ANL/DIS/TM-59

Proceedings of the

Workshop on Agent Simulation: Applications, Models, and Tools

October 15-16, 1999
The University of Chicago

Edited by Charles M. Macal and David Sallach

Co-sponsored by

Social Science Research Computation
The University of Chicago

and

Decision and Information Sciences Division
Argonne National Laboratory

September 2000

ISBN 0-9679168-0-1

Available by request from

Decision and Information Sciences
Division Office
Argonne National Laboratory
9700 South Cass Avenue
Argonne, Illinois 60439-4832
Telephone: (630) 252-5464
Home page: http://www.dis.anl.gov/

Argonne National Laboratory

Argonne National Laboratory, with facilities in the states of Illinois and Idaho, is owned by the United States Government, and operated by the University of Chicago under the provisions of a contract with the Department of Energy.

Disclaimer

CONTENTS

Modeling Electrical Networks

SATURDAY, OCTOBER 16, 1999

Agent Toolkits: Strategies and Tradeoffs

Agent Methodologies and Model Validation

The Agent Research Horizon

Closing Panel

Foreword

Agent simulation is a new approach to the study of social, economic, and physical systems. Following the publication of Thomas Schelling's groundbreaking book, *Micromotives and Macrobehavior* (W.W. Norton, 1978), many scholars have made pivotal contributions that demonstrate the potential of complex, adaptive models for representing large-scale and emergent social processes. While many important effects have been demonstrated, this approach to studying complex systems is not yet a fully realized methodology. This workshop addressed several questions related to this new field of inquiry.

- What has been accomplished so far?
- In what areas does agent research have the potential to contribute?
- What kind of information can agent models yield, and how should it be interpreted and used?
- What constitutes validation of an agent model, and, in particular, what permutations and scaling of validation are needed as models become more complex?
- What are the strengths and limitations of available agent toolkits, and what features are proposed — and needed — in the next generation of tools?
- What is the research horizon?

In addition to presentations and discussions on these larger issues, the workshop included presentations on specific applications in computational economics and agent models of electrical networks.

The workshop was intended to provide a meeting ground for a stimulating exchange of diverse views. Indeed, a topic reiterated throughout the workshop was the importance of continued conversation between experts concerned with the content, theories, and conclusions of individual subject domains and experts concerned with advancing the art and science of simulation. From such conversations there will surely arise many unexpected and fruitful applications of the concepts and tools of complex adaptive systems. The proceedings are presented in that spirit.

David Sallach
Director, Social Science Research Computation
The University of Chicago

Thomas Wolsko
Director, Decision and Information Sciences Division
Argonne National Laboratory

Acknowledgments

Victor Lofgreen coordinated the logistics and managed the audiotaping of the workshop. Mika Robinson and Lisa Reyes also helped with logistics. Kathy Ruffatto handled registration and administration, and Bob Baker managed the workshop web site. Jane Andrew assembled and copyedited the proceedings, including presentations, abstracts, papers, and discussions, with help from Curt A. Strating and others from Alanwood Enterprises, Inc., for transcription of the discussions and Judith Robson for much of the word processing.

ORGANIZING COMMITTEE

David Sallach, University of Chicago (co-chair)
Thomas Wolsko, Argonne National Laboratory (co-chair)
Victor Lofgreen, University of Chicago
Charles Macal, Argonne National Laboratory
Michael North, Argonne National Laboratory
John Padgett, University of Chicago
Randal Picker, University of Chicago

Agent-Based
Social Science Models

WHY AGENTS? ON THE VARIED MOTIVATIONS FOR AGENT COMPUTING IN THE SOCIAL SCIENCES

R.L. AXTELL, The Brookings Institution*

ABSTRACT

The many motivations for employing agent-based computation in the social sciences are reviewed. It is argued that there exist three distinct uses of agent modeling techniques. One such use — the simplest — is conceptually quite close to traditional simulation in operations research. This use arises when equations can be formulated that completely describe a social process, and these equations are explicitly soluble, either analytically or numerically. In the former case, the agent model is merely a tool for presenting results, while in the latter it is a novel kind of Monte Carlo analysis. A second, more commonplace usage of computational agent models arises when mathematical models can be written down but not completely solved. In this case the agent-based model can shed significant light on the solution structure, illustrate dynamical properties of the model, serve to test the dependence of results on parameters and assumptions, and be a source of counter-examples. Finally, there are important classes of problems for which writing down equations is not a useful activity. In such circumstances, resort to agent-based computational models may be the only way available to explore such processes systematically, and constitute a third distinct usage of such models.

1 THE NEED FOR COMPUTATIONAL MODELS IN GENERAL AND AGENTS IN PARTICULAR

The ideal gas, the perfect fluid, the Eulerian beam — such ideal types are commonplace in science. Each ideal type plays an important role in its field, both pedagogically and practically. Students are taught about ideal types in order to build their intuition about fundamental relationships between key variables. Knowledge of such ideal types is also helpful in making approximate, order-of-magnitude calculations. But reality is not ideal: real gases depart systematically from the ideal gas law, real fluids have viscosity, and real beams buckle in all kinds of ways that Euler never imagined.

Rational agents are an ideal type. They are introduced to undergraduates in order to teach first principles, e.g., strategic behavior, incentives, expectations, substitution effects, moral hazard, adverse selection. To their credit, the undergraduate textbooks are quick to circumscribe the domain of the rational agent. It is something of a curiosity that only in graduate school is this ideal type reified into the main object of study.

* *Corresponding author address*: Robert Axtell, Center on Social and Economic Dynamics, The Brookings Institution, 1775 Massachusetts Ave. NW, Washington, DC 20036; e-mail: raxtell@brook.edu; web: http://www.brook.edu/es/dynamics.

Or so it has been up until recently. There are now a variety of approaches to political economy that attempt, each in its own way, to move beyond the rational agent. These include experimental economics, evolutionary economics, and certain computational approaches. This paper is concerned primarily with the latter approach, although it is also relevant to the others insofar as computational economics makes use of evolutionary ideas and results from experiments.

There is a branch of computational economics in which relaxing the dependence on rational agents plays no important role. This stream of thought is well represented by Amman *et al.* [1996] and Gilli [1996], and concerns itself primarily with efficient numerical solution of equation-based models involving rational agents, or, with bringing new optimization techniques to bear on classical economic problems.

The other main branch of computational economics involves agent-based models. In such models, individual agents are explicitly represented. These agents interact directly with one another, and social macrostructure emerges from these interactions.[1] A very common motivation for such models is, broadly speaking, a basic dissatisfaction with rational agents. Thus, essentially all agent-based models that have appeared to date involve some form of boundedly rational agent.[2] This paper is concerned with various uses of agent-based models in the social sciences generally and in political economy in particular.

1.1 Agent-Based Computation: Strengths and Weaknesses

An agent-based model consists of individual agents, commonly implemented in software as objects. Agents have states and rules of behavior. Running such a model simply amounts to instantiating an agent population, letting the agents interact, and monitoring what happens. Stated differently, executing the model is all that is necessary in order to "solve" it.[3] Furthermore, when an agent-based model, call it A, produces result R, one has established a sufficiency theorem, that is, the formal statement R if A [Newell and Simon, 1972: 13].[4]

There are, ostensibly, several *advantages* of agent-based computational modeling over conventional mathematical theorizing. First, as described above, it is easy to limit agent rationality in agent-based computational models. Second, even if one wishes to use completely

---

[1] Indeed, in their most extreme form, agent-based computational models will not make any use whatsoever of explicit equations. Thus, there is a definite sense in which the two distinct branches of computational economics are nearly anthithetical. Stated differently, about the only thing that they have in common is that both use computers. Note that there is a close analogy between these two types of computational economics and the situation in computational physics, where the numerical solution of continuum (differential equation) models is only weakly related to particle and cellular automata models.

[2] Epstein and Axtell [1997] give a fairly comprehensive bibliography of agent-based models in the social sciences that were either in working paper form or published by 1996. Since then there has been a rapid expansion of agent-based modeling efforts, and anything like a complete listing of this work would reference several hundred papers.

[3] Of course, if the model is stochastic, then multiple realizations are necessary in order to characterize R.

[4] "To take sufficiency as a first requirement of a theory is simply to adopt a particular approximating sequence in science's progress. Since not all things can be done first, a particular theoretical orientation gets some of its flavor from what it puts first" [Newell and Simon, 1972: 13].

rational agents, it is a trivial matter to make agents heterogeneous in agent-based models. One simply instantiates a population having some distribution of initial states, e.g., preferences. That is, there is no need to appeal to representative agents. Third, since the model is "solved" merely by executing it, there results an entire dynamical history of the process under study. That is, one need not focus exclusively on the equilibria, should they exist, for the dynamics are an inescapable part of running the agent model. Finally, in most social processes either physical space or social networks matter. These are difficult to account for mathematically except in highly stylized ways. However, in agent-based models it is usually quite easy to have the agent interactions mediated by space or networks or both.

However, the agent-based modeling methodology has one significant *disadvantage* vis-a-vis mathematical modeling. Despite the fact that each run of such a model yields is a sufficiency theorem, a single run does not provide any information on the robustness of such theorems. That is, given that agent model A yields result R, how much change in A is necessary in order for R to no longer obtain? In mathematical economics such questions are often formally resolvable via inspection, simple differentiation, the implicit function theorem, comparative statics, and so on. The only way to treat this problem in agent computing is through multiple runs, systematically varying initial conditions or parameters in order to assess the robustness of results. While the curse of dimensionality places a practical upper bound on the size of the parameter space that can be checked for robustness, it is also the case that vast performance increases of computer hardware are rapidly converting what was once perhaps a fatal difficulty into a manageable one.

1.2 Agent-Based Computation: Architecture and Implementation

Before moving on to discuss distinct motivations for agent-based models, it will serve as useful background to first describe their basic computational structure. Of course, agents are the key ingredient. Each agent possesses both states (i.e., data, also known as instance variables) and rules of behavior (i.e., procedures or functions, aka methods) and are most conveniently represented in software as objects. Agent states can be either private or public; the latter are visible to other agents while the former are not. Similarly with agent behavioral rules — some are private and some public. This is exemplified by the following pseudo-code:

```
Agent object:
  private states:
     preferences;
     wealth_1;           /*         hidden wealth      */
     :
     ;
  public states:
     bid-price;
     wealth_2;           /*         other wealth       */
     :
     ;
  private behavior:
     compare_choices;
     compute_internal_valuations;
     draw;
     :
     ;
```

```
public behavior:
    initialize;
    seek_trade_partner;
    communicate_with(Agent i)
    :
    ;
end.
```

Code fragment 1: Typical agent object.

Just as each agent is an object, so with the agent population as a whole. It has both states — data relating to the agents — and functions — such as routines to compute population statistics. An example population object is:

```
AgentPopulation object:
   private states:
      internal_representation_of_population;
      currently_active_agent;
      :
      ;
   public states:
      number_of_agents;
      number_of agents_working_in_firms;
      :
      ;
   private functions:
      get_Nth_agent (N);
      randomize_the_agents;
      draw;
      :
      ;
   public functions:
       initialize;
       agents_trade;
       compute_average_bid_price(commodity j);
       :
       ;
   end.
```

Code fragment 2: Typical population object.

In order to track the performance of an agent model some statistical analysis must be hooked into the code. This can take the form of simple text output, graphical display or real-time, model-generated statistics, or even real-time econometric estimation. Then with all this in place an agent-based computational model becomes little more than:

```
program typical_agent_model;
   initialize agents;
      repeat:
         agents_interact;
         compute_agent_statistics:
      until done;
   end.
```

Code fragment 3: Typical agent-oriented program.

Now, much detail has been abstracted away in these code fragments. In actual implementations agent interactions occur either sequentially or in parallel, and if parallel then with some degree of synchrony. The timing of interactions implicitly specifies a global clock and the definition of a model "period." The order of agent activation must be systematically randomized from period to period in order to avoid the production of artifacts, phenomena in model output that arise due to accidentally imposed inter-agent correlations and that are not robust to seemingly innocuous code changes. It is sometimes necessary to give each agent its own random number generator. And so on. But these code fragments give a skeletal picture of agent-model architecture.

There is a definite sense in which agent-based computational models are relatively *easy* to create, in comparison to other computational models. This is so because the heart of the code are the agent behavioral methods, and you write these only once; the behavioral repertoire is the same for each agent. Consider the following example. An agent-based model of exchange is instantiated with 10^6 agents. Each agent has preferences, an endowment, and a current allocation. Depending on the size of the commodity space, each agent might consume $O(10)$ to $O(1000)$ bytes. Overall, the amount of memory used by the agent-model is $O(10^7)$ to $O(10^9)$ bytes, that is from 10 megabytes to perhaps a gigabyte. But such a program can be specified in fewer than 1000 lines of C/C++ code, perhaps 100 lines of Ascape.[5] So a relatively short "program" at compile-time is actually a very large "program" at runtime.[6]

This architecture, in which very little source code effectively controls a much larger amount of execution code, is the basis for the highly scaleable nature of agent-based models. The number of agents or commodities, for instance, are conveniently specified as user-defined constants in the source code or read in from the command line, and thus the scale of the model can be specified at compile or run-time. Typically, no significant rewriting of the code is needed in order to change the scale of the model.[7]

It is also the case that the "small source, large execution code" character of agent computing is partially responsible for the production of artifacts, an important class of systematic problems that can arise in agent models, as alluded to above. When a small amount of code — say a single commodity exchange rule, for example — controls so much other code, then it will sometimes be the case that an idiosyncrasy in the rule code will produce output that one takes as

[5] For more on SWARM, a high-level language for agent-based modeling, see Minar *et al.* [1996] and Daniels [this volume]. For a description of Ascape, see Parker [this volume].

[6] The standard notion of "program" seems problematical in this context.

[7] Of course, this is not the same as saying that the number of computations (clock cycles, agent-agent interactions) is a linear function of the scale of the model. The computational complexity of such models is a much more complicated issue and will be addressed below.

a significant result of the model. A common route to phenomena of this type occurs when the agent interaction methods impose some spurious correlation structure on the overall population — say agents are interacting with their neighbors more than with the population overall in what is supposed to be a "soup" interaction model — then an ostensibly systematic result — large price variance, say — is clearly artifactual.[8] There is no real solution to this problem, aside from careful programming. One can, however, look for the existence of such artifacts by making many distinct realizations of an agent model, perturbing parameters and rules. When small perturbations in the code produce large changes in the model output, then artifacts may be present. Sometimes, large changes in output are realistic and not signatures of artifacts. For example, imagine that a small change to a threshold parameter makes an agent die earlier than it otherwise would, and therefore induces at first a small change in agent exchange histories (i.e., who trades with who), that over time is magnified into a wholesale change in the networks of agent exchange. Perhaps this is not unrealistic. But when such large scale changes have origins that are unrealistic empirically, then one should be instantly on guard for undiscovered flaws in the source code.

In the next section we describe a simple use of agent-based models and argue that the term "simulation" is best applied to this use. Then, in section 3 certain commonly encountered difficulties with mathematical models are described, as is the use of agent-based models to circumvent these problems. Finally, in section 4 a third use of agent-based models is described in relation to problems for which mathematical representation is simply not useful.

