AI: WHAT SIMULATIONISTS REALLY NEED TO KNOW

David P. Miller
R. James Firby

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109

Paul A. Fishwick

Department of Computer and
Information Science
University of Florida
Gainesville, Florida 32611

Jeff Rothenberg

The RAND Corporation
1700 Main Street
Santa Monica, California 90406

David W. Franke

Microelectronics and Computer
Technology Corporation
3500 West Balcones Center Dr.
Austin, Texas 78759-6509

ABSTRACT

As simulationists strive to make their simulations more accurate, and more efficient, they are forever looking for new, more advanced programming techniques. Artificial Intelligence (AI) is basically the field of advanced programming techniques. While these techniques were originally developed for modeling cognitive processes or the behavior of cognitive beings, many of these techniques are applicable to the more general simulation audience. This paper presents five short essays by researchers is simulation, AI, and a couple who have feet in both camps.

1. DAVID P. MILLER: AI FOR SIMULATION IS MORE THAN JUST A BAG OF TRICKS

1.1 AI Techniques Have Their Place

Simulation's opinion of AI, like most people's, ranges from believing that AI can solve everything to believing that AI is just a crock. In general though, certain practices of AI such as hierarchies of abstraction, object oriented programming, and specialized programming environments or shells have become pervasive throughout simulation (though some might argue that those ideas originated in simulation and have become pervasive throughout AI).

Though these techniques are pervasive, they are not universally adored. OOP usually runs slower than do traditional programming techniques. Shells allow you to do the things they were designed for very easily, but if you are trying to do something else, there can be a huge learning curve to overcome. For people who have had these problems, they have tried AI and found it wanting. For those who have avoided those pitfalls there response to the AI question is too often "Yes, we use AI techniques in our simulations" (translation: we use KEE or C++ to write our simulations) "for example, we use advanced inferencing techniques" (translation: if statements) "and models of our domain." (translation: numerical databases).

AI has many things to offer the creators of simulation. Some of these things like data-dependencies, backward chaining, and abduction can be used to greatly simplify the creation of simulations. However, they are not a universal panacea, and must be used to sparingly to avoid a combinatorial black hole. Other techniques, such as qualitative modeling should allow things to be simulated that simply no one knew how simulate before. Behaviorally directed autonomous agent techniques is a new area of AI. However, it holds great promise for simulation to allow the simulator to perform an actual event simulation (where there are multiple independent agents being modeled) rather than relying on a traditional statistical or exhaustive model. Finally there exist AI technologies such as Neural nets. Neural Nets offer the promise of greatly increased computing and self tuning systems. However, in most current implementations neural nets operate only in simulation, so to try and get them to solve some of simulation's practical problems is almost certainly premature.

1.2 AI Philosophy for Simulation

New techniques may be helpful, but the most important contribution AI can make to simulation, is a philosophical one. Unlike most of the rest of computer science, AI has largely abandoned the idea of deterministic algorithms, in favor of heuristic techniques. The problems that AI systems are usually designed to handle are either to poorly understood to solve, or have known solutions that take geologic time to run. The lesson of AI is that techniques that handle eighty percent of the possible situations can work one-hundred percent of the time; if the right part of the problem space is the one always under consideration. The odd situations, which make otherwise simple algorithms NP-complete, may never actually occur. Or they occur so seldom that you can catch those situations, and handle them specially.

Most of the problems that crop up in the world have cropped up before, and they can usually be handled in a similar way. Case-based and explanation-based reasoning techniques offer direction on how to "solve", in the AI sense, these problems.

In real life, optimality is often desirable, but seldom necessary. Near optimal solutions are usually indistinguishable from the real thing, and the speed of the answer usually more than makes up for any increased inefficiency. Heuristic search techniques, qualitative models, and approximation techniques can solve the "real" problem much of the time.

The real-world is a very complex place. Research in robotics has shown that things seldom go as planned. Things seldom go exactly as simulated either. It therefore seems wise for simulationists to adopt a more AI-like philosophy. By lowering your standards of *precision*, it may be possible to increase the *accuracy* of your simulations. While the fidelity of each iteration might be slightly lower, the increased number of situations that could be explored might prove more valuable, in many situations.

