Why Monte Carlo Simulations are Inferences and not Experiments

Claus Beisbart
Institute for Philosophy and Political Science
TU Dortmund
http://www.claus-beisbart.de
and
John D. Norton
Department of History and Philosophy of Science
Center for Philosophy of Science
University of Pittsburgh
http://www.pitt.edu/~jdnorton

Version June 11, 2012
currently under review

Monte Carlo Simulations arrive at their results by introducing randomness, sometimes derived from a physical randomizing device. Nonetheless, we argue, they open no new epistemic channels beyond that already employed by traditional simulations: the inference by ordinary argumentation of conclusions from assumptions built into the simulations. We show that Monte Carlo simulations cannot produce knowledge other than by inference; and that they resemble other computer simulations in the manner in which they derive their conclusions. Simple examples of Monte Carlo simulations are analyzed to identify the underlying inferences.

1. Introduction

Monte Carlo simulations exploit randomness to arrive at their results. Figuratively speaking, the outcomes of coin tosses repeatedly direct the course of the simulation. These Monte Carlo simulations comprise a case of special interest in the epistemology of simulations, that is, in the study of the source of the knowledge supplied by simulations. For they would seem, at first look, to be incompatible with the epistemology of simulation we hold. Following Beisbart (forthcoming) and Stöckler (2000), we hold that simulations are merely arguments, albeit quite elaborate ones, and their results are recovered fully by inferences from the assumptions presumed.

Tossing coins, rolling dice, spinning roulette wheels, drawing entries from tables of random numbers or taking the outputs from computational pseudo-randomizers all seem quite remote from the deliberate inferential steps of an argument. Indeed, this reliance on randomness makes these Monte Carlo simulations look more like real experiments. For the outcome of a real experiment is learned only by doing the experiment. We cannot know in advance whether a test group will fare better under the treatment, which particles will be produced at novel energies in an accelerator, or which micro-organisms will be found to be thriving when we look through our microscope. We need to do the experiments to discover the results. They are not inferred.

Correspondingly, the whole point of randomizing seems to be the injection of an element into the simulation that cannot be known in advance. Just as we cannot know antecedently into which direction the particle in the accelerator will be scattered, we cannot know whether the randomizer coin will fall heads or tails. The Monte Carlo roulette wheel must be spun before we can know the random number it will inject at that step into the simulation. It is to be discovered. It is not to be inferred.

Considerations such as these suggest the view that Monte Carlo simulations are conceived more properly as experiments. Dietrich (1996, pp. 344-47) argues that Monte Carlo simulations share the same basic structure as controlled experiments and reports the

view among geneticists who use them that they were "thought to be much the same as ordinary experiments" (ibid., pp. 346-47).[1]

Even if Monte Carlo simulations do not form experiments, they may challenge our view that simulations are merely arguments. Humphreys (1994, pp. 112-113) allows that Monte Carlo simulations are not experiments properly speaking. However, he does not allow the natural alternative that they are merely a numerical technique of approximation, such as truncation of an infinite series of addends (p. 113). Rather, he denies that the simulations form a method of "abstract inference" (p. 112) because they are more experiment-like and generate representations of sample trajectories of concrete particles. Humphreys' conclusion is that Monte Carlo simulations form a new scientific method, which occupies the middle ground between experiment and numerical methods and which he dubs "numerical experimentation" (p. 112). Humphreys' analysis is not entirely clear on whether this middle ground employs novel epistemic modes of access to the world, distinct from those of the two extremes of experiment and the inferences of numerical methods.

Our purpose in this paper is to reaffirm that, as far as their epistemic access to the world is concerned, Monte Carlo simulations are merely elaborate arguments or inferences. In this regard they do not differ from other simulations. Our case is twofold. First, indirectly, Monte Carlo simulations could not be anything else. Simulations, Monte Carlo or otherwise, do not gain knowledge of parts of the world by interacting with them, as do ordinary experiments. They are closed systems that can only return knowledge of the world external to them in so far as that knowledge is introduced in the presumptions used to set up the simulation. They can then exploit that knowledge to yield their results by an inferentially reliable procedure, that is, by one that preserves truth or the probability of truth. Second, directly, an inspection of Monte Carlo simulations shows them merely to be a sequence of inferences no different from an ordinary derivation, with the addition of some complications. These are: there are very many more individual inferences than in derivations normally carried out by humans with pencil and paper; the

[1] Morrison (2009) maintains that many computer simulations, Monte Carlo or not, are experiments because they draw on models in the same way as experiments do.

choice of which inferences to make is directed by a randomizer; and there are meta level arguments that the results are those sought in spite of the random elements and approximations used.

Our thesis is a narrow one. We are concerned solely with the epistemological problem of how Monte Carlo simulations can give us knowledge of the world.[2]. In other ways, Monte Carlo simulations are like experiments that discover novel results. We might say that the random numbers they use are introduced as many, little novel discoveries. The simulations proceed at a high level of opacity so that their results cannot easily be anticipated, much as we cannot anticipate the outcomes of truly novel experiments.[3] We will argue, however, that these sorts of similarities are superficial. They may make Monte Carlo simulations look like experiments in some aspects. But they do not and cannot make them function like real experiments epistemically. It is this epistemic aspect that is of concern in this paper.

In Section 2, we review briefly how Monte Carlo simulations work. In Section 3, we begin our main argument by reviewing the two modes of epistemic access open to experiments and simulations: discovery and inference. Sections 4 and 5 make the indirect and direct cases for our thesis. In Section 6, we illustrate our thesis by reconstructing some instances of Monte Carol simulations explicitly as arguments. In Section 7, we respond to objections.

2. Monte Carlo Simulations

What are Monte Carlo simulations and how do they work? In a broader sense, Monte Carlo simulation is a method that uses random numbers to carry out a calculation.[4]

---

[2] There are many other fascinating aspects to Monte Carlo simulations. See e.g. Humphreys (1994) and Galison (1997, Ch. 8). See also Anderson (1987), Eckhardt (1987), the papers in the first part of Gubernatis (2003) and Hitchcock (2003) for the history of Monte Carlo simulations.

[3] The epistemic opacity of computer simulations is stressed by Humphreys, 2004, pp. 148–50.

