# THE NATURE OF MODELING

by

Jeff Rothenberg

Chapter for "AI, Simulation & Modeling"

Lawrence E. Widman, Kenneth A. Loparo, and Norman R. Nielsen, editors

John Wiley & Sons, Inc., August 1989, pp. 75−92

(Reprinted as N−3027−DARPA, The RAND Corporation, November 1989.)

## Introduction

Modeling is one of the most fundamental processes of the human mind. Yet it is often misunderstood in ways that seriously limit our ability to function coherently and effectively in the world. The use of inappropriate models (or the inappropriate use of modeling itself) is responsible for countless disasters of personal, technological, and historical proportions. Modeling is the quintessential human conceptual tool. Yet it is rarely examined from a theoretical point of view and therefore rarely mastered.

This chapter attempts to define modeling precisely. It surveys the kinds of models human beings use and discusses their motivations, advantages, and limitations. It places simulation in this context and surveys various kinds of computerized simulation. It then discusses Artificial Intelligence in the broadest terms, highlighting a few of its most relevant aspects, and attempts to show how AI can contribute to−−and how it depends on−−modeling. Finally, it suggests that the traditional view of simulation is too narrow and should be expanded to encompass more of modeling, leading to "Knowledge−Based Simulation". This is illustrated in terms of ongoing research at The RAND Corporation.

## Overview of Modeling

Modeling in its broadest sense is the cost−effective use of something in place of something else for some purpose. It allows us to use something that is simpler, safer, or cheaper than reality instead of reality for some purpose. A model represents reality for the given purpose; the model is an abstraction of reality in the sense that it cannot represent all aspects of reality. This allows us to deal with the world in a simplified manner, avoiding the complexity, danger and irreversibility of reality.

Modeling underlies our ability to think and imagine, to use signs and language, to communicate, to generalize from experience, to deal with the unexpected, and to make sense out of the raw bombardment of our sensations. It allows us to see patterns, to appreciate, predict, and manipulate processes and things, and to express meaning and purpose. In short, it is one of the most essential activities of the human mind. It is the foundation of what we call intelligent behavior and is a large part of what makes us human. We are, in a word, modelers: creatures that build and use models routinely, habitually−−sometimes even compulsively−−to face, understand, and interact with reality.

Using an inappropriate model to deal with reality can do considerable harm. How many patients were killed by Medieval blood−letting who might otherwise have recovered? How many children have been locked into Procrustean roles by models like Girls aren’t good at math or Boys aren’t intuitive? How many computer programs do the wrong thing correctly? These are all cases of using incorrect or inappropriate models. There are even cases where using any model at all is inappropriate. For example, relating to someone in terms of a model precludes the comprehension and appreciation of the richness and unpredictability that distinguish living beings from rocks and tractors.

In order to avoid inappropriate choices and uses of models, it is vital to formulate a clear definition of what a model really is, what constitutes a good, appropriate model, and how to judge when using a particular model (or any model) is justified.

## Other Models of Modeling

Before developing our own definition, it is useful to discuss some of the literature on modeling. The subject is as broad as human intellectual endeavor itself, since modeling is the intellect’s tool of choice. At one extreme lie fundamental points of view on how we approach reality, tracing their roots to Plato and Aristotle. For example, Kant contends that reality in and of itself (the "noumenon") is unknowable, and that the "forms of our perception" constitute what is effectively an inherent and unavoidable modeling process (though he does not use this term) that separates us from the noumenon [25]. A survey of this subject would amount to a synopsis of much of philosophy, which is beyond the scope of this chapter.

At another extreme, formal model theory in mathematical logic defines the semantics of a propositional language as a "model" that specifies what can be concluded validly from what [13]. Model theory (despite its name) is somewhat esoteric to the discussion at hand, in that it focuses on formal properties of one particular kind of model (i.e., logical) rather than on modeling as a whole [49].

Of greater relevance is the literature on the application of specific modeling techniques in application areas like systems analysis and decision support. These discussions tend to be concrete enough to be relevant to real-world modelers while being abstract enough to provide insight into modeling in general.

From this applied perspective, modeling is often seen as a way of gaining control over the world [47], or of making decisions or answering questions about the world [7,11,14,18,23,33,35,39,43,46]. It is widely recognized that the purpose of a model must be understood before the model can be discussed [12,33]. The purposes to which models may be put are frequently categorized as being either descriptive (describing or explaining the world) or prescriptive (prescribing optimal solutions to problems) [14,34,39]. Prescriptive uses of a model are sometimes further distinguished from normative uses (such as identifying feasible goals or standards [18]) and from idealization (allowing the construction of hypothetical, ideal entities that illuminate real-world phenomena [21]). Specific uses of models include projection (conditional forecasting), prediction (unconditional forecasting), allocation and derivation (e.g., of expected demands for resources or services) [18], as well as hypothesis-testing, experimentation, and explanation [21].

Models are generally assumed to have an analogous or imitative relationship to some real-world phenomenon or system, though this assumption is often only implicit. Even where explicit, this assumption usually remains vague and intuitive [18,33,46]. Since most work in modeling is carried out for a particular purpose within a particular application domain, most discussions touch only lightly on abstract modeling issues before elaborating specific techniques.

Some writers [18,39,46,50] point out that models can be characterized in many alternative ways, but most suggest categorizations specific to the application areas under consideration. Models may be characterized in terms of their form, their relationship to reality, their purpose, the way they interact with their users, the way they are used, their assumptions about the certainty of their data (i.e., deterministic vs. probabilistic models), their treatment of time (static vs. dynamic and continuous vs. discrete-state models), the kinds of questions they can answer, the kinds of answers they give, etc.