1.3 Conclusions

AI is both a set of programming techniques, and a state of mind. Most of the transfer of AI to simulation has only been in the form of programming techniques. This has been due to the transfer being done almost exclusively by simulationists. It is much easier to adopt another fields techniques than it is their philosophy.

One need to be selective in both the AI techniques used, and how and when they are applied. One must avoid the AI wannabe syndrome where you try and put every technique you know about into your system. Likewise, one should not avoid all AI because you talked to an AI researcher one time and he was a real flake. Through the careful application of AI techniques, simulation can improve its performance, fidelity, and scope of applicability. The techniques currently used by simulationists have only scratched the surface of what is possible.

However, simulationists will continue to have some difficulty profitably applying AI techniques to their simulations unless they also adopt a bit of AI philosophy. AI offers the possibility of more than just allowing simulationists to do what they have been doing, only better and faster. AI philosophy, combined with new techniques offer simulationists the chance to tackle problems they would not have considered addressing before.

2. JEFF ROTHENBERG: EXPANDING THE BOUNDS OF SIMULATION

In the context of AI, the term "simulation" must be freed from its own tradition, where it often denotes a very limited form of modeling. There is a strong tendency in simulation circles to view simulation narrowly as a way of making predictions by running an encoded behavioral model ("winding it up and letting it run") to answer "What-if?" questions. This can be thought of as the "toy duck" view of simulation [Rothenberg, et al. 1989].

Since a great deal of effort is required to encode the knowledge needed to build a simulation, one should attempt to derive the maximum benefit from this knowledge. In particular, in addition to "running" a 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. These methods should be able to explain why a given sequence of events occurred and answer definitive questions such as "Can this event ever happen?" or "Under what circumstances will this happen?" and goal-directed questions such as "Which events might lead to this event?". This broad view of simulation is sometimes referred to as Knowledge-Based Simulation.

The major impact of AI on simulation is (or should be) to encourage simulation to make use of a wider range of modeling techniques: the result will still be a phenomenological model, but one that can take full advantage of additional techniques to answer a wider range of questions that are of interest to its users. I refer to this natural, long-overdue extension of simulation as going "Beyond What-if?".

Discrete-state simulation has derived great benefit from many of the techniques developed in AI. The object-oriented paradigm, which first appeared in Simula [Dahl and Nygaard 1966], owes its present state of refinement to AI language efforts like Smalltalk [Goldberg and Kay 1976] and ROSS [McArthur, Klahr, and Narain 1984]. The object-oriented approach has many advantages, despite many shortcomings [Rothenberg 1986]. For example, the appropriate use of inheritance hierarchies (or lattices) greatly simplifies the specification of a complex simulation, producing highly comprehensible models [Klahr 1985]. Searching and planning techniques developed in AI have made feasible models that simulate the behavior of human decision makers in environments involving "command and control", while backward chaining can help answer questions about how to achieve a given result. Techniques for representing goals and beliefs have helped build simulations that can explain the behavior of simulated entities. Some of the current outstanding problems in discrete-state simulation, such as the problem of representing and computing continuous information like weather and terrain, may also yield to AI solutions.

Analytic simulation has tended to look to mathematics rather than AI for its methods, but here too AI offers some new approaches. One example is recent work in sensitivity analysis (a sorely neglected problem in simulation), where AI techniques are used to represent and propagate sensitivity information through a computation, so that it need not be recomputed for every function call whenever some higher-level function is perturbed to probe its sensitivity to changes in its parameters [Rothenberg, Shapiro, and Hefley 1990]. Similarly, symbolic algebra programs developed by AI, such as REDUCE [Hearn 1985], may allow applying expert algebraic manipulation to analytic functions within a simulation.