[4] A more appropriate expression is “Monte Carlo method” or “Monte Carlo technique”. See e.g. James, 1980, p. 1147 for a definition. Consult Hammersley & Handscomb (1967), Halton (1970) and James (1980) for reviews about this method.

Monte Carlo integration is the prime example of this technique.[5] In a narrower sense, Monte Carlo simulations trace physical processes. Simulations of both kinds are arguments, or so we will argue.

Suppose our task is to evaluate the expectation value of a random variable *f*. Assume that we have a uniform probability distribution over the interval [0,1] and that our random variable returns $\sqrt{1-x^2}$ for every x from [0,1]. We can estimate the expectation value, $E(f)$, from the average over independent realizations of the probability model. We use N random numbers x_i following a uniform distribution over [0,1], apply f and take the average. Our estimate is:

$$E_N(f) \equiv \frac{1}{N}\sum_{i=1}^{N} f(x_i)\,.$$

This is the most basic Monte Carlo method.

Random variables are extremely useful in the natural sciences. A pollen particle suspended in certain liquids undergoes a zigzag motion that looks random. The motion is called Brownian, and so is the particle. Brownian motion is described by random variables. For each time t, the position of the particle is the value of a random variable $X(t)$. A model can relate the probability distribution over $X(t)$ to the distribution at an earlier time t'. In a simple discrete random walk model, time is discrete, the motion of the particle is confined to the nodes of a grid, and at every instance of time $t = 1, 2, \ldots$, the particle jumps to one of the neighboring nodes, following some probability distribution.[6] We can use this model to make predictions, for example, about the expected position of the particle at later times or about the probability that the particle is within a certain region of space. We start from the initial position of the particle, $X(0)$, and use a sequence of random numbers to determine positions $X(1)$, $X(2)$ and so on successively. This produces one sample path/sample trajectory, that is, one possible course the particle could take. To estimate the expected position or the probability of the particle being in a certain region of space, we average over a large number of sample trajectories. These

---

[5] See e.g. James (1980, Sec. 2) for an introduction to Monte Carlo integration.

[6] See Lemons (2002) for a readable introduction to the related physics.

calculations undertaken for pollen grains are simulations in the narrow sense because they trace a physical process. They are computer simulations in the sense defined by Humphreys (2004, p. 110) because they evaluate a model of a physical process in the world.

Monte Carlo methods can also be applied to problems that do not involve randomness. To see this, note that the expectation value $E(f)$ can be written as

$$E(f) = \int_0^1 dx \sqrt{1-x^2}\, p(x) = \int_0^1 dx \sqrt{1-x^2} \; .$$

The second equality holds because we have assumed a flat probability density over [0,1]. Consequently, our Monte Carlo method has estimated the value of the integral

$$\int_0^1 dx \sqrt{1-x^2} \; , \tag{1}$$

which is known to equal π/4. More generally, the value of an integral

$$\int_a^b dx\, g(x)$$

for an integrable function g and real numbers $a < b$ can be approximated using random numbers. The integral can be rewritten as

$$(b-a) \times \int_a^b dx\, g(x) \times \frac{1}{b-a} \, .$$

The last factor, $1/(b-a)$, is the probability density for a probability model with a uniform distribution over $[a,b]$. To obtain the value of the integral, we use N independent random numbers x_i following a uniform distribution over $[a,b]$ and take the average over $g(x_i)$:

$$\frac{(b-a)}{N} \sum_{i=1}^{N} g(x_i) \, .$$

This method is called Monte Carlo integration. It differs from other methods of approximating the value of an integral numerically, such as the trapezoidal rule,[7] because

[7] Consult Press et al. (1992, Ch. 4) for details.

it uses random points instead of a regular grid to sample the interval. Monte Carlo integration can be generalized and refined in many ways; it is particularly successful for higher-dimensional integrals. It has been shown that every Monte Carlo method reduces to Monte Carlo integration (James, 1980, p. 1148).

In applying a Monte Carlo method, we use random numbers that follow a certain probability distribution. To obtain such numbers, we can draw on the outcomes of a physical random process.[8] Alternatively, we can use a pseudo-random number generator. This is a deterministic algorithm that returns a series of numbers following a certain probability distribution.[9] In practice, it does not matter how we obtain the random numbers as long as they follow the intended probability model. Whether or not they do so is often ascertained using statistical tests (see e.g. James, 1980, pp. 1170–72). Often, the tests check whether the frequencies of the numbers in certain intervals match those predicted by the probability model. If we have 100 random numbers with a uniform distribution over [0,1], approximately 50 random numbers should be in [0, .5]. The random numbers generated by some pseudo-random generators have even been characterized analytically (ibid., p. 1172).[10]

Since a Monte Carlo integration uses random numbers, its results are random. If we run the method twice using different random numbers, the results will differ. In either case, the result will not coincide with the exact value of the integral. The difference between the outcome of the method and the exact value of $\pi/4$ is called error, and, in the

---

[8] See Tocher (1975, Ch. 5) for the generation of random numbers via physical random processes.

[9] See e.g. Hammersley & Handscomb (1967, Ch. 3), Tocher (1975, Ch. 6), James (1980, pp. 1167–1165) and Knuth (2000, Ch. 3) for the generation of pseudo-random numbers.

[10] Here we define random numbers probabilistically. That is, a sequence of numbers is random just if they fit the requisite probabilistic model. It is a matter of considerable discussion just what it is for random numbers to fit some probabilistic model. These issues go beyond the scope of the present paper and will not be addressed here. It is sufficient for our purposes that, as a practical matter in Monte Carlo simulations, a large sequence of numbers would not be accepted as conforming to a uniform distribution over [0,1] if all the numbers are clustered close to some particular value, such as 0.25, even though there is some very small probability that just such a clustering may happen. This sequence and other pathological sequences like it would be rejected in favor of ones whose members are more uniformly distributed over [0,1].

case of Monte Carlo simulations, the error is statistical. In practice, the error is not a problem since the method converges probabilistically (James, 1980, p. 1151). Suppose that we can only tolerate an error smaller than $\varepsilon > 0$ in the evaluation of the integral, and that we want the outcome to be within this error bound with a probability of p, say 99%. Probabilistic convergence assures us that, if we use more than some specific number of random points, we can keep our error within the tolerable bounds with a probability higher than p. What that number is depends on the details of our problem.