It is tempting to try to describe models in terms of their form, but the definition of form is subjective. For example, models can be described as physical or symbolic; however, physical (i.e., material) models are sometimes divided into "iconic" (or "schematic" [18]) and "analog" models, whereas symbolic models may be thought of either as strictly mathematical [7,14,18,34] or as non-material (i.e., including conceptual, or "mental", models) [21,33,39].

Furthermore, the physical/symbolic dichotomy is sometimes extended to include disparate terms, thereby subverting a simple material/immaterial interpretation. For example, simulation [14] or role-playing (i.e., "gaming") [18] may be added to the categories "physical" and "symbolic", producing heterogeneous classification schemes. In addition, many terms have multiple meanings in the literature; for example, "iconic" may include physical models of houses, engineering drawings, and maps [7,27,34], or it may be restricted to the former meaning (physical miniatures), while a different term (e.g., "analogue" [24], or "physical" model [18]) is used to include things like maps. On the other hand, the term "analogue" may be used to denote a physical model that uses analogue processes like water or electrical flow to model dynamic phenomena [7,27,34].

Mathematical models can themselves be classified as continuous vs. discrete and as deductive (proceeding from a priori knowledge or axioms), inductive (generalizing from observed behavior), or pragmatic (relying on a means-end oriented engineering approach) [47]. Analytical techniques (for which closed-form solutions exist, permitting optimization) are sometimes contrasted to numerical techniques [7]. Because of their formality, analytical techniques are seen as capable of representing only limited aspects of the real world [46]. Explicit computerized models (as compared to implicit mental models) are seen as having potential advantages including rigor, accessibility, comprehensiveness, logic, and flexibility [33,39].

Mathematical modeling encompasses the use of qualitative interaction (or "impact") matrix techniques like Leopold matrices, as well as numerical optimization techniques [22]. In addition to optimization (or "mathematical programming") based on network theory, PERT, calculus, etc., there are a wide range of stochastic techniques drawn from areas including queuing theory and inventory theory, as well as general statistical techniques including multivariate analysis (factor, principle component, discriminate analysis, etc.), statistical inference and decision theory [34]. Finally, no discussion of real-world modeling can ignore the issues of data availability, reliability, quality and relevance [18,22,23].

There is little consensus on how simulation relates to modeling, or even what the word "simulation" means. It is either thought of as (1) a way of using models that is more general than any particular kind of model [14,18,23,27,39] or (2) a specialized kind of model that makes use of a particular subset of the available modeling techniques [7,9,22,24,34]. Nevertheless, there is some consensus that simulation is a dynamic, imitative kind of modeling [9,14,18,24,27] that tends to be a technique of "last resort" used to model phenomena that are poorly understood or for which more rigorous techniques are unavailable [9,22].

The following discussion attempts to synthesize a coherent definition of modeling and simulation.

## Definition of Modeling

Precisely what do we mean by modeling? Modeling is a way of dealing with things or situations that are too "costly" to deal with directly (where "cost" is interpreted in the broadest sense). Any model is characterized by three essential attributes:

1) REFERENCE: It is of something (its "referent").

2) PURPOSE: It has some intended purpose with respect to its referent.

3) COST-EFFECTIVENESS: It is more cost-effective to use the model for this purpose than to use the referent itself.

To model, then, is to represent a particular referent cost-effectively for a particular purpose. ( Verbal forms like "modeling" and "to model" are used here to denote the entire enterprise of building and using models. (Some authors reserve these forms for the process of developing a model as opposed to using one [23]; however, this usage precludes saying that a model "models" its referent.) Phrases like "model building" will be used here to achieve the above distinction.

The referent and purpose of a model must be well-defined; otherwise all three criteria become meaningless. For example, a video game need not represent anything real; there may be some video games based on models (e.g., flight simulators), but most are "pseudo-models". The notion of a "game" implies an incidental relationship to reality, or the fabrication of a "pseudo-reality". It is tempting to "back-project" a pseudo-model into a corresponding pseudo-reality, thereby becoming convinced that the pseudo-model is a bona fide model [44]. However, in addition to being misleading, this process is also indeterminate, since the modeling abstraction cannot be reversed deterministically. Even if a purpose and cost-effectiveness criterion are fabricated for a pseudo-model, the fact that it can be back-projected into any number of equally possible pseudo-realities makes it worthless as a model.

The referent of a model need not actually exist, but it must be objectively testable in order to serve as "reality" for the model. It is reasonable to model a fictitious or hypothetical reality (for example, the psyche of Oedipus, the terrain of Camelot, or the flight characteristics of a proposed airplane), but only if the referent has some objective form against which the validity of the model can be verified.

The purpose of a model may include comprehension or manipulation of its referent, communication, planning, prediction, gaining experience, appreciation, etc. In some cases this purpose can be characterized by the kinds of questions that may be asked of the model. For example, prediction corresponds to asking questions of the form "What-if...?" (where the user asks what would happen if the referent began in some initial state and behaved as described by the model). This is analogous to applying "if-then" rules in the forward direction (i.e., "forward chaining").

On the other hand, goal-directed questions are concerned with finding an initial state or condition of the referent (along with constraints or conditions of the model itself) that can lead to a given result. This is analogous to the use of if-then rules in the backward direction (i.e., "backward chaining") or to mathematical optimization techniques.

There are also definitive questions that ask whether certain states, conditions, or actions are ever possible for the referent. These correspond to proving assertions about the referent by using the model. Finally, there are explanatory questions that seek to explain the behavior of the referent by showing how some state is reached or what the referent's reasons are for acting in a certain way.