The relationship between AI and simulation is bilateral: AI has produced many systems that use models as sources of internal expertise. One of the earliest examples of this was Gelernter's Geometry Machine [Gelernter 1959], which embedded a model of a geometry student's diagram (itself a model), and used a virtual "diagram computer" to test hypotheses against this internal diagram. This has become a classic AI paradigm that expresses AI's recognition of the importance of models to intelligent agents: in seeking to model such agents, AI is naturally driven to model their use of models! In the case of the Geometry Machine, whose stated motivation was to solve problems generally considered to require intelligence, 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 [Brown, Burton, and DeKleer 1982], which taught electronic circuit diagnosis by means of an interactive dialogue (in English). In order to allow students to ask hypothetical questions such as "What would happen if I measured the voltage across points A and B?", SOPHIE used a simulator of the electronic circuit being diagnosed. This simulator was treated as a source of expertise about electronic circuits. The AI program that conducted the dialogue with the student did not encode answers to all possible questions the user might ask; instead, it answered those questions by consulting its internal model, i.e., running its embedded simulation.

There is considerable evidence 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 give programs real "understanding" of the world, at least within their domains. There are many possible approaches to providing such understanding, but they all essentially involve giving a program a model of its world that it can use to answer a wide range of unanticipated questions about that world.

3. DAVID W. FRANKE: WHAT SIMULATIONISTS NEED TO KNOW ABOUT THEIR PROBLEMS

While many AI techniques or technologies have established communities concerned with research into and extension of their respective techniques, it should be noted that the original motivation for each of these techniques has been the solution of specific problems. For example, production systems are used to solve problems (e.g. diagnosis) in domains for which there is no concise theory of the domain. Work in qualitative physics was originally motivated by the desire to reason about physical objects and real world phenomena exhibited by mechanical systems or physical processes (such as boiling liquids or kidney function).

This perspective of "What problem is being addressed?" can also be applied to the body of simulation techniques. It is claimed that this is the appropriate perspective when evaluating particular simulation techniques, whether they be AI techniques applied to simulation, or simulation techniques used in AI problem solving. Some of the ultimate goals of model specification and construction and subsequent simulation of these models in AI and other disciplines are:

- Construction of models of existing systems or hypothetical systems
- Derivation of dynamic descriptions from a static (structure and behavior specification of primitives) description of a system or mechanism. This has also been called behavior generation and functional evaluation (e.g. digital circuit simulation).
- Analysis of the dynamic description
  - ° Performance evaluation (temporal properties)
  - ° Properties of distributed systems (deadlock, fairness, ...)
- Evaluating theories about the real system/mechanism being modeled. Theory evaluation or validation is used in diagnosis, monitoring, and design applications.

- Prediction of system/mechanism behavior
- Explanation of system/mechanism behavior in causal and/orteleological terms

One can consider the goals described above in the contexts of the problem solving tasks of system design, system diagnosis, and system monitoring. In each task domain, a scenario of model development from initial, general qualitative models through mixed qualitative/ quantitative models to detailed, specific quantitative models can be applied. This scenario reflects a top-down refinement approach to problem solving, but can also incorporate a bottom-up approach via mixed qualitative/quantitative models. (It should not be inferred that every quantitative model is more specific or finer-grain than a qualitative model, but merely that across the full spectrum of models, qualitative models tend toward generalization, while quantitative models tend toward specialization.) While the obvious application of AI techniques is in 1) the simulation of qualitative and mixed qualitative/ quantitative models and 2) interpretation of the simulation results (design, diagnosis, explanation, or monitoring), another current research thread in the model-based reasoning community is the development and refinement of the models themselves. Techniques which assist the modeler interpret dynamic behaviors (simulation results) and take appropriate actions, particularly to modify or refine the model, should be applicable to the entire spectrum of models and simulation techniques, from qualitative to quantitative.

A Case For Qualitative Simulation:

Abstraction: In human problem solving, abstraction is an important technique for managing complexity. One characterization of human expertise is the ability to make the most appropriate abstraction in a particular domain, domain situation, and problem solving situation. Qualitative representations of system/mechanism primitive behaviors, constraints, state and behavior are one dimension in which abstraction applies (as opposed to eliminating specific components from a system/mechanism and the associated variables from the state). This abstraction dimension is in fact very useful, as demonstrated in human reasoning and programs that reason from such qualitative representations. For example, digital circuit simulation abstracts the actual voltages that exist in the circuit to logic values 0, 1, and X.

