A Decision-Theoretic Approach to Defining Use for Computer Simulation

Eric W. Weisel
General Dynamics Information Technology
Eric.Weisel@GDIT.com

Keywords: decision theory, validation theory, validation, simulation use, intended use

Abstract
Although computer simulation shares a common beginning with the advent of the digital computer, our science still lacks sufficient foundation in rigorous mathematics. An understanding of the complex relationship between simulation and reality is a defining characteristic that differentiates simulation science from science that uses simulation. Significant work has been done in relating simulation to system with respect to accuracy but the consideration of how a simulation is, or is to be, used has not been rigorously defined. Foundational mathematical structures suitable for reasoning about use are presented within the context of a framework for rigorously relating simulation to reality. Classes of artifacts in the simulation activity are identified and relations are drawn between them. Decision theory is applied to define a mathematical structure for use that is sufficient to inform decision-making using models and simulations in a rigorously defensible way.

1. INTRODUCTION
The objective to understand the complex relationship between simulation and reality is a defining characteristic that differentiates simulation science from science that uses simulation. Significant work has been done in relating simulation to system with respect to accuracy. Although broadly accepted by the community as an important consideration, how a simulation is, or is to be, used has not been rigorously defined within this context. Although this question is usually confined to discussions of validation, construction and validation of a simulation are in many ways two sides of the same coin. This observation matches the “begin with the end in mind” philosophy of simulation best practice which proffers the importance of considering use and validation early in the development process. A mathematical structure for use is needed to connect the accuracy analysis to the suitability for use decision. Without this progress it will be impossible to move forward in developing rigorous techniques and tools that provide defensible measures and metrics that can be readily used by a decision-maker to assess suitability for use.

The broad objective then is to lay the foundation, or at least bits of it, for a rigorous theory for decision-making using models and simulations. Any mathematical theory will include structures, the conceptual artifacts that the theory is about, as key building blocks of the theory. To be useful, a structure must be defined with sufficient rigor to enable reasoning on its properties to develop new bits of knowledge. The specific objective in this paper is to propose a mathematical structure for use that is sufficient to inform decision-making using models and simulations in a rigorously defensible way.

CONSTRUCTION

VALIDATION

Figure 1. The generalized simulation activity

2. THE SIMULATION ACTIVITY
In order to inform the approach to use, the discussion will begin with a general framework for relating simulation to reality. Figure 1 describes the simulation activity framework at the most general level. Classes of artifacts in the simulation activity are identified and relations are drawn between them with sufficient rigor such that we can: (1) define the artifacts that make up the class, that is, carefully describe a set of rules for inclusion in the class; (2) define relations between classes; and (3) assess the mathematical properties of the collected elements of the now defined classes and relations. Classes of artifacts should be context-free with respect to the relations defined and the experiment that the simulation is applied to.

In Figure 1, and the diagrams that follow, clouds and boxes represent classes of artifacts. Clouds represent abstract artifacts or mathematical structures. Boxes represent tangible artifacts. Arrows represent relations between the classes. Note that in this diagram an M&S (noun) means a model and/or simulation.
3. FORMALIZING THE SIMULATION ACTIVITY

Formalism within a particular science, in the strictest sense, refers to the application of formal logic and proof theory to the structures and theorems of that science; although in practice most mathematicians don’t go quite this far, including only a rigorous application of informal logic in their published proofs and relying on peer review to confirm the results. In this paper the term formal means the application of formal logic. The term rigorous means the application of practical mathematical proof. As such, within simulation science, formalism, or a formalism, more often refers to a mathematically rigorous approach to defining the simulation object and presenting properties and theorems related to it. It is inherent in this latter approach, as it is in most published mathematical proof, that given sufficient time, patience, and effort, a rigorous informal proof could indeed be made formal.

3.1. Formalizing computation in computer simulation

Zeigler et al [1]; Barrett et al. [2] [3] [4] [5]; and Weisel, Petty, and Mielke [6] present rigorous approaches to developing simulation theory, or formalizing simulation. The objective of these approaches, in the end, is to develop theories with broad usefulness for simulation theoreticians and practitioners alike. Each of the approaches noted above describe structures for simulation executed on a digital computer. Theories that describe live simulation, a military field exercise for instance; or analog simulation, flight training using an analog flight simulator, engineering analysis using a scale-model of a ship hull in a wave tank, or a medical procedure practiced on a cadaver, would be in many ways fundamentally different. These theories compare systems to systems. Theories described in this paper apply to simulation systems utilizing computation on a digital computer. This will be called computer simulation. So, we will look to theoretical computer science to formalize the structure and process of computer simulation and provide a theoretical structure that will allow us to rigorously compare the computer simulation to the system.

