Association for Computing Machinery
Advancing Computing as a Science & Profession

Ubiquity, an ACM publication
January, 2011

Computation, Uncertainty and Risk

Jeffrey P. Buzen

In this eleventh piece to the Ubiquity symposium discussing What is computation? Jeffrey P. Buzen develops a new computational model for representing computations that arise when deterministic algorithms process workloads whose detailed structure is uncertain.

Peter J. Denning
Editor

Association for Computing Machinery
Advancing Computing as a Science & Profession

Ubiquity, an ACM publication
January, 2011

Computation, Uncertainty and Risk

Jeffrey P. Buzen

1.Introduction

Computation can be viewed from a variety of perspectives. For many observers, computation is simply the process that underlies the operation of deterministic algorithms (i.e., algorithms that always produce the same result when provided with a given set of input values). Classical Turing machines and finite state automata are formal abstractions that capture the essence of this perspective.

In contrast to this purely deterministic point of view, analysts who study the performance of computer systems and communication networks must account for variability and uncertainty in the workloads that drive these systems. The unpredictable nature of such workloads is evident to anyone who has examined the flow of packets through a communication network, the arrival of transactions at a web server, or the competition for resources among active processes running within a computer system.

One way to inject variability and uncertainty into Turing's original formalism is to assume that the operation of the algorithm itself is unpredictable. Probabilistic Turing machines are capable of representing algorithms that operate in this manner. With computational models of this type, the same set of input values can generate different results each time the associated non-deterministic algorithm is executed.

Probabilistic Turing machines can be used to investigate a number of fundamentally important questions. However, the unpredictable behavior associated with real world systems is seldom the result of non-deterministic algorithms. In fact, the step-by-step operation of real world systems is usually controlled by algorithms that operate in a purely deterministic manner. When unpredictable behavior arises, it is typically due to a lack of certainty regarding the detailed properties of workloads that drive these systems.

This paper presents a computational model that preserves the deterministic core of Turing's original formalism, assigns the source of uncertainty to workloads rather than algorithms, and characterizes uncertainty through assumptions that are based entirely on observable quantities. The result is a formal entity that will be referred to as a loosely constrained deterministic system (LCDS). The observable phenomena that are associated with an LCDS can be classified as computation even though detailed properties of these phenomena remain completely uncertain.

2. Loosely Constrained Deterministic Systems

Essentially, loosely constrained deterministic systems represent the behavior of systems, algorithms and computations as the product of two separate elements, one deterministic and the other non-deterministic. The deterministic element is represented by a conventional finite state machine (finite state automaton) and its associated state transition diagram. The non-deterministic element is represented by a set of “loose constraints” on the workloads (input streams) that are processed by the finite state machine.

To understand the rationale behind loose constraints, suppose that a conventional finite state machine is being used to model the behavior of a system, an algorithm or a computation. In cases where the detailed structure of the workload (input stream) driving this finite state machine is known, it is possible to reproduce the entire trajectory (sequence of states that the finite state machine passes through) with complete accuracy. Such models can be characterized as being “tightly constrained.”

In other cases, information concerning the detailed step-by-step structure of the workload may be impossible to obtain or may simply be missing. Although the detailed structure is unknown, statistics that summarize or aggregate certain observable properties of the workload may be readily available. For example, it is often reasonable to assume that the average values of certain key quantities are known. Variances, correlations, percentiles, and other statistics may also be available.

The observed or predicted values of such summary statistics can be regarded as “loose constraints” on the structure of workloads that drive finite state machines. Clearly, a large number of workloads that differ in their detailed properties can share the same set of summary statistics and thus satisfy the same set of loose constraints.

When only loose constraints are specified, the structure of the resulting trajectories cannot be reproduced with complete accuracy. Nevertheless, mathematical expressions that characterize important properties of these trajectories can still be derived. The correctness of the derivations can be rigorously established despite the uncertainty surrounding detailed properties of the trajectories. Moreover, the derivations do not require the assumption that random forces have generated the workload being processed by the finite state machine. Thus, loosely constrained deterministic systems differ from conventional stochastic models and from probabilistic Turing machines in their approach to representing uncertainty.

The next few sections illustrate these concepts using an especially simple example: a random walk in one dimension. The example is first analyzed using traditional methods. The standard solution is derived, and its application to the original problem is examined.

The random walk is then reformulated as a loosely constrained deterministic system. This makes it possible to characterize uncertainty using a concept known as empirical independence [BUZE09]. The discussion demonstrates that empirical independence is intuitively plausible, easily verifiable, and sufficiently powerful to generate the same solution that is obtained using traditional probabilistic analysis

Although the specific details of this particular example are not important, the nature of the assumptions that are employed and the solutions that are obtained are of general interest since they (or their variants) are common to all loosely constrained deterministic systems. A number of generalizations and extensions are discussed in subsequent sections of this paper. Applications to Monte Carlo simulation and to the analysis of risk are also noted.

3. A Simple Random Walk

Random walks, which are discussed in many texts on probability theory, provide a useful starting point for comparing conventional stochastic models with models that are based on the concept of a loosely constrained deterministic system.

For the specific example being considered here, assume that a walker travels along a route marked by four stations that are positioned along a straight line as shown in Figure 1. As the walk proceeds, the walker departs from the current station, turns left or right, and moves to the next station in the path.

To prevent the walker from disappearing off the end of the path, assume there are “reflecting barriers” at each end. Thus a walker exiting from station 3 and moving to the right encounters a reflecting barrier that directs him back to station 3. Similarly, a walker exiting from station 0 and moving to the left encounters a reflecting barrier that directs him back to station 0.

