Computation, Uncertainty and Risk

JEFFREY P. BUZEN

Corresponding author: buzen@post.harvard.edu

A model is developed for representing computations that arise when deterministic algorithms process workloads whose detailed structure is uncertain. The new computational model, which is based upon the notion of loosely constrained deterministic systems (LCD systems), incorporates a characterization of uncertainty that is expressed entirely in terms of observable properties of the computation itself. This provides a number of advantages over traditional probabilistic models. In addition to providing a more direct link to problems that arise in practical settings, LCD models offer a new perspective on the analysis of risk and support a new method for improving the efficiency of certain computer driven simulations.

Keywords: computational model; probabilistic model; operational analysis; empirical independence; shaped simulation; trans-distributional properties; uncertainty; risk

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.

Although probabilistic Turing machines can be of value in certain cases, the unpredictable behavior exhibited by real world systems is seldom the result of non-deterministic algorithms. In fact, the step-by-step operation of real world systems is almost always 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 (LCD system). The observable phenomena that are associated with LCD systems can be classified as computation even though detailed properties of these phenomena remain entirely uncertain.

2. LOOSELY CONSTRAINED DETERMINISTIC SYSTEMS

Essentially, LCD models regard the behavior of systems, algorithms and computations as the result of interactions between two separate components, one deterministic and the other non-deterministic. The deterministic component is represented by a conventional finite state machine (or finite state automaton) and its associated state transition diagram. The non-deterministic component is represented by a set of “loose constraints” on the workloads (strings of input symbols) that are processed by the finite state machine.

To understand the motivation for loose constraints, begin by considering models that are entirely deterministic in nature. Such models incorporate parameters that are sufficiently detailed to identify every element of the workload being processed. As a result, the analysis of the model can be based...
on the complete set of observable properties exhibited by
the trajectory that the workload generates (i.e., the complete
sequence of states that the finite state machine passes through
as the workload is processed). Deterministic models with this
property can be regarded as being “tightly constrained.”

Now consider cases where the detailed step-by-step struc-
ture of the workload cannot be determined from the model’s
parameters. The parameters of such models typically corre-
spond to average values or other statistics that that summarize
properties of the entire workload and its associated trajectory.
These parameters can be regarded as “loose constraints” on
the structure of model’s workload. In effect, the results derived
from an analysis of such models will apply to the entire class
of workloads that are constrained to share a common set of
parameter values.

When only loose constraints are specified, the structure of
the associated trajectories cannot be reproduced in complete
detail. However, 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, such derivations do not require the
assumption that random forces have generated the workload
being processed by the finite state machine. Thus, LCD
models 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 a traditional Markov model.
The standard solution is derived, and its applicability to the
original problem is examined.

The model of the random walk is then reformulated as a LCD
system. This makes it possible to characterize randomness in
terms of a simpler concept known as empirical independence
[1]. The discussion demonstrates that the assumption of empir-
ic independence is intuitively plausible, easily verifiable,
and sufficiently powerful to generate the same solution that is
obtained using a traditional Markov model.

Although the specific details of this particular example are
not intrinsically important, the nature of the assumptions that
are made 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 discussed.

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 LCD models.

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.

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 only station-to-
station 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
core that can be specified using a diagram of this type.

The next issue to consider is the nature of the workload
driving this system. In this case, a workload is simply a series
of left and right turns represented symbolically by a sequence
of the letters R and L. Once such sequence where right turns
are twice as common as left turns is RRLRRLRRLRRLRL.
solution:

\[ P(n) = P(0) \left( \frac{r}{1-r} \right)^n, \quad \text{for } n = 0, 1, 2 \text{ and } 3 \]  

(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 the 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 observed proportion of right turns should be equal to \( r \), the probability of a “head” in the stochastic model (based on coin tosses). By observing the trajectory generated by the walker, we can also determine the proportion of visits to states 0, 1, 2 and 3. These observed 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 were accurate for this case, we would also 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,

\[ P(0) = 1/15 \]  

(7)

\[ P(1) = 2/15 \]  

(8)

\[ P(2) = 4/15 \]  

(9)

\[ P(3) = 8/15 \]  

(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 are 1/3 and 2/3 respectively.
The success of equations (5) and (6) for the trajectory generation 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 exhibit. 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 observable properties of workloads driving real world systems can be regarded as samples from underlying random variables. There is no way to verify the correctness of such assumptions by direct observation of actual workloads or their associated trajectories.

To develop an alternative characterization of uncertainty, consider a workload that has actually been generated by large number of idealized coin tosses. It’s clear that many mathematical properties can be extracted from such a workload and its associated trajectory. However, a small subset of these properties may in fact 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 property is, in fact, realized by Workload 2. Recall that Workload 2 has the following form:

\[
RRRRRRRRRRRLRRLRLRLRLLLRR
\]

As already noted, the trajectory generated by this workload is illustrated in Figure 4 and shown below:

23333333333333232323232323212121212121212121210101012

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 a sketch of the proof.

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 LCD 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 LCD 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 determine the proportion of right turns the walker makes upon exiting from stations 0, 1, 2 and 3. If these four observed proportions 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 traditionally considered to be driven by forces that operate in a purely random fashion. In such 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 an attained distribution with the same mathematical form as the steady state distribution of the corresponding stochastic process.