Computation in computer simulation can be represented by a deterministic labeled transition system consisting of a non-empty set of states, a set of boundary values or inputs, and a computable simulation transition function that computes a new state for each combination of state and input. A labeled transition system is represented by a directed multi-graph called a transition graph. Figure 2 is a conceptual diagram of computation in computer simulation.

3.2. Comparing transition systems

Milner and Park [7] [8] [9] developed bisimulation to study processes that appear similar by external observation. A rigorous approach for comparing all possible trajectories of a simulation for a given input set is provided by the mathematical structure of bisimulation. Tanner and Pappas [10]; Weisel, Petty, and Mielke [6]; and van der Schaft [11] have used the concept of bisimulation as a basis for the comparison of computer simulations to the systems they are intended to represent.

If a relation on two labeled transition systems is a bisimulation, then for all related pairs of states and for each input, if one transition system transitions then so does the other and the new states are also related. Each transition system matches the transitions of the other. This structure is strong bisimulation. A related form of bisimulation called weak bisimulation allows one or both transition systems to have additional internal transitions as long as the important states are related. Weak bisimulation might be applied to simplify the theoretical analysis of computer simulation by ignoring floating-point operations, register shifts, and the like. The bisimulation relation is denoted by the symbol \( \equiv \). \( T_1 \equiv T_2 \) means \( T_1 \) and \( T_2 \) are bisimilar. A simulation relation is a one-way version of a bisimulation relation. The simulation relation is denoted by the symbol \( \subseteq \). \( T_1 \subseteq T_2 \) means \( T_2 \) simulates \( T_1 \). That is, \( T_2 \) matches all of the transitions of \( T_1 \).

Structures that are suitable for efficient theory are not necessarily best suited for efficient computation or even intuitive model building. For a theoretical structure to be
useful, however, computation and model building paradigms should be readily transformable to the theoretical structure. There is no need for theoretical structures to look like structures used in practical simulation development as long as the structures can be mapped one to another.

3.3. Interpreting artifacts
Model theory is concerned with relations between sentences in formal languages and interpretations (assignments of values to variables) that make those sentences true [12]. A formal theory \( K \) consists of: (1) an alphabet – a set of abstract symbols; (2) a grammar – a set of rules specifying the ways in which the symbols of the alphabet may be formed into finite strings of symbols and the ways in which the strings may be formed into statements. Statements that submit to these grammatical rules are called well-formed; (3) axioms – a set of well-formed statements accepted as true without proof; and (4) rules of inference – a set of rules specifying the ways in which axioms and other well-formed statements may be changed into new well-formed statements called theorems.

\[
\text{SPECIFIES}
\]

\[
\begin{array}{c}
\text{NATURAL SYSTEM} \\
\Downarrow \\
\text{INTERPRETS} \\
\end{array}
\]

\[
\begin{array}{c}
\text{FORMAL THEORY} \\
\end{array}
\]

Figure 3. Simulation artifacts and relations [13]

For example, consider a first-order model. A first-order language \( L \) consists of the prepositional connectives \( \neg \) and \( \rightarrow \), and the universal quantifier \( \forall \) (note that the existential quantifier \( \exists \) can be constructed from \( \forall \)); the punctuation marks left parenthesis, right parenthesis, and comma; a set of variables; a set of function letters; a set of constants; and a set of predicate letters. A first-order theory in the language \( L \) is a formal theory \( K \) whose symbols and well-formed statements are the symbols and well-formed statements of \( L \) with specific logical axioms, proper axioms which vary from theory to theory, and specific rules of inference. In the context of model theory, a model of \( K \) is an interpretation of \( L \) for which all the axioms of \( K \) are true.

So model theory allows us to formally associate a set of well-formed statements, such as mathematical equations, with an interpretation within the universe of discourse, provided that the well-formed statements are described in the context of a formal theory. Figure 3 demonstrates the relationship. The formal theory specifies the system. The system provides meaning for the formal theory.