2 FIRST USE — WHEN EQUATIONS CAN BE FORMULATED AND COMPLETELY SOLVED: AGENT MODELS AS CLASSICAL SIMULATION

Here we describe a simple, perhaps some will say trivial, use of agent-based computational modeling. It is almost certainly *not* the most important use of agents, but it is the use that best fits the conventional meaning of "simulation."

Imagine that some social process is under study, and that it is possible to write down one or more mathematical relationships that fully describe the process. Furthermore, imagine that the resulting model may be solved explicitly, either symbolically or numerically. Then what role is there for an agent-based computational model of the process?

2.1 Agent Computational Modeling as an Instance of Monte Carlo Simulation

There are many answers to this question. First, if the solution is not available symbolically, but only numerically, then an agent-based model serves as a useful check on the numerical solution. That is, the solution obtained by numerically solving the equations should agree with that which emerges from the agent model. As an example of this, consider solving some set of equations for an income distribution. Alternatively, one creates an agent-based model of the process, runs the model, queries each agent for its income, sorts the data, and builds-up an 'empirical' income distribution, which should agree with the numerically solved equations. Because these methods are equivalent, it would seem that researchers never perform both

[8] For a discussion of such artifacts see Huberman and Glance [1993].

numerical and agent-based solutions.[9] In fact, this is not a common use of agent-based computational models.

Second, if the model is stochastic then the numerical solution will be some distribution of outcomes. This is classical simulation as practiced in operations research, a well-known variety of which is Monte Carlo simulation.[10] In particular, imagine that the output, Y, of a stochastic model is given by $f(X)$, where $f(\bullet)$ is a deterministic function of a random variable, X. With X and f given, Y is completely specified although often cannot be computed symbolically. Thus one resorts to Monte Carlo analysis, in which many realizations of $X = x$ are made and for each one $y = f(x)$ is computed. In this way Y is built up progressively.

There is a one-to-one relationship between this kind of Monte Carlo analysis and agent-based modeling. In particular, if X is a known distribution over a heterogeneous agent population — e.g., agent preferences — and f is some specified social process — yielding equilibrium allocations of goods, say — then each realization of X can be thought of as an agent. Therefore, it is only this narrow usage of agent-based modeling — when the model is intrinsically stochastic and the equations governing it cannot be solved analytically — that deserves to be called *simulation*, or so it seems to me. Two examples attempt to flesh out this usage of agent-based models as simulation.

Example 0: Consider the classical OR simulation of a bank teller line. This is a queuing model and no general analytical solution is known for arbitrary distributions of arrivals and service times. Therefore, the queuing process is commonly simulated via the Monte Carlo method and distributions of waiting times and server utilization result. However, this is completely equivalent to actually instantiating a population of agents, giving them heterogeneous arrival times according to some distribution, and then literally running the agent-based model in order to build up the waiting time distribution function.

Example 1: Young [1993a, 1993b] has described a class of evolutionary models in which a population of agents repeatedly interacts in the context of a bargaining game. Each agent has finite memory and plays a best reply strategy, based on its idiosyncratic memory. Agents play randomly with a small probability. He has shown that on the many Nash equilibria in such games, only some have positive probability asymptotically, the *stochastically stable equilibria*. An agent-based computational version of this model has been created [Axtell *et al.*, forthcoming] that (1) visually displays the path of a population to Nash equilibrium configurations, (2) illustrates transits between equilibria, and (3) achieves the stochastically stable equilibria eventually. Furthermore, the way in which these results depend on the noise level (i.e., probability of playing randomly) has been investigated using this model. This agent-based code is a variant implementation of classical simulation of the Markov process that underlies this game model. Instead of generating random state vectors — i.e., agent memories — and iterating forward to a stochastically stable state, each agent can be thought of as a state vector, and the interactions of the agents as the way in which initially random state vectors get transformed into stochastically stable states. There is even a sense in which the agent-based model is a relatively efficient simulation technique, for classical simulation of the actual governing equations requires storage of a very large, albeit sparse, transition matrix.

[9] I am unaware of any papers that do both.

[10] See, for instance, Bratley *et al.* [1987] or Banks and Carson [1984].

So, whenever stochastic governing equations of a social process can be written out and their solution space characterized, so that all that remains to be done is generate numerical realizations, then the use of the term *simulation* to describe agent-based computational models corresponds to its traditional usage in operations research.

2.2 The Efficacy of Agent-Based Modeling as a Tool for Presenting Mathematical Results

Consider the (increasingly unusual) case in which a model of a social process can be solved explicitly. Here it would seem that there is no role whatsoever for agent computing, since the solution is completely specified. However, even in this case there is utility for creating an agent-based implementation of the formal model. Most people outside the academy have limited training in mathematics and therefore have a difficult time interpreting regression results, for example. But people are very good at visual interpretation and analogical reasoning. Because the output of agent models tends to be visual, such models can be very effective at depicting formal results from mathematical models. Such uses of agents are especially relevant for demonstrating technical results to policy-makers and business decision-makers.

3 SECOND USE — PARTIALLY SOLUBLE EQUATIONS: ARTIFICIAL AGENTS AS *COMPLEMENTARY* TO MATHEMATICAL THEORIZING[11]

The second distinct use of agent-based computational models occurs when it is not possible to completely solve a mathematical model analytically. Here, theory yields mathematical relationships, perhaps even equations, but these are not directly soluble, or perhaps no appropriate solution concept is readily available. Or it may be that an equilibrium configuration can be figured out, but that its stability is ambiguous. Or perhaps the dependence of the equilibrium on a distinguished parameter is of interest but cannot be readily computed. There are a variety of ways in which formal models resist full analysis. Indeed, it is seemingly only in *very* restrictive circumstances that one ever has a model that is completely soluble, in the sense that everything of importance about it can be obtained solely from analytical manipulations.

In such circumstances it is common to resort to numerical instantiations of the symbolic model, in order to glean some additional understanding. It is also generally possible to build agent-based computational models in order to gain insight into the functioning of the model. Now, if the agent-based computational model is merely an instantiation of the mathematical model, then we are back to agents-as-simulation, described in the previous section. However, it is often the case in the process of formalizing a theory into mathematics that one or more — usually more! — assumptions are made for purposes of simplification; representative agents are introduced, or a single price vector is assumed to obtain in the entire economy, or preferences are considered fixed, or the payoff structure is exactly symmetrical, or common knowledge is postulated to exist, and so on. It is rarely desirable to introduce such assumptions, since they are not realistic and their effects on the results are *a priori* unknown, but it is expedient to do so.

---

[11] For a related discussion in the context of non-agent computational models, see Judd [1997].

As described in the first section of this paper, it is typically a relatively easy matter to relax such ‘heroic’ assumptions-of-simplification in agent-based computational models: agents can be made diverse and heterogeneous prices can emerge, payoffs can be noisy and all knowledge local. One can then “dock” the agent-based computational model with what may now seem like the highly stylized analytical model by creating an instance of the computational model with homogeneous agents, fixed preferences, etc. Then, once the “docked” computational model is shown to reproduce the known analytical results — thus providing a crude, first-order validation of its performance — it can be instantiated with fully heterogeneous agents, etc., and then used for systematic study.[12] In this usage agent-based computational modeling turns out to be very powerful at advancing one’s understanding of a formal theory. It is as if the agent model is a “prosthesis for the imagination” and a complement to formal theory.

An argument sometimes used against agent-based modeling is that individual realizations (runs) are just special cases and nothing very general can really be known about the process under study until analytical results are obtained. We have offered the ‘computer programs as sufficiency theorems’ argument of Newell and Simon, in the introduction to this paper, as a partial refutation of this criticism, but more can be said. One role for agent computing is to check whether mathematical results yet obtain when specific assumptions are relaxed. In fact, if a single instance of an agent-based model produces results that violate a theorem that holds in more restricted circumstances, then the computational model stands as a counter-example to the wider applicability of the theorem. This use of agent computing, while usually not explicit, is implicit in much of the work that has appeared to date.

In this section, usage of agent-based computational models as a complementary to social theory will be described. Mathematical models for which analytical results are incomplete will be stated. Then the ways in which agent-based computational models can be used to further explore the solution space will be presented.

3.1 Equilibria Exist But Are Effectively Uncomputable

Fixed-point theorems were apparently introduced into economic theory by von Neuman in his input-output model [1945–46]. Now, many domains of economic theory depend on fixed-point theorems for the existence of equilibria. The original Brouwer and Kakutani theorems did not have constructive proofs, but such proofs are now known, e.g., through Sperner’s lemma. Of course, the existence of equilibrium is not the same as its achievement. That is, without some mechanism for converging to a fixed point in a bounded amount of time then there is little reason to believe that a fixed point could ever be realized; i.e., equilibrium would not be plausible.

It is now known that the Brouwer theorem has worst case complexity that is exponential in the dimension of the problem [Hirsch *et al.*, 1989] — the dimension being the size of the commodity space in the Arrow-Debreu version of general economic equilibrium, for example. Furthermore, it has recently been shown that computation of Brouwer and Kakatani fixed points are computationally hard problems [Papadimitriou 1994]. Taken together with Scarf’s empirical estimate that the number of computations required to equilibrate a computable general equilibrium (CGE) model scales like the size of the commodity space to the fourth power, one is left to believe that the Walrasian model is not a particularly credible picture of how a real exchange economy works.

---

[12] For more on “docking” models, see Axtell, Axelrod, Epstein, and Cohen [1996].

Example 2: Recently, an agent-based computational model of Edgeworth barter has been created and its computational complexity investigated [Axtell 2000]. In this model agents from a heterogeneous population are paired at random and engage in Pareto improving bilateral exchange. It has been demonstrated that the number of agent-agent interactions required to equilibrate the economy is linear in the number of agents and quadratic in the number of commodities. These results are robust over five decades of agent population size (10 to 10^6 agents), and nearly four decades of commodity space dimension (2 to 10^4 goods). Now, it turns out that it is possible to prove some theorems for this process — that it converges, that there exists a wealth effect, that it is possible to obtain Walrasian equilibria via bilateral exchange but bilateral exchange equilibria cannot be achieved by a Walrasian mechanism. The stimulus toward these results was due directly to studying the output from the agent-based model.

As a final point about the existence of fixed points, it would seem useful to point out that in the bilateral trade model described above, there exist unimaginably vast numbers of equilibria.[13] Existence of equilibria is a relatively trivial matter in these models and the important question becomes equilibrium selection. The equilibria that actually obtain in these agent-based models are indiosyncratic with respect to the agents, in the sense that from realization to realization any particular agent's allocation may vary substantially. But the macrostatistics of these models are quite robust from run to run. Thus we have a weak form of *path dependence* in which the history of agent interactions is important for the individuals, although not for the economy overall. If one takes these models seriously then one is left with the impression that sociology — in particular, social networks determining who interacts with whom — should be given a more prominent place in economic theory, while fixed point theorems should be concomitantly demoted.

3.2 Equilibrium Not Attained by Boundedly Rational Agents

There are many ways to model bounded rationality.[14] Papadimitriou [1993] has argued that computational complexity is a useful framework for thinking about rationality: full rationality implies exponentially difficult computations, while bounded rationality means that the requisite computations are bounded by a polynomial in the relevant problem parameters. There have appeared a variety of models in which bounded rationality is treated in this way, including Board [1994] and Spear [1989]. The main results of these mathematical models is that boundedly rational agents have a difficult time of, for example, learning rational expectations equilibria in non-trivial economic environments.

A related way of modeling boundedly rational agents is through the use of finite automata. In game theoretic settings this approach was first adopted by Rubinstein [1986] and Neyman [1985]. There is a large and growing literature on this subject [cf. Binmore and Samuelson [1992]. However, it has recently been shown, in the context of two person games, that the process of learning the strategy of an automaton opponent generally requires an amount of time exponential in the number of states of the automaton [Mor *et al.* 1996; see also Prasad 1997].

---

[13] This is also true in the model of coalitions of DeVany [1994] and the computational Tiebout model of Kollman *et al.* [1994].

[14] See the recent review article by Conlisk [1996].

Bounded rationality is a common feature of agents in agent-based computational models. The failure to achieve equilibrium is a characteristic result of such models. However, while it is often quite difficult to characterize non-equilibrium phenomena analytically, the ability to systematically study dynamics is one of the powerful features of agent-based computational models, as exemplified by the following model.

Example 3: The Santa Fe artificial stock market is an agent-based model in which heterogeneous agents allocate their assets between a risky stock paying a stochastic dividend and a risk-free bond. The agents formulate their expectations adaptively, based on past market performance, and these are therefore endogenous to the market. What emerges is an ecology of beliefs among the agents that coevolves with time. There exists a regime in the model in which rational expectations equilibria emerge, a regime characterized by limited exploration of alternative expectations models by the agents. However, when agents are actively engaged in exploring the space of expectations models then the market self-organizes into a more complex structure, prices and volumes have statistical features characteristic of actual market behavior, ‘technical trading’ arises in the agent population, and the market is no longer efficient (i.e., speculative opportunities exist).

It would seem that computational agent models have much to contribute to the study of bounded rationality.

3.3 Equilibria Obtained Asymptotically But Not Realized over Long Periods

In the agent-based model of bargaining described in example 1 above, the dynamical system is formally ergodic but, depending on the noise level, the memory length, and the number of agents, the population can get stuck away from the stochastically stable equilibria for arbitrarily long periods of time. Systems with this behavior are said to display “broken ergodicity.” The computational implementation of this model was informative in illustrating this phenomenon but also served to characterize it quantitatively. More generally, agent computation is particularly useful for depicting the transient, far-from-equilibrium behavior of social systems.

3.4 Equilibria Exist But Are Unstable

The stability of equilibria can often be inferred from the structure of a problem, either by qualitative considerations or explicit calculation. A problem may have multiple equilibria, not all of which are stable, and sorting the stable from the unstable becomes an important question. What happens when all equilibria are unstable? One’s first impulse is to reject such models out-of-hand, thinking that no social process would be well represented by such a model. However, the following example indicates that models with unstable equilibria may be relevant to some social institutions, and that agent-based computational models may be the most efficacious route to developing a thorough understanding of such models.

Example 4: Canning [1995] has analyzed a simple model for the formation of groups. In it, agents must choose how much effort to put into their group, and there is an associated cost to the agent of her effort. There are local economies of scale such that the effective effort of two agents working together in the same group is greater than their individual efforts; this leads to cooperative behavior. Each member of a group shares the total output equally; this leads to free riding behavior as the group size grows. That is, utility maximizing agents find it in their self-interest to put in large effort levels when they are receiving direct compensation for their labors, and little effort when their share of output is insensitive to how hard they work. It is possible to analyze this model as a dynamical system. This reveals that the equilibrium group size and effort level are unstable: groups are constantly growing and shrinking — all groups are meta-stable — the exact nature of the dynamics depending on parameters of the model.[15] It appears that it is not possible to say more about this model analytically, but important questions remain, for instance, “What is the distribution of group sizes that arises in the model?” But how might one go about analyzing the out-of-equilibrium structure of such a model?

An agent-based computational model of the Canning mathematical model has been created [Axtell 1999]. It displays the essential instability of the mathematical model insofar as firms are born, grow, and then succumb to the free rider problem and either shrink significantly or vanish altogether. It also yields a skewed distribution of firm sizes that closely resembles the empirical firm size distribution. The agent model displays quantitatively correct firm growth rate distributions, dependence of growth rate variance on firm size, and a wage-size effect of approximately the right magnitude. Here the agent-based model has contributed much beyond mere replication of analytical results.

