Randomness By Design

William A.Dembski, Ph.D.

A mathematician and a philosopher, William A. Dembski is associate research professor in the conceptual foundations of science at Baylor University and a senior fellow with Discovery Institute's Center for the Renewal of Science and Culture in Seattle. Dr. Dembski previously taught at Northwestern University, the University of Notre Dame, and the University of Dallas. He has done postdoctoral work in mathematics at MIT, in physics at the University of Chicago, and in computer science at Princeton University. A graduate of the University of Illinois at Chicago where he earned a B.A. in psychology, an M.S. in statistics, and a Ph.D. in philosophy, he also received a doctorate in mathematics from the University of Chicago in 1988 and a master of divinity degree from Princeton Theological Seminary in 1996. He has held National Science Foundation graduate and postdoctoral fellowships. Dr. Dembski has published articles in mathematics, philosophy, and theology journals and is the author/editor of seven books. In *The Design Inference: Eliminating Chance Through Small Probabilities* (Cambridge University Press, 1998), he examines the design argument in a post-Darwinian context and analyzes the connections linking chance, probability, and intelligent causation. The sequel to *The Design Inference* is due out December 2001 with Rowman's & Littlefield's and critiques Darwinian and other naturalistic accounts of evolution. It is titled *No Free Lunch: Why Specified Complexity Cannot Be Purchased without Intelligence*.

1. Introduction

"Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."1 John von Neumann's famous dictum points an accusing finger at all who set their ordered minds to engender disorder. Much as in times past thieves, pimps, and actors carried on their profession with an uneasy conscience, so in this day scientists who devise random number generators suffer pangs of guilt. George Marsaglia, perhaps the preeminent worker in the field, quips when he asks his colleagues, "Who among us has not sinned?" Marsaglia's work at the Supercomputer Computations Research Institute at Florida State University is well-known. Inasmuch as Marsaglia's design and testing of random number generators depends on computation, and inasmuch as computation is fundamentally arithmetical, Marsaglia is by von Neumann's own account a sinner. Working as he does on a supercomputer, Marsaglia is in fact a gross sinner. This he freely admits. Writing of the best random number generators he is aware of, Marsaglia states, "they are the result of arithmetic methods and those using them must, as all sinners must, face Redemption [*sic*] Day. But perhaps with better understanding we can postpone it."2

Despite the danger of being branded a heretic, I want to argue that randomness entails no moral deficiency. I will even advocate that random number generators be constructed with reckless abandonthough a reckless abandon that is well thought out. Randomness, properly to be randomness, must leave nothing to chance. It must look like chance, like a child of the primeval chaos. But underneath a keen intelligence must be manipulating and calculating, taking advantage of this and that expedient so as systematically to concoct confusion. I am reminded of the photo-journalists in Vietnam who rearranged scenes of carnage simply to enhance the sense of indiscriminate violence. Here, of course, there was a moral fault, but not with randomness per se. Suffice it to say, randomness, to be randomness, must be designed.

In his now classic, though somewhat dated, study of random numbers Donald Knuth (1981, pp. 4-6) describes his naive attempt to construct a foolproof random number generator. His "Super-random" number generator (the shudder quotes are his) was a tangled web of subroutines that built complication upon complication. His rationale was that an incredibly complicated algorithm which no one could follow ought to produce an incredibly complicated sequence of numbers which, again, no one could follow, i.e., for which no systematic pattern could be found. Failure to find such patterns would be taken to signal randomness. Inscrutability in, inscrutability outthis was Knuth's rationale. His rationale proved dead wrong. Instead of finding disorder and chaos, Knuth discovered the worst sort of non-randomness: his algorithm took a particular seed (i.e., an initial input that launches the random number generator) and just kept repeating it. The seed was 6065038420. Knuth's random number generator repeated 6065038420 over and over again:

6065038420 6065038420 6065038420 6065038420 6065038420

6065038420 6065038420 6065038420 6065038420 ....

Whatever is meant by randomness, it certainly can't be this. Knuth (1981, p. 5) was quick to draw the right conclusion: "The moral of this story is that *random numbers should not be generated with a method chosen at random*. Some theory should be used" (the italics are his).

Knuth and I agree that generating randomness involves forethought and design. Knuth, however, still suffers from a guilty conscience, which I do not. Random number generators must be carefully designed. On this point there is no controversy. Randomness is fundamentally a question of design. This point is more far reaching and open to controversy. Randomness supervenes on design, not probability. Herein lies a departure from precedent. The typical way of understanding randomness is as follows: an object supposed to exhibit randomness is proffered (e.g., a sequence of numbers). Next one examines the object against a collection of patterns (e.g., statistical tests). If the object fits any pattern in the collection, it is non-random. If it violates all the patterns in the collection, it is random. I propose to reverse this. Consider *first* a fixed collection of patterns. Any object which violates all the patterns in this collection is random. Those which satisfy some one pattern in the collection are non-random. In this way randomness becomes a relative notion, i.e., randomness *with respect to* a collection of patterns.

In practise the first approach to randomness is fundamentally probabilistic: strings of digits constitute the random objects, and statistical tests set the patterns. When the pattern induced by a statistical test is violated, we say the string passes the test.3 When the string passes sufficiently many tests, we say the string is random. The tests, however, are formulated so that most strings generated according to a fixed probability distribution pass the test. This poses a problem. For any string there is some statistical test which the string fails to pass. Thus we can always cook up tests which render a supposedly random string non-random.[4] It is within this context that von Neumann uttered his dictum. Truly random strings are supposed to be generated according to some probability distribution and for this reasonand this reason alonepass statistical tests. Random number generators, on the other hand, are purely deterministic and can only mimic the passing of statistical tests. According to von Neumann, strings generated by computer algorithms can at best pretend to randomnessthey are impostors.

But when probability is repudiated, randomness is no longer a question of imitating chance. When randomness supervenes on design, patterns become the fundamental object of study. A random object is then an object which systematically violates a fixed collection of patterns. In contrast to the conventional probabilistic approach, this alternate approach is without pretense. With premeditated randomness one does not try to imitate chance as one does with probabilistic randomness. Rather, one conducts a methodical search for an object satisfying certain constraints. The constraints comprise the patterns which must all be violated.

To clarify these thoughts I shall need to review a little probability theory as well as some past thoughts on randomness. In analyzing concrete instances of randomness, I shall limit myself to sequences of 0's and 1's. This limitation involves no real loss of generality. At the outset let me stress that probability is a well-defined mathematical theory. Randomnesswhat I have called probabilistic randomnessis not. At an interdisciplinary conference on randomness attended, among others, by statisticians George Marsaglia and Persi Diaconis as well as philosophers Brian Skyrms and Richard Jeffrey, the broad conclusion was this: *We know what randomness isn't, not what it is*. I attribute this unattractive conclusion to the wedding of randomness with probability. The two experience irreconcilable differences. Probabilistic randomness has consistently withstood a precise theoretical formulation. On the other hand, the premeditated randomness I shall sketch does lend itself to a theoretical formulation.

2. A Little History and Motivation

Probability's appeal to the popular imagination has always resided in the law of large numbers. Ever since Thomas Huxley's simian typists gave the world a complete set of Shakespeare, people have stood in awe of this law. Its basic contention is that if an event has a positive probability of occurring, no matter how small, and if one repeats the circumstances under which that event can occur *often enough*, then that event will definitely occur.[5] Of course, if the event has probability zero of occurring, it will never occur.

As an example, suppose you are confined to prison and handed a fair coin. You are informed that if you flip the coin and get 100 heads in a row, you will be released. Since individual coin tosses are independent, probabilities multiply. Thus you expect heads on the first toss with probability 1/2, two heads in a row with probability (1/2) x (1/2) = 22, ..., and 100 heads in a row with probability (1/2) x (1/2) x x (1/2) [100 times] = 2100, which is approximately 1 in 1030. This probability is so small as to leave you little hope of getting out of jail soon. If you could, for instance, make 10 billion attempts each year to obtain 100 heads in a row, then you stand only an even chance of getting out of jail in 1020 years. But take heart, the strong law of large numbers guarantees that *eventually* you will be free.[6]

Suppose next you are handed a standard deck of playing cards. This time to get out of prison you have to deal yourself a royal flush in the suite of spades, each time thoroughly shuffling the deck. This event has probability on the order of 1 in a million. And so in about a million tries you should be out of jail. Your jailer, however, likes your company, and wants to keep you around. Consequently, he decides to remove the ace of spades from the deck. This move shatters your hopes of freedom. With the doctored deck your probability of getting the appropriate royal flush is precisely zero.