4. RELATIONS ON SIMULATION ARTIFACTS
Sargent [15] and Petty [16] diagram the relationships between concepts encountered in the simulation construction and validation processes. The general approach in this paper is to define classes of artifacts in the simulation activity by the properties of the elements that make up the class. Once defined, the classes are linked through a collection of relations on the classes.

4.1. Creating a proxy for reality
In any meaningful sense, reality is inaccessible for this assessment. A proxy for reality is provided by an idealized version of reality and measurements taken by observations of that same reality. The ideal system is conceptual, an abstracted, generalized, and bounded representation of the natural system, which exists within some universe of discourse. Individual perceptions of the natural system may be biased by the world-views of one or more observers. The observed system is a collection of data of the same natural system.

4.2. Constructing the M&S
The M&S consists of the source program and its compiled counterpart, the executable program. Compiler verification, is a formal assessment of compiler correctness. The conceptual system is a collection of formal and informal artifacts that both capture the ideal system from the stakeholders’ perspectives and intermediate artifacts that capture specification and interpretation steps from ideal system to source program code.

4.3. Connecting the M&S to reality
Figure 4 demonstrates how these classes work together to connect the M&S to reality in a general framework for drawing relations on simulation activities. The framework demonstrates three primary relationships: (1) theoretical; (2) empirical or experimental; and (3) analytic or logical. Each is identified by both construction and validation flowpaths. The inner circle, relationship (1), is most useful for describing theoretical relationships. The outer circle, relationships (2) and (3), is most useful for describing practical activities. There are two construction and validation activity flows for the practical activities.

The inner circle depicts theoretical validation. The artifacts identified in theoretical validation are the mathematical structures defined by the source program and the executable program. These structures are the transition systems \( T(M) \) and \( T(X) \) respectively. \( T(M) \) is the transition system that is the closest computable approximation to an idealized abstraction of the system of interest. These structures are related through simulation and bisimulation
relations. Although $T(M)$, $T(X)$, and $T(M^*)$ are rigorously defined and suitable for reasoning, tractability and decidability issues present a challenge to working directly with these mathematical structures in any practical sense. The inner circle is most useful in developing theory.

*Experimental* or *empirical validation* results from a direct comparison of trajectories calculated by the executable program and the observed system. Validation is achieved/methods are performed through observation or experimentation using simulation response. Empirical validation yields to rigorous statistical methods. The primary benefit of empirical validation is the direct assessment of accuracy.

*Logical* or *analytic validation* consists of analytic or logical comparisons of the source program to the ideal system primarily through informal but rigorous assessment of the conceptual model, with a specific use in mind, coupled with more formal program verification to assess program correctness with respect to some specification based on the same conceptual model. Analytic validation yields to a robust set of analysis techniques underpinned by the conjecture that with sufficient effort informal conceptual models and rigorous analysis could be formalized.

5. **DEFINING USE**

Sargent [15] and Petty [16] provide surveys of the state-of-the-art in validation of computer simulations. It is broadly accepted that neither M&S development nor validation analysis can be decoupled from use. [18] Although the importance of use is well-known, many treatments of validation speak primarily to accuracy. As early as 1967, Naylor and Finger [19] identified competing views on the proper perspective of validity of an M&S. Kleindorfer, O’Neill, and Ganeshan [20] provide a more contemporary treatment of validation in the context of the philosophy of science. Harmon and Youngblood [21] identify confidence as a necessary component of quality simulation validation. Validation theory must go well beyond accuracy. A rigorous connection between the M&S and its use must be made to enable a theory for validation. Likewise, our discussion so far only addresses accuracy.

5.1. **Acknowledging the performer**

The next step in addressing use in a rigorous way is to consider that the analysis includes both the M&S and the performer. A *performer* is an actor who takes some *action* on a system. The performer may be but is not necessarily
the decision-maker in the simulation use decision. More specifically, consider \( P \) is a class of performers and \( A \) is a class of actions in the same fashion as classes of artifacts are defined in the simulation activity. The intended result of the action is some consequence \( \xi \), described by a well-formed statement, such that \( \xi = \text{TRUE} \) if some measurable condition or state of the system or the performer is \( \text{TRUE} \). \( \neg \xi \) is the complement of \( \xi \). For a training use \( \xi \) and \( \xi' \) are observed in the performer. For an analysis use \( \xi \) and \( \xi' \) are observed in the natural system \( N \) and simulation \( N' \) respectively. Figure 5 describes this relationship.