Completeness: In deriving behavior via simulation, qualitative simulation (e.g. QSIM) is complete in that all possible behaviors are represented in the envisionment (assuming that generation of such an envisionment is tractable). For numerical simulation approaches, the same claim cannot be made. It should be understood that qualitative distinctions of behaviors are dependent upon the specification of the system/mechanism (e.g. introducing a landmark into a variable's quantity space can result in qualitatively distinct behaviors not observed before the landmark was added). This, however, is the price of abstraction.

Operating with Incomplete Knowledge of the Domain: The qualitative model specification and simulation techniques developed in the AI qualitative reasoning community have emphasized the ability to proceed in the face of incomplete knowledge (theory or model) of the system/mechanism and any initial conditions. This is exhibited not only in qualitative variable and state values, but also in the expression of primitive behaviors use in system/mechanism description. For example, QSIM provides monotonic increasing (M+) and decreasing (M-) constraints, and Qualitative Process Theory expresses influences between variables. The ability to develop a model and simulate it in the presence of incomplete knowledge is important in that some initial information can be collected and subsequently used in problem solving and model refinement.

If one considers qualitative and numerical models and simulation techniques as points or areas on an abstraction spectrum, the problem of developing, validating, and maintaining theories about the domains of interest (either for humans or for autonomous agents) can be viewed as building and validating a theory at some point on the spectrum, and then possibly modifying the theory in the direction most appropriate for the task at hand (i.e. more or less abstract). For a design activity, the modification must necessarily go to a very fine level of detail so that the associated mechanism can be constructed. A diagnosis or explanation capability however, may not require such a fine grain description. In fact, for explanation or prediction purposes, a more abstract description is often appropriate (e.g. cyclic, or remains with limits). Issues in model construction and selection are an active area of research in the qualitative modeling and model-based reasoning communities. The integration of quantitative and qualitative information is also being investigated.

One of the central issues in AI has been knowledge representation. Issues of expressive power, tractability and completeness of inference procedures, and conceptual integrity with respect to the problem domain have guided research in representation. The problem domain governs ontological issues for the objects examined in the problem solving process (e.g. components of a mechanism, observations such as medical data, physical processes) as well as objects/concepts of the problem solving process itself (e.g. design goals, explanations). These representation issues (domain objects, problem solving process concepts) plus the goals of the problem solving technique provide an understanding of the current AI approaches to and uses of simulation (e.g. qualitative simulation). Particular choices are sometimes motivated by models of human problem solving, not with the goal of accurately modeling human problem solving activity, but with the goal of giving programs better problem solving capabilities.

The goal of (AI's) qualitative modeling research has been much discussed, ranging from the desire to faithfully model human cognition to the ability to build and utilize precise, accurate models of the real world. To repeat an earlier message, I believe that the appropriate context is the pragmatic one, in which the particular simulation or modeling approach can best be judged by 1) its ability to solve a particular problem and 2) the ability for humans or other programs (autonomous agents) to evaluate and utilize the results of the modeling. Unfortunately many claims of the form "Yes, we use AI techniques" are made for systems and products. One must be careful in evaluating such claims, and examine the problem solving capabilities as well as any implementation approaches.

4. PAUL A. FISHWICK: AI & SIMULATION: SOME LESSONS LEARNED[1]

4.1 Overview

The fields of Artificial Intelligence and Simulation are fairly large in terms of literature and interdisciplinary tendencies. Discussing, therefore, how the two relate to one another is a formidable task; however, we have learned many key points or "lessons" especially during the AI and Simulation workshops, conferences and panel sessions over the past several years. In this panelist position paper, I will discuss some things that I have learned during my time studying the benefits of AI and Simulation to each other.