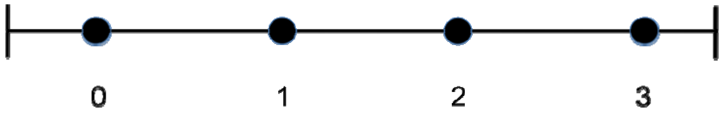

Figure 1 – Random Walk with Four Stations

Suppose we are interested in the walker’s path from station to station. We are not interested in how much time the walker spends at each station, or how much time it takes to travel between stations. Instead, we are concerned only with the sequence of stations the walker visits during the journey. This sequence will be referred to as the walker’s trajectory. The trajectory is clearly a function of the starting station (0, 1, 2 or 3) and the sequence of left and right turns the walker makes.

One of the simplest questions we can now consider is the relative number of times the walker visits each station during the course of a trajectory. For example, if right turns are twice as common as left turns, what is the relative frequency of visits to stations 0, 1, 2 and 3?

To begin an analysis of this question, note that the transitions that take place during the walk can be represented by the state transition diagram shown in Figure 2. The circles represent the four stations, and the arrows represent the transitions that can take place during a single segment of the walk. Even though we are analyzing a “random” walk, the diagram in Figure 2 is clearly deterministic. Most systems that are analyzed using stochastic models have a deterministic foundation that can be specified using a diagram of this type.

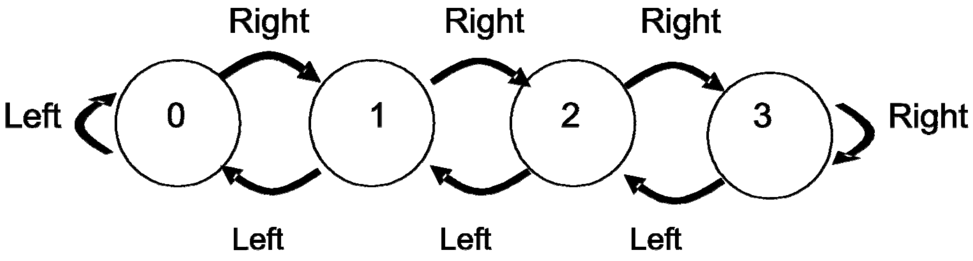

Figure 2 – Deterministic State Transition Diagram

The next step is to consider the nature of the workload that is driving this system. In this case, a workload is simply a series of left and right turns that can be represented by a sequence of the letters R and L. Once such sequence where right turns are twice as common as left turns is RRLRRLRRLRRLRRL. When this workload is used to drive the model shown in Figure 2 (with the walk beginning at station 2), the result is the trajectory shown in Figure 3.

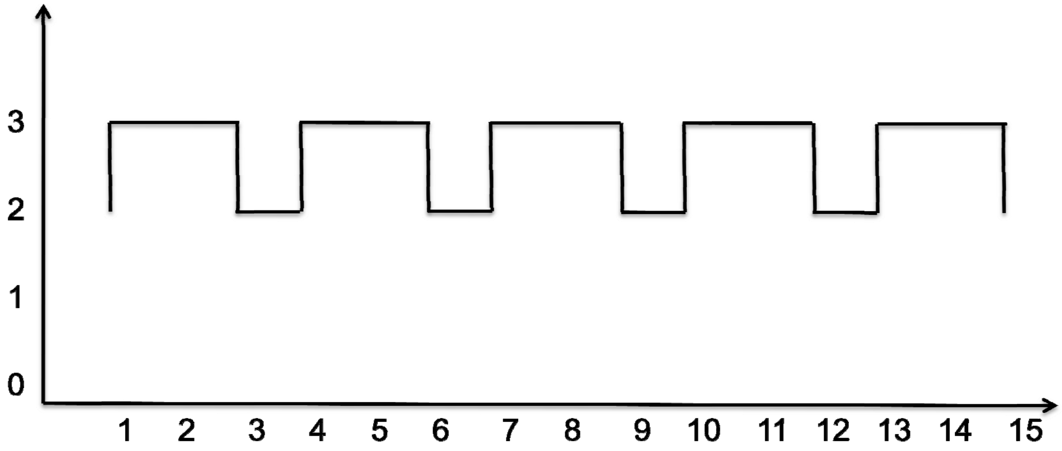

Figure 3 – Trajectory for Workload 1

Note that the walker simply cycles between stations 2 and 3, with 2/3 of the visits passing through station 3 and 1/3 of the visits passing through station 2. Stations 0 and 1 are never visited.

Although this analysis is clearly correct, it is not particularly interesting. The main problem is that the workload in this example is neither variable nor unpredictable. To resolve this issue, randomness and uncertainty must be incorporated into the specification of the workload.

One way to represent randomness is to assume that the walker tosses a coin before leaving each station. If the coin comes up "heads", the walker exits to the right; otherwise, to the left. Assume that the probability of getting a head and exiting to the right is equal to r on each toss. The coin is not required to be perfectly balanced, so r is not necessarily equal to ½ .

The probability of the walker being located at stations 0, 1, 2 or 3 after completing N segments of the walk can now be expressed as a function of r. Of course, these probabilities also depend on the station from which the walk originates. However, the influence of the starting station diminishes rapidly as N increases**.** In the limit as N approaches infinity, the influence of the starting station disappears entirely and the probability of the walker being in each station becomes independent of N.

These limiting probabilities are referred to as "steady state' or "stationary" probabilities. Let P(n) be the steady state probability that the walker is at station n (for n = 0, 1, 2 and 3). As discussed in the Appendix, the values of P(n) are given by the solution to the following set of "balance equations".

$$P(0) = (1-r)\ P(0) + (1-r)\ P(1) \qquad (1)$$

$$P(1) = r\ P(0) + (1-r)\ P(2) \qquad (2)$$

$$P(2) = r\ P(1) + (1-r)\ P(3) \qquad (3)$$

$$P(3) = r\ P(2) + r\ P(3) \qquad (4)$$

Combining these equations with the requirement that the sum of all four values of P(n) must be equal to 1.0 yields the following solution:

$$P(n) = P(0) \times [r/(1-r)]^n \text{ for } n = 0, 1, 2 \text{ and } 3 \qquad (5)$$

where $P(0) = 1 / [1 + r/(1-r) + [r/(1-r)]^2 + [r/(1-r)]^3]$ (6)

4. Application of the Solution

Equations (5) and (6) represent the classical solution for a random walk in one dimension with reflecting barriers. Although the solution is mathematically correct, it does not really answer the original question regarding the proportion of visits to each station in a (long) trajectory. Instead, this original problem has been transformed into a related question regarding the steady state distribution of a stochastic process that is being used as a model of the random walk.