5.2. Defining the performance decision problem

The performance decision defines use. Consider a decision such that a decision-maker \( \text{DM} \) chooses between (1) selecting an action \( A \) by performer \( P \) on system \( N \) with consequence \( \xi \) or \( \neg \xi \) and probability of occurrence \( P(\xi) \) or \( P(\neg \xi) \) respectively or (2) rejecting the action, i.e. selecting the complement \( \neg A \), and thereby accepting the status quo \( \phi \) with probability of occurrence \( P(\phi) = 1 \). So use is defined by a tuple \((P, N, A, \xi)\).

The performance decision can be represented by a decision tree. A decision tree is a structure for modeling complex decisions. It is represented by a graph with decisions represented by squares and events represented by circles. Decision trees are useful for representing the calculation of expected value or expected utility. Figure 6 is the decision tree that represents the performance decision. This particular representation values consequences at their asset position.

Posed in this way, use is essentially formulated in the style of a decision problem within the context of theoretical computer science. A decision problem is simply a problem formulated as a question having only two answers, YES or NO. A decision problem may have additional structure defining a class of instances based on a set of properties. The decision problem is also found underpinning theorems related to computability and computational complexity in theoretical computer science. [22]

![Figure 5. Observing consequence for training and analysis uses of simulation](image)

![Figure 6. Performance decision](image)

6. ASSESSING UTILITY AND RISK

A fundamental paradigm shift is required in the understanding of validation. In every simulation-use decision, some decision-maker, whether a high-ranking official in an organization with a strong M&S accreditation process or a teen-ager choosing whether to play a video game or not, makes a choice, informed or otherwise, based on perceived gain, risk, and confidence in the M&S. Elele and Smith [23] identify risk of using an M&S as a function of consequence and error and suggest a process for conducting a risk-based validation assessment. While this risk-based approach to validation does not eliminate the necessity to address accuracy, it may provide a way to rethink the analysis that will make the accuracy question more accessible. In order to avoid the all models are wrong fallacy, error in the context of computer simulation must be tied directly to a well-formulated use problem.
6.1. Defining the simulation decision problem

The simulation decision provides a structure to evaluate the suitability of the simulation for a specific use. Consider a decision such that DM chooses between (1) selecting to test an action $A'$ by performer $P$ on simulation $N'$ with response $\xi'$ or $\neg\xi'$ and probability of occurrence $P(\xi')$ or $P(\neg\xi')$ respectively and then behaving optimally or (2) selecting the performance decision directly and choosing between $A$ and $\neg A$ without testing. Expected value is calculated in the usual way using an asset position values for $\xi$, $\neg\xi$, and $\phi$. DM will choose to test in simulation if the expected value of testing and then deciding optimally $EV(A')$ is better than the alternative $EV(\neg A')$. This approach is similar to a textbook approach to evaluating a decision with sample information. See Winston [24] as an example.

In general, a practical application would consider simulation cost and devalue $\xi$, $\neg\xi$, and $\phi$ for the test alternative appropriately. In the discussion here, as an example, utility theory will provide a way to evaluate a sample simulation decision in a simplified case without valuing $\xi$, $\neg\xi$, and $\phi$. Within the broader decision theory, utility theory rigorously connects a decision to a decision-maker’s attitude toward risk. A comprehensive utility theory approach would incorporate risk tolerance but would be complicated by measuring decision-maker’s utility curve.

6.2. Assessing the utility of the status quo

For the example, consider $(P, N, A, \xi)$ and DM with perfect knowledge of the corresponding probabilities where $\gamma = P(\neg\xi)$ and $\gamma > 0$ such that DM is indifferent between $A$ and $\neg A$. Figure 8 is the decision tree for this discussion.
Since DM is indifferent between A and ¬A, then \( U(\phi) \) equals the expected utility of A so \( U(\phi) = (1-\gamma)U(\xi) + \gamma(\neg\xi) \). Since \( U(\xi) = 1 \) and \( U(\neg\xi) = 0 \) then \( U(\phi) = 1-\gamma \). Note that \( U(\neg\xi) < U(\phi) < U(\xi) \) since (1) if \( U(\phi) \leq U(\neg\xi) \) then \( \neg A \) is dominated for all \( \gamma > 0 \) and (2) if \( U(\phi) \geq U(\xi) \) then A is dominated for all \( \gamma > 0 \).