4.2 Code All the Knowledge

Perhaps the chief contribution of AI to all fields, including Simulation, is the realization that knowledge that is non-equational or quantitative in nature can still be used for useful problem solving. The primary example of this type of AI research is found within "expert systems." We should point out that expert systems, from a problem solving viewpoint, are not unique because they represent expert knowledge per se. After all, continuous models for aircraft flight or discrete event models for assembly lines are also reservoirs of knowledge --- specifically, "expert knowledge" about

1. This research is made possible in part by a grant from the National Science Foundation IRI-8909152.

the principles of flight and the operation of assembly lines. What, then, makes an expert system unique? Expert systems have been built in those areas where models have been either very weak or non-existent such as in medical diagnosis; we do not have a simple set of equations that accept symptoms as inputs and produce a correct diagnosis as an output. Mycin [Buchanan and Shortliffe 1984] provides an excellent example of a program that contains the deepest knowledge available in the domain for which Mycin was coded: the selection of antimicrobial drugs given specific symptoms of bacterial infection. Because we do not have such equations for the automatic calculation of drugs given medical symptoms, AI technology has suggested to us that models based on predicate calculus (of which expert system knowledge is a special case) are indeed useful if we can feed in inputs and obtain reasonable outputs. The AI approach suggests that we code *all* the knowledge that is available to us for our simulation models, and not only that knowledge which yields to numerical analysis. The high level knowledge that is coded within expert systems is usually of a "decision-making" or diagnostic type. How does this effect the field of computer simulation? It suggests that we code decision-making and planning components within our simulations.

4.3 Systems Modeling

One of the problems in the area of AI and Simulation is that many researchers have thought of expert systems (and other AI models) as being completely different than models that exist in simulation. In addition, the concept of simulation as being inherently numerical has been forced onto the simulation community by some AI researchers. For example, in Forbus' survey of qualitative physics [Forbus 1988], he briefly mentions the "alternative route of numerical simulation." The assumption is that simulation is intrinsically numerical; however, we disagree with this assessment. The initial proof for this statement can be found in the works of systems theorists such as Zadeh and Padulo/Arbib in the late sixties and early seventies --- a unified theory of systems includes discrete as well as continuous models. Simulation models that use finite state automata or Markov models also provide nice vehicles for expressing qualitative system knowledge -- states can have lexical interpretations. It is true that expert systems technology is new to Simulation because simulationists have not generally been concerned with simulating human minds or decision-making as the end product, but rather expert knowledge of physical processes in the many abstraction levels available for those processes. Where no models exist for physical processes, we should look toward human minds ("experts") for models.

The point I wish to make is that expert systems represent simulation models (founded on the predicate calculus formalism) at an *early evolutionary stage* of development. Expert system models of dynamic concepts are precursors to more intensely mathematical models reflecting greater degrees of validity between model and process. Therefore, let us not view expert system models in a completely different light. Instead, let us recognize that the systems problem solving process is highly iterative; we start with simple models and progress to complex models. Also, we are concerned with designing simulation models that incorporate multiple interacting levels of abstraction [Fishwick 1989a]. Expert rules are knowledge and numerical equations are also knowledge; let's not delegate the two to different categories.

4.4 Conflicting Terminology

I have attended several panels on the general subject of AI and Simulation where it is generally acknowledged that we should not be overly concerned about which concept belongs to which discipline (AI or Simulation). The claim "this is purely a territorial issue" often surfaces during discussion. The problems of conflicting terminology, though, are important and these must be carefully addressed [Fishwick 1989b]. One might claim that discussing how "landmarks" (in qualitative reasoning [Kuipers 1986]) and "discrete events" (in simulation) are related is a pointless enterprise. I would strongly disagree with this conclusion -- the issue of terminology is not one of pure territoriality. Instead, it is one that lies at the very foundation of how we structure a discipline and formulate theories. If the terminology is different then we should discuss the "hows" and "whys."