As an illustration, we have run Monte Carlo simulations to approximate the value of $\pi/4$ (Eq. 1). For the pseudo-random number generator, we choose the linear congruential generator

$$X_{i+1} = (1664525\, X_i + 1013904223) \pmod{2^{32}} . \tag{2}$$

The resulting set of random numbers is almost perfectly uniformly distributed over the interval $[0, 2^{32})$.[11] Hence, the numbers $r_i = X_i/2^{32}$ follow a uniform distribution on the unit interval. The pseudo-random number generator has to be initialized using a number in $[0, 2^{32})$. We have used different initial values and different values of N (the number of random numbers) to obtain Monte Carlo estimates of the value of the integral:

$$\pi / 4 = \int_0^1 \sqrt{1-x^2}\, dx \approx \frac{1}{N} \sum_{i=1}^{N} \sqrt{1-r_i^2} . \tag{3}$$

The results are shown in Table 1. They illustrate that the approximations to the integral tend to approach one another and to converge to the true value of the integral ($\pi/4 \cong .785398$), as N increases.

Initial value X_0	N = 100	N = 1,000	N = 10,000	value of $\pi/4$
0	.816315	.795922	.785052	.785398
1	.773266	.774889	.787492	.785398

[11] See Press et al. (1992, p. 284).

Caption of Table 1: Results of several Monte Carlo integrations of the integral in (Eq. 1), which equals $\pi/4$. We obtain random numbers following (Eq. 2) for various combinations of the initial value X_0 and the number N of random numbers used.

Monte Carlo simulations use random numbers to evaluate mathematical expressions. These expressions may have a probabilistic meaning or not. The method converges to the true result in a probabilistic manner as the number of random points is increased. The last few decades have seen many further developments of Monte Carlo methods, which we need not pursue in what follows, since they do not alter any matters of basic principle.

3. What Powers our Epistemic Activities

How can simulations give us knowledge of the world? We distinguish two modes as the sole options.

3.1 Discovery

Discovery—here always understood as empirical discovery—goes directly to the world. We observe the world to learn about it. In an experiment, we do this in a particular way. We contrive some artificial arrangement in nature and watch what happens.[12] In his celebrated experiments, Millikan (1913) suspended electrically charged oil drops in an electric field to determine the charge of a single electron. Cavendish (1798) used a torsion balance to determine the gravitational force of attraction between lead masses. While Millikan and Cavendish contrived the arrangements, the experimental outcomes depend essentially on what then manifests in the apparatus.

A complication is that most experiments are not purely discovery. What we call an experimental result is commonly obtained with the help of inferences. To declare an experimental result for the universal natural constant e, the "elementary electric charge," Millikan had to generalize the behavior of the few electrons measured in his apparatus to all electrons. Cavendish required some delicate inferences to calibrate his torsion balance.

[12] See Heidelberger (2005) and Radder (2009) for the notion of scientific experiment.

We will not pursue these inferences here since they merely mold and generalize what powers the experiment epistemically: the novel experience provided by the behavior of the apparatus. They are not essential to an experiment.

3.2 Inference

The second mode requires no contact with the world and thus is not self-sufficient. It transforms knowledge of the world already gained. If this transformation is to be reliable, it must be truth preserving, and hence a deductive inference; or preserving the probability of truth, and hence an inductive inference of great strength. Investigations in this mode are powered epistemically by the knowledge of the world introduced at the outset. They can nevertheless produce new knowledge simply because the results inferred were not known prior to the investigations.

Thought experiments, Norton (2004) maintains, operate this way. They are merely picturesque arguments that make inferences from presumptions implicit in the description of the thought experiment scenario. Since human imaginative powers do play a role, it is not immediately apparent that prosaic inference is all that is employed. It is fashionable and enticing to imagine that thought experiments are able to tap into some mysterious new epistemic channel. Norton requires fairly elaborate argumentation to establish that they do not do this.

As Beisbart (forthcoming) has argued, the case of computer simulations is more straightforward than that of thought experiments. How simulations work is transparent in the sense that they consist of a large number of simple steps, typically programmed into a computer. A sufficiently patient monitor could trace the entire simulation step by step from start to finish. No peculiarly human powers of imagination enter. A simulation does not differ in kind from a derivation from some suitably collected set of assumptions. Rather it differs in degree. It tends to use simpler and weaker steps, but compensates for their weakness by employing very many of them.

An example illustrates this. The angular displacement θ of a simple pendulum of length L at time t is governed by the differential equation

$$d^2\theta/dt^2 = -(g/L)\sin\theta,$$

where g is the gravitational acceleration at the surface of the earth. One can determine the motion for small displacements by approximating $\sin \theta \approx \theta$ and noting that the equation is reduced to

$$d^2\theta/dt^2 = -(g/L)\ \theta\ .$$

It is solved, for example, by

$$\theta(t) = \theta_{max} \cos((g/L)^{1/2}t)$$

for θ_{max} the maximum displacement. This argument is recognized as a classic derivation in mechanics. Meta-analysis restricts the domain in which the result can be used with acceptable error.

Alternatively, one can perform a step-wise integration. In the simple but inefficient Euler method, one starts from momentary rest at $t = 0$ at the maximum displacement $\theta(0) = \theta_{max}$, where the initial angular velocity is zero and the angular acceleration is $-(g/L) \sin \theta_{max}$. One can then approximate the motion over the next small time interval Δt by assuming that the acceleration is unchanged. One recovers

$$\theta(\Delta t) = \theta_{max} - (1/2)\,(g/L) \sin \theta_{max}\,(\Delta t)^2\ .$$

The process is iterated to recover the angular position and speed for later times $2\Delta t$, $3\Delta t$, $4\Delta t$, $5\Delta t$, … If this step-wise integration is performed on a computer, we have a simple computer simulation. Meta-analysis is also needed to assure us that the errors introduced by the approximations are acceptably small. For the inefficient Euler method, a standard theorem (Young and Gregory, 1972, p. 447) assures us that, for generically well-behaved systems, the error in integrating over some fixed time increases roughly linearly with the step size Δt. As a practical convenience, we can estimate the size of the error by halving the step size and checking how much the final result changes. That change is, very roughly, half the total error of the original estimate.