Even an exhaustive list of such questions could not characterize all possible purposes of a model. A model may be intended for appreciation of its referent, in which case its user may not ask any questions of it at all. The purpose of a model is constrained only by the ingenuity of its builder and user.

It is impossible to evaluate--or intelligently use--a model without understanding its purpose. Calling something "a model" of its referent without further qualification makes it impossible to know which aspects of the referent are being modeled and which are not. No model can faithfully reproduce all aspects of its referent (since only the referent itself can do this). Therefore, without specifying its intended purpose, it is almost impossible to prevent using a model for purposes for which it may be highly inappropriate. This can have dire consequences if decisions and actions are based on false predictions or understanding. Similarly, it is imperative to have a clear statement of the intended purpose of a model before trying to build it: otherwise it is impossible to decide which aspects of the referent must be modeled and with what fidelity.

Yet knowing a model's purpose is not enough. It must also be more cost-effective to use the model for the given purpose than to use its referent, either because it is impossible to use the referent directly or because using the referent would be dangerous, inconvenient or (generally) expensive in some relevant coin. This cost-effectiveness criterion is central to the notion of modeling: without it, there is never any reason to use a model in place of its referent. The cost-effectiveness criterion of a model must be known in order to judge the model's value.

Judging the cost-effectiveness of a model requires answering two questions: "What does it claim to buy?" and "Does it buy this?". In addition, building a model requires asking two prior questions: "Is this the most appropriate thing to be bought by the proposed model?" (i.e., "Is this the most appropriate cost-effectiveness criterion on which to base the proposed model?") and "Will the model's cost-effectiveness pay for the cost of building it in the first place?".

The cost-effectiveness criterion is a kind of Occam's Razor for modeling: it allows models of equal power to be compared and evaluated. However, whereas Occam's Razor applies the criterion of simplicity (or "parsimony"), here the costs to be compared and evaluated are stated explicitly as part of the cost-effectiveness criterion. Since the criterion is not necessarily simplicity, it follows that a model is not necessarily simpler than its referent. A model may actually be more complex than its referent, if in so doing it satisfies some valid cost-effectiveness criterion other than simplicity.

Since a model cannot be identical to its referent, it is always an abstraction of its referent, in the sense that it can never be completely faithful to it. The fact that a model may be more complex than its referent implies that abstraction does not necessarily result in simplification, as is usually assumed. Although we sometimes appear to model something by using the thing itself, this always involves using the referent in some unusual way or restricted mode that offers some advantage over using it directly (an example of this is the modeling of human behavior by asking human subjects how they would act in hypothetical situations).

The criteria of purpose and cost-effectiveness for that purpose together determine which features of the referent must be modeled (and with what accuracy) and which features can be ignored. These criteria provide a complete functional characterization of a model. In addition, they determine a number of key "pragmatic" characteristics such as who the intended users of the model are and how the results of using the model must be presented in order to be usable (i.e., understandable) by those users. These can be thought of as interface issues: for a model to fulfill its stated purpose cost-effectively, it must be appropriately useful to and usable by its intended users.

There is also the pragmatic issue of how a model is to be maintained. This depends on how likely the model is to change and evolve over its lifetime, how extensible it needs to be, and who will be maintaining it. Accounting properly for these maintenance issues requires that the purpose of the model allows for its evolution over its entire projected lifetime and that its cost-effectiveness criterion considers the cost of maintaining it over this lifetime.

The above definition clarifies what are often blurred distinctions: namely, those among models, symbols and representations. A representation can be any use of something in place of something else: it need not (though it may) have a purpose or a cost-effectiveness criterion. That is, a model is a special kind of representation. Similarly, names and symbols are representations but are not models.

To some extent these distinctions depend on how something is used. A model may be used degenerately as a symbol for the thing it models: for example, the formula E=mc** is a mathematical model for the relationship between energy and mass, but the formula has become a popular symbol for all of Einstein's work and even modern physics as a whole. Similarly, an object may be usable as a model whether or not it was intended as one. The megalithic structure at Stonehenge may be interpreted as an astronomical model [19], but it is unlikely that that was its intended purpose [10].

## Examples of Different Types of Models

There are many ways of modeling a given thing or phenomenon, including physical analogs, closed-form mathematical representations, conceptual metaphors, linguistic formalisms, simulation, and many others. A given model (of any of these types) has strengths and weaknesses depending on its fidelity, utility, "computational cost" (i.e., the amount of work required to use the model), and pragmatic considerations including its suitability for various kinds of users and its maintainability. Whereas a given type of model may tend to have certain characteristics (for example, physical analogs tend to be more static and harder to modify than mathematical models), there are no invariant rules about which types of model display which strengths and weaknesses. Some examples will make this more concrete.

A street map is a physical analog that provides some kinds of information (such as connectivity) but usually not others (such as elevation). It can answer some kinds of questions easily (such as "Can I get from A to B?")

while others (such as "How long is the shortest route from A to B?") may require considerable computation [3]. It is used for comprehension, communication and planning; its cost-effectiveness derives from the difficulty of comprehending the layout of a city directly. It is relatively inflexible and hard to "maintain", either by the cartographer who creates it or by its user, who can at most add annotations to it.

Mathematical models come in many flavors. The darling of modern physics is the theory called Quantum Electrodynamics ("QED"), which describes the quantum mechanics. This model has achieved unprecedented accuracy of prediction over a range of dozens of orders of magnitude in scale [37]. However, it is relatively inaccessible and incomprehensible to all but physicists, and the cost of using it is relatively high even for the most mathematically astute and computationally well-armed.