In any probabilistic interpretation time plays a key role. Coin tossing is really the basic example in probability theory; there is a sense in which if one understands coin tossing in all its ramifications, one understands all of probability theory.[7] Say you are given a fair coin. You are about to toss the coin. You are uncertain of the outcome. There is an even chance that it will come out heads or tails. Now you flip the coin. It lands heads. Suddenly all uncertainty is removed. Uncertainty and probability apply only to future, unrealized events. Once the event has occurred and been noted, all uncertainty is removed.

Rare events are a cause for surprise only if the timing is right. Imagine, for instance, that before you is a large, grassy field. You have 100 stones and 100 flags each marked from 1 to 100. With a helicopter you fly over the field, releasing the stones indiscriminately. After you have dropped your last stone, you land the helicopter safely away from the field, leave the helicopter on foot, and examine where your stones have landed, placing next to each stone a flag with the corresponding number. There are an exceedingly large number of ways the stones could have landed. They had to land in some one way. You are looking at it. You are not surprised or shocked. You don't think a miracle has occurred because you are witnessing an event of exceedingly small probability. Some improbable event had to occur. Placing the flags next to the stones *after* the stones have fallen does not change these conclusions.

Now modify the situation. As before you have a field, stones, flags, and a helicopter. As before you take your helicopter and stones, and fly over the field, dropping the stones indiscriminately. But before you take off you first walk around your field and stick the flags in the ground at will. Having dropped the stones, you land the helicopter and now examine the field. Lo and behold, all the stones are next to their matching flags. Do you have a right to be surprised? Absolutely. When an extremely unlikely event matches a preset pattern, there is cause for surprise. In fact when such an event becomes too unlikely, one looks for non-probabilistic factors to account for it.

To reinforce this point, let me offer another example. Suppose someone stands 50 meters from a large wall with bow and arrow in hand. The wall is sufficiently large that he cannot help hitting it. Every time he shoots an arrow at the wall, he paints a target around the arrow, so that the arrow is squarely in the bull's-eye. What can be concluded? Absolutely nothing about the archer's ability as an archer. But suppose now he paints a fixed target on the wall and then shoots at it. Behold, 100 times in a row he hits a perfect bull's-eye. Nobody in his right mind would attribute this performance to beginner's luck. In fact, one is obliged to conclude this is a world-class archer.

Temporal succession figures into any probabilistic interpretation. When the flags are placed *after* the rocks have fallen, and the archer paints the bull's-eye *after* the arrow has been shot, there are no surprises. But when the flags and target are preset, and the outcome matches the preset pattern, it is vain to appeal to the law of large numbers. It only tells us that eventually we can expect to see some incredibly rare event, not that we shall witness it as the next event. If we do witness it immediately, we should be shockedso much so that we should look beyond chance to account for these otherwise grotesque anomalies.

The examples I have just described fit neatly within Kolmogorov's foundational framework for probability theory which he developed in the 1930's (Kolmogorov, 1950). By the mid-1960's, however, Kolmogorov was concerned with the following problem for which his earlier work in probability provided no insight (Kolmogorov, 1965b): flip a fair coin 100 times and note the occurrences of heads and tails in order. Let us agree to denote heads by the number 1 and tails by the number 0. Thus a sequence of 100 coin flips could be represented as follows:

11000011010110001101111111010001100011011001110111 (R)

00011001000010111101110110011111010010100101011110

This is in fact a sequence I just constructed by flipping a penny 100 times. Now compare this with the following sequence:

11111111111111111111111111111111111111111111111111 (N)

11111111111111111111111111111111111111111111111111

This sequence corresponds to flipping heads 100 times in a row. Now the problem Kolmogorov faced with his standard probabilistic framework, the one he constructed in the 1930's, was his inability to say anything about which of these two sequences was *more random*. Sequence (R) and sequence (N) have been labeled suggestively, R for random, N for non-random. Kolmogorov wanted to say that (R) was more random than (N). But his probability theory from the 1930's only told him that each of these sequences have the same small probability of occurring, namely 2100, or approximately 1 in 1030. We analyzed this probability earlier for the sequence (N), but the analysis is true for any sequence of 100 coin tosses. Each sequence of 100 coin tosses has the same small probability.

To get around this difficulty Kolmogorov introduced some concepts from recursion theory, a subfield of mathematical logic concerned with computation and generally considered quite far removed from probability theory. What he said was that a string of 0's and 1's is more and more random as the shortest computer program that generates the string becomes longer and longer (Kolmogorov, 1965b). A computer program can be conceived as a collection of simple instructions to be executed sequentially. For our purposes we can think of a computer program as a short-hand description. Thus sequence (N) is not very random because it has a very short description, namely,

repeat '1' 100 times.

Note that we are interested in the shortest descriptions. Any sequence can be described in terms of itself. Thus (N) has the long description

copy '11111111111111111111111111111111111111111111111111

11111111111111111111111111111111111111111111111111'.

But this is of no interest to us since there is one so much shorter.

The sequence

11111111111111111111111111111111111111111111111111 (H)

00000000000000000000000000000000000000000000000000

is slightly more random since it requires a longer description, for example,

repeat '1' 50 times, then repeat '0' 50 times.

So too the sequence

10101010101010101010101010101010101010101010101010 (A)

10101010101010101010101010101010101010101010101010

has a short description,

repeat '10' 50 times.

The sequence (R) has no short and neat description. For this reason Kolmogorov would regard it as more random than sequences (N), (H), and (A).

As we noted, one can always describe a sequence in terms of itself. Thus (R) has the description copy '11000011010110001101111111010001100011011001110111

00011001000010111101110110011111010010100101011110'.

Because sequence (R) was constructed by coin flips, it is very likely that this is the shortest description of (R). It is a fact that the vast majority of sequences of 0's and 1's have as their shortest description just the sequence itself, i.e., most sequences are random in Kolmogorov's computational sense. In the language of statistical mechanics, there are lots of high entropy sequences, but few low entropy sequences. Thus the collection of all highly ordered sequences, those whose computational descriptions are very short, constitutes a rare event, and the observance of any such sequence as a result of chance alone is cause for surprise. Nay, it is cause to look for explanations other than chance.

Let us now consider a practical application of Kolmogorov's ideas. Consider some fellow who approaches you on the street and informs you he has just flipped a coin 100 times. If he hands you sequence (R), you examine it and try to come up with a short description (coming up with a short description is analogous to performing statistical tests). After repeated attempts you find you cannot describe the sequence any better than the sequence describes itself. Hence you conclude it is a genuinely random sequence, i.e., a type of sequence this fellow might well have gotten by flipping a fair coin. You are not particularly surprised or impressed.

Suppose next this fellow hands you sequence (R) on a slip of paper and then disappears. A week later he reappears and says, "Guess what? Remember that sequence I handed you a week ago. Well, last night I was flipping this penny. And would you believe it, I got the same sequence as on the slip of paper." You examine the coin and are convinced of its genuineness. Moreover, this fellow insists that each time he flipped the penny, he gave it a good jolt (these were not phony flips). What do you conclude now? As before, you will not be able to find any shorter description than the sequence itselfit is a random sequence. Unless you believe in miracles, however, you would be a fool to conclude this fellow is telling the truth. The timing is all off. When he handed you the sequence a week earlier, he preset the pattern. Thus the order is established. When he returns and says he *subsequently* reproduced the sequence he handed you, he perjures himself. For what he is really saying is that he knew what sequence he would be flipping later that week. This is prophecy. Lest anyone think that prophecy is not miraculous (read supernatural, strictly outside the material realm), he need only go to Wall Street or Las Vegas where all genuine prophets are billionaires.