Clearly there are great pragmatic differences between the two procedures of derivation and step-wise integration. The first is easy for a human to carry out with pencil and paper. The second requires assistance from some computing device. However they do not differ in the epistemic element of concern to us here: they both proceed as inferences from the presumption of the original differential equation. The derivation

proceeded from the approximation sin $\theta \approx \theta$. It converted the equation into one whose solution is in the standard repertoire of introductory mathematics texts. The simulation employed a coarser and less powerful approximation of the constancy of acceleration over a short time period. It enabled us to look forward by a small time interval after which the approximation needed to be repeated. Meta-analysis—sometimes trivial, sometimes not—is needed in both cases to determine the accuracy of the results.

3.3 The Only Possibilities

These two modes—discovery and inference—exhaust those suitable for the epistemology of simulation. To see that they are exhaustive, note that all suitable modes must belong to one of two classes: those that depend essentially on contact with the world; and those that do not. In the first class, we require that contact to be either passive or active. Passive contact is observation. Active contact, or at least active contact that improves epistemically on passive contact, is experimentation. So the first class is what we have called discovery. In the second class, we must proceed without any new contact of the world. That is, as noted above, we must transform what we already know by a procedure that preserves truth or its probability. That is, this second class is inference.[13]

That these two modes exhaust the relevant possibilities will provide a premise for our indirect argument below.

4. Monte Carlo Simulations are Arguments: The Indirect Case

Our claim is that, epistemically, Monte Carlo simulations are arguments. We assert this even though they are in many ways unlike other, more familiar arguments. They are hugely complex so that it is impractical for a human to follow through their steps, whereas arguments are normally carried out by humans. They are distinctive in that their steps are governed by random or pseudo-random processes, whereas ordinary

---

[13] One might imagine other epistemic modes. In the context of thought experiments, Brown (2004) has proposed that we can learn of the world through a Platonic perception of the ideal forms of the laws of nature. Because of the transparency of the individual steps of simulations, we believe that there is no role for these exotic modes.

argumentation does not include randomly chosen inferences. However these differences are unimportant for our concern of what powers Monte Carlo simulations epistemically. They are arguments. They deliver their results by transforming presumptions built into their set ups in a way that preserves truth (deductive inference) or a way that preserves its probability (strong inductive inference).

We make our case indirectly and directly. Indirectly, we argue in this section that they could be nothing else. Directly, in the next section, we review the operation of Monte Carlo simulations and find that they are merely many inferences assembled.

Indirectly, we arrive at the conclusion that Monte Carlo simulations are arguments by elimination, that is, by a disjunctive syllogism. The first premise is that Monte Carlo simulations are powered epistemically either by what we have called "discovery" or by what we have called "inference" in the last section, whose principal burden was to establish that premise. The second premise is that Monte Carlo simulations are not powered epistemically by discovery. Hence, by disjunctive syllogism, they are powered by inference.

The second premise requires a little discussion. For Monte Carlo simulations can employ real physical systems as randomizers. This is a type of contact with the external world. However it is not the type that is required for discovery. A method only includes discovery if hitherto unknown properties characteristic of a particular physical system are recorded. It is the incorporation of these previously unknown properties that furnishes discovery its epistemic power. In Monte Carlo simulations, by contrast, physical randomizers are used in a manner carefully designed to eradicate any influence of their peculiar properties. Monte Carlo simulations can use any of many possible physical randomizers without the change of randomizer making any difference to the outcome. The randomizer can vary from truly random processes provided by quantum systems, such as the random clicks of a Geiger counter near a radioactive source; to the effectively random processes supplied to thermal noise in an electronic circuit; to properly executed casino randomizers, such as roulette wheels. All that matters is that the devices provide numbers with the requisite distribution, where that distribution has been specified ahead of any physical contact with the randomizer. Thus the randomizers are designed to

suppress completely exactly that aspect of the contact with the world that powers discovery epistemically.

Finally, a physical randomizer can be replaced entirely by the computer code of a pseudo-randomizer without any harmful effect or even any discernible effect on the Monte Carlo simulation. That replacement displays most clearly that a Monte Carlo simulation does not require contact with the randomizer as an external physical object. Hence the simulation cannot implement "discovery" for that mode requires such contact.

5 Monte Carlo Simulations are Arguments: the Direct Case

In so far as Monte Carlo simulations carry out calculations that trace the dynamics of a system, direct examination shows them to be arguments or inferences, as sketched in Section 3.2 above. What remains to be shown here is that the introduction of randomization does not change this assessment.

We can break the problem down into two sub-problems according to whether the Monte Carlo simulation is used merely to compute some mathematical quantity or whether it is used to imitate a physical system. In both, following the discussion of Section 4, we will not distinguish physical randomizers from pseudo-randomizers since the difference between them is immaterial to the functioning of Monte Carlo simulations.

5.1 Computing a Mathematical Quantity

We have seen in Section 2 that Monte Carlo simulations can be used merely to compute a mathematical quantity. The simulation employs mathematical structures only: sets, functions and the like. The simulation consists of computations or mathematical derivations—inferences—that employ these structures. Randomization adds the complication that the final outcome must be interpreted probabilistically: the result is likely not exactly correct, but very likely close to the correct result. This assurance is recovered from a meta-level inference about the derivation. The entirety of the simulation, if a pseudo-random number generator is used, simply is a derivation in mathematics.

This is illustrated in the Monte Carlo computation of π/4 in Section 2. The basic structure is the set $\{r_i\}$, generated by the formula in (Eq. 2), and the summation over it of (Eq. 3). The resulting values of Table 1 are generated by direct computation. A meta level argument assures us that the numbers in the set $\{r_i\}$ are uniformly distributed over the unit interval for most of the initial values X_0 that we may use to seed the generator of (Eq. 2) and that, as a result, the summation of (Eq. 3) will be very close to π/4.

That the introduction of randomization is benign can be illustrated with a simple but computationally impractical example in which we introduce randomization in two steps. Consider the problem of summing all the integers from 1 to 1,000,000. We could find it directly by summing 1+2+3+…. Or we could use a randomizer to select a different order in which to sum, say 56,723 + 2 + 899,765 + …. In this second case, the intrusion of randomness does not alter the outcome; addition is associative and gives the same result whatever the order. Using a randomizer to select the order corresponds, figuratively, to putting all the numbers in an urn, drawing them randomly without replacement and adding them in the order they are drawn. A more efficient process requires us to draw a small random sample of the entire set, sum them and scale up the sum of the sample to estimate the sum of the totality. It turns out that a very small sample, drawn with replacement, is sufficient to get a good estimate. Merely drawing 10,000 numbers, which is just 1% of the totality, will yield a result that is likely within 1.15% of the correct answer.[14] This last procedure corresponds to a Monte Carlo simulation of the integral $\int_{x=0.5}^{1,000,000.5} dx\, x$.