There is an analogy to be made here. In fluid mechanics it is possible, for certain geometries, to solve a set of partial differential equations for the stationary distribution of fluid velocity as a function of spatial coordinates — the so-called laminar velocity profile. However, even undergraduates are taught that this laminar solution to the equations is unstable beyond a certain critical value of a dimensionless ratio of parameters known as the Reynolds number. Of course, just because the steady solution is unstable does not mean that the fluid does not flow! For super-critical Reynolds numbers the flow becomes turbulent, it is unsteady (time-dependent) microscopically, although perhaps stationary, and no general solution to the governing equations is known. If one wants to learn about the structure of turbulent flows, then computational models are essentially the only available methodology.[16] Such models well describe how the qualitative structure of the flow changes with increasing Reynolds number, from essentially laminar with small eddies to full-blown turbulence with eddies on all scales. It turns out that there is a definite way in which turbulent flows are self-organized in the sense that they expend less energy per unit volume than would a laminar flow having the same average velocity. Nature is thus not shy about involking non-equilibrium configurations when she finds it appropriate to do so, and economists should be equally intrepid!

3.5 Dependence on Assumptions and/or Parameters Unknown

When mathematical models are only incompletely soluble it may prove difficult to determine how the known results depend on particular assumptions or exogenous parameters. In such cases agent-based computational models may prove insightful, as illustrated by the following.

---

[15] Because the equilibria are unstable does not mean that the model is unrealistic. New firms start-up and old firms die every day. One wonders how many people leave GM for Ford or Daimler-Chrysler each day in Detroit, while others are moving to GM from the other two.

[16] As an aside we note that in computational fluid mechanics the term *simulation* is used to describe model instances of and not the general field. Similar usage holds broadly in computational physics.

Example 5: Axelrod [1995] describes an agent-based model for the evolution of generic cultural attributes. In the model, agents are arranged in a fixed position on a two-dimensional square grid having N cells on a side, thus N^2 cells total. Each agent is endowed with a set of cultural "tags." Agents are activated at random to interact with their neighbors, with the probability of interaction being proportional to the extent to which the agents are similar.[17] Regions of cultural homogeneity arise in the model, while from region to region cultures are heterogeneous. Once all agents are, with respect to all neighbors, either completely identical — share the same string of cultural attributes — or completely different — no attributes in common — then all cultural evolution has effectively stopped.[18] The number of distinct cultural regions, call it R, is an important summary statistic generated by the model. Axelrod argues that the model is relevant to the origin of distinct languages and language dialects.

Axelrod systematically studied the dependence of R on, among other things, parameter N. Interestingly, he finds that there is a non-monotone relationship between R and N. In particular, for N either small or large R is small, while for an intermediate value of N there exists a maximum value for R. This is shown in the following figure.

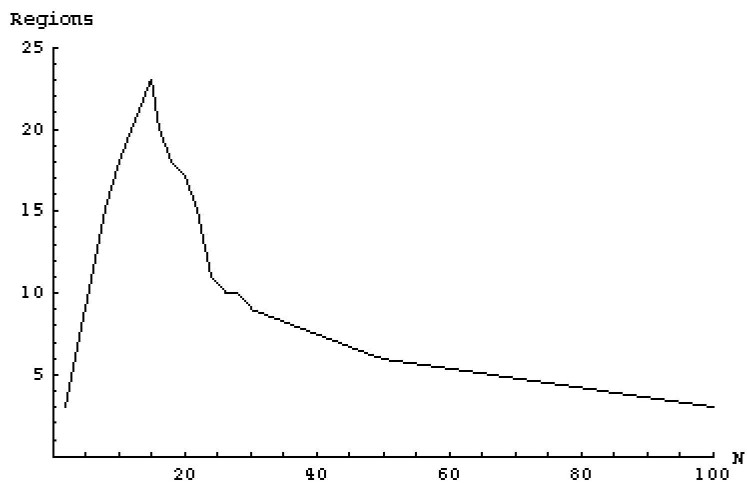


FIGURE 1 Dependence of the equilibrium number of culturally distinct regions on the size of the lattice.

An explanation for this relationship is offered in Axelrod [1995]. In order to assess the reasonableness of this explanation it would be useful have an analytical model of this agent-based model. Axelrod solicited the help of an eminent mathematician in order to formulate this model mathematically, but they were unsuccessful.[19] Therefore, the so-called "docking" experiment described in Axtell *et al.* [1996] was undertaken in order to verify this result.

While the illustration above describes how agent models can be used to understand parameter dependence, their use in assessing the importance of assumptions is directly

---

[17] This cultural transmission model generalizes that described in Epstein and Axtell [1996]. Some details of this are described in Axtell *et al.* [1996].

[18] There is no mutation in the basic version of the model. For some variations see Axtell *et al.* [1996].

[19] Personal communication.

analogous. For example, if a theorem is proved in a mathematical model under the assumption that all agents have the same number of memory states, one can simply give the agent population in an agent-based model some distribution of memory states, then run the program and observe whether or not the theorem still holds. If it does not then the agent-based model represents a counter-example to generalizing of the theorem with respect to heterogeneous agent memory length.

Relatedly, it is quite common to attempt an assessment of an analytical model's dependence on an assumption, to determine whether or not it is a "key" assumption. If several assumptions are made in a model, it may even be that the importance of each is assessed, and one is heartened if each seems to matter only a little, if at all. However, these assessments are usually independent — that is, it is unusual to try to relax multiple assumptions at once. But the overall validity of a result may depend on such simultaneous relaxation.

3.6 Equilibrium or Stationary Configuration Known Only for Simple Interaction Structures

The spatial aspect of social processes is a topic receiving renewed theoretical interest recently. The models that have appeared have a variety of forms, from random graph models to partial differential equation approaches. Typically, these models are analytically soluble only under rather restrictive conditions, requiring homogeneous agents or limited spatial dimension (e.g., 1D), for example.

Spatial processes are quite naturally represented in agent-based computational models. A physical location can be part of an agent's internal states. Likewise, its location in a social network can be easily represented internally, perhaps through a list in which each agent keeps so-called pointers, or pointers to pointers (handles), to other agents. Then, agents can be made to interact either through the physical space or the social network or both.

The effects of social network interactions have been widely studied theoretically [cf. Young 1998], but typically these results obtain for only very specific interation graphs. Human social networks seem to have a "small world" property, i.e., they are characterized by both highly "local" and certain long-range connections, and today not much is know about social processes on such graphs [Axtell 2000].

> **Example 6:** Glaeser, Sacerdote and Scheinkman [1996] build and estimate a simple model of criminality based on local social interactions. Large spatial variation is characteristic of crime rate data, and these authors demonstrate that such high variance can be produced by a simple model in which a large proportion of agents imitate their neighbors. They analyze their model mathematically and are able to develop closed form expressions for excess variance across space. Their estimation results support the hypothesis that certain kinds of crime are more social — in their model, require more imitation — than others. Auto theft, for example, is more social than murder. They even obtain crude approximations for the size of the neighborhoods of interaction by crime type.
>
> They analyze the model for agents arranged along one dimension, and in footnotes offer that they have studied a computational model in two and three dimensions. Now, while it is certainly true that humans live in a three-dimensional physical world, human imitative behavior occurs within social networks. It would be very interesting to know how their

> results stand up when agents behave as they postulate but within realistic social networks. This would seem to be a pregnant domain for agent computing. Furthermore, their behavioral model of the agents is so primitive — some agents are criminal for life, for example — that it would also be interesting to know how robust their results are to more realistic agent specifications. Once again, this is something easily manageable with artificial agents.

Lastly, for many models the effects of adding spatial dimensions so complicates the analysis that little progress is possible analytically, and resort to computational approaches seems to be the only way to make progress. The well-known Schelling model (see Epstein and Axtell [1996] and references therein) is an example of this.

3.7 Equilibrium Less Important Than Fluctuations and Extreme Events

In many stochastic dynamic models it is possible to characterize the equilibria and stability of the models asymptotically, but little can be said about their out-of-equilibrium behavior. Transit times between equilibria, expected time spent at particular equilibria, dependence of these times on the level and character of noise, these are all dynamical issues that go beyond mere existence of equilibria.

The importance of the out-of-equilibrium behavior of such systems is clear, since it may take such systems very long times to reach asymptotic equilibrium. In fact, it may be that one only cares about the extreme events of such systems. Agent-based models are good devices for systematically studying such dynamics. The following example illustrates this.

> **Example 7:** In models of traffic an important statistic is the distribution of jamming. Agent-based models have been created to study the dynamic aspects of traffic [Nagel and Rasmussen, 1993]. These models are capable of reproducing real-world data with extreme fidelity [Casti 1997]. In particular, on crowded roads it is known that local flowrate data are highly non-stationary. Differential equation models of traffic have a difficult time of capturing this feature of the data. However, large-scale, massively parallel (i.e., agent-like) computational models of traffic do a good job of capturing this phenomenon. Additionally, the jamming distributions that emerge in these models display a kind of universality also seen in statistical physics. That is, the macrostatistics of the systems are insensitive to the agent specifications. That is, many reasonable models of driving behavior produce the same distribution of traffic jams!

3.8 All, Or Nothing at All

It is a characteristic, although not often noted feature of agent-based computational models that once the model has been created it provides not merely one aspect of the solution — the equilibria, say, or the stability. Rather, as a functioning agent model spins forward in time it gives first the out-of-equilibrium dynamics, then approaches an equilibrium configuration and either becomes entrained in some basin of attraction or not, thereby indicating something of the stability of the equilibria. Repeated realizations yield the dependence of results on parameters and the importance of assumptions — all this from a single operational piece of code. Of course, if the code is not working properly, then nothing useful at all comes out.

4 THIRD USE — EQUATIONS OSTENSIBLY INTRACTABLE OR PROVABLY INSOLUBLE: AGENT MODELS AS *SUBSTITUTES* FOR ANALYSIS

When the equations governing a social process are completely intractable, either apparently or provably so, then there would seem to be little hope of progress with only mathematics. In either case, the equations would appear to be of little use. We investigate these two cases of intractability in turn.

4.1 Intractable Equations

It would seem that claims concerning a model's intractability are not often made — at least not in print — perhaps because people do not like to sound like they have given up. If one only had more time to read through the topology book, or perhaps next semester your teaching load will ease up and you will have time to explore the new probability journal, or you will save the problem for a bright student — these are all admissions that a problem is in some way intractable, say locally intractable. But, aside from actually proving that a problem is insoluble (see below) there are other ways that intractability can be made into a less subjective concept.

For example, it is well-known that there do not exist closed form solutions to certain relatively simple differential equations in terms of elementary functions. When a problem is intractability in this way it has nothing to do with its complexity. Rather, it is an artifact of the limited explorations undertaken to date in the infinite library of functions. In such circumstances one makes recourse to numerical solution. But there are also instances in which numerical solution would appear to be essentially intractable. This occurs when governing equations are highly nonlinear.[20] When such circumstances arise in computational physics, particle models can sometimes be advantageous. Perhaps the same is true of agent-based computational models in the social sciences, although I am unaware of any papers in which agent-based models are proferred as a substitute for an intractable analytical model.

---

[20] According to Feigenbaum [1988: 567]:

> Consider a cloudlike initial configuration of some fluid equation (a classical field theory). Imagine that the density of this configuration possesses rich scaling properties (e.g., a fixed spatial scale exponent over many decades). Moreover, imagine that at successive moments of time it also possesses these scaling properties, although possibly variable in time. From this we should surmise that the instantaneous velocities should also possess similar scaling properties. Imagine that these scalings are easily specified, that is, we have discerned in this complex spatial object some prescriptive rules that if iterated would construct it. Now let us contemplate how we advance this structure in time. By the locality of the field equations we must *actually* spin out this iterative construction in order to provide the equations with the sort of initial data they require. Now we can advance the structure a step ahead in time. But what do we now have? Simply an immense list of local density and velocity values of high local irregularity. Of course, if we possess a good algorithm, we could now from this new pabulum of data again discern the scaling information — perhaps evolved — that we know about anyway. This is obviously a foolish double regress. Since our informed understanding lay in the scaling description, we should obviously have transcribed our "true" local dynamics into one pertinent to these scalings, rather than mount a numerical program that strains the most powerful machines we possess. That is, the solution in the usual sense of our local field theories is apt to be a mindless enterprise when the solutions happen not to be simple. In this sense, our theories, while "true," are useful only to God, which seems not to be the hallmark of what humans adjudge to be truth.

4.2 Model Formally Undecidable

A conventional view of Gödel's incompleteness theorem is that it is largely irrelevant to the daily practice of mathematics.[21] Chaitin has argued for essentially the opposite perspective: the problems of incompleteness are ubiquitous and place a serious constraint on what can be accomplished with mathematics. In the following example decidability rears its head in a context closely related to a variety of social science problems.

Example 8 [Buss *et al.* 1991]: Consider the following system of coupled automata. There are A identical automata, each having some finite number of states, S. The initial state of each automaton is given; an A-vector specifies all states. There is a global control rule, G, which for simplicity take to be a first-order sentence. Finally, there is some time, T, at which we wish to know the state of the system. Let us say that this system is predictable if it is possible to determine its state at T in an amount of time that is polynomial in A, S, and the logarithm of T. Formally, if the system is not predictable in this way, then it is PSPACE-complete, and the only way to effectively predict the state vector at T is to build an automata (multi-agent) model and execute it; that is, agent-based modeling is the best one can do.

Definition: A poll, $P(x)$, gives the number of automata in state x.

Definition: G is *constant-free* if it does not refer to any particular state of the automata.

Instance of a constant-free rule: If $\forall x\ \exists y$ such that $P(x) = P(y)$ then TRUE else FALSE.

Instance of a non-constant-free rule (referendum on s_i): If $P(s_i) > A/2$ then TRUE else FALSE.

Instance of an apparently non-constant-free rule that is actually constant-free: If $\forall x\ P(x) = P(s_i)$ then TRUE else FALSE; this rule is actually constant-free since it is equivalent to: If $\forall x\ \forall y\ P(x) = P(y)$ then TRUE else FALSE.

Theorem 1: If G is constant-free then the system can be predicted in polynomial time.

Theorem 2: If G is not constant-free then the system is PSPACE-complete.

Theorem 3: The set of G that are constant-free is undecidable.

Many aspects of this example have much in common with modern, mathematical social science. Agents are interacting, they are changing state over time while engaged in a kind of voting behavior. It would seem that progress in social science will come from a better understanding of such minimal, abstract formulations of social processes.

[21] For a typical statement of this type see the October 1997 *SIAM News* and the review of "Logical Dilemmas: The Life and Work of Kurt Gödel." The reviewer writes "...it is an important fact that the brilliant, earth-shaking theorems of Gödel are of absolute unimportance to 99.5% of research mathematicians in their professional work."

4.3 Emergence

Up until now we have made very little of what is claimed by some to be one of the most important properties of agent-based models, their ability to generate emergent structures or properties. The reason for this is that as of today there exists no very satisfactory mathematical theory of emergence.[22] Thus, one cannot discuss in any definitive way the relationship between emergence in multi-agent systems and how this relates to mathematical theory.