Suppose finally this fellow comes to you and says, "Would you believe it? I just flipped this penny 100 times, and it came up heads each time." As before, the coin he shows you is a genuine penny, and he is emphatic that his were not phony flips. This time he did not preset the pattern. Rather the pattern is intrinsically given. Sequence (N) has about the lowest entropy possible. There are very few sequences with descriptions as short as "repeat '1' 100 times." Once again, except for a miracle you would be a fool to believe this fellow is telling the truth. Reasonable minds explain such events apart from chance. The problem is not that such sequences constitute exceedingly rare events. The problem is rather that there are too many other events which violate the few preset patterns humans are able to retain in their minds. Basic here is the notion of an intrinsic order. In the sense of our flags and stones example, our cognition presets the flags in a very limited number of ways. When the stones fall and land next to the preset flags, we are right to be surprised and look for explanations other than chance. Probabilistic arguments of this sort are circumstantial. Our coin flipping friend who claims to have flipped 100 heads in a row (with a fair coin, without phony flips) would be convicted of lying in polite society, much as a lottery manager whose relatives all win the jackpot would be convicted of fraud by a jury.[8]

3. Complexity and Randomness

Computational complexity theory is perhaps the hottest topic currently in theoretical computer science. Computational complexity addresses the computational resources needed for an algorithm to accomplish its task. The big question in computational complexity is whether the polynomial-time algorithms coincide with the non-deterministic polynomial-time algorithmswhether P equals NP (see Garey and Johnson, 1979). This is a question of time-complexity. The resource is time and the question is whether the problems in NP can be solved in polynomial time. But time is not the sole computational resource. Space, or equivalently memory, enters as well. How much memory is needed to solve a given problem? This too becomes a major consideration. In the construction of efficient algorithms, time-memory tradeoffs must always be kept in mind. Thus a polynomial-time algorithm may require too much memory to be practicable, whereas a program requiring little memory may run interminably.

Now what has all this to do with randomness? If we recall Kolmogorov's approach to randomness, we understand that within his framework a string of numbers is random to the extent that the program which generates it is maximal. But maximal in what sense? Maximal in the sense of program length. Kolmogorov's random generators are programs which satisfy two constraints: (1) no program of strictly shorter length must exist which generates the proposed random string, i.e., the program cannot be abbreviated and still generate the string. Let us call such programs *terse*. This requirement is essential since for any program it is possible to add in some vacuous loops which increase the length of the program, but leave the effective work of the program unchanged, i.e., leave input-output unchanged. (2) Among all terse programs the random generators are those of maximal length. Kolmogorov's random generators are really solutions to a minimax problem: among all terse programs (those satisfying the minimality condition) choose those of maximal length. Kolmogorov's notion of randomness hinges on space-complexitythe key parameter is program length. To generate random strings these programs must be stored in the memory of a computing device. Those which eat up the most memory, but cannot be abbreviated without affecting input-output, are Kolmogorov's random generators.

More recently, time-complexity has been used to define randomness. In this case one looks to strings of digits which polynomial-time algorithms cannot distinguish from truly random strings (i.e., strings whose digits are derived by sampling independently from a fixed probability distribution). One speaks of strings being P-*indistinguishable* from truly random strings. The basic idea here is that the only algorithms humans can legitimately wield are polynomial-time algorithms; non-polynomial time algorithms are beyond our ken. Thus if all our polynomial-time algorithms fail to distinguish a putative random string from a truly random string, then in fact no distinction exists. Leibniz's identity of indiscernibles is implicit heredistinctions arising through non-polynomial algorithms are indiscernible.

Mathematicians have found these space- and time-complexity approaches to randomness highly stimulating, at least initially. Without question the ideas are pretty. Moreover, there is something genuinely deep going on here. Martin-Löf (1966a), a student of Kolmogorov, derived a good deal of classical probability theory from the space-complexity approach to randomness (e.g., the law of large numbers and the law of the iterated logarithm). Andrew Yao (1982) and Silvio Micali (Goldreich, Goldwasser and Micali, 1986) have used the time-complexity approach to randomness with some success in cryptography (cf. the one-way and trapdoor functions of public-key cryptography).

Still, there are problems. After the initial enthusiasm and successes have worn thin, one finds that complexity approaches to randomness don't deliver on their promises. This is certainly true of Kolmogorov's approach via space-complexity. Time-complexity, being a much more recent approach to randomness, has yet to find disfavor. Nevertheless, similar difficulties face both approaches. Certainly space- and time-complexity supply wonderful intuitions for randomness, and without them it is unlikely this paper would have been written. But they fail to deliver a theory of randomness in the sense that one can point to any concrete sequence of 0's and 1's and call it random.

There are two reasons for this practical failure. The first has to do with the choice of programming language. By this I do not mean BASIC, Lisp, or Fortran, but rather how a computational device interprets a string of 0's and 1's as a program and then uses such a (program) string to generate the random (output) strings we are after. Alternatively, we can ask, Which universal Turing machine are we to use? Neither space- nor time-complexity approaches to randomness address this question. The technical results that derive from these approaches are fundamentally asymptotic, depending on ever-increasing input and output strings. As a result the actual choice of programming language becomes immaterial: one can say what general characteristics ever-increasing strings of 0's and 1's must have to be random, but one cannot specify the random strings of a given length.

To clarify this criticism let us reconsider an example from the last section. There we examined two strings,

1100001101011000110111111101000110001101100111011 (R)

00011001000010111101110110011111010010100101011110

and

11111111111111111111111111111111111111111111111111 (N)

11111111111111111111111111111111111111111111111111.

(R) was constructed by flipping a coin 100 times, whereas (N) was constructed without recourse to any chance mechanism. I claimed that (R) was more random than (N) because the shortest program for generating (R) was longer than the shortest program for generating (N):

copy '11000011010110001101111111010001100011011001110111

00011001000010111101110110011111010010100101011110'

versus

repeat '1' 100 times.

But in making this claim I engaged in some shameless handwaving. This is not to say I misled the reader. Rather, in stroking the reader's intuition I had to dispense with the usual standards of mathematical rigor. Let me now make things right. Our choice of programming language was imperative statements in English: do this, then do that, then go back to doing this, do such-and-such ten times, etc. This is a perfectly valid programming language as long as all commands are intuitively computable.9 Thus we must exclude commands which would allow us to solve the halting problem or would stop if Fermat's conjecture were in fact true.

Let us call this programming language Glish. If we restrict our attention to the terse programs of Glish, we can be sure that (R) will require a longer program than (N). But let us now consider a variant of Glish, the programming language Glish*. Glish* is identical with Glish, save the following modification: for programs longer than 100! (= 100 x 99 x 98 x ... x 2 x 1) Glish* is just Glish; for programs shorter than 100! those which in Glish produce (N) produce (R) in Glish*, and those which in Glish produce (R) produce (N) in Glish*; for programs shorter than 100! which produce neither (N) nor (R), Glish and Glish* are identical. Thus Glish and Glish* have identical output for all programs beyond a certain length and interchange output of strings (N) and (R) for programs of shorter length. Note that Glish and Glish* are both universal computers. Also observe that since these languages coincide once programs have achieved a certain length (100!), Glish and Glish* have identical asymptotic properties. Thus any computational approach to randomness which is machine independent will yield the same notion of randomness for both Glish and Glish*.

Glish and Glish*, however, give conflicting accounts of the randomness of strings (N) and (R). In Glish* the simple program

repeat '1' 100 times.

generates what to our intuition is the more random

11000011010110001101111111010001100011011001110111 (R)

00011001000010111101110110011111010010100101011110,

whereas the complicated program

copy '11000011010110001101111111010001100011011001110111

00011001000010111101110110011111010010100101011110'

now generates the intuitively simple

11111111111111111111111111111111111111111111111111 (N)

11111111111111111111111111111111111111111111111111.

Of course the move from Glish to Glish* is a cheap trick, but it is a trick fully sanctioned by recursion theory. Because the programming language can always be perverted in this way, complexity theory can tell us nothing about the randomness of a fixed, finite string. Only as we allow strings to become arbitrarily large do the complexity approaches to randomness give firm results. Kolmogorov's approach to randomness offers an intuition of why (R) is more random than (N), an intuition confirmed for concrete programming languages like Glish. But the theorems of theoretical computer science carry weight only if they are machine independent, i.e., only if they hold across all programming languages. Thus the computational complexity approaches to randomness at best yield asymptotic, limiting results.

The question of programming languages is not solely responsible for the failure of complexity theory to give a practical account of randomness. Equally responsible is the still unresolved role of probability. Random strings are, after all, supposed to resemble strings derived from chance processes. Thus any string that a computer outputs demands probabilistic validation. And this as we have seen lands us in a probabilistic bog, for we must subject a putative random string to statistical tests. Now a statistical test is among other things a decision procedure; it must decide between outcomes which pass the statistical test, and those which fail it. Neither of these categories must be empty, otherwise the statistical test is vacuous. Thus any such test must fail some strings and pass others. But how shall the tests themselves be chosen? Which tests suffice to guarantee randomness?