---

[14] The N random variables X_i, $i = 1, \ldots, N$ are uniformly and independently distributed over the integers $\{1, \ldots, M = 1{,}000{,}000\}$. The estimate of the sum $\sum_{i=1}^{M} i = M(M+1)/2$ is $\frac{M}{N}\sum_{i=1}^{N} X_i$, which has a mean $M(M+1)/2$ and a variance $\frac{M^2(M^2-1)}{12N}$. Two standard deviations, expressed as a fraction of the mean, is approximately $2/\sqrt{3N}$, which supplies the 1.15% error of the text when N=10,000.

5.2 Simulating a Physical System

Consider now the case of Monte Carlo simulations of real physical systems. This case can be dealt with quickly, for it reduces to the case of computing a mathematical quantity of Section 5.1. The preparation for a Monte Carlo simulation of some real physical system is the developing of a mathematical description of the system. The Monte Carlo simulation itself is merely the derivation within this mathematical description of some quantity, exactly akin to what we have seen in Section 5.1.

For example, we may model Brownian motion as a discrete random walk as described in Section 2. Then each possible trajectory of the particle over 1,000 time units is represented by adding 1,000 numbers, each number representing a possible step. The related addition is a purely mathematical operation. The statistical parameters of the motion, such as its mean, its variance, the mean time for return to the origin, and so on arise from sums over all possible trajectories. Monte Carlo simulations estimate the related sums by summing only over a randomly selected subclass of possible trajectories. This is analogous to summing over a random selection of numbers to obtain the value of a sum or an integral.[15]

6. What is the inference of a Monte Carlo simulation?

There is direct and indirect evidence then to the effect that a Monte Carlo simulation is an argument. But what exactly is this argument? We have argued that it has two parts: the inferences of the computation itself, typically carried out by an electronic computer; and the arguments of a meta-analysis that assure us that the intended results have been attained. The purpose of this section is to display the arguments in our main examples of Monte Carlo simulations.

We return first to the Monte Carlo integration that estimates the numerical value of $\pi/4$. Consider the integration with initial value $X_0 = 0$ and $N = 10{,}000$ random numbers (see Table 1). The primary argument consists in the computation of the members of the set of random numbers $\{r_i\}$ by the linear congruential generator (4) and its substitution

[15] This argument illustrates why every Monte Carlo method performs a Monte Carlo integration.

into the summation formula $\frac{1}{n}\sum_{i=1}^{N}\sqrt{1-r_i^2}$ to arrive at the result .785052. This argument is merely a large set of simple arithmetic inferences performed by an electronic computer. It would be impractical and unilluminating to display them here. The argument of the meta-analysis is more delicate and more interesting. It assures us that the result .785052 has the significance intended. That argument is reconstructed in outline as

P1. The 10,000 random numbers $\{r_i\}$ are uniformly distributed over [0,1].

P2. Substituting these random numbers into the Monte Carlo estimator $\frac{1}{n}\sum_{i=1}^{n}\sqrt{1-r_i^2}$ gives the value .785052 .

P3. If the set $\{r_i\}$ is large and uniformly distributed, the Monte Carlo estimator $\frac{1}{n}\sum_{i=1}^{n}\sqrt{1-r_i^2}$ of the integral $\pi/4=\int_0^1\sqrt{1-x^2}dx$ very probably approximately equals its true value π/4.

C. π/4 very probably is approximately .785052.

The conclusion C states the result of the Monte Carlo simulation. It is qualitative because the error tolerance and the probability with which the error tolerance is met are not specified numerically. If the error tolerance and the respective probability are known, C could be rendered more precise.

The conclusion C is probabilistic in character. While we cannot enter here into the thorny problem of explicating probability, it is sufficient for our purposes that an objective notion is used and that this objective notion assures us of a close association between the relative frequencies of random numbers and the corresponding probabilities for large sets of random numbers. That Monte Carlo integration converges probabilistically has been proven quite generally, at least if the function to be integrated is well-behaved (see James, 1980, p. 1150). This general result does not entail how quickly a particular Monte Carlo integration converges. For some applications of the method, it is possible to obtain exact bounds on the error and the respective probabilities. For other applications, no such bounds are known, but they may be estimated from simulations.

The Monte Carlo simulation that estimates $\pi/4$ is merely a mathematical computation. We argued in Section 5.2 above that Monte Carlo simulations of physical systems reduce to analogous mathematical computations, using Brownian motion as an illustration. But since the results of such Monte Carlo simulations have empirical meaning, we can reconstruct the arguments associated with such simulations in a more illuminating way. We consider a simulation that provides the expected position of a Brownian particle after 1,000 time steps.

A Monte Carlo simulation of Brownian motion depends on the probability model that specifies the probability distribution p for the Brownian particle moving to each accessible position at some time t, given its position at an immediately antecedent time $t - 1$. The computation within the simulation that delivers the position for one sample trajectory after 1,000 time steps implements an argument that can be summarized as:

P1'. The initial position of the sample trajectory at time $t = 0$ is $X(0) = 0$ (in suitable units and coordinates).

P2'(t). (t = 0 to 999) If the position of the sample trajectory at time t is $X(t)$, then its position at time $t + 1$, $X(t+1) = X(t) + r_t$, where the r_t sample the probability distribution p.[16]

Cc'. The position of the sample trajectory at time t = 1,000 is $r = \sum_{t=1}^{1,000} r_t$.

The premises P2'(t) follow the same scheme that yields 1,000 individual premises as t takes values from 0 to 999. The conclusion Cc' follows from these premises in conjunction with P1'.[17]

The computation of a single sample trajectory is, in general, not useful by itself, since trajectories will vary considerably in their details. Our goal in the simulation is to obtain statistical features of the entirety of sample trajectories. These are recovered from

---

[16] As explained in Sec. 2 above, this is meant to say that certain frequencies in the random numbers r_i closely match those predicted by the probability model p.

