

Steven Stern

STS.035

Artículo de respuesta, semana nº 3

En los orígenes de la informática, los términos “aleatorio” y “pseudoaleatorio” eran intercambiables. Los informáticos de la época estaban de acuerdo en que sus funciones no eran aleatorias, ya que conocían un límite superior a partir del cual, cuando los números aleatorios se generaran, comenzarían a repetirse. En realidad, este límite superior era bastante pequeño. Si una función pseudoaleatoria que se repetía cada 10^{10} dígitos, se fuese a utilizar en un esquema criptográfico actual, este esquema podría romperse fácilmente en un ordenador personal. Quizás no sea razonable hacer una comparación con los ordenadores de hoy en día, puesto que posiblemente en el año 2050, los esquemas criptográficos actuales resulten algo trivial. No obstante, a pesar de que cumplía con sus propósitos por aquel entonces, creo que su definición de aleatoriedad era muy superficial.

Aceptaría una secuencia generada por ordenador como aleatoria. Los ordenadores utilizan muchos eventos físicos para generar números aleatorios. Por ejemplo, algunos ordenadores utilizan el tiempo entre pulsaciones, las variaciones de ruido blanco detectadas por el micrófono del mismo ordenador o la velocidad a la que gira el disco duro. Esto, por supuesto, se puede mejorar. Si la aleatoriedad real de los números generados al azar por el ordenador es de mucha importancia, al igual que cuando se genera la clave confidencial para el certificado de raíz VeriSign, se pueden utilizar entradas más extremas. La aleatoriedad pura, desde el punto de vista físico, se puede lograr simplemente incorporando un contador Geiger en el ordenador.

Mientras leía sobre las dudas que algunas personas tenían con el método Monte Carlo, me vino a la mente un suceso en concreto. Si usted pide a alguien que determine si una moneda no es falsa, resulta instintivo tirar la moneda miles de veces y grabar los resultados. Nadie se opondría a este método para calcular este resultado, a pesar de que es posible (aunque bastante improbable) que una moneda mala dé positivo en la prueba. No obstante, si el ordenador está dispuesto a realizar el equivalente de esta prueba, la gente se opone. La gente cree, por el contrario, que el ordenador debería estudiar la densidad y forma exacta de la moneda para determinar si se lanza bien o no. No veo ninguna diferencia entre este ejemplo y el modelado que los ordenadores estaban realizando con las reacciones nucleares.

La principal ventaja de una simulación por ordenador, comparado con un experimento de laboratorio, es que la simulación computacional es un entorno controlado. Un ordenador puede modelar una reacción en cadena nuclear que, si sucediera en realidad, allanaría una isla del Pacífico completa, sin causar daño real en absoluto. Creo que la simulación se aleja un paso de la realidad, pero estudiando la realidad adecuadamente, es posible reducir al máximo este alejamiento. Es posible que este argumento resulte un poco metafísico, pero cuando alguien lanza un balón, éste no puede asegurar si el movimiento del balón es “realidad” o es una simulación realmente buena. Quizás todos seamos parte de una simulación, y cuando lanzamos un balón, un ordenador tremendamente rápido calcula la trayectoria de éste, teniendo en cuenta la fuerza de lanzamiento, el giro del mismo y la resistencia del aire que rodea al balón.

Creo que la simulación computacional debería considerarse una ciencia. De hecho, existen 2 campos distintos de simulación por ordenador que deberían contar como ciencia: el empleo de la simulación y el perfeccionamiento de éstas. Cuando la simulación es bastante cercana a la realidad, no veo ninguna diferencia entre el trabajo realizado en una simulación y el llevado a cabo en la realidad. Creo además, que el trabajo realizado para perfeccionar simulaciones tendría que ser digno de un doctorado.

Se considera algo significativo que un ingeniero mecánico desarrolle un método básicamente superior para construir rascacielos, que reduzca considerablemente el coste de construcción de uno. Sin embargo, ese ingeniero tan sólo desarrolló un modo mejor de hacer lo que ya estaba hecho: desarrollar una simulación casi perfecta.