The two solutions are clearly linked: it is intuitively reasonable to assume that the probability of finding the walker at some specific station in the distant future is equal to the relative frequency of visits to that station over the course of the trajectory. A formal proof, which requires the Ergodic Theorem for the most general case, need not concern us at this time.

Let us turn instead to a set of more practical concerns. Specifically, consider how the mathematical solution specified by equations (5) and (6) is typically applied in practice. Imagine observing a walker who is moving from station to station along the walking path depicted in Figure 1. If the walk continues for a long time and all the assumptions of the underlying stochastic model are satisfied, the attained proportion of right turns should be equal to r, the probability of a “head” in the stochastic model (based on coin tosses). By analyzing the trajectory generated by the walker, we can determine the proportion of visits to states 0, 1, 2 and 3. These attained proportions should be equal to the steady state probability distribution {P(0), P(1), P(2) and P(3)}.

Applying this reasoning to the workload that generated the trajectory shown in Figure 3, we find that r, the attained proportion of time the walker turns to the right, is equal to 2/3. If the model is accurate for this case, we should expect the proportion of time the walker exits from stations 0, 1, 2 and 3 to be given by equations (5) and (6) with r = 2/3. In other words,

$$P(0) = 1/15 \tag{7}$$

$$P(1) = 2/15 \tag{8}$$

$$P(2) = 4/15 \tag{9}$$

$$P(3) = 8/15 \tag{10}$$

For the trajectory shown in Figure 3, these predictions are grossly inaccurate. As already noted, the actual proportion of visits to stations 0 and 1 is zero, while the proportion of visits to stations 2 and 3 are1/3 and 2/3 respectively.

This finding should come as no surprise. The workload that is driving the state transition diagram in Figure 2 certainly does not appear to be "random." In addition, the trajectory is much too short to expect the attained distribution (extracted from the trajectory) to match the steady state distribution (from the underlying stochastic process).

Now consider a second workload, Workload 2, which is slightly longer than the workload that generated the trajectory in Figure 3.

RRRRRRRRRRRRRRRRRLRLRLRLRLRLRLRLLRLRLRLLLRLRR

Of the 45 turns in Workload 2, 30 are to the right. Thus the proportion of time the walker turns to the right upon exiting from a station is once again equal to 2/3. Starting again from station 2 and following the deterministic process defined by the diagram in Figure 2, the walker now traces out the following trajectory:

2333333333333333332323232323232321212121001012

Extracting the attained distribution from this trajectory yields:

Proportion of time the walker exits from station 0 = 3/45 = 1/15

Proportion of time the walker exits from station 1 = 6/45 = 2/15

Proportion of time the walker exits from station 2 = 12/45 = 4/15

Proportion of time the walker exits from station 3 = 24/45 = 8/15

This is a puzzling result. Workload 2 does not appear to have been generated by a sequence of independent coin tosses [with probability of heads equal to 2/3 on each toss]. Nevertheless, the attained distribution is identical to the steady state distribution predicted by equations (5) and (6).

5. Variability, Uncertainty and Unpredictability

The success of equations (5) and (6) for the trajectory generated by Workload 2 is due to a specific set of properties that are exhibited by this workload. These properties reflect a representation of uncertainty that does not pre-suppose the existence of an underlying random variable, but instead relies on directly observable properties of workloads and the trajectories they generate. Before delving further into these technical issues, it is useful to step back and consider certain aspects of uncertainty that are purely intuitive in nature.

Whether or not an outcome is uncertain depends on how much the observer knows. For example, each successive value produced by a conventional random number generator may appear uncertain to an observer who knows nothing about its internal workings, but will be entirely predictable to another observer who knows the algorithm implemented by the random number generator and the starting seed. In other words, uncertainty is a relative - rather than absolute - concept.

For the random walk we have just been analyzing, there is an implicit expectation of uncertainty regarding the next station the walker will visit. However, if an observer knows the starting state and the precise structure of the workload, the state transition diagram in Figure 2 can clearly be used to predict the walker's next destination at each segment of the walk.

As noted earlier, observers of real world systems seldom have precise knowledge of the detailed structure of the workloads driving these systems. Nevertheless, observers are often able to determine the values of summary statistics that characterize the workloads: means, variances, correlation coefficients, etc. The challenge for analysts is to construct mathematical models that incorporate such summary statistics as symbolic variables while also allowing other less critical details to remain uncertain.

Conventionally, random variables are employed to achieve this goal. The analyst simply assumes the quantities that characterize the workload (right and left turns in this case) can be regarded as a series of independent samples from an underlying random variable. The parameters of this random variable can then be identified with summary statistics that the analyst is able to observe or estimate. (e.g., the proportion of right and left turns in a workload).

Given this assumption, the state of the system at any given instant (i.e., the current location of the walker) becomes a random variable whose distribution is a function of the starting state and the random variable that characterizes the workload. As shown in the Appendix, powerful mathematical tools can then be employed to derive equations that characterize the state of such systems under limiting conditions. Equations (5) and (6) are representative of the type of result that can be derived.

One problem with the conventional approach is that it begins with an unverifiable leap of faith: the assumption that the quantities that characterize a real world workload can be regarded as samples from underlying random variables. There is no way to prove definitively that this assumption is either correct or incorrect by observing the workload or its associated trajectory.

To develop an alternative characterization of uncertainty, consider a long workload that has been generated by an idealized coin tossing process. The number of detailed mathematical properties that can be extracted from such a workload and its associated trajectory is essentially unlimited. However, a small subset of these properties may actually be sufficient to ensure that trajectory's attained distribution satisfies the steady state distribution given by equations (5) and (6). Workloads that satisfy this subset of properties will be "random enough" to exhibit the "correct" attained state distribution. These same workloads may fail most conventional tests used to detect the presence of "pure randomness".