[17] A similar type of argument is carried out by a computer simulation that uses deterministic equations to follow the dynamics of a system (Beisbart, forthcoming).

statistical estimators. We may estimate the expected or mean final position by computing the average of the final positions of many—N, say—trajectories. An argument in the meta-analysis assures us of the final result:

P3'. The expected final position of the particle is very probably the average final position of the large number of N randomly chosen sample trajectories of the probability model.

P4'. The average over N such randomly chosen sample trajectories takes the value r_{av}.[18]

C'. The expected position of the particle, $\langle X(1{,}000)\rangle$, very probably is approximately r_{av}.

Premise P4' asserts the average of the results of the conclusions of the many arguments P1, P2',Cc' associated with the many simulations of the N individual trajectories.

Altogether, we have two kinds of argument: a series of arguments that derive the final positions of sample trajectories; and a meta-level argument that infers the result. The execution of the former, lower-level arguments requires the most computational effort. The meta-level argument is essential if we are to obtain the final result.

7. Distractions and Objections

We believe that we have established that Monte Carlo simulations are merely arguments like other simulations. Nonetheless, there are distractions that still may appear to press Monte Carlo simulations towards real experiments. We list some here and respond to them.

---

[18] As explained in Sec. 2 above, this is meant to say that certain frequencies in the random numbers r_i closely match those predicted by the probability model p.

1. Monte Carlo Simulations depend upon randomness, which is non-inferential. So how can they be arguments?

It is true that Monte Carlo simulations appear, initially, to be quite different from other simulations because of the essential role of randomness. This appearance is deceptive, for randomness can have no role or only a minimal one in the final results if they are to be credible. Otherwise Monte Carlo simulations would be little better than forecasting by reading tea leaves or shuffling tarot cards, whose results are mostly expressions of random chance. The results of a good Monte Carlo simulation are not. They are repeatable and reliable.

While one properly generated random number is unpredictable, the behavior of a large collection of independent random numbers is highly regular. This fact is well-known from the Laws of large numbers.[19] Exploiting this regularity is what enables Monte Carlo simulations to eliminate the vagaries of chance and to return reliable, repeatable results.

As we have shown above, a well-designed Monte Carlo simulation minimizes the role of randomness. Probabilistic convergence of Monte Carlo simulations guarantees that the errors become arbitrarily small. It is true that there is always a small probability of a Monte Carlo simulation producing an incorrect result. But this probability can be made arbitrarily small. In this sense, the role of randomness is minimized.[20]

---

[19] See e.g. Feller, 1968, Ch. 10.

[20] The point is nicely put by James (1980, p. 1151): The name "Monte Carlo simulation" is particularly fitting "because the style of gambling in the Monte Carlo casino, not to be confused with the noisy and tasteless gambling houses of Las Vegas and Reno, is serious and sophisticated." He further comments (ibid.): "The apparent contradiction between the unpredictability of the gambling process and the seriousness of the results is one of the fascinating aspects of the Monte Carlo method which has been responsible for a great deal of the interest shown in the method but has also resulted in considerable confusion and misunderstanding."

2. The randomness within Monte Carlo simulations is akin to the unpredictable, novel character of data supplied by real experiments. So why are Monte Carlo simulations not better thought of as real experiments?

In real experiments, the unpredictable and novel character of the data is the source of the final experimental result. In a Monte Carlo simulation, the unpredictable and novel character of the random numbers is minimized and plays almost no role in the final result.

We should also guard against an equivocation on the term "experiment." It may mislead us with the wrong sense of novelty. In one use of the term, "experiment" denotes any activity whose outcome is unknown at the outset. In that usage, generating a random number is an experiment, since we ought not to know which one will be recovered. Finding the square root of 234,398 would likewise count as an experiment. The use of "experiment" in this paper is narrower and epistemically richer. It denotes the activity of observing some arrangement in the world in order to discover how it will behave. It is empowered epistemically to provide knowledge of the world by a similarity of properties between the system on which we experiment and the target system of interest.

Thus, while the random numbers employed as intermediates in a Monte Carlo simulation may be unpredicted and unexpected, the sense of novelty is different from that associated with a real experiment.

3. Derivations and arguments are transparent in that they are carried out by humans and can be grasped by them. Monte Carlo Simulations are opaque in that we humans cannot grasp the course of the simulations. So how can they be arguments?

There is no requirement that an argument be humanly comprehensible. All that is required is that it is a sequence of propositions conforming to the rules of the applicable logic. That the entirety can be encompassed and comprehended in all detail by a human mind is not required.

Modern mathematics is now well into a revolution in its methods that is dispensing with this feature of proofs. Theorems are being demonstrated by proofs that

are so complicated that no human can comprehend them. Thomas Hales (2005), for example, provided a proof of Kepler's conjecture on the maximum density of the packing of spheres. The proof employed a computer to solve about 100,000 linear programming problems pertaining to different configurations of spheres. It was published only after scrutiny by a panel of 12 referees over four years, although they could not affirm the correctness of the entirety of the proof.[21]

> *4. Monte Carlo simulations can employ physical randomizers such as coin tossers, roulette wheels, Buffon's falling needles, Geiger counters, and so on. So why are these simulations not experiments on these physical randomizers?*

They fail to be experiments on these randomizers for two reasons. First, the point of an experiment is to discover something about the system upon which we experiment and, perhaps, to make an inference to other systems. But a Monte Carlo simulation does not produce any novel knowledge about the randomizer. Working scientists do not use the random numbers to learn about the randomizer.[22] In fact, a well-designed Monte Carlo simulation will return a result that is largely independent of the physical randomizer used. Approximately the same result will be returned whether the randomizer is a coin tosser or a Geiger counter. An essential part of the good design of Monte Carlo simulation is to suppress any peculiarity of the particular randomizer used, while ensuring the presence of some predetermined, desired property, such as the uniformity of distribution of the random numbers over some interval.

Second, most Monte Carlo simulations use a pseudo-random number generator implemented in software. These simulations are clearly not experiments. Further, *if*

---

[21] See Tymoczko, 1979 for philosophical reflections about computer-based proofs in mathematics.