Formal logic has made great strides in recent years, with the advent of efficient algorithms for the constructive proof of certain restricted classes of assertions [4,40]. This has resulted in a new generation of computer programming languages, typified by Prolog [48]. Models based on these formalisms have a compelling similarity to the "natural logic" of everyday language. For example, it is easy to write a Prolog program that defines intuitive models of the relationships in a family, and then to ask questions like "Who are John’s sisters, cousins, and aunts?". Unfortunately, such models are relatively opaque to all but Prolog programmers.

Conceptual metaphors are models consisting of ideas that shape the way we think about reality [16]. They introduce the intentional fiction that the referent is like some other better-known object or phenomenon. For example, the development of mechanical clocks had a profound influence on literary, philosophical and religious models of the universe. Similarly, the simplistic view of the atom as a miniature solar system provides comprehension, though it has poor predictive power. Closer to home, the metaphor of the human brain as a computer is one of the driving motivations for AI. Conceptual models like these form the paradigms that shape the thought of science and society as a whole [31].

Simulation is a form of modeling whose purpose is usually comprehension, planning, prediction and manipulation. It can be defined broadly as a behavioral or phenomenological approach to modeling: that is, a simulation is an active, behavioral analog of its referent. The essence of simulation is that it unfolds over time. It models sequences and (possibly) timings of events in the real world. Simulation is a process in which a model of any kind is used to imitate (some aspect of) the behavior of its referent. Simulation is a kind of modeling rather than a kind of model: it denotes an action (process) rather than a thing. However, the term is often used as a modifier of "model" (i.e., "simulation model"), with the word "model" itself often being omitted in such cases (for example, when speaking of a "weather simulation"). Nevertheless, what is meant in these cases is the use of a model as a simulation.

Simulation is generally used to answer "What-if...?" questions. It can also be used to answer questions of causality by generating a sequence of events from which one can attempt to infer what caused what. As traditionally conceived, simulation works only in this "forward" direction: the user "winds it up" and lets it run, to see what happens.

In some cases, one type of model may evolve or transmute into a different form. For example, there is some evidence that writing may have evolved from the use of physical analogs [42]. This putative evolution highlights the difficulty of distinguishing too sharply between physical analogs and symbols.

## Choosing Among Types of Models

The three criteria of reference, purpose and cost-effectiveness provide complete functional and pragmatic requirements for a model. But given a purpose with respect to some reality and a measure of cost-effectiveness, what determines which type of model should be used?

Most of the tradeoffs among the types are in terms of pragmatic issues (flexibility, extensibility, and suitability to different user groups) rather than among their functional abilities to fulfill various purposes cost-effectively. Furthermore, these pragmatic tradeoffs are often relative; for example, differential equations may be comprehensible to mathematicians, whereas intricate physical analogs may be more comprehensible to those with well-developed mechanical intuition. Conceptual metaphors have the advantage that they are immaterial and therefore require no apparatus for their use; in addition, they tend to be relatively simple and therefore accessible to a large community of users. Yet the lack of substance limits their computational power and may make them inaccessible to users who find abstractions hard to grasp. In contrast, physical analogs are highly tangible, but therefore have limited accessibility and may be difficult to modify and maintain.

Even with strong pragmatic constraints, the choice of which type of model to use is rarely determined by the requirements. The preferences and convenience of the model builders and users may ultimately dictate one type over another, but there are often several viable alternatives.

The following sections concentrate on one particular form of modeling--namely computerized simulation--not because it is best but because of its relevance in the context of this book.

## Computer Simulation

Implementing a simulation as a computer program results in unsurpassed flexibility; the malleability of the programming medium means that in principle (acknowledging the difficulty of producing programs without bugs) it is possible to refine, evolve, and extend a computer-based simulation in ways that are difficult to match in any other medium. Modern programming environments also facilitate the development of modular data and program code that (again, ideally) allow new simulations to be built using pieces of existing ones.

Computer simulation can be divided into analytic and discrete-state approaches. The analytic approach brings the power of mathematical analysis to bear on problems that can be understood or approximated analytically. For example, in cases where the reality being modeled can be accurately described by a set of differential equations (as in the flow of heat over a surface), analytic solutions of those equations can be used to generate the time-dependent behavior required for simulation.

Though closed-form solutions are often mathematically elegant, this very elegance may make them cryptic and incomprehensible. By reducing reality to an abstract mathematical relationship, they may obscure the understanding that is being sought. There are also cases in which analytic solutions are known, but feasible means of computing these solutions are not available. Nevertheless, analytic simulations are indispensable in many situations, particularly when dealing with complex physical phenomena involving vast numbers of relatively small and relatively similar entities whose individual interactions are relatively simple and whose aggregate interactions obey the "law of large numbers" (that is, permit statistical treatment). In such cases, analytic models often represent at least one form of "complete" understanding.

There remains a large class of problems, however, that are not well enough understood to be handled analytically, i.e., for which no formal mathematical solutions exist. These problems usually involve small to large (but not "vast") collections of interacting entities each of whose behavior is understood reasonably well in isolation and whose low-level, pairwise interactions with each other are known but whose high-level, group interactions are not well understood. The strategy of discrete-state simulation is to encode the known low-level interactions and "run" the resulting simulation in the hope that the overall behavior of the system will approximate that of its referent and (ideally) that higher-level interactions will reveal themselves.

Time is dealt with in discrete-state simulations as a succession of separate "states" in which entities interact; time advances discretely, either in fixed "ticks" of a simulated clock (referred to as "time-based" simulation) or whenever something significant happens (referred to as "event-based" simulation).