5 CONCLUSIONS

It has been argued that there exist three distinct uses of agent-based computational models. First, when numerical realizations are relevant agents can perform a variant of classical simulation. Second, when a model is only incompletely solved — its equilibria unknown, stability of equilibria undetermined, or the dependence on parameters opaque — then an agent-based model can be a useful tool of analysis, a complement to mathematics. Third, there are cases in which mathematical models are either apparently intractable or provably insoluble. In such circumstances it would seem that agent-based modeling is perhaps the only technique available for systematic analysis, a substitute for formal mathematical analysis.

There are two other reasons that augur for the increased prominence of agent-based models in the social scientist's toolkit. First and foremost, continued hardware evolution will soon place on our desks machines with capabilities beyond John von Neumann's wildest dreams. These machines will have the capability of modeling 10^6 agents of reasonable complexity, even larger numbers of simpler agents. Entire mini-economies will be able to be built in silicon, if we know enough about how to build agents in software, agents who trade in markets, who form firms, who engage in political activity and write constitutions and bribe other agents for votes and try to pass term limits. Today we do not know how to do all these things.

The second reason to work with computational agents is that this is a critical time for software agent technology. We are learning everyday a little more about how to get agents to cooperate, compete, and engage in conflict. There has been a sudden flurry of activity around agents, particularly in computer science, where engineers are busy populating the Internet with tradebots, automated markets, auction engines, and other agent-like devices.

There is an irony in this arrival of agents — in many ways a foreign methodology — to economics and political economy that is too pregnant to neglect. A key theme in agent computing is that of decentralization, that realistic social processes can be seen to emerge through the interactions of individuals. While decentralization is also an important theme in modern, mathematical social science, so is a competing "social planner" view, in which optimal outcomes are seen to be the result of a benign or perhaps benevolent auctioneer, who has perfect information and infinite computing power at its disposal, who can design perfect mechanisms and who can compute, implement and enforce optimal tax rates. This has given a peculiar

[22] Recent attempts to formalize this notion include Baas [1994]. For a cogent discussion of emergence, pro and con, see the new chapter 7 in the new edition of Simon [1996].

methodological orientation to certain sub-fields, especially in economics, of which general equilibrium is an example. But things are changing today, as Rust [1996] writes:

> "The reason why large scale computable general equilibrium problems are difficult for economists to solve is that they are using the wrong hardware and software. Economists should design their computations to mimic the real economy, using massively parallel computers and decentralized algorithms that allow competitive equilibria to arise as 'emergent computations'...[T]he most promising way for economists to avoid the computational burdens associated with solving realistic large scale general equilibrium models is to adopt an "agent-based" modeling strategy where equilibrium prices and quantities emerge endogeneously from the decentralized interactions of agents."

Today it is conventional practice among mathematical social scientists to state a model in mathematical terms and then attempt to solve it analytically. Only later, if at all, does one resort to a computational model, usually in the face of analytical difficulties. However, given that agent computational models are a powerful analytical tool as well as an effective presentation technology, one wonders whether such models might not someday be the first line of attack on new problems, with authors resorting to mathematics only to "tidy up" what the agent model has already demonstrated. Perhaps, as printed journals, with their static equations and figures, give way to electronic journals and dynamic, downloadable model animations, there will come a day when we all will wonder how we got along without agents.

ACKNOWLEDGMENTS

A preliminary version of this paper was circulated under the title "Three Distinct Uses of Agent-Based Computational Models in the Social Sciences." For useful comments I thank Art DeVany, Joshua Epstein, Fredrik Liljeros, and conference participants at Chicago and UCLA.

REFERENCES

Amman, H.M., D.A. Kendrick, and J. Rust, eds. 1996. *Handbook of Computational Economics*. North-Holland: New York.

Angeline, P., R.G. Reynolds, J.R. McDonnell, and R. Eberhart, eds. 1997. *Evolutionary Programming VI*. Springer-Verlag: New York.

Arthur, W.B., J.H. Holland, B. LeBaron, R. Palmer, and P. Tayler. 1996. Asset Pricing under Endogenous Expectations in an Artificial Stock Market. Working paper.

Axelrod, R. 1997. The Dissemination of Culture: A Model with Global Polarization. *Journal of Conflict Resolution*, *41*: 203-226. Reprinted as Chapter 7 in Axelrod [1997].

Axelrod, R. 1997. *The Complexity of Cooperation.* Princeton University Press: Princeton, New Jersey.

Axtell, R. 1999. The Emergence of Firms in a Population of Agents. Working paper. Brookings Institution. Available at www.brookings.edu/es/dynamics/papers.

Axtell, R.L. 2000a. "Effects of Interaction Topology and Agent Activation Regime in Several Multi-Agent Systems." Working paper. Brookings Institution. Available at www.brookings.edu/es/dynamics/papers.

Axtell, R.L. 2000b. The Complexity of Exchange. Working Paper. Brookings Institution. Available at www.brookings.edu/es/dynamics/papers.

Axtell, R., R. Axelrod, J.M. Epstein, and M.D. Cohen. 1996. Aligning Simulation Models: A Case Study and Results. *Computational and Mathematical Organization Theory, 1* (2): 123-141. Reprinted as Appendix A in Axelrod [1997].

Axtell, R., J.M. Epstein, and H.P. Young. 1999. Emergence of Classes in a Multi-Agent Bargaining Model. Working paper. Brookings Institution. Available at www.brookings.edu/es/dynamics/papers.

Baas, N.A. 1994. Emergence, Hierarchies, and Hyperstructures. In *Artificial Life III***,** C.G. Langton, ed. Addison-Wesley: Redwood City, Calif.

Banks, J. and J.S. Carson, III. 1984. *Discrete-Event System Simulation*. Prentice-Hall: Englewood Cliffs, New Jersey.

Binmore, K. and L. Samuelson. 1992. Evolutionary Stability in Repeated Games Played by Finite Automata. *Journal of Economic Theory, 57*: 278-305.

Board, R. 1994. Polynomially Bounded Rationality. *Journal of Economic Theory*, *63*: 246-270.

Bratley, P., B.L. Fox, and L.E. Schrage. 1987. *A Guide to Simulation*. Second edition. Springer-Verlag: New York.

Buss, S.R., C.H. Papadimitriou, and J.N. Tsitsiklis. 1991. On the Predictability of Coupled Automata: An Allegory about Chaos. *Complex Systems*, *5*: 525-539.

Canning, D. 1995. Evolution of Group Cooperation through Inter-Group Conflict. Working paper. Department of Economics, Queens University of Belfast, Northern Ireland.

Casti, J. 1997. *Would-Be Worlds*. Wiley: New York.

Chen, S.-H. and C.-H. Yeh. 1997. Modeling Speculators with Genetic Programming. In Angeline *et al.* [1997].

Conlisk, J. 1996. Why Bounded Rationality? *Journal of Economic Literature, XXXIV*: 669-700.

DeVany, A. 1994. Hard Cores, Soft Cores, and the Emergence of Self-Organized Coalitions. Working paper. UCLA, Center for Computable Economics: Los Angeles, Calif.

Epstein, J.M. and R. Axtell. 1996. *Growing Artificial Societies: Social Science from the Bottom Up*. Brookings Press and MIT Press: Washington, D.C., and Cambridge, Mass.

Feigenbaum, M. Presentation Functions, Fixed Points, and a Theory of Scaling Function Dynamics. *Journal of Statistical Physics*, *52* (3/4): 527-569.

Gilli, M., ed. 1996. *Computational Economic Systems: Models, Methods & Econometrics*. Kluwer: New York.

Glaeser, E., B. Sacerdote, and J. Scheinkman. 1996. Crime and Social Interactions. *Quarterly Journal of Economics*: 507-548.

Hirsch, M.D., C.H. Papadimitriou, and S.A. Vavasis. 1989. Exponential Lower Bounds for Finding Brouwer Fixed Points, *Journal of Complexity, 5*: 379-416.

Huberman, B.A. and N.S. Glance. 1993. Evolutionary Games and Computer Simulations. *Proceedings of the National Academy of Sciences, USA, 90* (August): 7716-7718.

Judd, K. 1997. Computational Economics and Economic Theory: Substitutes or Compliments? *Journal of Economic Dynamics and Control, 21* (6): 907-942.

Kollman, K., J.H. Miller, and S.E. Page. 1997. Political Institutions and Sorting in a Tiebout Model, *American Economic Review*, *87* (5): 977-992.

Minar, N., R. Burkhart, C. Langton, and M. Askenazi. 1996. The SWARM Simulation System: A Toolkit for Building Multi-Agent Simulations. Working paper. Santa Fe Institute: Santa Fe, New Mexico.

Mor, Y., C.V. Goldman, and J.S. Rosenchein. 1996. Learn Your Opponent's Strategy (in Polynomial Time)! In G. Weiss and S. Sen, eds. *Adaptation and Learning in Multi-Agent Systems*. Springer-Verlag: New York.

Nagel, K. and S. Rasmussen. 1994. Traffic on the Edge of Chaos. Working Paper 94-06-032, Santa Fe, Santa Fe Institute: N.M.

Neumann, J. von. 1945-46. A Model of General Economic Equilibrium. *Review of Economic Studies*, *13*: 1-9.

Newell, A. and H.A. Simon. 1972. *Human Problem Solving*. Prentice-Hall: New York.

Neyman, A. Bounded Complexity Justifies Cooperation in Finitely Repeated Prisoner's Dilemma. *Economics Letters*, *19*: 227-229.

Papadimitriou, C.H. 1994. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *Journal of Computer and System Sciences*, *48*: 498-532.

Papadimitriou, C. 1993. Computational Complexity as Bounded Rationality. *Mathematics of Operations Research*.

Prasad, K. 1997. On the Computability of Nash Equilibria. *Journal of Economic Dynamics and Control, 21* (6): 943-953.

Rubinstein, A. 1986. Finite Automata Play the Repeated Prisoners' Dilemma. *Journal of Economic Theory, 39*: 83-96.

Rust, J. 1996. Dealing with the Complexity of Economic Calculations. Working paper. Department of Economics, Yale University: New Haven, Connecticut.

Simon, H. 1996. *The Sciences of the Artificial*. Third edition. MIT Press: Cambridge, Mass.

Spear, S.E. 1989. Learning Rational Expectations under Computability Constraints. *Econometrica*, *57*: 889-910.

Young, H.P. 1993a. The Evolution of Conventions. *Econometrica, 61* (1): 57-84.

Young, H.P. 1993b. An Evolutionary Model of Bargaining. *Journal of Economic Theory, 59*(1): 145-168.

Young, H.P. 1998. Diffusion in Social Networks. Working paper. Brookings Institution: Washington, D.C. Available at www.brookings.edu/es/dynamics/papers.

DISCUSSION:

AGENT-BASED SOCIAL SCIENCE MODELS*

Fredrik Liljeros (discussant): It has been a pleasure to listen to your presentation and to read your paper. I think it has the potential to be a classic introductory text about agent-based computational models. I do not find anything in it that I don't agree with you about, except one minor thing, so I'm going to focus my discussion on things that maybe you can take even a bit further.

Let me start with your classification scheme for agent-based models that you propose, where you classify models into three distinct types: one where the problem can be formally represented in mathematical equations and solved in total, one where equations can be written down but not completely solved, and lastly one where writing down equations is not a useful activity. I think this makes a lot of sense.

But I don't understand the need for only defining agent-based models in the first group as "true" simulations. As I understand it, your motive for this is grounded in the way the concept is used in operations research. And even though I'm far from being a native speaker, I'm rather sure that the word "simulation" already has a wider use in everyday language and academia. Aren't, for example, agent-based traffic models like TransSim, very often called simulation models in the scientific community?

I totally agree with you that this first case is a specific kind of agent-based model, but my suggestion is that, instead of arguing that this is the only true simulation, you should try to come up with another, more specific concept for this specific type of simulation. My suggestion is based on the fact that I'm not aware of any case where someone has succeeded in redefining an already widely used concept like simulation into a narrower definition like you are trying to do here. Okay, that was my only critique.

So what I'm going to do now is to discuss things in the article where I think you are totally right and where, at least from my point of view as a sociologist, there are reasons for being even more positive than you are in the text when looking to the future for agent-based models. I think the way you used Newell and Simon's discussion about sufficiency theorems seems very fruitful, though I think agent-based modeling could have an even more important role to fill than just convincing people about sufficiency. The English philosopher of science Roy Bhaskar has, for example, pointed out that in science there is not so much a question of whether a theory is true or false, but whether a theory explains a phenomenon better than any competing theory. And I'm rather convinced that this is how the scientific community works in practice. I think that translating a social theory into the form of an agent-based model increases its rationalistic content, and consequently its competitive capacity, over the nonformalized theories that are very common in sociology.

* [Editor's note: The question-and-answer sessions were recorded and transcribed, and the transcripts were edited for continuity and ease of reading. Every effort was made to identify speakers and to interpret comments accurately.]

I also think, or at least I hope, that you're right in your speculation that agent-based models in the future will be the first line of attack on new problems, with authors resorting to mathematics only to tidy up what the agent model has already demonstrated. With a little bit of luck, this will also be the case in sociology too, where we do not have this strong tradition of building mathematical deductive models. One reason for believing why this has the potential to be the case is that it takes a much shorter time to learn, for example, object-oriented programming than it takes to learn enough math to build models that others will find interesting.

But I don't think that agent-based models will replace mathematical models in sociology. If sociologists start to use agent-based models more frequently, I think these models can function as an interface between sociologists and mathematically skilled people, and in this way stimulate the use of mathematical models by increased cooperation between sociologists and mathematicians.

The rare use of mathematical models in sociology today can, as I see it, partly be explained by the fact that sociologists in general lack enough mathematical knowledge to formulate their problems in ways that attract mathematically skilled persons, and I think that agent-based models can be a solution to this problem.

In the end of the article, you briefly discuss the concept of emergence. Your conclusion is that the lack of a satisfactory mathematical theory of emergence makes it impossible to discuss the relationship between emergence and multiagent systems and mathematical models. The problem of emergence has been a key problem in sociology ever since Durkheim founded sociology as an academic discipline. I am aware that you maybe are restricted from speculating about emergence by the purpose of your article, but anyway I think it would be interesting to hear a little bit about your thoughts about this concept and agent-based models as a tool for solving it.

Let me finish with some few short remarks. When you discussed the problem of artifacts, I think you missed mentioning the possibility of letting different people code the same model or having the same person write the model in different programming languages to safeguard from phenomena generated by the way the code is implemented. I'm not sure how much this is practiced in reality today, but I've tried this in cooperation with others, and it makes me a little bit more secure about my own results.

You also mentioned that agent-based models make it possible to study the effects of social and geographical networks, and I think also that it would be worth mentioning that agent-based models are very well suited also for studying the dynamics of the networks themselves.

So finally, I think that your article has an important role to fill as a presentation of different kinds of agent-based models, and if you think that the current form is too narrow for more speculative discussion, for example about emergence, then I really want to encourage you to also write a more speculative article based on the same concept.

Richard Gaylord: I'm Richard Gaylord, University of Illinois. The one thing that bothers me about agent-based modeling in particular is the mantra that's continually repeated that mathematics can't deal with these systems and therefore we have to use computers. And I'm not saying that that's not true, but it seems to me that it's an incredibly premature judgment to make. Mathematics has been developed to meet the needs that science poses for mathematics. Mathematicians would undoubtedly object to that, but scientists, including physicists, would certainly agree, although certainly heterogeneity is a major problem that in physics we tend not to have. But there is mathematics to be done, and mathematics indeed may give us a weapon that we don't have. And I would much prefer that people in the social sciences, and that includes economics, would simply say that we don't have these tools available to us now, so we really can't wait for mathematics to catch up with us and instead we use what we can.