6.3. Assessing Type I error

In the statistical sciences, Type I and Type II errors describe classes of poor outcomes of a decision made subject to the analysis of a null hypothesis. Type I error \( \alpha \) is characterized by a statistical test that rejects the null hypothesis but would have resulted in a desirable outcome had DM chosen otherwise.

![Type I error](image1)

For the simulation decision if DM selects to test and the response is \( \neg\xi' \) then DM will choose optimally afterward and select \( \neg A \). Type I error will not affect the calculation of expected utility at the simulation decision provided that \( U(\phi) > \alpha U(\xi) + (1-\alpha)U(\neg\xi) \). In this case, \( \alpha < 1 - \gamma \) as long as \( \gamma \) is small, i.e. as long as DM is willing to accept only a small \( P(\neg\xi) \). Figure 9 is the decision tree for this discussion.

6.4. Assessing Type II error

In this analysis expected value ties directly to risk. Risk is commonly understood as a function of likelihood of

![Type II error](image2)

Type II error and some loss function, or consequence, of that error. Type II error \( \beta \) is characterized by a statistical test that fails to reject the null hypothesis and yet the outcome is undesirable. For the simulation decision if DM selects to test and the response is \( \xi' \) then DM will choose optimally and select A provided that \( U(\phi) < (1-\beta)U(\xi) + \beta U(\neg\xi) \). So \( 1 - \gamma < 1 - \beta \) and \( \beta < \gamma \). We will see in the next section that \( \beta < \gamma \) is always the case for our simplified example for DM to select \( A \). Figure 10 is the decision tree for this discussion.

6.5. Analyzing the simulation decision

Lastly, we analyze the example using utility theory to determine a set of rules establishing when DM will select to test in simulation. Again considering \( U(\xi) = 1, U(\neg\xi) = 0 \), and the constraint on Type I error already described, and evaluating the expected utility of A and \( \neg A \), then DM chooses A (note that in both cases \( \beta < \gamma \)):

1. if \( P(\neg\xi) < \gamma \) then \( (1-\beta)P(\xi') + (1-\gamma)[1-P(\xi')] > 1 - P(\neg\xi) \) so test in simulation if \( \beta < \gamma \) and \( \gamma \leq P(\neg\xi) \); and
2. if \( P(\neg\xi) \geq \gamma \) then test in simulation if \( \beta < \gamma \)

7. CONCLUSION

Foundational mathematical structures suitable for reasoning about use are presented within the context of a framework for rigorously relating simulation to reality. Classes of artifacts in the simulation activity are identified and relations are drawn between them. Within a broader objective to develop theory for decision-making using models and simulations, decision theory is applied to define a mathematical structure for use that is sufficient to inform decision-making in a rigorously defensible way. A decision structure is proposed in the form of an example that demonstrates a way forward to rigorously assess a simulation’s suitability for a specific use using statistical methods. By incorporating a structure that is directly tied to use to compare expected value or utility of alternatives, this method provides not only a connection from accuracy to use in an actionable way, but also a way ahead to value simulation within a broader investment analysis paradigm.
References


Biography

Dr. Eric Weisel is Chief Scientist for Training and Simulation Solutions at General Dynamics Information Technology. Prior to joining General Dynamics, he led WernerAnderson, Inc., a small technology research company that provided modeling, simulation, and analysis services to partners in academia, government, and industry. Before entering the technology research field, he served as a U.S. Navy submarine officer on three Los Angeles class attack submarines and various Navy and joint staffs with expertise in nuclear engineering; navigation; and submarine, battle group and joint operations. Dr. Weisel earned the second Ph.D. in Modeling, Simulation, and Visualization Engineering awarded at Old Dominion University and holds an M.S. in Operations Research from the Florida Institute of Technology and a B.S. in Mathematics from the United States Naval Academy. He is an Adjunct Professor at Old Dominion University teaching courses in operations research and modeling and simulation.