Discrete-state simulation can be viewed as a last resort for modeling certain kinds of intractable problems. Its power lies in its ability to reveal high-level patterns of interaction that cannot be recognized in other ways. It is often possible to enumerate and describe a collection of entities and their immediate interactions without knowing where these interactions lead; if this knowledge is encoded in a discrete-state simulation and the behavior of the resulting model is observed, deeper understanding will often emerge.

## A Modeling Perspective on AI

Artificial Intelligence is one of the frontiers of computer science. It has traditionally been concerned with problems that have not yet yielded to solution by "conventional" means. The quest for computer intelligence has two distinct motivations, which might be referred to (guardedly) as "modeling" and "engineering". The modeling approach seeks to model the way we perform tasks that require intelligence; it therefore attempts to identify problems that we recognize as requiring intelligence, and it seeks to elucidate the mechanisms we employ in our own solutions of those problems. The engineering approach, on the other hand, is concerned with producing systems that solve useful problems, regardless of whether their solutions require "intelligence" or involve mechanisms parallel to our own.

The modeling approach to AI begins from a psychological or philosophical departure point: given a conceptual theory of intelligence, can we embody that theory in a computer model? Computer models make such theories concrete, allowing them to be tested, validated, and refined. The modeling approach to AI therefore views the implementation of computerized models as a key technique for understanding intelligence. In addition, these models often suggest novel mechanisms that may become part of the conceptual theory itself. It is this kind of feedback that has led to the popular conception of the brain as a computer. In addition, insights gained from AI models (as often from their failures as from their successes) have contributed to major theoretical revisions in areas ranging from linguistics to cognitive psychology.

The engineering approach to AI has a different departure point: since computers are not organisms, why not use them to their best advantage to try to solve useful problems, without worrying about whether they are solving them the way we would? This approach works in symbiosis with the modeling approach; when a given model fails to work, sound engineering often suggests a solution. While these solutions are sometimes ad hoc, they may reveal flaws in the conceptual theory that engendered the model, thereby suggesting revisions to the theory. The engineering approach to AI has the sometimes frustrating attribute that, whenever it succeeds in solving a problem (or even approaches success), its results tend to be appropriated by "conventional" computer science or engineering, with the result that AI receives no credit for the eventual solution. This has occurred repeatedly in AI's history (examples include list processing, character recognition, speech synthesis, and demons), contributing to the only partially facetious adage that "AI never solves any problem--by definition".

AI has made many contributions to computer science and software engineering. It is arid to try to distinguish too sharply between what is AI and what is not; at any given time there are a set of unsolved problems in computer science to which AI has laid claim. Often these problems are also attacked from other quarters of computer science, and it is not always easy to assign credit for the solutions that eventually emerge. It is sufficient to note that AI has had some part in solving--or is currently attempting to solve--a number of problems that have direct bearing on simulation and modeling. Of particular relevance are the object-oriented programming paradigm, demons, planning, search techniques, taxonomic inference via inheritance hierarchies, forward and backward chaining, qualitative reasoning, truth maintenance, proof procedures for formal logic, neural nets, and the representation of spatial and temporal phenomena, uncertainty, plans, goals, and beliefs.

The next section can only hint at some of the most important areas of overlapping research and cross-fertilization between AI and simulation.

## AI in Simulation and Simulation in AI

The term "simulation" is traditionally taken to mean only a very specific kind of modeling. Having gone to the trouble of encoding the requisite knowledge for building a simulation, one should attempt to derive the maximum benefit from this knowledge. That is, in addition to "running" the simulation to answer "What-if...?" questions, one should be able to utilize the full range of inferencing, reasoning, and search methods that are available in AI. This broad view of simulation is referred to as Knowledge-Based Simulation.

There is a well-entrenched tendency to view simulation narrowly as a way of making predictions by running an encoded behavioral model. The major impact of AI on simulation should be to encourage simulation to make use of other kinds of modeling as well: the result will still be a phenomenological model but one that can take full advantage of additional modeling techniques to answer questions that are of interest to its users. This natural, though long-overdue, extension of simulation can be referred to as Beyond "What-if...?".

Discrete-state simulation has derived great benefit from many of the techniques developed in AI. The object-oriented paradigm, though it first appeared in Simula [8], owes its present state of refinement to AI language efforts like Smalltalk [17] and ROSS [32]. The object-oriented approach to discrete-state simulation has many advantages, despite its shortcomings [41]. For example, the appropriate use of inheritance hierarchies (or lattices) greatly simplifies the specification of a complex simulation, producing highly comprehensible models [29]. Searching and planning techniques developed in AI should make it feasible to simulate the behavior of human decision makers in environments involving "command and control", while backward chaining should help answer questions about how to achieve a given result. Techniques for representing goals and beliefs should help build simulations that can explain the behavior of simulated entities.

Analytic simulation has tended to look to mathematics rather than AI for its methods, but here too there are possibilities on the horizon. One example is recent work at The RAND Corporation in sensitivity analysis (a sorely neglected problem in simulation), which uses AI techniques to represent and propagate sensitivity information through a computation. This avoids the need to recompute sensitivity for every nested function call whenever some higher-level function is perturbed to probe its sensitivity to changes in its parameters. We also foresee the use of symbolic algebra programs like REDUCE [20] to apply expert algebraic manipulation to analytic functions within a simulation.

AI programs have a long history of using models as sources of internal expertise. An early example is Gelernter’s Geometry Machine [15] which embedded a model of a geometry student’s diagram, and used a "diagram computer" to test hypotheses against this internal diagram. The Geometry Machine’s stated motivation was to solve problems generally considered to require intelligence; here the "engineering" approach converged with the modeling approach in choosing a solution based on a model of how we ourselves solve geometry problems: being inveterate modelers, we use a model (i.e., a diagram).