What bothers me the most about agent-based modeling is that it seems to me that there's an awful lot of brute force — certainly a lot more brute force than thought — that's going into the development of models. It seems to me that in agent-based models, we have very ill defined systems; or, if not that, we have overly defined systems where we can't be sure that we're not getting an artifact of exactly what we've set up. In fact, the biggest problem I've had with all the agent-based modeling in the social sciences is the fact that the results we get are incredibly bound to initial conditions. And one of the things that I discovered in my own work — as I left the use of cellular automata and started to realize that social networks were in fact a key to doing any sort of analysis of social systems — was that I have no idea how to do those models. And I know that people are working here and elsewhere in setting up social networks, and they're very specific, and I don't know that I can take anything from those results and go beyond that to say, "Well, it doesn't matter whether you have weak ties or strong ties; what's going on?" I've talked to economists, and what they say is, "Well, just do it with a whole bunch of different networks and hope that the result is robust," which means it doesn't matter what kind of initial condition you have. That would probably be a very bad result to get, it seems to me. But I think there's a lot of work to be done in defining our systems.

And in terms of mathematics, just to conclude, even in doing mathematics, we are continually frustrated by trying to identify what boundary conditions we should use in the solution of partial and regular differential equations, not because we don't know how to do the math, but because we don't know what the conditions mean physically. And I myself worked in a field where they used the wrong boundary condition for 20 years and had very nice results, and it agreed with what the data said, and it turned out that it was wrong. They were just the wrong boundary conditions. I think it's the same problem with social networks. I think if you don't set up your boundary conditions, and your initial conditions, we don't really know that we're doing anything that's more profound than just playing a game and getting the results.

Robert Axtell: It may not have come across in the paper, but I'm all for more math in these areas, to help people understand these systems. I guess I'm primarily expressing skepticism that that's going to be rapidly forthcoming. Now, if I'm proved wrong about that, I'll be very happy. But it just seems like these are new areas, and there's new math to be done, of course, but it just seems like, at least from a purely rate-process point of view, the growth of the hardware is outpacing the growth of the new math. But I've heard [mathematician] Steve Smale say that we live in the golden age of mathematics where there are more well-trained mathematicians today than ever before.

So I'm not exactly sure what the long-term prognosis for math is. Maybe a different way to say it is that it's easier to project the long-term development of Intel hardware than it is the development of the math community.

When it comes to whether agent models are too specific, you're worried, Richard, that they're too tied to details that may or may not be important, and you're not sure about what that means. There again, the point of comparison has to be conditional modeling in the social sciences, not necessarily physics. I mean, when you say social networks matter and the result you get depends on what social network you employ, I say, fine, I agree with that. I'm sure that's going to be the case. But would you rather have this kind of noisy model, with a lot of different results coming out, or would you rather have the model that the CBO computed so everybody can interact with everybody else with equal probability?

So I guess my main point is that there is a network model underlying all social science, and to date that's largely equal probability of interaction, or, what's worse, to take the extreme case of [inaudible] equilibria, nobody interacts with anybody else. There's one auctioneer who figures out the best price and then everybody interacts with the price vector. I think that's very abstract. We started out the *Growing Artificial Societies* book with a quote from Herb Simon that says that now, in fact, the social sciences are the hard sciences because they're the *difficult* sciences. And I have no doubt that these are very difficult problems, and I don't believe that the agent-based modeling perspective is some kind of silver bullet that's going to cut through them quickly. But I do believe that it is a technique that already has shown promise early on, and I think that it is now starting, in some domains, to rapidly pull ahead of the conventional theorizing.

Michael North: I thought it was very interesting that you talked about computers and computation as sort of, I won't say a replacement for math, but at least a stopgap. That makes sense. From another perspective — from the artificial life view — computation is an extension to experiments, rather than a replacement or substitute for math. And that may have a lot of value, because the big danger with a lot of the agent-based modeling is that it will draw too much from one model, and you'll be picking up artifacts.

You mentioned that we probably should reimplement models in another language, hopefully by someone else, so maybe we'll have different artifacts, at least. It sounds more like an experiment.

Axtell: You know, I agree 100% with that. In fact, things like docking or coding in different languages, like having someone else do the coding, is crucial to knowing the robustness of the results. I have a kind of a vignette to say about this — my colleague Epstein and I recently had a paper published, a model of retirement decision making, in which there was a famous economist who was the discussant. We described the agent-based technique in some detail, along with the results. And he — we actually thought he would — actually bought in completely to our economic modeling, that is, our behavior modeling of the process of retirement decision making. But then when it came to describing the computation, he said, "I find the fairly significant treatment of computational details in this paper to be completely obfuscatory," or something to this effect. And he went on to say that he thought it was as irrelevant to describe the details of computation employed in the paper as it was to say that you composed your document using Microsoft Word. So I would say that his opinion is the opposite of yours. [*Laughter from audience.*] But I think you're closer to the truth.

Political Agents

ADAPTIVE MODELS AND ELECTORAL INSTABILITY

S. de MARCHI, Duke University*

ABSTRACT

This paper investigates the possibility of constructing measures of the underlying difficulty of a particular electorate's preferences and details how different levels of difficulty may affect electoral outcomes.

[*Scott de Marchi was unable to attend the workshop. Session discussant Meredith Rolfe summarized his working paper as it appeared on the web at the time of the conference. What follows is a summary of her remarks from notes taken by an attendee.*]

De Marchi's paper concerned electoral behavior, an area in which formal sociological models already exist. However, most of these formal models say that an incumbent should never win. In de Marchi's agent models, voters have different levels of information about the candidates and pay different amounts of attention to the political process. Also, the candidates have different levels of information about voters. De Marchi uses a rugged landscape to represent the range of opinions held by voters; the candidates, in turn, can be influenced by these opinions to different extents. In this model, incumbents can indeed be elected, under certain conditions, as in the real world. Other model results suggest some additional observations. For example, when people are not paying attention to the process, "opportunistic" candidates, who are trying to "sneak" their ideologies past the voters, could in fact be elected, but then the people "wake up" and elect someone else.

* *Corresponding author address*: Scott de Marchi, Department of Political Science, Box 90204, Duke University, Durham, NC 27708.

AGENT-BASED MODELING OF COLLECTIVE IDENTITY: AN ALTERNATIVE LARGE-N APPROACH

I.S. LUSTICK, University of Pennsylvania*

EXTENDED ABSTRACT**

The dominant theoretical approach to the emergence and transformation of collective identity is constructivism. Constructivists reject the view that group identities are inherited or otherwise primordially "given." Instead identities are portrayed as fluid, but not perfectly so. They are seen as chosen, subject to change as incentive structures change, and affected by manipulative entrepreneurs of culture. But the constructivist consensus has failed to produce work that goes beyond rejection of older, "primordialist" views. Agent-based modeling can be used as an effective means of refining, elaborating, and testing hypotheses drawn out of the basic constructivist position.

This paper presents the ABIR (Agent-Based Identity Repertoire) model, which seeks to refine, elaborate, and test constructivist theories of identity and identity change. In this model, agents with activated identities interact on a landscape. These agents have repertoires of latent identities. They respond to pressures toward conformity within their neighborhoods and to global changes in external biases advantaging or disadvantaging different identities over time. A simple set of micro-rules, conforming to constructivist theory's standard propositions about the fluidity, multiplicity, and institutionalizability of identities, as well as agent responsiveness to changing incentive structures, determines in any particular interaction what identities will be activated, deactivated, or maintained. Macro-patterns that emerge from these myriad micro-interactions can then be systematically studied.

Statistical monitors associated with the model allow data to be gathered from repeated runs under strictly controlled parametric conditions, under randomized initial conditions, and combined with analytically strategic variation. Experiments reported in this paper focus on how variation in the size of agent repertoires can affect tension reduction and aggregation across the landscape. We find that overall tension levels, measuring amounts of difference in the encounters across the population among activated identities, change in curvilinear patterns with increases in the size of agent identity repertoires. These results are robust but significantly affected when heterogeneity across the population is increased, when entrepreneurs (more sensitively responsive agents with repertoires larger than basic agents) are introduced into the population, and when the environment, or incentive structure, is made to change in more turbulent ways. Roughly the same results were observed when Herfindahl Index measures of "identity aggregation" (the level of concentration in the "market shares" occupied by different identities) were examined. These results suggest that cultural groups can most stably institutionalize themselves when individuals across the population have neither a very small repertoire of possible identities nor a very large one.

---

* *Corresponding author address*: Ian S. Lustick, 217 Stiteler Hall, University of Pennsylvania, Philadelphia, PA 19104-6215; e-mail: ilustick@sas.upenn.edu.

** Full paper appears in the proceedings of the 1999 Annual Meeting of the American Political Science Association (Atlanta, Georgia, September 2-5, 1999).

Of particular interest were the results of introducing a relatively small amount of skewness into the initial distributions of latent (subscribed) identities. Identities were initially distributed randomly at the activated level and at the subscribed level. In the experimental condition, the random distribution of activated identities was accompanied by a mild degree of skewness in the distribution of subscribed identities. In the moderately uneven, or skewed, initial subscription condition, 20% of identities appearing in the initial landscape were present in significantly fewer agent repertoires than the other 80%. In this condition, we observed a curvilinear pattern in the aggregation of identity that was considerably more pronounced than when the initial distribution at the subscription level was even — an effect that was robust across increases in the heterogeneity of the population and with the introduction of entrepreneurs (although entrepreneurs did moderate the effect).

We analyze this effect as the result of an increase in tipping or cascade effects that arise from the subtle asymmetries present in the populations with skewed subscription distributions. These asymmetries produce opportunities for quick and decisive expansion when biases temporarily become favorable. This relationship is suggestive of a somewhat paradoxical implication of the presence within a population of exclusivist identities. If exclusivist identities are those which are relatively less able to coexist in the same agents who also have (unspecified) clusters of other identities, then their presence represents the skewness in the distribution of latently available identities. The presence of these exclusivist identities then explains the marked tendency for some inclusivist identities to more often extend themselves over very large proportions of the population.

The paper concludes with a discussion of plans for further research.

DISCUSSION:

POLITICAL AGENTS

Meredith Rolfe (discussant): In thinking about these papers, I tried to look more from the point of view of someone who is substantively interested in the problems and how they affected my thinking about the substantive concerns of identity and elections, rather than purely from the point of view of someone who is interested in agent modeling. And what I found was that a lot of my questions and concerns were about the meeting between empirical work — empirical findings, things that I know about how the world works — and the agent systems themselves. And I tried to break that down into categories, because I think that's a tension, and it's always going to be a tension in models like this. How much do you go toward having your models perfectly simulate the real world? I want to just reflect on that a little bit in light of these two papers.

So one question is in problem selection and hypothesis formation itself. One of the things that I found is that in problem selection, it's a whole lot easier when you go in, as Scott [de Marchi] did, and have a nice, easily-laid-out terrain of formal issues and attack them rather than having to just sort of dive in and do a completely exploratory model, like Ian [Lustick] did. And it ended up making differences in what it had you think about.

Ian, I ran into problems with your hypotheses about the repertoire size, because it in essence seemed to be saying, "When people have stuff in common, they're less likely to conflict," which seems obvious. I think that actually as you went on in your discussion it made more sense. But it would be nice to hear, whenever a hypothesis is made about something, where it comes from in the real world, what type of research is informing the hypothesis, instead of just having it be something that could be an artifact of the model.

Model assumptions, boundary conditions — I think they actually do pose a huge problem, because you can get too simple or too complex. Some comments on bias numbers: I immediately wanted to know where the bias numbers came from, how robust the model was to different bias numbers, and what it was supposed to be in the world. There's also the question of social networks and diffusion, and I know that this was brought up earlier today, but from network analysis, we know that the diffusion of innovations really depends on the network itself — that if you have a very centralized network things are going to diffuse, and if you have a less centralized one, the density of the network matters. And in all of these agent-based models, the networks are stuck. It's just one person with eight other people around him. And that's not really what the world looks like, and it can really affect how innovations diffuse across the landscape.

[*At this point technical difficulties interrupted the taping of the discussion for the rest of this session. What follows is a summary from notes taken by an attendee.*]

Rolfe continued by saying that she liked Lustick's use of the concept of opinion leaders/entrepreneurs [outspoken advocates of certain positions] because it "gets at the idea that there are some people with more influence." She raised a concern about de Marchi's "issue space" for representing opinions, noting that there were only 10 opinion states, so the issue space was discontinuous. She proposed looking at the question of what effect a continuous issue space would have on the ruggedness of the landscape. She also suggested that the effect of voter attention on election outcome might be an artifact, and wondered why de Marchi chose to have voter attention drive candidate behavior. Continuing the earlier discussion about model validation and correspondence between models and the real world, Rolfe suggested two conclusions from these papers that could easily be checked against the world: (1) Lustick's conclusion that opinion leaders help retain diversity of views and (2) de Marchi's conclusions about the conditions under which incumbents can be elected.

Ian Lustick commented that he didn't consider that hypotheses should necessarily come from the world, noting that part of the problem with using "folk theorems" is that one may be building the effect into the settings. He questioned how much is learned if one starts with a freestanding theorem, and "*mirabile dictu*," out come the expected results. He said that with his model, he didn't anticipate that the decline in population diversity would be asymptotic. This question had not occurred to anyone, although it is implicit in the theory. They now have a means of generating predictions and hypotheses: they can make individuals apathetic, fanatic, or charismatic, and compare their effects to social data — it's a new direction for empirical studies.

Michael North, commenting on the issue of experimental work and bias in models, noted the extensive efforts in the social sciences to create experimental designs to avoid bias.

Lustick noted that selection bias enters into his model when deciding on the "standard" setting for the individuals' biases, that is, their identities. In the real world, a person's biases have specific referents. Another question is how should one consider the idea from evolutionary theory that variation in scale is more difficult to adapt to than variation in rate. Also, what is the effect of a cylindrical, closed landscape versus an open one? His approach is to change the settings in the directions that theory indicates are important, run every condition many times, and calculate several indexes of diversity.

Computational Models of Social and Economic Processes

THOUGHTS ON DIVERSITY

S.E. PAGE, University of Iowa*

INTRODUCTION

Diversity takes many forms. People differ. Products differ. Organizations differ. Norms, languages, and cultures differ. These differences can be beneficial, creating an abundance of activities, arts, and commodities, and they can be harmful, leading to warfare and other less tragic, but nevertheless unfortunate, events such as the crash of a space probe because one group of scientists describes distances in meters while another uses feet.

In these comments, I discuss diversity in the context of several simple models that, with the help of co-authors, I have constructed over the past few years. I hope through these examples to convince you that diversity matters. This rather modest goal fits within a larger agenda: to analyze the benefits of a complex systems approach to modeling social systems. Other scholars have advanced and outlined this broader agenda more thoughtfully than I do here, and I encourage the interested reader to read further (see Axelrod 1997, Axelrod and Cohen 1999, Epstein and Axtell 1996, Holland and Miller 1991, and Tesfatsion 1997). These advocates of computational approaches to social science emphasize the inclusion of diversity, together with interactions, geography, and dynamics, as an advantage of complex systems modeling. I offer these simple models as preliminary evidence in support of the importance of diversity.