4.4 Choosing the Right Model and Level

Simulation theory and science offers many models for different purposes, and we must not lose sight of the fact that models are chosen to yield specific *types* or *levels* of answers. If a simulationist is not using an expert system to model a physical process (such as one in mechanics) then it is most likely because an expert system would not be able to answer the kinds of questions that the simulationist seeks (i.e. "at what time does the flywheel reach its maximum speed?"). There is no "magic" to an expert system — it yields answers built upon purely inferential knowledge. Ask yourself "What problem (precisely) am I solving" and "What type of answer do I want?" Don't choose a model based on inferential reasoning if you want precision, and don't choose an equational model if you are modeling a decision making process. Choose the right modeling language at the right level of abstraction.

Recently, within AI, several researchers have pointed to two classes of knowledge: shallow versus deep knowledge. Shallow knowledge is of the inferential kind whereas deep knowledge reflects the use of model based reasoning. One kind of knowledge is not more important than the other; shallow reasoning is useful for decision and control knowledge whereas deep knowledge is for representing more mechanistic knowledge. In the simulation field, we do model based reasoning as well in the sense of formulating predictions, and so it is important that the Simulation and AI fields form a common ground for further discussions of "deep knowledge."

4.5 Tests of Validity for AI Models

In Simulation, tests for verification and validity are known to be an important part of the field. Is this also true in Artificial Intelligence? For many AI researchers, validity is extremely important. Consider an expert system whose rule base grows incrementally to yield results that are more valid than the last time it was tested. Expert systems researchers are very concerned with validity. If the expert system does not solve the problem better than conventional methods (if they exist) then it is a poor expert system. If a computer chess program always loses, then its expert heuristics (no matter how entailed) are flawed. Using various measures, such as ratings for chess programs, we can validate AI systems. Now, what about those models that are supposed to be models of human thought — how can these be validated? This kind of validation is common in the psychological literature; however, within AI it is not as common. Is it reasonable to create a model without the slightest concern for validation? If it is not, then we should be extremely wary of theories or models that claim to represent human reasoning. Note, for instance, that Hayes [Hayes 1985] develops an elaborate theory for commonsense reasoning about physical systems and then omits any validation of his theory. It is pointless to formalize system models that have little hope of being validated or invalidated; one must eventually compare one's process theory against 1) physical theory, or 2) psychological protocol.

4.5 Toward Automated Systems Problem Solving

Many researchers in both AI and Simulation are after the same goal: the automation of the systems problem solving process. In a recent paper with Bernard Zeigler [Fishwick and Zeigler 1990], we discuss this objective as a common goal of both disciplines. The efforts that include using expert systems knowledge, heuristic knowledge, and qualitative knowledge in Simulation are all based, effectively, on the underwritten theme of systems problem solving automation. With this theme in mind, we need to carefully analyze what is being said in each field. It is not fruitful for both AI and

Simulation to ignore one another, even though this has been suggested from time to time. There needs to be more researching the basic issues that arise when AI researchers and Simulation researchers talk about the same topic using different languages. We are all after the same goal of automation. Achieving that goal will require cooperation, and cooperation will require critical analysis of each others' results.

5 R. JAMES FIRBY: AI CONTRIBUTIONS TO SIMULATION

5.1 Introduction

It would seem that before we can evaluate the contributions AI can, and should, make to Simulation, we must try and define the two fields. I generally balk at discussions that begin with "AI is ..." because AI, like Simulation, is primarily computer science. The only real differences are the particular problem aspects being emphasized and the resulting specialization of the problem solving techniques being employed. Object oriented programming is no more an AI technique than it is a Simulation technique; it is a computer programming technique which offers advantages in any field. Debate about whether object oriented programing, or LISP, or expert system shells belong to AI or Simulation is superficial and misses the real contributions each field can make toward solving problems.

For the purposes of this panel, it is probably most useful to distinguish between the *problems* that AI and Simulation researchers work on so that we can exchange ideas in a meaningful way. I think that there is only one major characteristic that separates problems in the two fields and it is the reason that different solution techniques are employed. Simulation concentrates primarily on predicting a system's behavior when its *fine structure* is well understood, whereas AI concentrates on predicting behavior when a system's *course structure* is most important.