Another classic example of an embedded model in an AI system is SOPHIE [5], which taught electronic circuit diagnosis by means of an interactive dialog. In order to allow students to ask hypothetical questions (e.g., "What would happen if I measured the voltage across points A and B?") SOPHIE used a simulator of the electronic circuit being diagnosed. Here the simulator was treated as a source of expertise about electronic circuits. The AI program that conducted the dialog with the student did not attempt to know all the answers to all possible questions the user might ask; instead, it answered those questions by consulting its internal model of reality, i.e., running its embedded simulation.

It is generally acknowledged that in order to exhibit more than superficial intelligence, AI systems must make use of "deep structures", or models of reality like those described above. Simple action-response rules can produce programs that perform impressively up to a point, but beyond that point there is no escaping the need to imbue programs with real "understanding" of the world, at least within their domains. The way to provide such understanding is to endow a program with a model of the world that it can use to answer a wide range of unanticipated questions arising from its need to act (or reply to queries) appropriately in that world. These ideas are discussed further in the next section.

## Knowledge-Based Simulation at The RAND Corporation

A number of research efforts are currently attempting to blend AI and simulation in a new discipline called Knowledge-Based Simulation. In order to elaborate the ideas of the previous section, the following describes our current research in this area at The RAND Corporation.

Artificial intelligence and simulation have been major areas of research at RAND for many years [30]. The work of Newell, Shaw and Simon at RAND in the 1950s [36] was one of AI's earliest successes and defined many areas that continue to be focal points for AI research. More recently RAND's research in expert systems produced the languages RITA [1,2] and ROSIE [26,45] as well as several expert system applications (including LDS [51], TATR [6] and SAL [38]). Similarly, RAND's long history of simulation research produced the Simscript language [28] as well as both theoretical and experimental results in game theory and monte carlo simulation. RAND began applying AI to simulation in the late 1970s and early 1980s. The development of the object-oriented ROSS language clearly demonstrated that AI could benefit simulation technology. The Knowledge-Based Simulation project continues this tradition.

The goal of the KBSim project is to make simulations both more powerful and more comprehensible by (1) allowing modelers to build, validate, evolve and maintain more powerful and realistic simulations that model a wider range of relevant phenomena, and (2) allowing users to interact with these simulations in ways that provide deeper understanding of the phenomena being modeled. Making simulations more powerful requires extending the kinds of modeling they can perform and the kinds of questions they can answer (as discussed above). Making simulations more comprehensible requires developing techniques for what we call intelligent exploration and explanation, i.e., allowing users to modify both the model and the course of events in a simulation, and making the simulation explain its behavior in useful ways. Our context is the object-oriented, discrete-state simulation of objects in a geographical setting.

This research has spawned a number of distinct tasks, the first of which involves reasoning about simulation behavior. This includes being able to ask goal-directed questions, questions about whether or how an initial state can produce a desired result, questions about the possible values of variables in a simulation, questions about the interactions of objects or factors, questions about the goals of an object, and questions about why an object performed an action. The inability of current discrete-state simulations to answer such questions derives from limitations in their representational and inferential capabilities stemming from the fact that knowledge is represented implicitly in procedural code and is therefore not amenable to inference. Support for reasoning requires representing the behavior of objects in ways that allow the use of automated reasoning techniques (like forward and backward chaining) and integrating these with other forms of inference, such as those based on the use of object taxonomies.

In addition to the explicit use of reasoning, it is important to allow implicit reasoning based on multiple relations. Complex simulations require the representation of multi-dimensional relationships among objects, such as "A is a-kind-of B", "A is a-part-of B", "A is in-control-of B", "A is in-communication-with B", or "A is near B". It is vital for the simulation user to be able to define relations freely, examine the state of the simulation in terms of these relations, and modify them dynamically. Most object-oriented systems support only minor variations of the "class-subclass" (also called "IS-A" or "taxonomy") relation along with a corresponding "inheritance" mechanism to maintain taxonomic relationships (i.e., specialized inferential support for the class/subclass relation). We are attempting to provide a true multiple relation environment in which different kinds of relations are supported by appropriate specialized inference mechanisms and to provide a general facility to allow the simulation developer to define new relations with appropriate inferential support.

In order to be comprehensible to users, simulations must make intelligent use of highly interactive graphics interfaces. These should allow: graphic querying of the simulation state; being able to roll the simulation back to a previous state, change a parameter, and rerun the simulation; saving multiple simulation states for later analysis and comparison; being able to build or modify simulation scenarios graphically; and being able to build or modify simulation objects graphically (e.g., defining and exercising new behaviors graphically). We have defined a highly interactive graphics environment of this sort that emphasizes the ease of manipulating simulation objects, minimizes redundant display updating, and facilitates animation of sequences of events (i.e., "causal chains"). We are also investigating the use of graphically interactive diagrams or pictorial representations of relations, which users can display and edit graphically.

Sensitivity analysis is one of the great abandoned areas of simulation. Yet without it there is no guarantee that the results of a simulation would not be drastically different if some small change were made to some initial parameter. Sensitivity analysis is also important for indicating which parameter values are the most important to verify (by real-world means) for a simulation to be valid and believable.