Confusion here has led to droves of abysmal random number generators, which because of their wide use in experimental research have filled the scientific literature with type I errors. This is a well recognized fact. Often it has been blamed on programmers who while competent at the computer left much to be desired as statisticians. Nevertheless, the problem of bad random number generators persists even among highly competent workers in the field. Thus Donald Knuth touts an additive number generator which George Marsaglia later discredits. How does Marsaglia accomplish this? He concocts a statistical test which strings produced by the additive generator should pass if they derived from a chance process, but in fact fail to pass.10

The picture is that of a game where programmer and statistician fight it out. The programmer wants an efficient program that generates random numbers. The statistician wants a simple statistical test which discredits the random numbers so generated. The programmer proposes, the statistician disposes. As long as the statistician has no statistical test to discredit the random strings generated by the program, the programmer wins; as soon as a successful statistical test is cooked up, the statistician wins. The game is no doubt fun, and responsible for countless research articles. But it can never offer a conclusive theory of randomnessthe game has no resolution.

4. Randomness as a Theory

Throughout this essay I have deliberately distinguished randomness, probability, and chance. Chance I leave to coin tossing and quantum events. Whether chance is reducible to a determinism or fundamentally indeterministic or simply illusory is a debate I will not venture upon here. Probability, the measure theoretic probability of Kolmogorov from the 1930's, is a well-defined mathematical theory inspired by chance processes and designed to model chance. Randomness, to date, has been the scientist's attempt to mimic chance using deterministic methods.11

Let us now repudiate all pretensions to chance and probability, and require but one thing of randomness: the systematic violation of a fixed set of patterns. What will such a theory look like? First we need to delimit a collection of potentially random objects. Let us call such a collection a *candidate space* and denote it by . The elements of are candidates running for officethe honor of being called random. Next we need to delimit a collection of patterns. The patterns are, if you will, hurdles which the candidates must jump in order to receive the distinction of being called random. More precisely, a candidate w in is random if it violates all the patterns from a fixed collection of patterns. Let us call such a collection of patterns a *pattern space* and denote it by P. Observe that this is a relative notion of randomnessw is random relative to P.

For each pattern p in P, a candidate w will either fit or violate the pattern. Thus a pattern is nothing more than a separation of the space into two nonempty, disjoint, and exhaustive subsets, where inclusion in one of the subsets signifies fitting p, inclusion in the other, violating p. Now this can make for some exceedingly dull mathematics, if we're not careful. For, starting with the candidate space , we can reduce patterns to nothing more than a collection of subsets of , like say A1, A2, ..., An. Then for some object w to violate all these patterns is simply for w to fall outside each of A1, A2, ..., and An. Thus w is random if it lies in the complement of A1 » A2 » ... »An. Moreover, if this complement is empty, then has no random elements with respect to the pattern space {A1, A2, ..., An}. At the highest level of generality this is all we are doing when constructing or finding a random object. Thus, if the framework I am proposing for randomness offers any interesting possibilities, it must do so at a lower level of generality, where some rationale justifies the choice of patterns relative to which candidates in are deemed random (e.g., complexity considerations).

Nevertheless, even at the purely set theoretic level some useful insights into randomness can be gained. We are looking for random objects in the candidate space relative to the pattern space P. We take the patterns in P as subsets of so that fitting a pattern p in P coincides with membership in p. Let us denote the random objects of relative to P by

/P := {w|wpforallpP}. (4.1)

Consider now two pattern spaces P and P¢. If P¢ includes P, then the /P¢ cannot contain more random elements than /P. This accords with intuition, for the more patterns a potentially random element must violate, the less likely it is to attain this distinction. The patterns set up hurdles which the candidates in must jump to qualify as random. Since P¢ contains more hurdles than P, the candidates have a harder time qualifying relative to P¢ than to P.

It is also clear from this general formulation that there can be too many patterns, or that the patterns might be ill-chosen, so that /P is empty. Thus we might set up too many hurdles so that no candidate can qualify as random. This was precisely the problem with von Mises's (1936) *collectives*. His idea was to delineate the random infinite sequences of 0's and 1's modeled on the endless tossing of a fair coin. The candidate space was therefore {0,1} and a proposed random sequence was to have 0's and 1's evenly distributed (i.e., same proportion of 0's as 1's). von Mises wanted to push this notion of even distribution as far as he could. Thus he wanted to require even distribution of 0's and 1's across all subsequences of a potentially random sequence. This proved too stringent a requirement.

More formally, von Mises entertained the following hope: his candidates w comprised all functions from the natural numbers **N** = {0,1,2,...} to the binary set {0,1}, i.e., the infinite sequences of 0's and 1's. His patterns were induced by infinite subsets of **N** like S = {s0 < s1 < s2 < ...}. As von Mises saw it, for w to be random it should be evenly distributed on any such Srandomness after all was to mimic the tossing of a fair coin. Thus a random w was to satisfy

(4.2)

for all infinite subsets S of **N**.

But this presents a problem. There are simply too many such subsets S for any candidate w to satisfy (4.2) for all S. This is readily seen. A random w must certainly be evenly distributed on all of **N** and must therefore satisfy

(4.3)

Now if we choose S to be that (infinite) subset of **N** on which w is identically 1, then on S

(4.4)

w certainly fails to be evenly distributed on this S. Hence for any purportedly random w we can always find a subset of **N** on which w looks anything but random.

By permitting too many patterns, we in effect commit the by now familiar post hoc fallacy of randomness, i.e., we concoct patterns to test the randomness of an object after the object has already been presented. In the preceding example, to obtain the limit in equation (4.4) we needed to constructed S on the basis of the purportedly random object itselfw. This, as we have observed, is analogous to the old statistical fallacy of selecting statistical hypotheses after the experiment is over and its results have been examined. Such a methodology is always disingenuous.

Because von Mises's original idea could not be made to work, attempts were made to salvage it. The obvious move was to restrict the subsets S for which w had to satisfy (4.2). Thus it was suggested that (4.2) be required only for the infinite subsets of **N** that were recursively enumerable (r.e.) (cf. Church, 1940). Since there are only countably many programs to generate these sets, the collection of r.e. sets itself is countable. Moreover, measure theoretic considerations imply that almost every candidate w satisfies (4.2) for all S's in such a collection.12 Thus the patterns induced by the infinite r.e. sets leave plenty of infinite sequences that are random with respect to this countable pattern space.

While this example illustrates the theory of randomness I am after, it is not the best advertisement for my theory. The problem with infinite random sequences is that they remain random irrespective of their finite initial segments. Thus for an infinite sequence of 0's and 1's, one can change the first 101000 entries all to 0 without affecting the randomness of the string. The randomness of an infinite string can only be ascertained by taking into account the entire limiting behavior of the string. This is bad news for anyone interested in the practical applications of randomness. Thus in the sequel I shall concentrate on randomness in finitary contexts.

So what should a theory of randomness look like? Certainly we must start with a collection of potentially random objects, the candidate space . Next we must find a pattern space P with respect to which the objects in can be random. P is both straightforward and problematic. P is straightforward because its patterns enable us quickly to decide whether a purportedly random object fits the pattern or not (on this view the patterns reduce to binary partitions of). P is problematic because its patterns must be selected according to a rationale which justifies calling the elements of /P random. Set theoretic considerations enter here: P must be big enough and small enough. It must be small enough to keep /P from being emptyP can always be augmented to make /P empty. On the other hand, if P¢ includes P, and if /P¢ is nonempty, then P¢ is preferable P. Thus P must contain all the patterns which random objects cannot legitimately fail to break.

5. Randomness in Practise

Randomness as the systematic breaking of fixed patterns has been implicit in past research. Just before introducing his computational complexity approach to randomness, Kolmogorov (1965a) wrote a paper entitled "On Tables of Random Numbers," whose mathematical content was pure combinatorics. In this paper, Kolmogorov addressed the problem of constructing random numerical sequences of a fixed finite length. Having decided on a fixed length n (some positive natural number), he then proceeded systematically to rule out sequences which could not be random according to a certain frequentist criterion of randomness. These systematic exclusions constituted the patterns which the nonrandom sequences failed to violate. In this section I shall incorporate Kolmogorov's work on finite random sequences into the framework I am developing. My treatment will introduce simplifying assumptions that involve no loss of generality, but will also extend certain ideas implicit in Kolmogorov's original work.