To identify this subset, note first that the idealized coin tossing process incorporates the assumption that the probability of "heads" on each toss is equal to r. This assumption is clearly intended to be valid regardless of the walker's current location or past history. Thus, if we restrict our attention to those turns made upon exiting from station 0, it will still be true that the probability of turning right is equal to r. Obviously, the same conclusion also applies to turns made upon exiting from stations 1, 2 and 3.

Analysts who are concerned only with observable properties of the workload and the trajectory cannot "see" these probabilities. However, they will be able to calculate the proportion of time the walker turns to the right after exiting from stations 0, 1, 2 and 3. If the walker's movements are controlled by a random mechanism that is independent of the walker's current location, these four observable proportions will, in the limit, be identical to one another.

This expectation is, in fact, realized by Workload 2. Recall that Workload 2 has the following form:

RRRRRRRRRRRRRRRRRLRLRLRLRLRLRLLRLRLRLLLRLRR

As already noted, the trajectory shown below will be generated when this workload is processed by the finite state machine shown in Figure 2 (with the walker starting from station 2).

2333333333333333332323232323232121212121001012

Direct inspection of this trajectory shows the walker exits from station 3 a total of 24 times. Similarly, the walker makes 12 exits from station 2, 6 exits from station 1, and 3 exits from station 0. For each of these stations, the proportion of right turns the walker makes upon exiting is exactly equal to 2/3. This finding is entirely consistent with the assumption that random coin tosses are controlling the walker's trajectory. However, this assumption expresses a directly observable relationship that is easy to verify.

Surprisingly, this simple relationship provides the key to proving that equations (5) and (6) must be satisfied by the trajectory generated by Workload 2. To complete the proof, it is also necessary to assume the trajectory ends with the walker in the same station he occupied at the start of the walk (station 2 in this case). This second assumption can be relaxed without materially affecting the conclusion. See the Appendix for further discussion of these points.

6. Empirical Independence

When a random walk is modeled as a stochastic process, the position of the walker at any instant is characterized by a random variable. The direction the walker turns after exiting from each station is also characterized by a random variable. Since each coin toss is assumed to be an independent event, the random variables representing the walker's current position and the direction of the next turn must be statistically independent.

Within the framework of an LCDS model, the position of the walker at any instant is a specific value: the walker is either at station 0, 1, 2 or 3. Similarly, the direction of the walker's next turn also has a specific value: either right or left. Random variables are not involved.

Since random variables are not part of the specification of LCDS models, the concept of statistical independence is not directly applicable. However, there is still a simple way to express the idea that the direction of the walker's next turn (left or right) is independent of his current location (station 0, 1, 2 or 3). For each location, simply compute r_j, the proportion of right turns the walker makes upon exiting from station j. If the four values of r_j are all equal (as they are in the case of Workload 2), we will say that the walker's next turn is "empirically independent" of his current position.

The term empirical independence has been chosen to emphasize that this relationship is closely related to, but still distinct from, the traditional concept of statistical independence. Empirical independence is a relationship among observable properties of workloads and trajectories. In contrast to statistical independence, its definition does not pre-suppose the existence of random variables.

Although it may not be readily apparent, empirical independence plays a central role in modeling systems whose behavior incorporates properties that are uncertain and unpredictable. In these cases, the assumption that certain observable properties are empirically independent of the state of the system (or empirically independent of some subset of system states) leads to equations for the attained distribution that have the same form as the equations that characterize the steady state distribution of the corresponding stochastic process.

7. Loosely Constrained Deterministic Systems: Review

The particular LCDS that has been presented here can be generalized to model a wide range of systems. The first step is to recognize that the operation of most real world systems is regulated by deterministic algorithms that can be represented using finite state machines and their associated state transition diagrams.

The second step is to characterize the workloads that drive these finite state machines through the use of summary statistics (means, variances, etc.). In effect, these summary statistics represent loose constraints on the family of workloads that lie within the scope of the analysis. Equations derived during the analysis will apply to all workloads that satisfy these constraints.

The third step, which is not always necessary, is to introduce additional constraints that reflect assumptions about uncertainty and randomness in specific aspects of system behavior. These constraints are generally expressed in terms of empirical independence. In particular, symbolic variables that represent observable properties of workloads and trajectories are assumed to be empirically independent of certain states the system may be in. Such constraints can also be characterized as “loose” since they can be satisfied by multiple workloads.

The abstract model characterized by steps 1, 2 and 3 is referred to as a loosely constrained deterministic system. The dynamic action sequence evoked by a workload in such a system is called a trajectory of the system. A trajectory is analogous to a computation evoked by an algorithm on a computer. The LCDS model assumes that every trajectory satisfies the model’s constraints. Thus, if we say that the system has mean network delay of 1 millisecond, we are constraining the model to only those trajectories for which the mean network delay is measured as 1 millisecond.

The LCDS is based upon the formal representation of deterministic computation developed by Turing and his successors, but also incorporates characterizations of uncertainty that are expressed as observable properties of individual trajectories. In contrast to probabilistic characterizations of uncertainty, this representation yields results that can be applied with absolute certainty to every trajectory that satisfies the associated set of loose constraints. We will discuss this again in a later section.

8. Shaped Simulation

Monte Carlo simulation is a well established and widely used technique for the numerical evaluation of stochastic models. LCDS models, which provide an alternative to stochastic models, have a number of implications for analysts who employ this technique.

For example, suppose the random walk we have been discussing here is being evaluated using Monte Carlo simulation. This requires a simulation program that implements the deterministic behavior depicted in Figure 2. Of course, the program must also call upon a random number generator to regulate the direction the walker will turn each time he exits from a station. The simulation can be regarded as a realization of an underlying stochastic process. The goal of the simulation is to evaluate certain properties of that process: in this case, the goal is to evaluate the process's steady state distribution.