The straightforward approach to sensitivity analysis requires running a simulation many times perturbing individual parameters to see how the results differ. This is prohibitively expensive in most cases, as a consequence of which it is rarely done. Our research seeks to provide a means of computationally feasible sensitivity analysis in a simulation environment, utilizing a new approach that propagates and combines the sensitivities of composite functions through a computation. Viewing a simulation as a top level function that invokes many levels of nested subfunctions, most of these invocations normally involve a relatively small number of distinct subfunctions, each of which is called many times. For example, a sine function may be called thousands of times in computing the geographical positions of objects. Normally, perturbing a top level parameter involves executing the top level function several times, each time executing the nested sine function thousands of times. Our approach instead computes a representation of the sensitivity of the sine function the first time it is executed, and propagates this sensitivity information through the computation rather than recomputing it each time it is needed.

Another major shortcoming of current simulation models is their inability to vary the level at which they are aggregated (also referred to as their "resolution"). It is generally necessary to choose a desired level of aggregation in advance and design a simulation around that level. Changing this level typically requires considerable reprogramming of the simulation; changing it under user control or dynamically is generally unthinkable. The fact that the level of aggregation of a model gets "frozen in" early in its design is a major impediment to the reusability of models and the utility of simulation in general. Users should be able to vary the level of aggregation of a simulation and to indicate which aspects of the model are of particular interest, running those aspects of the simulation disaggregated while running peripheral aspects at higher levels of aggregation. Users should also be able to run highly aggregated simulations to identify interesting cases and then examine those cases in more detail by rerunning them disaggregated. Our goal is to develop a methodology for building simulations whose level of aggregation can be varied either statically or dynamically, as appropriate to the user’s purpose. This requires mechanisms for representing "vertical slices" of objects in an aggregation hierarchy and for allowing interactions between objects at different levels of aggregation. It is also necessary to address problems of inconsistency that can arise between different levels: that is, running a simulation at an aggregated level should produce results that are consistent with the results of running the same simulation at a disaggregated level.

In order to model real-world environments that include human decision makers, we are attempting to build simulations that embed models of intelligent agents possessing varying degrees of awareness, authority, initiative, intelligence, etc. This also requires hierarchical planning so that, at each level, plans will be translated into objectives for agents at the next lower level.

Finally, there are a number of "pseudo-objects" or phenomena that are not modeled well by the current object-oriented paradigm. For example, terrain, roads, rivers, and weather defy easy representation by conventional object-oriented means. These pseudo-objects seem to require representations and manipulations that are different from those used for more compact objects, either because they traverse or interpenetrate other objects (without actually being "part" of them), or because they are best described by continuous models (such as partial differential equations). We are exploring a number of AI techniques to represent such pseudo-objects and their interactions.

The above gives a brief summary of our current research in this area. The wedding of AI and simulation is still in progress; its consummation promises to be of great value to both fields of endeavor.

## Summary and Conclusion

Modeling is one of the foundations of intelligence and is an essential aspect of being human. It is nothing short of the primal lever with which we move the Earth to suit our needs. However, the creation and use of models is so instinctive that we rarely analyze it or question its appropriateness. This chapter has attempted to define modeling precisely in order to provide a framework within which models can be compared and evaluated according to the three criteria of reference, purpose and cost-effectiveness. This should make it possible to design and choose more effective models, avoid inferior ones, recognize pseudo-models, and acknowledge that the use of any model may sometimes be inappropriate.

Simulation is a powerful modeling strategy for understanding complex phenomena. Artificial Intelligence provides new techniques that will infuse simulation with unprecedented power. At the same time, simulation has a great potential for contributing to AI, providing phenomenological models that can be used by reasoning programs as a deep source of expertise about the world. Some of the most exciting potentials of this new marriage of disciplines are exemplified by current research in Knowledge-Based Simulation at The RAND Corporation.

It is apparent that the future of civilization depends, at least in part, on our ability to devise new and more effective tools for making intelligent decisions. Whatever form these tools may take, they must necessarily be based on appropriate models of ourselves and our reality. It is perhaps not too much to ask--or at least to hope--that the informed use of AI and simulation may play a major role in producing these vital tools. It is in any case clear that current developments in this area will greatly enhance the efficacy of simulation as a means of understanding the world.

## REFERENCES

Anderson, R.H., J.J. Gillogly, 1976. RAND Intelligent Terminal Agent (RITA): Design Philosophy, The RAND Corporation, R-1809-ARPA.

Anderson, R.H., M. Gallegos, J.J. Gillogly, R.B. Greenberg, R.V. Villanueva, 1977. RITA Reference Manual, The RAND Corporation, R-1808-ARPA.

Anderson, R.H., N.Z. Shapiro, 1979. Design Considerations for Computer-Based Interactive Map Display Systems, The RAND Corporation, R-2382-ARPA.

Bibel, W., 1982. Automated Theorem Proving, Viewweg, Braunschweig.

Brown, J.S., R.R. Burton, J. DeKleer, 1982. "Pedagogical, Natural Language and Knowledge Engineering Techniques in SOPHIE I, II, and III", in D. Sleeman, J. S. Brown (eds.) Intelligent Tutoring Systems, New York: Academic Press, pp. 227-282.

Callero, M., D.A. Waterman, J.R. Kipps, 1984. TATR: A Prototype Expert System for Tactical Air Targeting, The RAND Corporation, R-3096-ARPA.

Catanese, A.J., 1972. Scientific Methods of Urban Analysis, University of Illinois Press.

Dahl, O-J., K. Nygaard, 1966. "Simula--An Algol-Based Simulation Language", Communications ACM, Vol. 9, pp. 671-678.

Dalkey, N.C., 1968. "Simulation" in Systems Analysis and Policy Planning: Applications in Defense, E.S. Quade & W.I. Boucher eds., Elsevier.

Daniel, G., 1980. "Megalithic Monuments", Scientific American, vol. 243, No. 1, pp. 78-90.