Consider any given simulation problem domain. In general, the problem will consist of a large number of simple interacting processes, such as airplanes landing, fluids moving, mixing and boiling, or photons scattering. The classic Simulation problem is to figure out the way the system as a whole will behave by modeling all of the simple interactions individually (or in small groups if enough mathematics is known). However, in many situations the individual processes making up a system are not well understood and the system's behavior cannot be predicted by simulating basic components. AI researchers typically focus on the problem of predicting the way a system will behave given only a course understanding of its structure. Instead of detailed knowledge about the physics of a system, a course structure problem solver must rely on information like: what did this system do last time, is this system similar to a system I already know, and what do systems with this sort of structure usually do.

This is obviously an over-simplification but, within the areas where AI and Simulation problems overlap, I think it is a fair characterization of the difference in emphasis between the two fields.

5.2 Simulation and Fine Structure

Simulation researchers are concerned with trying to accurately predict the behavior of a system given a detailed understanding of its fine structure. The advantages of predicting a system's behavior using simulation of its fine structure are rigor and precision; analytical science is grounded in the idea that a system can be understood completely by understanding its simplest parts. Simulation researchers need this rigor and precision because they are primarily interested in using simulation during the system design process.

5.3 AI and Course Structure

AI researchers have typically been concerned with trying to understand a system's behavior at a coarse structure level. Either the fine structure of the system is unknown, or the goal is to *automatically* derive the coarse structure. In the former situation there is no way to simulate the system's individual components and hence there is no way to systematically predict its overall behavior. In the latter case, the problem requires rigorous simulation but AI researchers generally assume that the simulator is given and concentrate on developing techniques for deriving the coarse behavior.

I will claim that the primary reason AI researchers are more interested in the coarse structure of a system is because AI is interested in prediction for the purpose of taking action rather than for the purpose of design. An abstract characterization of a system's behavior (*i.e.* an understanding of its coarse structure) usually points toward an appropriate response to that behavior much more clearly than a very rigorous understanding of the system's details. As a result, for AI problems, precision is often less important than abstraction.

Furthermore, in many situations of interest to AI researchers, the fine structure of the problem domain cannot be used anyway. Consider the problems of reasoning about a system when:

- Information about the system is very uncertain or simply unknown. For example, suppose an AI system is trying to decide what will happen if a slug is put into a Coke machine. The internal workings of the machine will almost certainly be unknown and simulation of their behavior will be impossible. However, it should still be possible to reason that the machine might not deliver a Coke.
- Information about the system is extremely complex. Suppose an AI system is trying to fix a computer network that has become too slow. It is certainly possible to simulate traffic over the network to suggest reasons for the problem but such a simulation is likely to be too complex and time consuming to be worth the trouble. However, the AI system should still be able to consider unplugging one or more nodes from the network as a possible fix.

The basic techniques that AI researchers use to address these problems is to characterize the coarse structure of the system and predict its behavior in an approximate way based on that abstraction.

5.4 What are the Contributions?

In general, AI and Simulation researchers are working on different aspects of the same problem: how to predict the behavior of a system for design (Simulation) and response (AI). Simulation researchers demand rigorous answers based on a detailed understanding of the system's fine structure. AI researchers, on the other hand, are more interested in classifying modes of behavior and can often do that with an understanding of only coarse system structure. AI solution techniques are usually only approximate but they can be used even when a detailed understanding of a system is unavailable.

Given this view of Simulation and AI, the main contribution that AI can make to Simulation is a collection of techniques to predict the behavior of systems when:

- Rigorous simulations get so complex that it makes sense to trade some accuracy for speed. Appropriate techniques include qualitative reasoning, approximate reasoning, expert systems, and planning.
- Rigorous simulations are impossible do to a lack of detailed knowledge. Appropriate techniques include probabilistic temporal reasoning, case-based reasoning, and reasoning about explanations and similarity.

The latter problem will arise more often as simulation moves out of engineering design and into fuzzier areas like military war gaming.

