófico en sí mismo. Romper con la tradición, mirar el mismo objeto de estudio pero desde un ángulo distinto, es lo que posibilita nuevas formas de hacer filosofía. Así pues, el prejuicio filosófico de creer que una filosofía de la ciencia más familiar permitirá comprender mejor el uso de las simulaciones computacionales en la ciencia moderna carece de fundamento. Las simulaciones computacionales, en ese sentido, se han arraigado tan profundamente en la práctica científica que ignorarlas mediante el ninguneo filosófico desacredita al propio filósofo. Esto no significa, de ningún modo, que se “requiera reescribir la filosofía de la ciencia”, pero sí que un entendimiento más profundo de lo que es la práctica científica actual requiere un estudio filosófico adecuado de las simulaciones computacionales. Creo, en ese sentido, que entender su naturaleza es el primer paso obligatorio.

#### **Agradecimientos**

Este trabajo fue realizado durante mi estadía en el Centro de Investigaciones de la Facultad de Filosofía y Humanidades, Universidad Nacional de Córdoba (Argentina). Mis agradecimientos van a toda la comunidad filosófica de esta universidad. Asimismo, este trabajo fue realizado con el apoyo de una beca postdoctoral de CONICET.

## Referencias

- Ajelli, M.; Gonçalves, B.; Balcan, D.; Colizza, V.; Hu, H.; Ramasco, J. J.; Merler, S.; Vespignani, A. 2010. Comparing Large-Scale Computational Approaches to Epidemic Modeling: Agent-Based versus Structured Metapopulation Models. *BMC Infectious Diseases* **10**(190): 1–13.
- Bailer–Jones, D. 2009. *Scientific Models in Philosophy of Science*. University of Pittsburgh Press.
- Frigg, R.; Hartmann, S.; Imbert, C. 2009. Editorial **169**(3): 425.
- Galison, P. 1996. Computer Simulation and the Trading Zone. In: P. Galison; D. Stump (eds.) *Disunity of Science: Boundaries, Contexts, and Power*. Stanford University Press.
- Guala, F. 2002. Models, Simulations, and Experiments. In: L. Magnani; N. J. Nersessian (eds.) *Model-Based Reasoning: Science, Technology, Values*, pp.59–74. Springer.
- Hartmann, S. 1996. The World as a Process: Simulations in the Natural and Social Sciences. In: R. Hegselmann; U. Mueller; K. G. Troitzsch (eds.) *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View*, pp.77–100. Springer.
- Humphreys, P. W. 1990. Computer simulations. *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association* **2**: 497–506.
- . 2004. *Extending Ourselves: Computational Science, Empiricism, and Scientific Method*. Oxford University Press.
- Knoll, D.; Morel, J.; Margolin, L.; Shashkov, M. 2005. Physically Motivated Discretization Methods. *Los Alamos Science* **29**: 188–212.
- Morgan, M. S. 2003. Experiments without material intervention. In: H. Radder (ed.) *The Philosophy of Scientific Experimentation*, pp.216–235. University of Pittsburgh Press.
- . 2005. Experiments versus models: New phenomena, inference and surprise. *Journal of Economic Methodology*, **12**(2): 317–329.
- Morrison, M. 2009. Models, Measurement and Computer Simulation: The Changing Face of Experimentation. *Philosophical Studies* **143**(1): 33–57.
- Parker, W. S. 2009. Does matter really matter? Computer simulations, experiments, and materiality. *Synthese* **169**(3): 483–496.
- Radder, H. (ed.) 2003. *The Philosophy of Scientific Experimentation*. University of Pittsburgh Press.
- Rohrlich, F. 1990. Computer Simulation in the Physical Sciences. *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association 1990* **1990**(2): 507–518.
- Winsberg, E. 1999. Simulation and the Philosophy of Science: Computationally Intensive Studies of Complex Physical Systems. *PhD thesis*. Indiana University.
- . 2001. Simulations, Models, and Theories: Complex Physical Systems and Their Representations. *Philosophy of Science* **68**(S1): S442.

JUAN M. DURÁN  
HLRS – University of Stuttgart  
ALEMANIA  
juanduran@gmail.com

## Notas

<sup>1</sup> Entiendo un *modelo simulacional* como el modelo apto para ser implementado y ejecutado en la computadora.

<sup>2</sup> Debemos notar que esta postura representa un cambio significativo de su artículo del año 1990, donde sostiene que “una simulación computacional es cualquier método implementado computacionalmente para explorar las propiedades de modelos matemáticos donde los métodos analíticos no están disponibles” (Humphreys 1990). En este sentido, este artículo es muy cercano a las ideas de Hartmann discutidas anteriormente.

<sup>3</sup> Nótese que esto no siempre es una cuestión cuantitativa. Muchas veces es necesario un cambio cualitativo en la forma en la que se presenta la información. Por ejemplo, presentar un gráfico que dibuje la curva de una función siempre será más comprensible que contar con todos los valores de la función.

<sup>4</sup> El concepto de *tratabilidad computacional* se debe entender como la plantilla computacional que puede ser resuelta en tiempo finito por una computadora digital.

<sup>5</sup> Aquí es posible incluir leyes.