Davis, M., S. Rosenschein, N. Shapiro, 1982. Prospects and Problems for a General Modeling Methodology, The RAND Corporation, N−1801−RC.

Evans, G.W., G.F. Wallace, G.L. Sutherland, 1967. Simulation Using Digital Computers, Prentice−Hall.

Frisch, A.M., 1985. "Using Model Theory to Specify AI Programs", Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'85).

Gass, S.I., R.L. Sisson, 1974. A Guide to Models in Governmental Planning and Operations, U.S. Environmental Protection Agency.

Gelernter, H., 1959. "Realization of a geometry theorem−proving machine", Proceedings of an International Conference on Information Processing, Paris: UNESCO House, pp. 273−282.

Gentner, G., A.L. Stevens (eds.), 1983. Mental Models, Lawrence Erlbaum.

Goldberg, A., A. Kay, 1976. Smalltalk−72 Instruction Manual. Report SSL 76−6, Xerox PARC, Palo Alto, California.

Greenberger, M., M.A. Crenson, B.L. Crissey, 1976. Models in the Policy Process, Russell Sage Foundation, NY.

Hawkins, G.S., 1965. Stonehenge Decoded, Doubleday.

Hearn, A.C., 1985. REDUCE User's Manual, Version 3.2, The RAND Corporation, CP78.

Hogwood, B.W., L.A. Gunn, 1984. Policy Analysis for the Real World, Oxford Univ. Press.

Holling, C.S., 1978. Adaptive Environmental Assessment and Management, John Wiley.

House, P.W., J. McLeod, 1977. Large−Scale Models for Policy Evaluation, Wiley.

Hughes, W.P., 1984. Military Modeling, The Military Operations Research Society, Inc.

Kant, I., 1783. "Prolegomena to Any Future Metaphysics", Liberal Arts Press 1950.

Kipps, J.R., B. Florman, H.A. Sowizral, 1986. The New ROSIE Reference Manual and User's Guide, The RAND Corporation, R−3448−ARPA.

Kiviat, P.J., 1967. Digital Computer Simulation: Modeling Concepts, The RAND Corporation, RM−5378−PR.

Kiviat, P.J., R.V. Villanueva, H.M. Markowitz, 1968. The Simscript II Programming Language, Prentice−Hall, Englewood Cliffs, New Jersey.

Klahr, P., 1985. "Expressibility in ROSS: An Object−Oriented Simulation System". AI Applied to Simulation, Proceedings of the European Conference at the University of Ghent, 136−139.

Klahr, P., D.A. Waterman, 1986. "Artificial Intelligence: A Rand Perspective", in Expert Systems Techniques, Tools and Applications, Addison−Wesley, pp. 3−23.

Kuhn, T.S., 1962. The Structure of Scientific Revolutions, University of Chicago Press, Chicago.

McArthur, D., P. Klahr, S. Narain, 1984. ROSS: An Object−Oriented Language for Constructing Simulations, The RAND Corporation, R−3160−AF.

Meadows, D.H., J. Richardson, and G. Bruckmann, 1982. Groping in the Dark, Wiley.

Meta Systems, Inc., 1971. Systems Analysis in Water Resource Planning, Water Information Center, Inc.

Minsky, M., 1965. "Matter, Mind and Models" Proceedings of the IFIP Congress 65, pp.45−49.

Newell, A., J.C. Shaw, H. Simon, 1957. "Empirical Explorations with the Logic Theory Machine", in The Proceedings of the Western Joint Computer Conference.

Pagels, H.R., 1983. The Cosmic Code, Bantam books.

Paul, J., D.A. Waterman, M.A. Peterson, 1986. "SAL: An Expert System for Evaluating Asbestos Claims", in The Proceedings of the First Australian Artificial Intelligence Congress, Melbourne.

Quade, E.S., 1985. "Modeling Techniques" in Handbook of Systems Analysis, H.J. Miser, and E.S. Quade eds., North−Holland.

Robinson, J.A., 1983. "Logic programming−−Past, present and future", in New Generation Computing, Vol. 1, pp. 107−124.

Rothenberg, J., 1986. "Object−oriented Simulation: Where Do We Go From Here?", Proceedings of the 1986 Winter Simulation Conference, Washington, D.C., pp. 464−469.

Schmandt−Besserat, D., 1978. The Earliest Precursor of Writing, Scientific American, June 1978, pp. 50−59.

Simon, H.A., et al., 1986. Scientific Futures: Selected Areas of Opportunity, Report of the Research Briefing Panel on Decision Making & Problem Solving.

Smith, B.C., 1986. (Personal communication).

Sowizral, H.A., J.R. Kipps, 1985. ROSIE: A Programming Environment for Expert Systems, The RAND Corporation, R−3246−ARPA.

Specht, R.D., 1968. "The nature of models" in Systems Analysis and Policy Planning: Applications in Defense, E.S. Quade & W.I. Boucher eds., Elsevier.

Spriet, J.A., G.C. Vansteenkiste, 1982. Computer−aided Modeling and Simulation. Vansteenkiste (U of Ghent, Belgium) Academic Press.

Sterling, L., E. Shapiro, 1986. The Art of Prolog, MIT Press.

Tarski, A., 1956. "The Concept of Truth in Formalized Languages" in Logic, Semantics, and Metamathematics, Oxford at the Clarendon Press.

Walker, W.E., J.M. Chaiken, E.J. Ignall eds., 1979. Fire Department Deployment Analysis, North−Holland.

Waterman, D.A., M.A. Peterson, 1981. Models of Legal Decisionmaking, The RAND Corporation, R−2712−ICJ.