Our candidate space is the collection of 2n sequences of 0's and 1's having length n. A candidate w is therefore a function from {1, 2, ..., n} into {0,1}. As with von Mises's collectives, our motivation for randomness is even distribution: the proportion of 0's and 1's for random candidates w should be about the same. Hence, insofar as the frequencies fail to be evenly distributed, patterns are matched and nonrandomness is evidenced. The totality of patterns that might interest us is induced by the collection S which comprises all the nonempty subsets of the indexing set for , i.e., the nonempty subsets of {1, 2, ..., n}. For any S in S the extent to which a candidate w is random corresponds to how close

(5.1)

is to 1/2. In expression (5.1) |S| denotes the cardinality of S (which is greater than zero because of how we defined S). Expression (5.1) is the proportion of 1's w has on the set S.

Now to require that expression (5.1) exactly equal 1/2 is too stringent a condition. If for example the cardinality of S is a prime other than 2, then no candidate w can be random with respect to Sexpression (5.1) could then never take the value 1/2. Thus we want (5.1) close to 1/2 while at the same time permitting some slack. We therefore fix a positive e and stipulate that a candidate w breaks the pattern prescribed by S if

13 (5.2)

These observations are at the root of Kolmogorov's (n,e)-random binary sequences.

A natural question now arises: Given n and e, for which subcollections of S and candidates of is inequality (5.2) satisfied? Really two questions are involved here: (1) Given a collection of S's, can we find a candidate w that satisfies (5.2) for each of these

S's? (2) Given w, for which S's is (5.2) satisfied? The first question asks if we can find a random object with respect to a preset collection of patterns. The second asks for the patterns which render a fixed candidate w random. The second question is new and does not arise in the work of Kolmogorov and his successors. Kolmogorov does address the first question, though from a limited perspective. Let us examine these questions in turn.

For a fixed collection of S's is there any candidate w that violates all the induced patterns and therefore is random? A number of constraints are struggling against each other. If e is bigger than 1/2, (5.2) is always satisfied and everything is random. Thus we shall want e less than 1/2. Once e is fixed it will generally be true that the number of sets S with respect to which a candidate w is random (i.e., breaks the pattern indicated in (5.2)) will increase with the sequence length n. But if e is too small, then we become guilty of requiring (5.1) to equal 1/2 (e too close to zero in inequality (5.2) is equivalent to expression (5.1) equaling 1/2 exactly).

Other constraints are less obvious. For instance, sets S whose cardinality is very small relative to n will generally be unsuitable for checking the randomness of a candidate. To take an extreme example, if S is a singleton (i.e., contains only one element), then expression (5.1) will be either 0 or 1 implying that for any reasonable e inequality (5.2) will be violated. Thus, with respect to S's that are singletons no candidate can be random. Within our framework, any pattern space P that includes at least one singleton has no random elements; in this case /P is empty.

For a more complicated example, consider sets S containing two elements. To simplify calculations let us assume that n is even (n = 2k) and let us restrict our attention to candidates w which have the same number of 0's and 1's (i.e., k). (These conditions can be eliminated without affecting our general conclusions.) We find that,

, (5.3)

sets S have 2 elements, (5.4)

sets S with 2 elements satisfy , (5.5)

k2 sets S with 2 elements satisfy , and (5.6)

= + k2. (5.7)

Thus for about half the sets S with two elements the frequencies are exactly correct (when (5.6) obtains), whereas for the other half the frequencies are completely off (when (5.5) obtains). Moreover, by a trivial inclusion-exclusion argument one can choose k such sets S (e.g., {1,2}, {1,3}, ..., {1,k}, and {1,k+1}) for which at least one of these sets will satisfy (5.5)regardless of candidate. In other words, one can find k patterns induced by sets S of cardinality 2 which render all candidates nonrandom. If we relax our initial assumptions, we observe that for arbitrary n and e < 1/2, we can find approximately n/2 sets S with 2 elements for which no candidate can be random (no candidate can violate all the induced patterns). Within our framework, for such a pattern space P, /P is empty.

The sets S in S which really interested Kolmogorov were those which, unlike the two preceding examples, included a substantial portion of the indexing set {1, 2, ..., n}. Such sets S were generated algorithmically, and tended to induce patterns one would like to see "genuinely random" sequences break. Thus the first S to be considered was the entire indexing set {1, 2, ..., n}any random object w should be evenly distributed within e on this set. Next, one should consider sets S containing alternate terms of the indexing set: {1, 3, 5, ..., 2[(n+1)/2] 1} and {2, 4, 6, ..., 2[n/2]} (brackets here indicate the greatest integer function). Kolmogorov found that by generating sets in this way he could get

(5.8)

sets in S for which at least one candidate w was random.14 Thus the number of patterns for which random objects exist is exponential in the sequence length n.15

With (5.8) Kolmogorov determined an upper bound on the number of patterns he could get away with and still obtain a random candidate. His algorithm fixed the patterns, (5.8) bounded the number of patterns, and with this information Kolmogorov proceeded to search for a random candidate. Our second question reverses all of this: given a fixed candidate w for what patterns (S's) is w random? Which pattern spaces P render w random? Kolmogorov failed to address this question. Nevertheless, it offers new insights into randomness and underscores the distinguished role permutations (and more generally group actions) play in any theory of randomness based on patterns.

To indicate why this second question is important consider the following example. Suppose the sequence

w = 0011100101 (5.9)

is an (n,e)-random sequence for n = 10 and e > 1/10. We find that on So = {1,2, ..., 10}, w is evenly distributed. On S1 = {1, 3, 5, 7, 9}, S2 = {2, 4, 6, 8, 10}, S3 = {1, 2, 3, 4, 5}, and S4 = {6, 7, 8, 9, 10} w is within 1/10 of being evenly distributed. Consider now the following permutations of the indexing set So = {1,2, ..., 10}:

s = (1 8)(2 10) (5.10)

t = (2 3)(5 6) (5.11)

s, for instance, permutes {1,2, ..., 10} by interchanging 1 and 8, as well as 2 and 10. If we now modify w by applying s and t, we find that the resulting sequence of 0's and 1's is anything but random:

wos = 1111100000 (5.12)

wot = 0101010101 (5.13)

On S3 and S4 the wos fails in the worst possible way to be evenly distributed; on S1 and S2 the same holds for wot. But the permutations which altered w also alter the sets (patterns) S1 through S4. Thus s transforms S3 and S4 into sS3 = {3, 4, 5, 8, 10} and sS4 = {1, 2, 6, 7, 9} on which wos is evenly distributed within 1/10, whereas t transforms S1 and S2 into tS1 = {1, 2, 6, 7, 9} and tS2 = {3, 4, 5, 8, 10} on which wot is evenly distributed within 1/10.

There is a lesson to be learned. Among 0-1 sequences of length 10 having the same number of 0's as 1's, wos is as nonrandom as they get. And yet with respect to some S's wos is just as random as w. In fact, whenever w is random with respect to S, wos is random with respect to sS, and wot is random with respect to tS. Randomness really depends on how one looks at things. Patterns So, S1, S2, S3, S4 are the sorts of patterns humans are comfortable with, to which our visual and perceptual apparatus resonates. We expect random sequences to be evenly distributed across such nice patterns. If on the other hand our perceptual apparatus were so configured that some permutation of these patterns appeared more natural (e.g., sSo, sS1, sS2, sS3, sS4), then our sense of randomness would be altered.16

6. The Role of Group Actions

Let me now summarize our work on randomness from an abstract point of view. We are given a collection of objects, the *candidate space* , where we want to find random objects. Randomness is understood as violating patterns. Generally there will be a collection comprising all conceivable patterns that might interest us (cf. S in the previous section). Let us refer to such a collection as a *complete pattern space* and denote it by F. While a complete pattern space will contain all patterns that might conceivably interest us, it will usually be so broad as to leave no room for randomnessevery candidate in is sure to fit some pattern in F so that no candidate can be random with respect to all of F. Thus typically /F is empty (if not, specify /F and your problems are over).

For this reason we shall normally want to consider *pattern spaces* P that are proper subsets of F. If we are confident that the pattern space P adequately captures what we want of randomness in , and if it is true that /P is nonempty, then our task reduces to specifying /P, i.e., finding those candidates w which violate all the patterns in P. In the last section Kolmogorov's algorithm for generating patterns provided just the pattern space P which Kolmogorov considered relevant to the randomness of finite 0-1 sequences. The bound given in expression (5.8) reflected how large P could be taken while keeping /P nonempty.