[22] The only exception is when the random numbers from a randomizer are tested for their statistical properties. If an independence test fails, we do learn something about the randomizer, viz. that the trials are not independent. Such tests are necessary to underwrite the premise that the random numbers follow a certain probability model. However, such a test is not a proper part of every Monte Carlo simulation. Once we are confident that a certain randomizer produces random numbers with a certain distribution, we can use the randomizer without testing the random numbers actually produced.

physical randomizers are used in a simulation at all, they are entirely dispensable. They can always be replaced by a suitably designed pseudo-random number generator without any epistemic loss.[23] Consequently, the physical randomizer is not the subject of the inquiry, and Monte Carlo simulations that use such a randomizer cannot be experiments on them.

> *5. A Monte Carlo simulation produces its result using sample trajectories of concrete particulars. This is very different from other numerical techniques used to approximate solutions to equations. Thus, while other numerical methods can be understood as carrying out an inference, Monte Carlo integrations cannot.*[24]

This objection misuses an analogy between real experiments and Monte Carlo simulations. Real experiments lead to results by generalizing from particulars, where the particulars are individual runs of a physical experiment. Monte Carlo simulations lead to results also by generalizing from particulars, where these particulars are computational surrogates of the individual runs of a physical experiment. The difference between the particulars is key. In the first case, the particulars are instances of the physical systems whose properties are sought. Hence their investigation conforms to the epistemic mode we have called "discovery" that is used by real experiments. In the second case, the instances are mere surrogates. As we have shown in Sec. 6, the surrogates are generated inferentially from assumptions about the target system. When they are combined with

---

[23] A possible exception is a case in which it is known that a physical randomizer produces the right sort of random numbers, but in which the pertinent probability model is unknown such that one cannot generate the random numbers using software. This case is rather peculiar though and not typical of Monte Carlo simulations. In this peculiar case, we would not object to the suggestion that we have a real experiment.

[24] Humphreys (1994, 112) puts the objection thus: "It is the fact that they [Monte Carlo simulations] have a dynamical content due to being implemented on a real computational device that sets these models apart from ordinary mathematical models, because the solution process for generating the limit distribution is not one of abstract inference, but is available only by virtue of allowing the random walks themselves to generate the distribution." Humphreys is here concerned with Monte Carlo simulations that trace a probability distribution. The particular interest is the final distribution.

further inferences in the Monte Carlo simulation, they yield the result of the simulation. Consequently, epistemically, the whole process of a Monte Carlo simulation is fully inferential.[25]

6. Monte Carlo simulations must be experiments because they can take the same role as experiments.[26]

That two things can take the same role does not show them to be identical. It only shows them to be similar in some aspect relevant to the role substitution. Experiments and Monte Carlo simulations still differ in the epistemic aspect of interest here. Experiments are powered epistemically by discovery and simulations by inference.

7. Monte Carlo simulations allow one to manipulate the initial conditions as do experiments. Hence, Monte Carlo simulations are experiments.[27]

Once again, two things that share a feature need not be the same. Hence, even though we can explore differing initial set-ups in experiments and in Monte Carlo simulations, this does not show that the latter are experiments. In simulations, the initial conditions are only set up in a model, and not in reality as it is the case in experiments.

8. In a Monte Carlo simulation, things can go wrong in a physical sense. For instance, the physical randomizer or the hardware may fail to function in the way

---

[25] It is true that drawing random numbers (particularly with the aid of a physical gambling device) is not an argument. But a Monte Carlo simulation is much more than drawing random numbers, and what is epistemologically decisive is inference, as we have argued in Sec. 4.

[26] For instance, Dietrich (1996) reports that certain Monte Carlo simulations were used as a benchmark for theoretical work in the same way experiments are (e.g. p. 341).

[27] As Dietrich (1996, p. 346) puts it, "Monte Carlo experiments share this basic structure of independent variables, dependent variables, and controlled parameters" known from experiments.

they are supposed to. This dependence on the physical implementation makes Monte Carlo simulations experiments.[28]

We agree that we can only trust a simulation if we can trust that the physical randomizer and the hardware to function properly. But this does not imply that the simulation is an experiment on the randomizer and the computer hardware. We have already shown that the randomizer is not the subject of empirical discovery during a simulation. Nor is the computer hardware. The working scientist does not seek to learn about the processors of computers. Rather, the randomizer and the hardware are technical means to carry out a calculation. Clearly, the means have to work properly if the calculation is to be successful, and this explains why scientists who run simulations are concerned about failures of the randomizer or the hardware.

The objection that we consider would also lead to implausible consequences. When a computer program like "Mathematica" calculates 4668570242 plus 6980836254, things can go wrong due to a hardware failure. But this does not show that the summation is an experiment.

9. In analog simulations we carry out an experiment on one system to learn about a system to which it is (supposedly) formally analogous. So why are not Monte Carlo simulations analog simulations? They include an experiment on a randomizer; a formal analogy is then used to infer something about the target system.[29]

Monte Carlo simulations are quite different from analog simulations. In a paradigmatic example of a formal analogy (Humphreys, 2004, pp. 125–29), two systems are well described using the same type of differential equation, e.g. an equation for a

---

[28] This argument applies not just to Monte Carlo simulations, but to every computer simulation. See Parker (2009, pp. 488–91) for a similar argument.

[29] To outline the objection, we rely on a proposal due to Guala (2002, pp. 66–67) and an emendation by Winsberg (2010, pp. 57–58). A formal analogy contrasts with a material one. The distinction between both analogies goes back to Hesse (e.g. Hesse, 1966, p. 68). Consult also Trenholme (1994) for analog simulations.

damped harmonic oscillator. This fact can be used to make an inference from one system to the other. Things are different in typical Monte Carlo simulations. The randomizer (say, a roulette wheel) and the target of the simulations (say, a Brownian particle) do not obey the same type of equation. It need not even be the case that the outcomes of the randomizer and some aspect of the target follow the same probability distribution, for the random numbers from the randomizer are often transformed considerably by the simulation program. For instance, random numbers with a uniform distribution over [0,1] are transformed into random numbers with a normal distribution (e.g. James, 1980, pp. 1179–83). The outcome is that a formal analogy between the target and the randomizer is not required for the functioning of Monte Carlo simulations.

8. Conclusion

Monte Carlo simulations are, we urge, merely rather complicated arguments and have no epistemic powers beyond those of an argument. We may well ask what we gain from this deflationary conclusion. For, we might say, Monte Carlo simulations are interesting in so far as they differ from our normal modes of investigation. Our deflationary conclusion merely identifies how they are the same.