Once the simulation has been initialized by assigning a specific numerical value to r, it must be run until it has converged to the "correct" answer. Deciding exactly when convergence has occurred is a difficult problem.

LCDS models shed a new light on this problem. Suppose the simulation is modified to keep track of the individual values of r_j as they vary over the course of the simulation. As already noted, these values can all be expected to converge to r in the limit. However, they may all be equal to r at any time during the course of the simulation. At any such point, the trajectory traced out by the simulation is guaranteed to have produced the correct steady state distribution, provided the initial and final states are the same. This fact, which follows directly from the analysis of the corresponding LCDS model, provides a new rationale for deciding when a Monte Carlo simulation has run "long enough".

Requiring that the initial and final states of a trajectory be the same is analogous to one of the requirements that must be satisfied for a simulation to reach a regeneration point [ALTI07]. In both cases, this objective eliminates "end effects" that complicate the analysis by creating imbalances. However, equality of the initial and final states is a substantially weaker condition since any state can serve as the initial and final state of the trajectory.

Suppose the simulation has run for a while without reaching a point where all the values of r_j are equal to r. In principle, it is possible to put aside the random number generator at this point and replace it by an adaptive algorithm that selects left and right turns with the explicit goal of forcing all the values of r_j to become equal to r. Such an algorithm will clearly invalidate the assumptions of the stochastic model that underlies the simulation. Nevertheless, it can also cause the simulation program to generate a trajectory that yields the exactly correct solution to the original problem (i.e., determining the steady state distribution of the underlying stochastic process for the specified value of r). This approach, which is referred to as shaped simulation [BUZE10], can be applied to any discrete time or continuous time Markov process. Preliminary studies have demonstrated it is capable of dramatically reducing the execution time of a simulation program (up to 98%) while also providing guarantees of accuracy that are not normally available.

9. Distributional and Trans-Distributional Properties

Although shaped simulation offers the promise of improved speed and accuracy, it also raises some perplexing questions. In particular, how can the results produced by a shaped simulation be trusted when the random number generator driving the simulation is replaced by a decidedly non-random algorithm?

To reconcile this apparent paradox, note that the mathematical properties of any steady state stochastic process can be partitioned into two main categories: distributional and trans-distributional. Distributional properties include the steady state distribution itself, plus all properties that can be expressed as direct functions of that distribution and the parameters of the associated stochastic process. All other properties of the stochastic process are trans-distributional [BUZE08, BUZE09].

For the case of a random walk, answers to the following questions depend on trans-distributional properties:

1. What is the probability of the walker making seventeen or more right turns in a row?

2. Given that the walker has just exited from station 0, how many visits will the walker make to other stations before returning to station 0?

3. Given that the walker is at station 2, what is the probability that his trajectory takes him to station 0 before his next visit to station 3?

None of these questions can be answered using only information that can be extracted from the steady state distribution. For the same reason, none of these questions can be answered using shaped simulation. However, shaped simulation can still provide accurate answers to questions involving distributional properties. In effect, shaped simulation sacrifices the ability to evaluate trans-distributional properties in order to gain speed and ensure accuracy when evaluating distributional properties.

The distinction between distributional and trans-distributional properties plays a critical role in understanding the strengths and weaknesses of loosely constrained deterministic systems. As already noted, the principal objective in typical analyses of loosely constrained deterministic systems is to derive equations that characterize the attained state distribution. Once this distribution has been derived, all properties that are direct functions of this distribution can be readily computed. From the perspective of the corresponding stochastic process, these are all distributional properties.

On the other hand, quantities that correspond to trans-distributional properties cannot, in general, be derived. In particular, the empirical independence assumption employed in the LCDS model of the random walk is sufficient to derive an expression for the attained state distribution, but too weak to derive the answers to questions 1, 2 and 3.

For a mathematician, this form of weakness is actually a strength. Determining the most general conditions under which a particular result is valid has always been a primary goal in mathematics. By proving that equations having exactly the same form as equations (5) and (6) can be derived under assumptions that are weaker than conventional stochastic assumptions, the LCDS model demonstrates that these equations are valid under a wider range of conditions than conventionally believed.

10. LCDS Models and Risk

The weakness of the assumptions used in LCDS models also has implications for categorizing and quantifying the nature of risk. As already noted, the assumptions traditionally employed to formulate stochastic models are powerful enough to support the analysis of these models to an almost limitless level of detail. Even though all the resulting equations may be mathematically correct, some are likely to be riskier than others.

The distinction between distributional and trans-distributional properties provides a new way to look at this old problem. Simply put, distributional properties are less risky than trans-distributional properties because they are valid under a wider range of conditions. Thus, practitioners who limit their predictions to distributional properties are less likely to encounter situations where their predictions are incorrect.

In the random walk example, the probability of seventeen consecutive right turns in a row is easy to compute, given the assumption of statistical independence that is incorporated into the stochastic (Markov) model of the random walk. On the other hand, determining the relative likelihood of this trans-distributional property is simply beyond the scope of the weaker assumption of empirical independence incorporated into the LCDS model. As illustrated by Workload 2 (comprised of 45 segments), the appearance of this apparently rare event in a relatively short workload is not inconsistent with a trajectory being “random enough” to exhibit the correct attained distribution (from the perspective of the stochastic model).

The distinction between distributional and trans-distributional properties extends to continuous time queuing models, where it can be shown that average response times are distributional while response time percentiles are trans-distributional [BUZE08]. This has important implications for datacenters that provide services to clients in both conventional and cloud-based environments: service level agreements (SLAs) specified in terms of response time percentiles are inherently riskier than service level agreements based on average response time. In other contexts where probabilistic models are employed, events such as a catastrophic economic collapse are likely to be associated with trans-distributional properties, making them riskier to predict.