Although the complete pattern space F will be sure to contain all patterns of interest, generally it is not clear whether a given pattern space P will provide the "right" notion of randomness for a set purpose, much less a universally correct notion of randomness. Pattern spaces are not etched in stone. They do not come with a natural rank ordering enabling us to decide which pattern space offers "better" randomness than another. They do not come with flags which mark them as the true carriers of randomness. If for some reason P were etched in stone, then the only remaining task would be to delineate the members of /P. But since this is generally not the case, it is convenient to reverse the picture. Thus we may begin with a candidate w and ask for which patterns is w random. Denote the patterns in F for which w is random by F(w). Call this the *pattern space on* F *induced by* w. w violates all the patterns in F(w) and is a member (possibly the only one) of /F(w).

The obvious problem now is to relate the induced pattern spaces F(w) for various candidates w. This I believe is best accomplished by means of group actions. We consider the action of a group G on the candidate space . Let us represent the group G multiplicatively, denoting the identity element by e. By saying that G acts on , we mean that every element of the group induces a function from to itself such that

1) e is the identity transformation on .

2) for every g and h in G g(hw) = (gh)w, i.e., composition of the functions induced by G mirrors the group multiplication.

It is immediate from 1) and 2) that the induced functions are actually permutations (bijections) on .17

From our perspective the group action of G on becomes interesting when it in turn induces a group action on the complete pattern space F. To see that a group action on will induce a group action on patterns and pattern spaces, it is enough to note that an individual pattern p is ultimately just a subset of . Thus for a group element g in G it is natural to consider the pattern gp={ gw | w p }. The pattern spaces P and the complete pattern space F are of course composed of such patterns p. Thus for g in G and a pattern space P it makes sense to consider gP={ gp | p P }. Since F's distinguishing characteristic as a pattern space is its completenessit must contain all patterns conceivably relevant to randomnessthere is no problem in choosing F so large that it is closed under the group operation. Thus we may assume that for all g in G, and all p in F, gp is also in F. With this closure property G does indeed induce a group action on the complete pattern space F, and sends pattern spaces P to pattern spaces gP.

With a group G acting on both and F, it becomes possible to compare the randomness of candidates w and w¢ with respect to induced pattern spaces F(w) and F(w¢). If for instance w¢ is in the orbit of w (i.e., if there is some group element g for which gw = w¢), then we can ask how F(w), gF(w), and F(w¢) = F(gw) all compare. If G is transitive on (i.e., if any candidate can be accessed from any other candidate via the group action), then all candidates can be compared in this way. An interesting question is whether gF(w) equals F(gw). If so, then the randomness of w and that of w¢ = gw are entirely symmetricalthe patterns which w breaks to be random and those which w¢ breaks to be random are mirror images under the group action.

Note that this abstract account of group actions was implicit in Kolmogorov's example of finite random sequences described in the last section. There was the collection of 0-1 sequences having a fixed length n. The group acting on was the symmetric group on n characters, Sn, which serves as our G. An element g in G (= Sn) is of course just a bijection on {1, 2, ..., n}. Thus for g to induce a function on it must be interpreted as follows: g(w) = wog. In effect, g takes any sequence w of 0's and 1's and rearranges these 0's and 1's in a different order.

G also induces a group action on the complete pattern space S, which comprises the nonempty subsets S of {1, 2, ..., n}. Under the action of a group element g, S is sent to its natural image under the symmetric group, namely gS. Note that S in S is not itself a subset of the candidate space . But when such an S is used to pick out candidates w via inequality (5.2), S specifies a pattern on (i.e., subset of) , which we can denote by p(S). We find a perfect consistency in the way the group action transforms the elements S of S, and the way the action transforms the patterns induced by such S's: gp(S) = p(gS) for all g in G, i.e., the pattern induced by gS is just the pattern induced by S and translated by g.

This concludes our summary of randomness. I have described from an abstract point of view our theory of randomness as it currently stands. My aim has been to make explicit the unspoken intuitions motivating the examples in Sections 4 and 5. With this abstract exposition in hand, I want now to focus on group actions and argue that they can be used to extend our notion of randomness. A prime intuition for randomness is the idea of mixing. A fresh deck of cards, for instance, is not "random" until it has been thoroughly shuffled, i.e., until the cards have been adequately mixed.18 In ergodic theory one considers mixing transformations which take distinct events and so intertwine them that they become probabilistically independent.19 In both these examples probabilistic considerations come to the fore, making it impossible to speak of a given fixed object as random in the way I am proposing. But the intuitions here are strong, and it is worth considering how these intuitions can work for us.

Let us for the moment think of a group G as a bag of gadgets for mixing things up. For concreteness, one might imagine a collection of blenders. Some of the blenders are broken and do no effective mixing at all. Some can only chop and grate. Others can liquefy. But the blenders best at mixing are the industrial strength blenders which operate at 20,000 rpm. Similarly, the group elements of G will vary in how well they mix under a group action. For instance, the identity e will be utterly useless for mixing things up. Throughout these musings I disregard the actual objects G is mixing. In the end we shall want G to mix the candidate space . But for now I am interested in establishing objective criteria for how well the elements of G mix, independent of what space G is acting upon. Suppose this is the casesuppose we are able to rank the elements of G by how well they mix. Furthermore, let us assume that whatever we mean by mixing in G, this notion is well-defined and intuitively plausible. In particular, our intuitions for mixing and randomness should correspond. How then can we exploit the mixing properties inherent in G to extend our theory of randomness?

For concreteness, let us imagine a bounded function m from the group G into the nonnegative reals [0,) which takes on higher values as group elements become increasingly good at mixing. Thus for group elements g and h, if m(g)<m(h), then h is better at mixing than g. Since m models intuition, m attains its lowest value at m(e), and is symmetric with respect to group inversion, i.e., m(g)=m() for g in G. Let us call m a *mixing measure* on G.20 Since our intuitions about mixing and randomness correspond, we want to specify those elements h which are best at mixing, i.e., those h for which m(h) equals or is very close to

(6.1)

Observe that this supremum exists inasmuch as m is assumed to be bounded. With a mixing measure like m the problem of finding the best blenders in G, if you will, becomes a straightforward optimization problem.21

Suppose now we have solved the optimization problem and found an optimally mixing element of G, call it h. Suppose further that G is acting on the candidate space . Our task is to find a random element in (random taken in its intuitive sense with no explicit reference to patterns yet). How shall we do it? A naive first attempt might be to take an arbitrary candidate w, apply h to it, and call the result hw random. But this presents a problem: if w is (intuitively) nonrandom and if w¢equals(w), then hw¢ is just the nonrandom w. By this trick, any optimally mixing group element h has images under the group action which are nonrandom. Yet surprisingly, this trick indicates a way of using h to obtain random elements from . If we can find a candidate w that is intuitively as nonrandom as possible, and if we apply an optimally mixing group element (by symmetry both h and will do) to w, then we get a candidate w¢ which I claim is random.

Certainly applying h to an arbitrary candidate can produce nonrandomness, but why should applying the optimally mixing group element h to a definitely nonrandom candidate w yield a random hw? Applying an optimally mixing h to an arbitrary candidate can in effect undo whatever randomness (still speaking intuitively) was already in the candidate. But an optimally mixing h applied to a definitely nonrandom w must issue in a random candidate hw because h cannot undo any of w's randomness. In effect, mixing will take something ordered and render it confused, but may take something confused and render it intelligible. It is worth recalling the conclusion of that interdisciplinary conference on randomness: We know what randomness isn't, not what it is. If we know what randomness isn't, then we know some definite, prototypical instance of nonrandomness (epitomized in the candidate w). For such an instance its mixture with an optimally mixing transformation must be random.

Let us formulate these ideas within our framework: we are given the candidate space , the complete pattern space F, and a group action of G on which extends to F. Our task is to find a random object in . We find a prototypically nonrandom candidate w in often this is easy. Next we find an optimally mixing group element h in G. w is intuitively nonrandom, but formally random relative the induced pattern space F(w). On the other hand, hw as an optimal mixing of a nonrandom object is intuitively random, and at the same time formally random with respect to the translated pattern space hF(w) (which under suitable symmetry conditions of the group action on F can be just F(hw)).