We agree that there is much interesting novelty in Monte Carlo methods and much to be found in investigating how they differ from other modes. In particular, they deliver results that other methods find difficult or even impossible in practice; and they are distinctive in their dependence on the novel computational technology of our modern era.

The danger is that we are swept up by the excitement of this novelty into believing that we have found some qualitatively new mode of investigation. We read our deflationary result as placing a bound on these enthusiasms. It shows that the epistemic gains supplied by Monte Carlo simulations are pragmatic. They open no new epistemic channels.

References

Anderson, H. L. (1987). Metropolis, Monte Carlo, and the MANIAC. *Los Alamos Science*, 96–107

Beisbart, C. (forthcoming). How can Computer Simulations Produce New Knowledge? *European Journal for Philosophy of Science*

Brown, J. R. (2004). Why Thought Experiments Transcend Empiricism (In: C. Hitchcock (Ed.), *Contemporary Debates in the Philosophy of Science* (pp. 23–44). Malden, MA: Blackwell, 2004.)

Cavendish, Henry (1798). Experiments to Determine the Density of the Earth. *Philosophical Transactions of the Royal Society of London*, 88, 469–526 (reprinted in A. S. MacKenzie (Ed.), *The Laws of Gravitation: Memoirs by Newton, Bouguer and Cavendish* (pp. 59–106). New York: American Book Company 1900.)

Dietrich, M. R. (1996). Monte Carlo Experiments and the Defense of Diffusion Models in Molecular Population Genetics. *Biology and Philosophy*, 11, 339–356

Eckhardt, R. (1987). Stan Ulam, John von Neumann, and the Monte Carlo Method. *Los Alamos Science*, 131–141

Feller, W. (1968). *An Introduction to Probability Theory and its Applications, Vol. II*. (New York: Wiley and Sons, third edition)

Galison, P. (1997). *Image and Logic. A Materical Culture of Microphysics*. (Chicago: University of Chicago Press)

Giere, R. N. (2009). Is Computer Simulation Changing the Face of Experimentation? *Philosophical Studies*, 143, 59–62

Guala, F. (2002). Models, Simulations, and Experiments (In: L. Magnani & N. Nersessian (Eds.). *Model-Based Reasoning: Science, Technology, Values* (pp. 59–74). New York: Kluwer.)

Gubernatis, J. (Ed.) (2003). *The Monte Carlo Method in the Physical Sciences: Celebrating the 50th Anniversary of the Metropolis Algorithm*. (Melville, NY: AIP, Conference Proceedings, Volume 690)

Hales, T. (2005). A Proof of the Kepler Conjecture. *Annals of Mathematics*, 162, 1065–1085.

Halton, J. H. (1970). A Retrospective and Prospective Survey of the Monte Carlo Method. *SIAM Review*, 12, 1–63

Hammersley, J. M. & Handscomb, D. C. (1967). *Monte Carlo Methods*. (London: Methuen)

Heidelberger, M. (2005). Experimentation and Instrumentation (In: D. Borchert (Ed.), *Encyclopedia of Philosophy*. *Appendix* (pp. 12–20). New York: Macmillan.)

Hesse, M. (1966). *Models and Analogies in Science.* (Notre Dame, IND: University of Notre Dame Press)

Hitchcock, D. B. (2003). A History of the Metropolis-Hastings Algorithm. *The American Statistician*, 57, 254–257

Humphreys, P. (1994). Numerical Experimentation. (In P. Humphreys (Ed.). *Patrick Suppes. Scientific Philosopher*. (Volume 2, pp. 103–118) Dordrecht: Kluwer.)

Humphreys, P. (2004). *Extending Ourselves: Computational Science, Empiricism, and Scientific Method*. (New York: Oxford University Press)

James, F. (1980). Monte Carlo Theory and Practice. *Rep. Prog. Phys.*, 43, 1145–1187

Knuth, D. E. (2000). *The Art Of Computer Programming. Vol. II: Seminumerical Algorithm*s. (Upper Saddle River, NJ: Addison-Wesley, third edition)

Lemons, D. S. (2002). *An Introduction to Stochastic Processes in Physics*. (Baltimore and London: Johns Hopkins University Press)

Millikan, R. A. (1913). On the Elementary Electrical Charge and the Avogadro Constant. *Physical Review*, 2, 109–143

Morrison, M. (2009). Models, Measurement and Computer Simulation: The Changing Face of Experimentation. *Philosophical Studies*, 143, 33–57

Norton, J. D. (1996). Are Thought Experiments Just What You Thought? *Canadian Journal of Philosophy*, 26, 333–366

Norton, J. D. (2004). Why Thought Experiments Do Not Transcend Empiricism. (In C. Hitchcock (Ed.). *Contemporary Debates in the Philosophy of Science* (pp. 44–66). Malden, MA: Blackwell)

Parker, W. S. (2009). Does Matter Really Matter? Computer Simulations, Experiments, and Materiality. *Synthese*, 169, 483–496

Press, W. H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T. (1992). *Numerical Recipes in C: The Art of Scientific Computing*. (Cambridge: Cambridge University Press, second edition)

Radder, H. (2009). The Philosophy of Scientific Experimentation: A Review. *Automatic Experimentation*, 1, http://www.aejournal.net/content/1/1/2 .

Stöckler, M. (2000). On Modeling and Simulations as Instruments for the Study of Complex Systems (In M. Carrier, G. J. Massey, & L. Ruetsche (Eds.). Science at the Century's End: Philosophical Questions on the Progress and Limits of Science (pp. 355–373). Pittsburgh, PA: University of Pittsburgh Press.)

Tocher, K. D. (1975). *The Art of Simulation.* (London: Hodder and Stoughton)

Trenholme, R. (1994). Analog Simulation. *Philosophy of Science*, 61, 115–131

Tymoczko, T. (1979). The Four-Color Problem and its Philosophical Significance. *Journal of Philosophy*, 76, 57–83

Young, D. M. and Gregory, R. T. (1972). *A Survey of Numerical Mathematics. Volume 1*. (Reading, MA: Addison Wesley. Reprinted New York: Dover, 1988)

Winsberg, E. (2010). *Science in the Age of Computer Simulations*. (Chicago: University of Chicago Press)