In certain cases, trans-distributional properties can be converted into distributional properties by expanding the complexity of the model (i.e., adding states) and introducing additional empirical independence assumptions [BUZE09]. These additional assumptions can then be reviewed carefully to assess whether or not they are likely to be valid in practice.   The additional assumptions reflect the extra risk that must be accepted when deriving predictions (especially those involving rare events) from such a model.

LCDS models can also be used to explore the robustness of results that are mathematically correct. For example, rather than assuming that the proportion of right turns is empirically independent of the walker's current location and is always exactly equal to r, it is possible to assume instead that the proportion of right turns associated with stations 0, 1, 2 and 3 are all within ⍰⍰ of the overall value r. As discussed in the Appendix, this relaxed form of empirical independence introduces a moderate increase in the level of mathematical complexity; nevertheless, it is still possible to derive a closed form analytic expression for the attained state distribution.   The sensitivity of the distribution to the value of ⍰ can then be regarded as an indicator of the risk associated with assuming empirical independence.

11. Relationship to Operational Analysis

Operational analysis [BUZE76a, DENN78] is a framework for analyzing the performance of computational systems in typical real world settings. It is based on the perspective of practitioners who work with such systems on a regular basis.

Operational analysis begins with the assumption that an observer is studying the performance of some real or hypothetical system as it operates over an interval of time. The observer has access to measurements that characterize the behavior of the system during the interval. In this setting, the goal is to derive mathematical relationships among variables that represent observable aspects of system behavior. These relationships will, in general, depend on assumptions that are expressed in terms of other observable properties (e.g., the assumption of empirical independence).

By dealing exclusively with observable or potentially observable properties, operational analysis is able to generate results that apply with absolute certainty to individual trajectories that exhibit the necessary properties. This distinguishes operational analysis from traditional stochastic modeling, where results concerning individual trajectories can only be expressed in probabilistic terms.

Operational analysis has proven useful to practitioners, researchers, educators and students. The approach, which combines mathematical simplicity with straightforward applicability, has been integrated into a number of texts on performance modeling [FERR83, JAIN91, LAZO84, MENA94, MENA02, MORR82]. Nevertheless, legitimate concerns have always existed regarding a lack of mathematical rigor in the intuitive notion of "an observer studying the performance of a system as it operates over an interval of time." In particular, when a complex system can be modeled at multiple levels of detail, the "observable state" of the system is not always definable in intuitively meaningful terms.

These concerns can be resolved by re-casting operational analysis as the study of trajectories generated by loosely constrained deterministic systems. This shift provides analysts with a well-defined formal abstraction that supports all the definitions, assumptions, derivations and results that lie at the core of operational analysis. In addition, because there is a clear intuitive link between loosely constrained deterministic systems and observable properties of real world systems operating over intervals of time, all the practical implications of operational analysis are retained.

The real world systems that provide the original motivation for operational analysis all operate as continuous time processes. For such systems, time is assumed to advance smoothly and continuously from one instant to the next. In contrast, all the models we have been discussing here are based on discrete time processes. Time is assumed to advance in a series of step-by-step jumps. We have not been concerned with the amount of time that elapses between jumps, or what the state of the system may be at those intermediate points.

Loosely constrained deterministic systems are equally capable of modeling the behavior of both continuous time and discrete time systems. Although the underlying principles remain unchanged, terminology has evolved. In particular, specific examples of empirical independence have been referred to as "homogeneity" and as "online behavior = off-line behavior" in earlier publications [BUZE76b, DENN77, DENN78, BUZE08].

12. Relationship to Sample Path Analysis

Traditionally, the term "sample path" is used to denote an individual trajectory associated with the operation of a stochastic process. A sample path can also be thought of as a trajectory generated by a Monte Carlo simulation program that realizes the stochastic process. In effect, sample paths provide a natural link between the abstract mathematical concept of a stochastic process and the observable behavior of a real world system being modeled by that process.

When the length of a sample path is finite, this link is much weaker than one might hope. To see why, consider the stochastic model of a random walk presented in Section 3. The underlying stochastic process has a single parameter, r, which represents the probability of a right turn. However, the observed proportion of right turns in a sample path of finite length is not necessarily equal to r. In fact, it is entirely possible for this observed proportion to take on any value in the interval [0,1]. This makes it difficult to express observable properties of finite length sample paths as functions of stochastic parameters such as r.

To address this issue, analysts who employ stochastic models note that the results they obtain can be regarded as averages over the ensemble of all sample paths associated with the underlying stochastic process. However, these ensembles typically comprise infinitely many sample paths, and it is never possible to ensure that such results will actually apply to any particular member of the ensemble. In other words, the results generated by a single run of a Monte Carlo simulation program can never be guaranteed to agree with the solution obtained through a mathematical analysis of the corresponding stochastic process.

As the length of the sample path approaches infinity, the Strong Law of Large Numbers tightens the relationship between stochastic parameters and their corresponding values in individual sample paths. Specifically, this law states that, in the limit as path length approaches infinity, stochastic parameters and their observable counterparts will *almost surely* be equal to each other (i.e., will be equal to each other with probability one). This provides the foundation for deriving a number of other limiting case results.

A comprehensive treatment of these limiting cases is presented in [TAHA99], where results are obtained under very general conditions: sample paths need not be associated with any particular stochastic process; however, their observable properties are required to converge to well-defined limits as path length approaches infinity.

From the perspective of pure mathematics, there is legitimate interest in the limiting properties of sample paths as their length approaches infinity. In addition, such results will be applicable (*almost surely*) to long trajectories generated by systems whose operation is inherently random.

While it is possible to construct physical mechanisms such as Roulette wheels that display inherently random behavior, phenomena that are classified as computation are seldom inherently random. Instead, they are often best characterized as loosely constrained deterministic systems. Moreover, trajectories encountered in the real world are always finite and do not require limiting case analysis.

Operational analysis can be regarded as the study of finite length trajectories generated by loosely constrained deterministic systems. There is no need to assume that these trajectories realize a stochastic process, or even that their properties converge to well-defined limits as their length approaches infinity. This enables operational analysis to avoid all the mathematical complexities noted above, while still providing results that are of direct practical value.