BACKGROUND

The models that I describe all are either borrowed from or applied to political and economic problems. In each, I think of a political economy as a complex evolving system. People, firms, organizations, and institutions interact in an environment that, though predictable in places and at times, often changes in novel and unexpected ways. This partial predictability is important. An argument can be made that any functioning social system should balance the predictable with the novel. A complete lack of predictability precludes long-term and even short-term planning. Total predictability would be stifling (Dubos 1968). No news every day would not be “good” news.

The challenge for social scientists is to model these two tendencies: the groping toward equilibrium and the creation of novelty. An established equilibrium theory based on rational, optimizing actors has proven partially effective at modeling the former, but it has failed at the latter. The newer theory of complex adaptive systems has made some encouraging first steps in creating dynamic, perpetually novel worlds. But the gap between creating and explaining novelty is a large one. And complex adaptive systems research has a long way to go.

Complex adaptive systems models typically include diverse, adaptive agents with geographic locations interacting in time — real-world features often missing from equilibrium models. This is not to discredit equilibrium theory. It has led to deep and important insights about economic, political, and social phenomena. I enjoy equilibrium theory. I have not come to praise

* *Corresponding author address*: Scott E. Page, W380 PBAB, University of Iowa, Iowa City, IA 52242-1000; e-mail: scott_page@uiowa.edu.

it, nor shall I try to bury it. I just want to emphasize that any theory must (1) rely on simplifying assumptions, (2) have limited domains of applicability, and (3) be at the mercy of the level of technique development. For example, equilibrium models usually rely on individual agents as the units of analysis. Obviously, this may not always be appropriate or accurate. Often people are best defined contextually. Someone may play several roles: mother, daughter, teacher, student, and partner. The influences that pulse across these connections and not some single utility function may be the key determinants of behavior. In other words, revealed preferences may be revealed connections. And "rationality" may only be an accurate assumption provided that the connections are unchanging. That said, the methodological individualism that has underpinned economic theory has allowed for powerful, elegant theorizing that routinely has withstood empirical testing.

Complex systems models have their own strengths and weaknesses. A strength, as I hope to demonstrate in this paper, is their ability to admit diverse agents. As in economic models, we can endow agents with different preferences, distinct endowments, and even different information. But we can also include a diversity of world views, sets of human capital, and geographic locations. In the simple models that follow, I explore various forms of agent diversity.

DIVERSITY OF PROBLEM SOLVERS

Some problems, such as adding two plus two, are easy; other problems, say, developing a clean, renewable source of energy, are more difficult. The former, we tend to solve optimally. On the latter, we do the best we can. And typically, when confronted with a hard problem, two people are not likely to solve it in exactly the same way. These differences could result from distinct ways of seeing the world — what computer scientists call encodings. They could also stem from people possessing unique sets of problem-solving techniques and tools. In a series of papers, Lu Hong and I construct a formal model of diverse problem solvers (Hong and Page 1998, 1999). We characterize a problem solver along two dimensions: perspectives and heuristics. A perspective is an internal representation of the problem. Heuristics are how we manipulate candidate solutions within our perspectives.

When confronted with a challenging problem, a problem solver encodes her problem using her perspective and applies her heuristics in an attempt to locate an improving solution. She probably will not locate the optimum, but so what. Human progress is a story of improvement, not of optimality. In our framework, a collection of agents will find a sequence of solutions, each better than its predecessor, but each a local optima — a solution that when represented in the agent's perspective is not near (via her heuristics) any better solutions. Metaphorically, this process can be described as a walk on a rugged landscape.

Two agents are not likely to take identical walks or get stuck on the same solutions, and this diversity proves beneficial. We prove that a collection of diverse agents can solve even the most recondite of problems. Any local, but nonglobal, peak eventually will be surmounted by someone. In contrast, if all agents are bounded in the same way, then they would all get stuck at the same locations and a collection of agents would do no better than any one person working alone.

This basic insight requires some clarification. First, it does not mean that a group will always outperform an individual. The result has two hidden assumptions. Everyone in the group must assign the same value to every solution, and everyone must be able to communicate. If either of these assumptions fails, then a group could make a lousy decision. Second, it differs from the standard explanation of how boundedly rational people solve difficult problems. Alternatively stated, it's an explanation of "If we are so stupid, how did we manage to put someone on the moon?" Finally, it need not be interpreted as what would result from an empowered, communicative group working together. It can also be viewed as what occurs over time in an economy in which people work sequentially and independently, as scientists and mathematicians have, in pursuit of solutions to important problems.

More interesting than this basic result — that diversity leads to optimality — are the corollaries. In a rather technical model, we show the following result: Given some rather mild assumptions, if you rank problem solvers by their ability to locate good solutions individually and take the best twenty problem solvers and let them work collectively, their performance will be inferior to that of a random collection of intelligent problem solvers. The reason why is that the first group tends to be homogeneous. Imagine a problem defined on a two-dimensional surface. Suppose that problem solvers use the same heuristic: they climb local gradients, but they differ in their perspectives: they use alternative bases for two-dimensional Euclidean space. So, one problem solver's perspective might be the canonical basis: the vectors (0,1) and (1,0). Another problem solver might use the basis vectors (2,1) and (1,4). Suppose the best-performing problem solver uses the basis vectors (3,4) and (5,1). Provided the underlying problem to be solved has some exploitable structures, the second-best problem solver is likely to use basis vectors similar to those of the best problem solver, perhaps (3,4) and (9,2). Using a similar argument, it follows that the "best" problem solvers tend to rely on similar basis vectors. In other words, they tend to look at the world similarly. Thus, collectively they are not much better than they are individually. The group of random, intelligent agents performs better because they see the world from all sides.

The formal theorem relies on technical assumptions requiring that the set of problem solvers be large, diverse, and intelligent. A collection of stupid problem solvers would not be collectively brighter than the proverbial monkeys at typewriters. The result accords nicely with evidence of the multidimensionality of intelligence. Yet it says something slightly different. It says that intelligence is contextual. Someone's contribution to a problem depends on the collection of other people working on the problem.

A second corollary pertains to the role of consultants. The U.S. economy has experienced a sharp rise in the number of consultants. This is a bit of a puzzle. Economists often talk about how a market economy exploits returns specialization. Bridge designers know bridges. Chefs know food. The former need only know how to eat and the latter how to walk to benefit from the expertise of the other. The prevalence of consultants calls into question this standard intuition. How can consultants know more about bridges than engineers do and more about food than chefs do?

Answer: they don't. Within any firm, a common language or set of languages is required in order to facilitate group problem solving, a point that we made in our main result. This adherence to a common perspective or set of perspectives has limitations. Someone need not be smarter to tell a firm how to make a better mousetrap: they need only come equipped with a different way of looking at the problem. This is part of what consultants do — though if you were to ask them, they would support the conjecture that they possess supreme intelligence.

DIVERSITY OF ROUTES

In his important book *Micromotives and Macrobehavior*, Thomas Schelling (1978) describes how amazing it is that the economy works at all. After all, millions of people make billions of decisions every day. Yet, these all manage to co-exist with surprisingly little conflict (Kirman 1997). Yes, stores sometimes run out of food, traffic jams occur in Los Angeles and even Kansas City, and English Ph.D.s face a tough job market, but for the most part the economy works. Many attribute this organization to the magic of incentives and markets. But general equilibrium theory and computational models do not describe the geographic coordination of economic activity, only the formation and efficiency of market prices. So, in an attempt to reduce the amazement, I constructed a simple model of geographic organization (Page 2000).

In the model, agents must visit K locations in K periods. You can think of this as a daily route: I must go to the hardware store, the laundromat, the bank, the grocery store, the post office, and the movies. Or, preferably, you can think of it as a weekly organization. Tuesdays, I bowl. Wednesdays, I play poker with friends. Saturdays, I go to movies. That sort of thing. The model is not meant to be overly realistic, just suggestive of a reality in which we choose places and times for our activities.

The driving assumption of the model is that people hate crowds. We would like to minimize the number of people at the bank so that we don't waste the afternoon waiting in line. (Incidentally, that's why I go to the bank at 8:45 on Wednesdays.) In the model, agents choose an ordering of the K locations. If there are three locations, an agent might choose the route 132, or the route 321. In an economy with three agents, if one chose the route 123, another the route 231, and the third the route 312, then each would be alone at each location in each period.

A collection of routes is organized if the number of agents at each location in each period is equal and minimal. A minimal organized collection of routes needs only K distinct routes. The notion of organized routes begs some mathematical questions. For example, can any organized collection of routes be decomposed into disjoint sets of minimal organized collections? Answer: no. With a little effort, you can construct some rather baroque nondecomposable organized collections.

One question is whether adaptive agents could evolve organized collections of routes. The models shows that they often can. Another question is whether the system could get stuck at a disorganized collection, where no agent could improve her route, but the collection was not organized. In the paper, I show that regardless of the set of possible routes that an agent could choose, this is always a possibility. But, as the agents get smarter, as they are allowed to choose from a larger set of possible routes, the probability that this occurs decreases.

The agents in this model are not optimizing and fully informed. They have only local information and limited route-generation ability. I considered two distinct learning rules that I called BRO and SIS. BRO agents were best responders. BRO agents looked at their friends and copied the best route they saw. SIS agents performed an iterative search over nearby routes — they switched pairs of locations. If they got better performance, they stuck with the new route.

Both BRO and SIS agents proved adept at evolving organized collections of routes, and at doing so quickly. SIS proved better than BRO. BRO suffered from being limited to the set of routes in the initial population. SIS allowed for any route to be evolved. This increased flexibility explains why the SIS agents did better.

Equally interesting was the type of organized collections that tended to evolve. They were not minimal. Under SIS, each agent tended to evolve a unique route. This could be seen as a weakness. Mathematically, the minimal organized routes are far more elegant then the elaborate collections that the agents evolved. The minimal collections look like something that would be planned centrally. The evolved collections appear more organic, more diverse.

This diversity, though messy and interesting, has an unexpected benefit: robustness. To see how diversity improves robustness, imagine that one location and one period are dropped. Assume that agents visit the remaining K – 1 locations according to their previous order. In other words, if an agent began with the route 53214 and location 2 were dropped, she would then visit the four remaining locations in the order 5314. A collection of routes is robustly organized if it remains organized after a location and period are dropped. It can be shown that there exist robustly organized collections of routes. An example is the permutation group on K elements. If you drop one location and one period, you end up with K copies of the permutation group on K – 1 elements, which is also organized.

The evolved collections of routes were rarely robustly organized. That would be too much to expect. However, they were far more robust than minimal organized collections of routes. And an analysis of all of output suggests that the diversity of routes, as measured by entropy, appears to be positively correlated with the robustness of the system to dropping locations.

This finding, that diversity and robustness are related, echoes Norman Johnson’s research (Johnson 1998). He constructs a model where agents evolve paths on a graph and then edges disappear. He finds that the agents demonstrate collective robustness. The agents, by following their bread crumbs, happen upon another good route. Incidentally, in his Pulitzer-Prize-winning book *Annals of the Former World*, John McPhee describes how explorers often relied on buffalo paths to find routes across mountains. Buffalo probably relied on a process similar to the one that Johnson simulates.

DIVERSITY OF PREFERENCES

People differ in their preferences regarding public policies. Some of us care about school quality, others want a clean environment, and still others want both good schools and a cleaner environment. These differences in opinion create political difficulties. As Ken Arrow showed in *Social Choice and Individual Values*, we may not be able to aggregate the preferences of individuals into a social welfare function. Only when there exists substantial agreement and symmetry can we expect voting to lead to a single equilibrium. Otherwise, we should expect the policy to be near the center of voters’ preferences but to roam around a bit.

This ebb and flow of policies is evident in the political history of many nations. In democracies, coalitions form only to fall apart. This complexity is partly due to the fact that any fixed policy can be defeated. A challenger just needs to find the right collection of policy positions. This political instability has been trotted out as a weakness of democracy. However, Ken Kollman, John Miller, and I have shown that in a larger system, the instability brought about by diversity can be beneficial. More precisely, within a democratic system with multiple jurisdictions, diversity creates problems but also contains its own solution to those problems.

Imagine a society with K towns. Each town has to make a decision on N binary public policies, such as whether to have a recycling program, whether to impose a curfew, and whether to build a new park. Citizens have diverse preferences over these policies and possess the ability to move to a different town if that town offers a preferable vector of policies. In this system, the preference diversity creates policy instability, which, in turn, creates movement between locations.

Ideally, citizens would sort into towns where all had similar preferences. If this were to occur, then each town would have a stable policy. Now, as it turns out, this is what happens. Why? Suppose that citizens are sorted poorly. As I just described, this means more sorting and more policy experimentation. Thus, the system settles down only when the sorting is good.

This idea — instability at low values and stability at high values — underpins a search algorithm known as simulated annealing. In simulated annealing, a temperature is lowered over time. At high temperatures, the search for solutions can take steps downhill. At low temperatures, simulated annealing is hill climbing. Democracy naturally anneals. Bad sorts are unstable. Good ones are not.

THE END OF DIVERSITY

In my last example, I describe preliminary research on the growth of chain stores. As chains grow, they destroy diversity according to a process I call bootstrapping homogeneity. The idea is simple. Suppose that a chain store enters in several towns. If it is the first chain to enter, it must be an impressive idea in order to succeed in two distinct towns. Suppose that two or three chains have entered the towns. Now, the competitors in the towns are similar, and if an idea works in one of the towns, it is likely to work in the others as well. In other words, similarity begets similarity. So long as communities remain diverse, succeeding in one town will not guarantee success in another. But if the two towns are identical, then success can be replicated across towns.

This diversity destruction might be interpreted as a good thing. After all, these are high-quality firms that enter. Yet, if there are multiple ways to organize firms, the chaining could lead to premature convergence. And waiting could yield a better equilibrium. Such logic underpins the French rebellion against McDonalds. Further, if idea generation depends upon the set of existing firms, then homogeneity could stifle long-term innovation. Finally, in that diversity appears correlated with system robustness, rampant homogeneity may lead to a brittle system.

CONCLUDING COMMENTS

The simple models described in this essay emphasize the importance of heterogeneity in social science models. We have seen how diversity in problem-solving approaches can help to solve hard problems, how diversity leads to robustness, and how, at least in one case, diversity can create instability. Computational models of the sort presented at this conference naturally admit diversity and its exploration.

REFERENCES

Axelrod, Robert (1997) *The Complexity of Cooperation*, Princeton University Press, Princeton.

Axelrod, Robert and Michael Cohen (1999) *Harnessing Complexity*, book manuscript, University of Michigan.

Dubos, René (1968) *So Human an Animal*, Scribners, New York.

Epstein, Joshua and Robert Axtell (1996) *Growing Artificial Societies: Social Science from the Bottom Up*, Brookings Institution.

Holland, John and John Miller (1991) "Artificial Agents in Economic Theory," American Economic Review Papers and Proceedings, *81*:365-370.

Hong, Lu and Scott E. Page (1998) "Problem Solving by Heterogeneous Agents," manuscript, University of Michigan.