5.5 Simulating Multiple Autonomous Agents

An intriguing example using of using AI and Simulation techniques together arises when simulating multiple, complex, context dependent agents such as units on a battlefield. One would like to be able to study the progress of a battle or campaign by simulating each of the basic units to get as much realism as possible. However, the behavior of a given unit is very difficult to model rigorously because it depends on a host of local details and the ability of the unit to sense and analyze those details.

My specific interest within the field of AI is that of controlling autonomous agents. In trying to control an autonomous agent almost all reasoning must be done based on the coarse structure of the domain at hand. The agent will lack considerable knowledge about the workings of machines and other agents; noisy sensor data coupled with lack of knowledge makes prediction of the future extremely uncertain; and any fine detail that is known about the world introduces far too much complexity into the reasoning process to be useful for anything except very focussed control problems. However, it is still possible to generate appropriate actions in a given situation using abstraction and prepackaged context sensitive rules.

I think it is possible to combine the course structure AI reasoning used to control agents with fine structure simulation of environmental factors to produce a hybrid simulation of a battlefield. Each unit on the field would acquire information about the world, analyze it, and generate actions using AI techniques while rigorous Simulation techniques would link the units together and model unit-independent effects like weather and terrain. The AI supplies a realistic mechanism for modeling the decision making within each basic unit and the Simulation then treats each unit as a building block connected by the physics of the battlefield.

6 REFERENCES

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

Buchanan, B. (1984), and E. Shortliffe, *Rule-Based Expert Systems*, Addison Wesley.

Dahl, O-J. and Nygaard, K. (1966), "Simula–An Algol-Based Simulation Language," *Communications of the ACM*, 9, pp. 671-678.

Forbus, K. (1988), "Qualitative Physics: Past, Present and Future" in *Exploring Artificial Intelligence* edited by Howard E. Shrobe, pp. 239 - 296, Morgan Kaufmann.

Fishwick, P.A. (1989a), "Abstraction Level Traversal in Hierarchical Modeling," in *Modeling and Simulation Methodology*, edited by Bernard Zeigler, Maurice Elzas and Tuncer Oren, pp. 393-429, Elsevier North Holland.

Fishwick, P.A. (1989b), "A Study of Terminology and Issues in Qualitative Simulation," *Simulation*, January 1989, 51(7), pp. 5-9.

Fishwick, P.A. and B. P. Zeigler (1990) "Qualitative Physics: Toward the Automation of Systems Problem Solving," Submitted to *Communications of the ACM*.

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

Goldberg, A. and Kay, A. (1976), *Smalltalk-72 Instruction Manual* Report SSL 76-6, Xerox PARC, Palo Alto, California.

Hayes, P.H. (1985), "The Second Naive Physics Manifesto," in *Formal Theories of the Commonsense World* edited by J. Hobbs and B. Moore, pp. 1-36, Ablex Publishing Corporation.

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

Klahr, P. (1985), "Expressibility in ROSS: An Object-oriented Simulation System," in: *AI APPLIED TO SIMULATION: Proceedings of the European Conference* at the University of Ghent, pp. 136-139.

Kuipers,, B. (1986), "Qualitative Simulation," *Artificial Intelligence*, September 29(3), pp. 289-338.

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

Rothenberg, J. (1986), "Object-oriented Simulation: Where Do We Go from Here?" *Proceedings of the 1986 Winter Simulation Conference*, Washington, DC, pp. 464-469.

Rothenberg, J., Narain, S., Steeb, R., Hefley, C., and Shapiro, N. Z. (1989), *Knowledge-Based Simulation: An Interim Report*, The RAND Corporation, N-2897-DARPA.

Rothenberg, J., N. Z. Shapiro, and C. Hefley (1990), "A 'Propagative' Approach to Sensitivity Analysis," *Proceedings of the AI, Simulation and Planning in High Autonomy Systems Conference*, (Tucson, Arizona, March 26-27), B. Zeigler and J. Rozenblit, eds., IEEE Computer Society Press, Los Alamitos, CA, pp. 10-16.