It remains to spell out what we mean by an optimally mixing group element h in G. An example will help. Let be the candidate space of all 0-1 sequences having length 100 and having the same number of 0's as 1's (50 of each). Take the complete pattern space F to be all nonempty subsets of . Take G to be the symmetric group on 100 characters, S100. For g in G and w in , gw is the composition wog, which is just w with its indices rearranged. In fact, because each candidate has the same number of 0's as 1's, for any two candidates w and w¢ there is a group element g in G which takes w to w¢. Thus G is transitive on .

Next we must find a prototypically nonrandom object from . I suggest a sequence we have seen before (see Section 2, sequence (H)):

11111111111111111111111111111111111111111111111111

00000000000000000000000000000000000000000000000000.

Call this sequence of 50 1's followed by 50 0's, w. Whatever we mean by random elements of , w must certainly lie at the other end of the spectrum. Whether w is the most nonrandom sequence in , or whether some other candidate is more nonrandom, depends on criteria for judging nonrandomness which will be situation specific. I don't take this to be a problem inasmuch as acute cases of nonrandomness are obvious. In the present example a complexity approach à la Kolmogorov will offer one way of seeing that w is simple and therefore nonrandom. Since we know what randomness isn't, I take finding prototypically nonrandom elements to be the least of our problems.

This leaves us with having to find an optimally mixing element h in G. What will such an element look like and how shall we go about finding it? I leave a general theory of optimally mixing group elements for another time, but let me offer some heuristics for the present case. G (=S100) is by definition the set of all permutations on {1,2,3,,100}. Thus to think of G as mixing is to ask how its group elements mix this set. Since {1,2,3,,100} is the indexing set for the sequences in , it is plausible to connect randomness in with the mixing of {1,2,3,,100} by G.

Now there are many ways to understand permutations as mixing {1,2,3,,100}. Since permutations can be written as the product of transpositions, one may ask what is the minimal number of transpositions for representing an arbitrary permutation g. Let us call this minimal number t(g). The induced function t is bounded by 99 (= n1) on G,[22] takes values in the natural numbers, assumes its lowest value of 0 at the identity (t(e) = 0), and is symmetric with respect to inverses (t(g) = t()). For permutations different from the identity t is strictly positive. Thus one measure of how well h mixes is how close t(h) is to

(6.2)

t is a mixing measure, but not an effective one. Essentially, t makes sure that its optimally mixing elements move all the elements of {1,2,3,,100} to points other than themselves. Thus for the permutation h which sends i to i+1 mod 100 (i.e., which shifts all numbers less than 100 up 1 and takes 100 down to 0), t(h) will assume the supremum in (6.2).23 Under this h the transformed sequence hw is almost as nonrandom as the original w. hw is just

11111111111111111111111111111111111111111111111110

00000000000000000000000000000000000000000000000001.

A more promising approach to mixing is through a type of mixing measure I call an *explosive measure*. If a group acts on some structured set (like {1,2,3,,100} which is ordered, has a natural metric, etc.), it is natural to think of mixing as the breaking or exploding of this structure.24 For instance, {1,2,3,,100} possesses a metric structure d given by the absolute value of the difference: d(m,n) = |mn|. One can imagine a permutation g in G exploding the metric structure d if it takes m and n close together (resp. far apart) and sends them to numbers far apart (resp. close together), i.e., if d(m,n) is small (resp. large), then d(gm,gn) is large (resp. small). This explosive property can be captured by the following mixing measure:

(6.3)

which defines x for all g in G.25 x is minimal at the identity e and gets big precisely for those g which break the metric structure. An optimally mixing group element h according to this mixing measure is one which satisfies

(6.4)

Still other mixing measures can be proposed. On {1,2,3,,100} consider the metric d¢(m,n) = min(|mn|,100|mn|). This alternate metric on {1,2,3,,100} treats the natural numbers between 1 and 100 as evenly spaced points around a circle. With this metric 1 and 100 are adjacent. In equation (6.3), if we substitute d¢ for d we obtain an alternative mixing measure, which we can denote as z. Other modifications can be introduced as well. The group G may include a subset D which we definitely want to exclude from consideration as mixing elements. Thus in G (=S100) we may want to exclude permutations with certain cycle structures. In this case finding optimally mixing group elements in G entails finding suprema for t, x, and z over the reduced set GD.

It is evident that any weighted average (convex linear combination) of mixing measures on a given group is again a mixing measure. Thus we may combine the mixing measures t, x, and z into a super-mixing measure w1t+w2x+w3z, where the weights are positive real numbers summing to 1. Just how the weights should be chosen will depend on the relative importance of the measures t, x, and z to mixing, as well as the relative sizes of the mixing measures (x is always at least n2 n whereas t is never more than n). Having chosen the mixing measures, the weights, and the set D with care, we now search for h in G which satisfies

(6.5)

and thereby transforms an intuitively nonrandom w into an intuitively random hw. Of course, w will be formally random with respect to F(w), whereas hw will be formally random with respect to F(hw) = hF(w). This concludes the example.

In closing this section I want to say a word about constructing mixing measures, and more generally about criteria for optimally mixing group elements. My approach in the last example was strictly ad hocI imagined properties I thought optimally mixing group elements should possess for the given group G, and then constructed mixing measures to model these properties. Such mixing measures set up criteria for optimal mixing. How good these criteria are, how good they can be made, and how to implement these criteria computationally are questions I leave for another time. In the preceding example I have not even computed an optimally "explosive" h in line with (6.3). The solution to these problems is not straightforward and requires a deeper analysis than is possible in this expository paper. Still, I hope to have convinced the reader not only that groups can possess intrinsic mixing properties relevant to randomness, but also that these mixing properties can be effectively specified.

7. Philosophical Postscript

Whatever happened to von Neumann's allegation of sin? It has frankly lost its sting. Redefinition is always an effective way to alter moral strictures, and the present case is no exception. von Neumann's guilty conscience derived from a paradox: deterministic systems were to model random systems, and yet random systems insofar as they were modeled by deterministic systems could not by definition be random. In this paradox von Neumann conflated randomness and chance. With this identification the paradox is indeed unresolvable. But when randomness is redefined as the breaking of patterns, the paradox disappears. Questions of determinism, chance, and probability no longer enter. At issue now is whether an object exists and can be found that breaks the patterns.

Something like Kant's Copernican revolution is going on here. Certainly I don't mean to place this essay in the company of Kant's first *Critique*. But there is a parallel in the way Kant's revolution changed the relation between object and knowledge, and the way my redefinition changes the relation between random object and pattern. Prior to Kant knowledge had conformed to object with object causally influencing knowledge. But with Kant (1927, p. 22) objects must henceforth conform to knowledge. As Allison (1983, p. 30) observes,

The point to be emphasized is that this "changed point of view" brings with it a radically new conception of an object. An object is now to be understood as whatever conforms to our knowledge, and this ... means whatever conforms to the mind's conditions (both sensible and intellectual) for the representation of it as an object. Consequently, an object is by its very nature something represented....

Similarly, the random objects I advocate reflect a changed point of view. In times past random objects were random because they mimicked chance. Forgeries they were. As long as the counterfeit looked specious, one could pretend it was the product of chance. But the technology for uncovering these forgeries was always improving. The latest statistical test was ever threatening to expose the "well-established" random object. However, within the new framework, the "conditions for the possibility" of such objects, to use a Kantian phrase, henceforth rests with the patterns that render these objects random, and not with the objects themselves. Patterns become strictly prior to random objects. Without patterns, objects are just objects, not random objects.

REFERENCES

Allison, Henry E.

1983 *Kant's Transcendental Idealism*, (New Haven, Conn.: Yale University Press).

Bauer, Heinz

1981 *Probability Theory and Elements of Measure Theory*, (London: Academic Press).

Church, Alonzo

1940 "On the Concept of a Random Sequence," *Bulletin of the American Mathematical Society*, Vol. 46, 130-135.

Dembski, William A.

1990 "Uniform Probability," *Journal of Theoretical Probability*, In press.

Diaconis, Persi

1981 "On the Statistics of Vision: The Julesz Conjecture," *Journal of Mathematical Psychology*, Vol. 24, 112-138.

1988 *Group Representations in Probability and Statistics*, (Hayward, Calif.: Institute of Mathematical Statistics).