Hong, Lu and Scott E. Page (1999) "Diversity and Optimality," manuscript, University of Michigan.

Johnson, Norman L. (1998) "Collective Problem Solving: Functionality Beyond the Individual," Los Alamos National Laboratory Working Paper LA-UR-98-2227.

Kirman, Alan (1997) "The Economy as an Interactive System," in *The Economy as a Complex Evolving System II*, W. Brian Arthur, Steven Durlauf, and David Lane, eds., Addison Wesley, Reading, Massachusetts, pp. 491-533.

McPhee, John, *Annals of the Former World*.

Page, Scott E. (2000) "On the Emergence of Coordination," forthcoming, *Journal of Economic Dynamics and Control*.

Schelling, Thomas (1978) *Micromotives and Macrobehaviour*, W.W. Norton, New York.

Tesfatsion, Leigh (1997) "How Economists Can Get A-Life," in *The Economy as a Complex Evolving System II*, W. Brian Arthur, Steven Durlauf, and David Lane, eds., Addison Wesley, Reading, Massachusetts, pp. 533-565.

THE EFFICIENCY OF AN ARTIFICIAL DOUBLE AUCTION STOCK MARKET WITH NEURAL LEARNING AGENTS — A SUMMARY*

J. YANG, Bank of Canada**

OVERVIEW

This paper investigates the convergence of a double auction market where two types of agents trade a risky asset that pays a stochastic dividend each period. The first type of agent, a *value* agent, forms expectations about the future return following a rational expectations formulation via an artificial neural network (ANN) and places orders based on those expectations. The second type of agent does not operate under a rational expectations rule, but trades on *momentum*. Market prices are set endogenously by trading among agents, with the efficiency of this artificial market measured by the convergence of the price to the rational expectations equilibrium (REE). Market dynamics under double auction converge to the REE in experiments with the ANN agents but not in experiments including momentum traders.

SIMULATION FRAMEWORK

The paper sets up a series of experiments, or simulations of a market, which are run for 1,000 periods, each consisting of 40 trading rounds. Four experiments, comprising differing mixes of traders, are conducted. The underlying behavior of the value traders is described according the standard REE framework. The simulation rules of the market are then based on this underlying behavior.

Standard REE

Agents trade a risky asset, paying a stochastic dividend d_t, and are assumed to behave according to a constant absolute risk aversion (CARA) utility function. They decide on their desired asset composition between the risky stock and a risk-free bond paying a constant interest rate r. Under the normality and negative exponential utility assumptions, the homogenous REE price can be solved as a linear function of the dividends:

$$P_t = fD_t + g \tag{1}$$

where $f = \dfrac{\rho}{1+r-\rho}$ and $g = \dfrac{1}{r}(1+f)\left[\bar{d} - \lambda(1+f)\sigma_\xi^2\left(\dfrac{\bar{Q}}{N}\right)\right]$

 This is a summary of a paper that will appear in *Evolutionary Computation in Economics and Finance*, S.-H. Chen (ed.). Studies in Fuzziness and Soft Computing, J. Kacprzyk (ed.), Springer-Verlag (Physica-Verlag), Heidelberg (forthcoming).

** *Corresponding author address*: Jing Yang, Bank of Canada, 245 Sparks Street, Ottawa, ON K1A 0G9, e-mail: jyang@bank-banque-canada.ca.

and D_t is the stochastically determined dividend, which follows AR(1) process; λ is the constant coefficient of risk aversion; $\overline{Q}$ is the units of stock; and N is the number of traders. Equation 1 is the benchmark used to compare the market simulations conducted in the paper.

One formulation of the optimal forecast in the full-revealing REE is given by

$$E_t\left(P_{t+1} + D_{t+1}\right) = \rho(1+f)D_t + \left[(1+f)\overline{d} + g\right] \tag{2}$$

where E_t is the expectation of the trader. The parameters in Equation 2 are the basis for the ANN learning the agents undergo.

Learning and Trading Strategies for Value Traders

We assume each value trader possesses an ANN(1-3-1) model. They have similar structure, but there are differences in the initial values of the parameters. The parameters start with initial values drawn from a uniform distribution whose range is set at [-1, 1]. The lagged dividend is the input to the ANN input layer. At the end of each trading period, agents update their estimated conditional variances according to an exponentially weighted average of squared forecasting error. This variance will be used in the traders' demand function, the reservation price function (Equation 3), and the spread function S.

Trading strategies for the value trader are based on the trader's reservation price, i.e., the price at which trader j has no incentive to buy or sell, given by:

$$P^{R,j} = \frac{\hat{E}^j{}_t\left(P_{t+1} + D_{t+1}\right) - \lambda\hat{\sigma}^2_{j,P_{t+1}+D_{t+1}} Q^j{}_t}{1+r} \tag{3}$$

Note that the reservation price is based on the forecast the trader has constructed from learning the parameters in Equation 2.

Each of the value traders operates according to the following rules.

1. Post a market buy order, if $a < P^R$.
2. Post a market sell order, if $b > P^R$.
3. Do nothing, if a (or b) = P^R.
4. Otherwise, post a limit order to bid or ask ΔQ shares at the price, P^R - SB.

In rule 4, B is an indicator variable where B = +1 for an ask order and B = -1 for a bid order, and S is a spread between the reservation price and the price quoted.

Trading Strategies for Momentum Traders

Momentum traders are chartists who believe that future price movements can be determined by examining patterns in past price movements as represented by various moving averages (MAs). The moving-average trading rule states that when the short-term (usually 1- to 5-day) moving average is greater than the long-term moving average (usually more than 50 days), a rising market is indicated. Thus, this trading rule would generate a buy signal. Based on such market trends, the momentum trader decides to enter or exit the market. The momentum traders are divided into two groups according to their choice of trading rules.

The first group of momentum traders compares the current market price P_t with MA(5). That is,

If $P_t > MA(5)$, they buy shares.
If $P_t = MA(5)$, they hold their current position.
If $P_t < MA(5)$, they sell shares.

The second group of momentum traders identifies a trading opportunity by comparing MA(5) with MA(10). Specifically,

If $MA(5) > MA(10)$, they buy shares.
If $MA(5) = MA(10)$, they hold their current position.
If $MA(5) < MA(10)$, they sell shares.

Simulations

Experiment 1: 10 value (ANN) traders with a double auction (experienced traders)

The purpose of this experiment is to test the convergence of prices to the REE with a double auction trading institution. Value traders in this experiment are trained on the pre-sample price and dividend data for 40 periods before they enter the market.

Experiment 2: 10 value (ANN) traders with a double auction (inexperienced traders)

The value traders in this experiment are not trained by the pre-sample data before they enter the market. This experiment is designed to check the effect of traders' experience on the convergence property.

Experiment 3: 10 value traders, 10 momentum traders with a double auction

The purpose of this experiment is to check the sensitivity of the convergence on this double auction market to the deviation from rationality. The presence of momentum traders should add some noise to the market. In this experiment, it would be interesting to observe whether the value traders can maintain rational expectation equilibrium given the existence of some erroneous signals caused by the presence of irrational momentum traders.

RESULTS

In general, ANN adaptive traders are able to learn rational expectation collectively. Convergence to the REE occurs in experiment 1, the identical trading strategy case. Such convergence occurs more slowly in experiment 2, where adaptive agents are not trained before they enter the market. In experiment 3, with the presence of momentum traders, the convergence is unattainable (Figures 1-6).

Under the null hypothesis of linear homogeneous REE, the price and dividend are a linear function of lag dividend. In the homogeneous REE, this estimated residual series should be independent and identically distributed, N(0, 1.89). The regression was run for experiments 1, 2, and 3, and the estimated residual series statistics are shown in Table 1.

The first column shows the variance of the residual from all three experiments. The parameters are chosen to give the theoretical value of 1.89. All three cases show a higher variability but to different degrees. The first two experiments are very close, but are consistently higher in the second experiment since the experienced traders converge more quickly to the REE. The next column gives the excess kurtosis, which should be zero under normal distribution. In the three experiments, the third one shows a significant amount of excess kurtosis. The third column presents the autocorrelation in the residual. In all three experiments, the autocorrelations are small; in the first two, the autocorrelation is close to zero, which is comparable to the low autocorrelations observed on the real market. The last column presents the average trading volume in each trading round in the three experiments. The last experiment shows very high trading activity.

The results from experiment 3 identify richer and more complex market dynamics. Prices diverge substantially from theoretical (fundamental) prices. The differences between the two price series provide systematic evidence of temporary price bubbles and crashes. This appearance

TABLE 1 Basic Summary Statistics for the Residual and Trading Volumes for Each of the Three Experiments [1]

Experiment	**Variance**	**Kurtosis**[2]	**ρ(1)**	**Trading Volume**[3]
Experiment 1	1.91 (0.12)	0.33 (0.31)	0.036 (0.007)	0.25 (0.03)
Experiment 2	2.93 (0.65)	0.71 (0.56)	0.078 (0.012)	0.96 (0.05)
Experiment 3	4.23 (0.97)	4.76 (1.01)	0.351 (0.098)	2.42 (0.11)

[1] The experiments are run for 1000 periods and 25 times for each. The statistics shown in this table are the average over 25 runs. Numbers in parenthesis are standard errors estimated using the 25 runs.

[2] The kurtosis reported is excess kurtosis.

[3] The trading volume is the average value for each period over all trading periods.

of bubbles and crashes suggests that momentum traders have affected the market. In this experiment, a large proportion of the variance in the price and dividend is not explained by the homogeneous rational expectation hypothesis. To see how the momentum traders affect the simulated market, the two other technical indicators used by momentum traders are added to the simple linear regression of price and dividend on the lagged dividend. The first indicator variable shows whether price is above or below a 5-period moving average, and the second indicator variable represents whether the 5-period moving average is above 10-period moving average.

Table 2 shows the regression for experiment 3. The numbers presented in the table are the estimated parameters and R^2 values. The standard errors are given in parentheses. The results show that the two technical indicators give significant extra predictability. The parameters are small but statistically significant. The R^2 values in the last column confirm the fact that adding technical trading indicators explains a higher proportion of the variance in price and dividend.

It is obvious that in the presence of momentum traders, ANN traders cannot drive the market price to the rational expectation equilibrium. They cannot pick up the part of the variation that can only be explained by the technical indicators, since the ANN formulation does not "know" of the presence of such traders.

SUMMARY AND FUTURE DIRECTIONS

Our artificial double auction market is capable of generating behavior close to REE under certain circumstances. The speed of convergence varies across traders' experience. When irrational traders emerge, the market is driven by more noisy factors than just fundamental trading. The convergence is unattainable in this case. The value trader cannot learn the chartists' strategies, and they may not even realize the existence of the chartists. It is still not clear whether the observed market inefficiencies should be attributed to the learning effects or to other differences in trading institution design.

When the intrinsic value of the risky asset becomes endogenous, the convergence is sensitive to the deviation from rationality, and the minimal rationality in this case is not sufficient. Some interesting hypotheses that can be tested on this market are informational efficiency, the role of market transparency, and market microstructure. Another interesting extension to this initial work is related to the choice of market mechanism. In this paper, we used a double auction. It would be appealing to simulate different types of auctions to investigate the effects of auction mechanism on the market efficiency.

TABLE 2 Regression of Price and Dividend on Fundamental and Technical Indicators, Experiment 3

Regression	**Constant**	**D**	$\mathbf{I_{MA(5)}}$	$\mathbf{I_{MA(10)}}$	$\mathbf{R^2}$
Fundamentals only	4.65 (0.08)[1]	1.85 (0.54)	NA	NA	0.43 (0.05)
With moving-average indicators	4.21 (0.13)	2.02 (0.87)	0.054 (0.023)	0.086 (0.034)	0.57 (0.08)

[1] The numbers in parentheses are the standard errors estimated in the 25 runs.

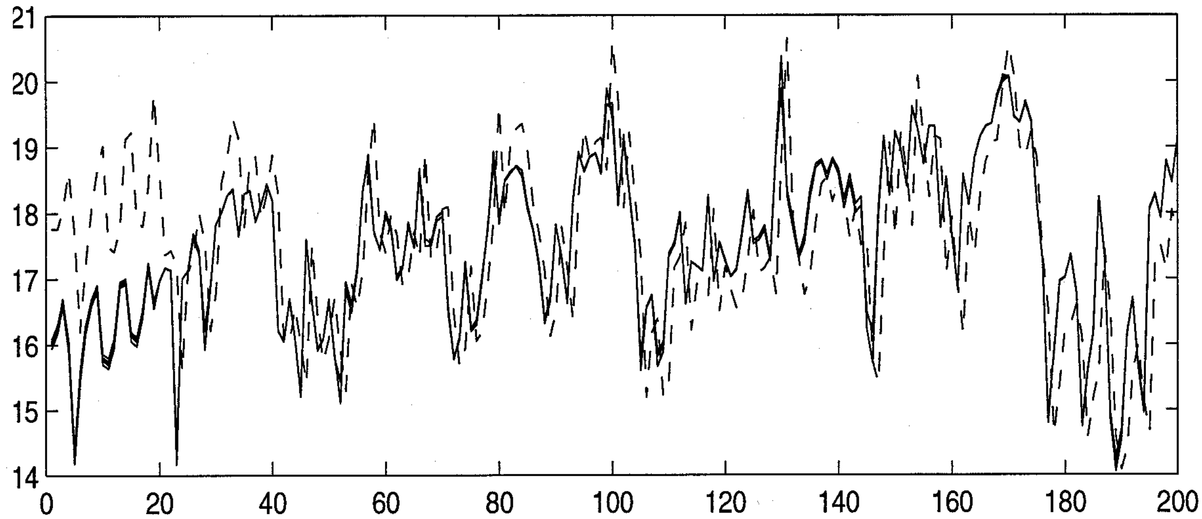


FIGURE 1 Experiment 0: Rational Expectation vs. ANN Expectation.

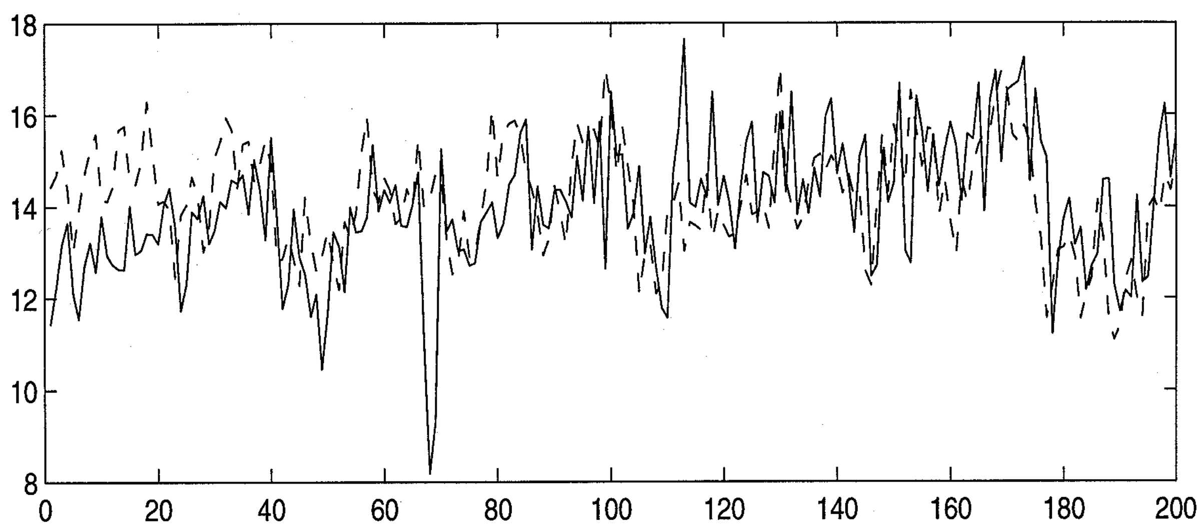


FIGURE 2 Experiment 0: Market Price vs. REE Price.