Within the framework of operational analysis, classical results that can be derived for limiting cases where the length of the sample path approaches infinity (e.g., Little's Formula [TAHA99]) become operational laws that apply with absolute certainty to finite length trajectories that are flow balanced (i.e., initial state = final state). In addition, classical expressions for steady state distributions of discrete time and continuous time Markov processes become operational theorems that apply with absolute certainty to finite length trajectories that are flow balanced and also satisfy suitable sets of empirical independence assumptions.

13. Conclusions

The class of observable phenomena that can be classified as computation is not limited to deterministic processes. Computation also encompasses phenomena that arise when deterministic algorithms process workloads (input streams) whose detailed properties are uncertain.

In such cases, the computational model must incorporate a mechanism for characterizing uncertainty. Traditionally, this issue is addressed by regarding observed phenomena as samples from a set of underlying random variables. Although this characterization yields a rich bounty of mathematical dividends, it also raises one very significant concern: there is, in general, no way to either prove or disprove the existence of these underlying random variables through direct observation of the computation itself.

The computational model developed here employs an alternative characterization of uncertainty that is expressed entirely in terms of observable phenomena. This alternative characterization, which is built upon the abstract notion of a loosely constrained deterministic system (LCDS), provides significant benefits: it leads to solutions that are directly applicable to practical problems, it provides a new perspective on the analysis of risk, and it supports a new method for improving the efficiency of certain computer driven simulations.

14. References

[ALTI07] Altiok, T. and Melamed, B. *Simulation Modeling and Analysis with Arena*. Academic Press, NY 2007

[BERT08] Bertsekas, D.P. and Tsitsiklis, J.N. *Introduction to Probability*. Athena Scientific, Belmont, MA, 2nd Edition, 2008.

[BUZE76a] Buzen, J.P. “Fundamental Operational Laws of Computer System Performance,” *Acta Informatica*, vol. 7, pp. 167-182, June 1976.

[BUZE76b] Buzen, J.P. “Operational Analysis: The Key to the New Generation of Performance Prediction Tools,” *Proc. of IEEE Compcon 76*, Washington, DC, pp.166-171, Sept. 1976.

[BUZE78] Buzen, J.P., Denning, P.J., Rubin, D.B. and Wright, L.S. “Operational Analysis of Markov Chains,” BGS-TR-79-001, April 1978, unpublished.

[BUZE08] Buzen, J.P. “The Improbable Success of Probabilistic Models,” *CMG 2008 Conference Proc*., Las Vegas, NV, pp. 21-32, Dec. 2008.

[BUZE09] Buzen, J.P. “Variability, Uncertainty and Workload Characterization,” *CMG 2009 Conference Proc*., Dallas, TX, Dec. 2009.

[BUZE10] Buzen, J.P. and Tomasik, M., “Shaped Simulation of Stationary Markov Chains,” Proc. 6th International Workshop on the Numerical Solution of Markov Chains, Williamsburg, VA, pp. 25-28, Sept. 2010.

[DENN77] Denning, P.J. and Buzen, J.P, “Operational Analysis of Queuing Networks”. In Beilner, H. and Gelenbe, E. (eds.), *Modeling and Performance Evaluation of Computer Systems*, North-Holland, Amsterdam, October 1977, pp. 151-172.

[DENN78] Denning, P.J. and Buzen, J.P. “The Operational Analysis of Queuing Network Models,” *ACM Computing Surveys*, 10, 3 (Sept. 1978), pp. 225-261.

[FERR83] Ferrari, D., Serazzi, G. and Zeigner, A. *Measurement and Tuning of Computer Systems*, Prentice-Hall, NJ, 1983.

[JAIN91] Jain, R. *The Art of Computer Systems Performance Analysis*, Wiley, NY, 1991.

[KOBA08] Kobayashi, H. and Mark, B.L. *System Modeling and Analysis.* Prentice-Hall, NJ, 2008.

[LAZO84] Lazowska, E.D., Zahorjan, J., Graham, G.S. and Sevcik, K.C. *Quantitative System Performance*, Prentice-Hall, NJ, 1984.

[MENA94] Menasce, D.A., Almeida, V.A.F. and Dowdy, L.W. *Capacity Planning and Performance Modeling*, Prentice-Hall, NJ, 1994.

[MENA02] Menasce, D.A. and Almeida, V.A.F. *Capacity Planning for Web Services*, Prentice-Hall, NJ, 2002.

[MORR82] Morris, M.F and Roth, P.F. *Computer Performance Evaluation*, Van Nostrand Reinhold, NY, 1982.

[STEW09] Stewart, W.J. *Probability, Markov Chains, Queues, and Simulation.* Princeton Univ. Press, 2009.

[SURI83] Suri, R., "Robustness of Queuing Network Formulas" *JACM* 30, 3 (July1983), pp. 564-594.

[TAHA99] El-Taha, M. and Stidham, S. *Sample-Path Analysis of Queueing Systems*, Kluwer, Boston, MA, 1999.

Appendix - Alternative Derivations of Equations (5) and (6)

Analysis of the Stochastic Model

Equations (5) and (6) represent the steady state distribution of the stochastic process associated with the random walk described in Section 3. This stochastic process can be represented as a finite state Markov chain whose state transition matrix is shown below in Figure A-1.

$$\begin{matrix} 1-r & r & 0 & 0 \\ 1-r & 0 & r & 0 \\ 0 & 1-r & 0 & r \\ 0 & 0 & 1-r & r \end{matrix}$$

Figure A-1 – Markovian State Transition Matrix

The interpretation of this state transition matrix is simple and intuitive. There are four rows, corresponding to the four stations in Figure 1. When the walker exits from any one of these stations, the probabilities in the corresponding row reflect his next destination. Each column in the matrix represents a different station. Since the walker can only move one station to the left or right, only two of the probabilities in each row are positive. The other two probabilities are zero, reflecting the fact that the corresponding stations cannot be reached in one step.