Garey, Michael R. and David S. Johnson

1979 *Computers and Intractability: A Guide to the Theory of NP-Completeness*, (New York: Freeman).

Goldreich, Oded, Shafi Goldwasser and Silvio Micali

1986 "How to Construct Random Functions," *Journal of the ACM*, Vol. 33, No. 4, 792-807.

Hungerford, Thomas W.

1974 *Algebra*, (New York: Springer-Verlag).

Kant, Immanuel

1929 *Critique of Pure Reason*, translated by N. K. Smith, (New York: St Martin's).

Knuth, Donald E.

1981 *Seminumerical Algorithms*, 2nd ed., in *The Art of Computer Programming*, Vol. 2, (Reading: Addison-Wesley).

Kolmogorov, Andrei N.

1950 *Foundations of the Theory of Probability*, (New York: Chelsea).

1965a "On Tables of Random Numbers," *Sankhya* (*The Indian Journal of Statistics*): Series A, Vol. 25, No. 4, 369-376.

1965b "Three Approaches to the Quantitative Definition of Information," *Problemy Peredachi Informatsii* (in translation): Vol. 1, No. 1, 3-11.

Kranakis, Evangelos

1986 *Primality and Cryptography*, (Stuttgart: Wiley-Teubner).

Lasota, Andrzej and Michael C. Mackey

1985 *Probabilistic Properties of Deterministic Systems*, (Cambridge: Cambridge University Press).

Martin-Löf, Per

1966a "The Definition of Random Sequences," *Information and Control*: Vol. 9, 600-619.

1966b "Algorithmen und zufällige Folgen," four lectures delivered at the Mathematical Institute of the Erlangen-Nürnberg University.

Mises, Richard von

1936 *Wahrscheinlichkeit, Statistik und Wahrheit*, (Vienna: Springer-Verlag).

Parthasarathy, K. R.

1967 *Probability Measures on Metric Spaces*, (New York: Academic Press).

Weihrauch, Kurt

1987 *Computability*, (Berlin: Springer-Verlag).

Wilder-Smith, A. E.

1975 *Man's Origin, Man's Destiny*, (Minneapolis, Minn.: Bethany House).

Yao, Andrew C.

1982 "Theory and Applications of Trapdoor Functions," *Twenty-third IEEE Symposium on Foundations of Computer Science* (FOCS), 80-91.

Department of Philosophy (M/C 267)

1524 University Hall

University of Illinois at Chicago

Box 4348

Chicago, Illinois 60680

NOTES

[1] Quoted in Knuth (1981, p. 1). It is surprising how this almost flippant remark has been elevated to a dogma. In addition to its canonical status, this remark functions as one of the computer scientist's stock inside jokes.

[2] These comments derive from the Interdisciplinary Conference on Randomness at Ohio State University, 11-16 April 1988. This event was significant for assembling philosophers, mathematicians, psychologists, computer scientists, physicists, and statisticians to share their insights into randomness. In referring to this event I shall use the initials *ICR*.

[3] The wording here may strike the reader as unnatural, for violating a pattern is equated with passing a statistical test. The two notions do in fact correspond: the passing of a statistical test is the normal, expected outcome; only when something unusual is going on do we expect a statistical test to fail. For a chance event to fit a pattern is unusual; any pattern is thought to be sufficiently restrictive that breaking the pattern constitutes the normal, expected outcome.

[4] Precisely because statistical tests abound and can disqualify any supposedly random string, von Mises unqualified notion of collective founderedno infinite sequence maintains the right frequencies across all subsequences. See von Mises (1936).

[5] For the strong law of large numbers see Bauer (1981, p. 172); for an unconventional look at Huxley's simians see Wilder-Smith (1975, p. 63).

[6] This example inspires a massive revision of the criminal justice system: with the requirement that all coin flips be fair and duly recorded, sentence a convicted criminal to serve time in prison until he flips n heads in a row, where n is selected according to the severity of the offence. Thus for a 10 year prison sentence, if we assume that the prisoner can flip a coin once every five seconds (this seems reasonable), eight hours a day, six days a week, and given that the average attempt at getting a streak of heads before tails is 2 (= S1i i2-i), then he will on average attempt to get a string of n heads once every 10 seconds, or 6 attempts a minute, or 360 attempts an hour, or 2880 attempts in an eight hour work day, or 901440 attempts a year (assuming a six day work week), or approximately 9 million attempts in 10 years. 9 million is approximately 223. Thus if we required of a prisoner that he flip 23 heads in a row before being released, we could expect to see him out in approximately 10 years. Of course specific instances would varysome prisoners being released after only a short stay, others never recording the elusive 23 heads!

[7] There are some deep isomorphism theorems about Polish spaces, of which the space that models coin tossing is a key example. Most of modern probability theory can be fitted into the abstract framework provided by Polish spaces. The reason coin tossing is fundamental is that all Polish spaces are (Borel) isomorphic to one another, and hence to the space that models coin tossing. See Parthasarathy (1967, pp. 7-15).

[8] It is significant that no lawyer in his right mind would defend such a lottery manager by appealing to the infinitesimally small probability of "things just happening that way."

[9] This is really an appeal to Church's thesis, i.e., the claim that intuitive and mathematical computability coincide. See Weihrauch (1987, p. 87).

[10] See Knuth (1981, p. 27) for his generally glowing remarks about the additive number generator. Marsaglia's disaffection with this generator was voiced at *ICR*.

[11] By deterministic methods I mean methods which are obviously deterministic, like running a computer program. Coin tossing is deterministic in the sense that Newtonian mechanics offers precise and accurate prediction. Nevertheless, I take coin tossing to be the paradigm for chance and ignore any underlying determinism.

[12] I am assuming the standard probabilistic model for coin tossing: the infinite product space of {0,1} together with the uniform product measure.

[13] It may seem counterintuitive to speak of w as breaking the pattern induced by S if this inequality is satisfied. Nevertheless, the underlying intuition derives from the probability of coin tossing which dictates that w should be evenly distributed if it is random. Since we have defined randomness as the breaking of patterns, for w to satisfy inequality (5.2) must therefore be identified with the breaking of a pattern. This point is strictly a question of terminology. See also note 3.

[14] The number in (5.8) is essentially the reciprocal of the probability bound in Bernstein's law of large numbers, a sharp combinatorial inequality arising from the binomial distributionsee Kranakis (1986, p. 94).

[15] Compare this with the time-complexity approach to randomness for which polynomial-time functions are insufficient to distinguish pseudo-randomness from genuine randomness. In the present example a potentially random sequence of length n must be checked against a collection of patterns whose cardinality is exponential in n, not merely polynomial in n.

[16] I should stress that I am after a mathematical, not a perceptual, theory of randomness. Still, there are parallelssee Diaconis (1981).

[17] See Hungerford (1974, pp. 88-92) for more details.

[18] The statistician Persi Diaconis, a key organizer of *ICR*, has done significant work in the area of group actions and randomness. As both a professional magician and statistician, he has obtained results in the mathematics of card shuffling (which is nothing but a group action in disguise) which has recently brought him and his colleague Dave Bayer to the public eye (cf. *Time Magazine*, 22Jan1990, p. 62). Their general finding was that 7 shuffles are necessary to take a nonrandom deck to a random state. See Diaconis (1988) for his general approach to randomness via groups. Let me stress that his approach is fundamentally probabilistic.

[19] See Mackey and Lasota (1985, pp. 63-65) for some striking computer generated pictures that reinforce the abstract intuitions motivating ergodic theory.

[20] Mixing measures are not measures in the sense of countably additive set functions. Rather, they are functions on a group whose extrema provide optimally mixing group elements.

[21] At least conceptually such optimization problems are straightforward. In practise they can prove tricky.

[22] This follows directly from the cycle-structure decomposition of permutations. See Hungerford (1974, pp. 46-51).

[23] The permutation (1 2 3 100) can be expressed most briefly as the product of the following 99 transpositions: (1 2)(1 3)(1 4)(1 100).

[24] This is clearly reminiscent of pattern breaking in randomness, but there are some differences.

[25] This summation has an integral formulation for compact metric spaces using (semi-) uniform probabilities. See Dembski (1990) for the appropriate measure to use in the integration.

Copyright/Reproduction Limitations

Christian Leadership Ministries
P.O. Box 129
4901 Keller Springs Road
Suite 106
Addison, TX 75001
Phone: 972-713-7130
Fax: 972-713-7670
Email: lu@clm.org