7. LOOSELY CONSTRAINED DETERMINISTIC SYSTEMS: REVIEW

The first step in creating an LCD model of a real world system is to identify the states of the system and the deterministic state-to-state transitions that characterize its dynamic behavior. This basic information is then mapped into the formal structure of a finite state machine. Figure 2 provides an example of such a mapping for the random walk depicted in Figure 1.

Once the finite machine has been specified, the next step is to identify the parameters that characterize the workloads that drive it. In general, these parameters are expressed in terms of the workload’s summary statistics (means, variances, and so on). In effect, summary statistics represent loose constraints on the set of workloads to which the analysis applies.

The third step, which is not always necessary, is to introduce additional constraints such as empirical independence that...
impose structure on the uncertainty that exists regarding the
details of system behavior. Such constraints, which reflect
intuitive notions of randomness, can also be characterized as
“loose” since they can be satisfied by multiple workloads.

The abstract model created in steps 1, 2 and 3 is 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.

Equations derived through the analysis of an LCD model will
be valid whenever the model’s loose constraints are satisfied.
For example, the workloads in the LCD model of a random
walk are characterized by a single parameter r, which is defined
as the proportion of right turns in the workload. The model
also assumes that the observed proportion of right turns is
empirically independent of system state. Thus, the predicted
state distribution obtained from the model when \( r = 1/2 \) will
be exactly correct in all cases where the observed proportion
of right turns is equal to one half, the initial and final states
are identical, and the assumption of empirical independence is
satisfied.

LCD models can be regarded as specialized extensions to
the general models of deterministic computation developed
by Turing and his successors. The LCD extensions character-
ize uncertainty in terms of directly observable properties of
workloads and trajectories. In contrast to probabilistic charac-
terizations of uncertainty, this representation yields results that
apply exactly to every trajectory that satisfies the associated
set of loose constraints.

8. SHAPED SIMULATION

Monte Carlo simulation is a well established and widely used
technique for the numerical evaluation of stochastic models.
LCD 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
determine 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.

LCD models shed a new light on this problem. Begin by
defining \( r_j \) as the proportion of time the walker turns right after
exit from station j. 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, the values of \( r_j \) can all be expected to
converge to \( r \) in the limit as the length of the simulation
approaches infinity. However, the values of \( r_j \) 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 LCD 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 \([2]\). 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 force 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 \([3]\), 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. By definition, 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 can be classified as trans-distributional [1,4].

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

(i) What is the probability of the walker making seventeen or more right turns in a row?
(ii) 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?
(iii) 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 LCD 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 LCD model demonstrates that these equations are valid under a wider range of conditions than conventionally believed.

10. LCD MODELS AND RISK

The weakness of the assumptions used in LCD 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 LCD 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 [4]. 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 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 [1]. These additional assumptions can then be examined carefully to assess whether or not they are likely to be valid in practice. Additional assumptions of this type reflect the extra risk that must be accepted when deriving predictions (especially those involving rare events) from such a model.

LCD 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 some small value 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 $a$ can then be regarded as an indicator of the risk associated with assuming empirical independence.

11. RELATIONSHIP TO OPERATIONAL ANALYSIS

Operational analysis [5,6] 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 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 [7–12]. 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 recasting 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 LCD systems and 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 [4,6,13,14].

12. 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 (strings of input symbols) 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 correctness of the characterization through direct observation of the computation itself.

The computational model developed here is based upon 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 an LCD system, 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.

REFERENCES

A.1. 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.

\[
\begin{pmatrix}
1 - r & r & 0 & 0 \\
1 - r & 0 & r & 0 \\
0 & 1 - r & 0 & r \\
0 & 0 & 1 - r & r
\end{pmatrix}
\]

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 the 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 [15–17]. 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 LCD 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. These arguments will be developed for the general case where an LCD system is being used to model the behavior of an entirely arbitrary system.

Consider a trajectory that is generated when an arbitrary LCD system processes a workload. Extracting the attained state transition matrix and 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 is also based on observed proportions rather than probabilities. In particular, the value in row j, column k of the matrix represents the proportion of exits from state j that are followed immediately by an entrance into state k.

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.

This surprising result is completely independent of any assumptions regarding the process that generated the trajectory being analyzed. In particular, assumptions regarding empirical independence are not required. This result was originally demonstrated for continuous time Markov processes: general birth-death processes in [13], and then queuing network models in [14]. Their application to discrete time Markov chains appears in an unpublished research note [18] that remains in draft form. Mathematical relationships having this degree of generality are referred to as operational laws [5,6].

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 LCD system 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 [3].

**Application to Random Walks**

The very general operational law derived above, which applies to all attained state transition matrices, can now be applied to LCD 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 equation (A.2).

\[
\begin{pmatrix}
1 - r_0 & r_0 & 0 & 0 \\
1 - r_1 & 0 & r_1 & 0 \\
0 & 1 - r_2 & 0 & r_2 \\
0 & 0 & 1 - r_3 & r_3
\end{pmatrix}
\]

(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.

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 LCD 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 equation (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 equation (A.1). A similar analysis involving a continuous time Markov process and a vastly more complex model is described in [19]. Sensitivity analyses of this type appear to have no direct counterparts in stochastic modeling.