In most cases of interest, the steady state distribution of a Markov chain is obtained by solving the "balance equations" that characterize the eigenvector of its state transition matrix. A proof of this well known result can be found in many standard texts [BERT08, KOBA08, STEW09]. For the random walk considered here, the "balance equations that characterize the eigenvector of the state transition matrix in Figure A-1 are given by equations (1) – (4). The normalized solution of this set of linear equations, which corresponds to the desired steady state distribution, is easily derived and is presented in equations (5) and (6).

Attained Distributions and State Transition Matrices – General Case

The LCDS model of the random walk is based upon observed values rather than random variables. Thus, the result derived above (for the stochastic model) is not directly applicable. However, an alternative set of arguments can be used to derive a solution that has the same algebraic form.

Begin by considering the general case where an LCDS is being used to model an arbitrary system. The LCDS model will be based on a state transition diagram of the type shown in Figure 2. At this point, however, there are no restrictions on the topology of the diagram.

Now consider a trajectory that is generated when this very general system processes a workload. Extracting an attained state transition matrix and an attained state distribution from the resulting trajectory is a routine task. The attained state distribution is simply the proportion of exits that are made from each possible state. All these proportions must, of course, be non-negative, and their sum must be equal to 1.

The attained state transition matrix corresponds to a Markovian state transition matrix of the type shown in Figure A-1; however, the elements of the matrix are observed proportions rather than probabilities. Since the topology of the state transition diagram is entirely arbitrary, there are few restrictions on the matrix: once again, all values must be non-negative, and the sum of the values in each row must be equal to 1.

In this very general setting, it is possible to prove that the attained state distribution will always be given by the solution of the corresponding "balance equations", provided the initial and final states of the trajectory are the same. That is, the attained state distribution will always be the normalized eigenvector of the attained state transition matrix in such cases. This surprising result is completely independent of any assumptions regarding uncertainty or variability. In particular, assumptions regarding empirical independence are not required. This result was originally demonstrated for continuous time Markov processes: general birth-death processes in [BUZE76b], and then queuing network models in [DENN77]. Their application to discrete time Markov chains appeared in an unpublished research note [BUZE78] that remains in draft form.

Mathematical relationships having this degree of generality are referred to as operational laws [BUZE76a, DENN78].

The proof of this operational law is based on a very simple argument that can be applied to both continuous time and discrete time processes. Begin by imagining that a token is being used to track the state of an LCDS as it processes some workload. During processing, the token moves from state to state. Clearly, the number of times the token enters each state must be equal to the number of times it exits from that state. The only exceptions are the initial state, from which there is one extra exit at the start of the trajectory, and the final state, into which there is one extra entrance at the end of the trajectory. If the initial and final states are the same, these two exceptions will cancel one another, guaranteeing an equal number of entrances and exits for each state.

If a_{jk} is defined as the number of times the token exits from state j and proceeds next to state k, the equality between the number of entrances to and exits from each state can be expressed as a set of equations involving summations of the a_{jk}. To construct these "token conservation equations" simply note that the number of times the token enters state n is the sum of all values of a_{jk} for which k=n. Similarly, the number of times the token exits from state n is the sum of all values of a_{jk} for which j=n.

The attained state distribution and the elements of the attained state transition matrix can easily be expressed as simple functions of the a_{jk}. Combining these expressions with the "token conservation equations" yields a set of equations that imply the attained state distribution must be the eigenvector of the attained transition matrix. For a complete proof, along with a treatment of cases where the initial and final states are not identical, see the Appendix to [BUZE10].

Application to Random Walks

The very general operational law derived above, which applies to all attained state transition matrices, can now be applied to LCDS models of random walks. It is clear from the structure of the state transition diagram in Figure 2 that each row in the corresponding state transition matrix will have two positive values (associated with left and right turns) and two values that are always equal to zero. Suppose for a moment that the empirical independence constraint is relaxed. The most general form of the attained state transition matrix is then shown in Figure A-2.

Note that r_j represents the proportion of times the walker turns to the right after departing from station j (for j = 0, 1, 2 and 3). Since it is not yet assumed that the proportion of left and right turns are empirically independent of system state, each distinct value of r_j is free to take on any value between 0 and 1.

$$\begin{array}{cccc} 1-r_0 & r_0 & 0 & 0 \\ 1-r_1 & 0 & r_1 & 0 \\ 0 & 1-r_2 & 0 & r_2 \\ 0 & 0 & 1-r_3 & r_3 \end{array}$$

Figure A-2 – Attained State Transition Matrix

The next step, of course, is to introduce the constraint that left and right turns are empirically independent of system state. This automatically forces all the values of r_j in the attained state transition matrix to become equal to r, the overall proportion of right turns in the workload. The attained state transition matrix in Figure A-2 then takes on the same form as the Markovian state transition matrix in Figure A-1. The operational law that requires the attained state distribution to be given by the normalized eigenvector of the attained state transition matrix then implies that the attained state distribution must satisfy equations (1) – (4). The solution is once again given by equations (5) and (6), even though the symbolic variables in these equations now have new interpretations.

This argument also demonstrates another benefit of LCDS models. In cases where the assumption of empirical independence is relaxed, it is still true that the attained state distribution is given by the eigenvector of the attained state transition matrix in Figure A-2. The error introduced by assuming empirical independence can then be expressed in terms of the difference between this eigenvector and the eigenvector of the matrix in Figure A-1. A similar analysis involving a continuous time Markov process and a vastly more complex model is described in [SURI83]. Sensitivity analyses of this type appear to have no direct counterparts in stochastic modeling.

About the Author

Jeffrey P. Buzen (buzen@post.harvard.edu) is an independent consultant and researcher specializing in the analysis of computer performance. He is best known as the co-founder and Chief Scientist of BGS Systems, a software company that developed tools for performance management and capacity planning from 1975 to 1998.