FIGURE 1. Solid Line = ANN Expectations
Dotted Line = Rational Expectations Forecasting
Period = 1:200 (First 200 periods)

FIGURE 2. Solid Line = Market Price with Walrasian Auction and ANN Expectation
Dotted Line = Rational Expectation Equilibrium Price
Period = 1:200 (First 200 periods)

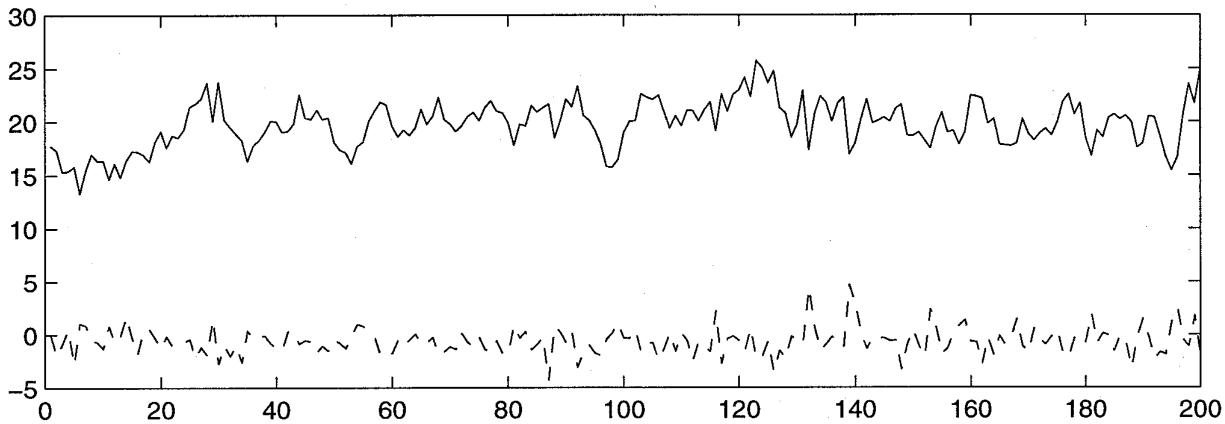


FIGURE 3 Experiment 1: Market Price vs. Price Deviation.

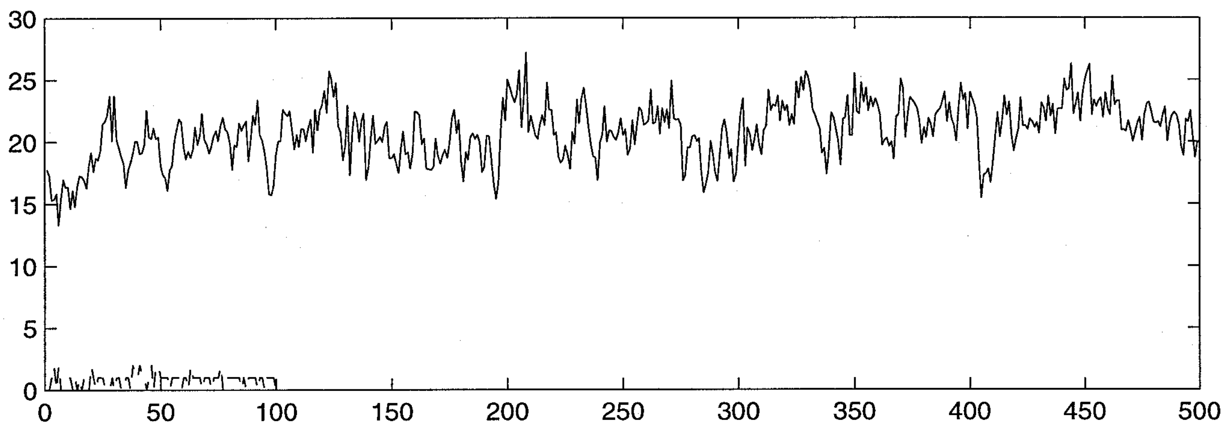


FIGURE 4 Experiment 1: Market Price vs. Trading Volume.

FIGURE 3. Solid Line = Market Price
Dotted Line = Price Deviation
Period = 1:200 (First 200 periods)

FIGURE 4. Solid Line = Market Price in Experiment 1
Dotted Line = Average Trading Volume in Each Trading Period
Period = 1:500 (First 500 periods)

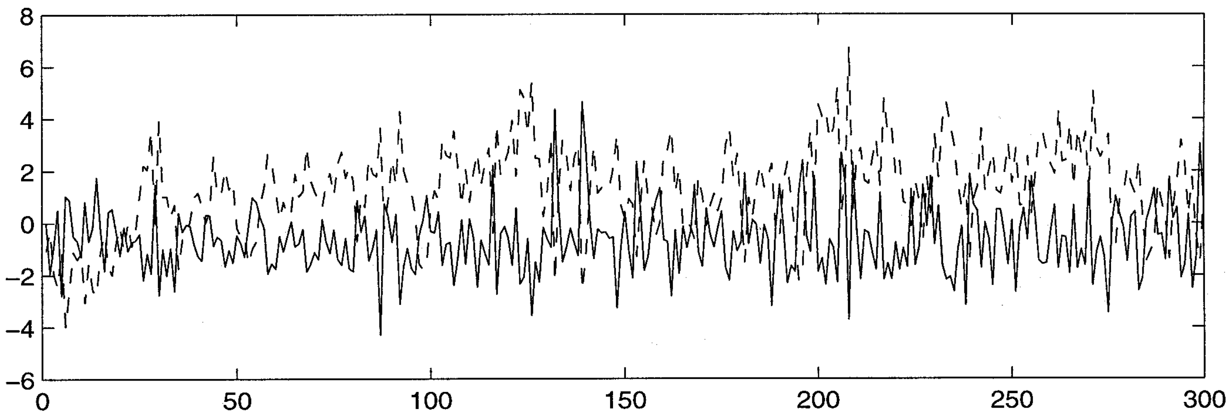


FIGURE 5 Price Deviation in Experiment 1 and Experiment 2.

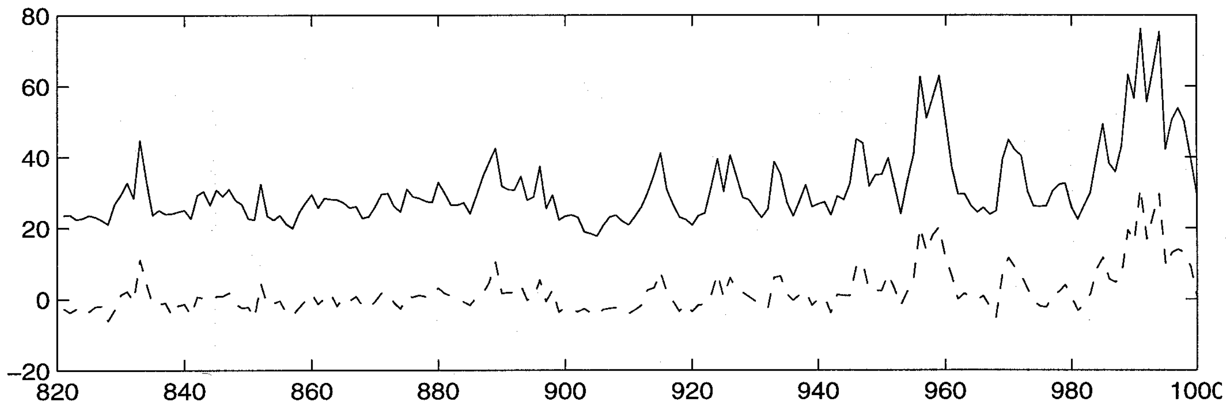


FIGURE 6 Market Price and Price Deviation in Experiment 3.

FIGURE 5. Solid Line = Price Deviation in Experiment 1
Dotted Line = Price Deviation in Experiment 2
Period = 1:300 (First 300 periods)

FIGURE 6. Solid Line = Market Price in Experiment 3
Dotted Line = Price Deviation in Experiment 3
Period = 820:1000 (Last 180 periods)

MODELING ORGANIZATIONS USING AGENT-BASED SIMULATIONS

M.V. NAGENDRA PRASAD, Andersen Consulting LLP*
D.A. CHARTIER, Andersen Consulting LLP

ABSTRACT

TalentSim is a prototype that embodies a modeling approach representing the middle ground between the analog modeling based on differential equations in system dynamics and the discrete rule-based modeling methods common in behavior-oriented modeling in the agent-based simulation community. We discuss our modeling approach and then illustrate our claims using examples from the TalentSim prototype that we built. The prototype was designed to help users improve their understanding of the dynamics of workforce and organizational change — specifically, how changing workforce styles can be made to match a new business strategy. A talent management practitioner can use TalentSim to visualize the dynamics and the impact of his or her decisions on the workforce transformation process. We end the paper with a discussion on the insights we developed in the context of knowledge acquisition for building models for our middle ground approach. We believe that these insights are more generally applicable to agent-based modeling.

INTRODUCTION

Simulation involves capturing, representing, and modeling some aspect of reality. Simulations are routinely used to gain insights into the behavior of engineering and physical systems. Of equal interest, but less often done is using simulations to gain insights into the behavior of social systems involving humans, man-made institutions and their interactions other parts of a system. There are three major approaches to social systems simulations:

- **Game theoretic simulations** use techniques from game theory to cast a problem domain as a set of agents strategically reasoning about one another based on the information available to them and the payoffs from their actions. Game theoretic techniques are based on a solid foundation of game-theoretic analysis [Bierman98] but are limited to a specific set of issues involving strategic thinking among various parties about pay-offs from a set of actions. Not all social systems simulations are concerned with posing questions in this format, but for those that are, game theoretic analysis can provide interesting insights. In this paper we will not be talking further about game-theoretic techniques. Interested readers are referred to [Bierman98].

* *Corresponding author address*: M.V. Nagendra Prasad, Center for Strategic Technology Research, Andersen Consulting LLP, 3773 Willow Road, Northbrook, IL 60062; e-mail: nagendra@cstar.ac.com.

- **System dynamics** models the world as feedback structures that generate complex dynamics. A model is represented in terms of two types of primitives — stocks and flows — and the relationships among them. The dynamics of the system are generated by capturing the relationships using differential equations and iteratively solving for them [Forrester61].

- **Agent-based modeling** requires that the domain be modeled as a set of behaviors. It is an "interaction-oriented" modeling paradigm, where the focus of the knowledge acquisition effort concentrates on defining the behaviors of the entities and the interactions among them [Epstein96].

AGENT-BASED VERSUS SYSTEM DYNAMICS MODELS

In agent-based behavior-oriented simulations, agents are entities (e.g., employees and company) that are endowed with certain behaviors, and the interactions among these entities executing their behaviors give rise to complex dynamics. Behavior-oriented simulations have certain advantages over system-dynamics-based simulations (system dynamics is perhaps the most popular and well-researched technique for social system dynamics so far).

System dynamics modeling requires that the domain be represented as levels and flows and relies on coupled differential equations that relate them to generate the dynamics. As noted previously, agent-based modeling requires that the domain be modeled as a set of behaviors. The latter is a more natural representation for many of the business problems because they are dominated by discrete decision making and symbolic reasoning tied to sense-and-respond behaviors (e.g., if most of my colleagues aren't planning to come to work today, I may take a day off). Converting such behaviors into levels and flows is difficult. Levels and flows cannot represent actions. They can only capture their aggregated effects and probabilities of occurrence. Representations that are behavior-oriented are also more natural to explain, understand and manipulate during the course of user experiments with the system.

In system dynamics, it is difficult to relate global parameters to local parameters. For example, how does the organizational culture affect the behavior of an individual? Parameters in system dynamics have to be modeled at similar levels of aggregation in order to make the modeling exercise practical and viable. In behavior-based modeling, micro and macro parameters routinely interact, and it is possible to model and study the effects of global parameters on individual entities.

Agent-based modeling techniques can handle more variety and heterogeneity in behaviors and domain descriptions. They are very amenable to data-driven modeling without the need for gross aggregations and averaging. For example, it is possible to feed the profiles, interests, and behaviors of music buffs obtained by extensive data gathering into an artificial agent-based model world to predict the probability of a particular kind of soon-to-be-released album becoming a hit. Winslow Farrell is doing this very thing [Farrell98].

However, system dynamics techniques have been extensively researched, and they are tools available to facilitate the model-building process. There is also a vast body of work on knowledge acquisition processes involved in building system dynamics models. Agent-based techniques are newer, and there is less collective experience using them. There are also not any well-tested off-the-shelf tools. In this work, we built our own tool for agent-based simulations.

We borrowed extensively from the vast body of insights that the system dynamics community developed over the last 40 years and adapted them to the agent-based modeling problems where appropriate. The models in TalentSim represent a middle ground between the analog modeling based on differential equations in system dynamics and the discrete rule-based modeling methods common in behavior-oriented modeling in the agent-based simulation community. At the individual agent-level, the modeling is more akin to system dynamics modeling. However, at the interaction level, the agents behave more like the behavior-oriented simulations of agent-based models. We represent the middle ground approach in yet another dimension too. Agent-based models are bottom-up, whereas system dynamics models are largely top-down. Our models combine these types. We have elements of bottom-up modeling in generating the interactions among agents, whereas we have elements of top-down modeling in capturing the right level of representation and abstraction for the individual entities. However, the tricky issue here is to be able to move from the analog world of differential equations into discrete world of behavior rules and vice versa. We don't have any systematic approaches to this problem. We use ad hoc thresholding techniques. We have more recently started exploring techniques from fuzzy set theory [Dubois80]. In fuzzy set theory, designers of fuzzy systems routinely convert relationships among soft variables with continuous membership functions into discrete decision rules. Below we briefly introduce our model and then illustrate our claims with examples from the prototype simulation that we built.

THE TalentSim SIMULATOR

According to Treacy and Wiersema [Treacy97] and Gubman [Gubman98], a company can have one of three types of strategic orientations: operationally efficient companies (OE) that derive their competitive advantage by more efficient or better execution, product companies that stay ahead of the competition by constant innovation in products, or customer-intimate companies (CI) that provide a more relationship-oriented service to the customers. We developed the model in the context of a major health management organization that is trying to change its strategic-orientation and the company culture from a traditionally OE environment to a more CI environment. Note that this case study was only used as a very rough guiding mechanism, and we have not yet done a thorough validation of the model in the context of the specific case study. The purpose of this prototype is to demonstrate the effectiveness of agent-based modeling methodology and convince other parties to become members of our expedition into this rich and promising direction.

Regardless of what the new culture looks like, changes in the strategic direction of an organization result in employees needing to modify the way they work. Additionally, any type of change has the potential to impair an employee's performance. It is likely that a reduction in performance will occur (as a result of a culture change) if, among other things, (1) the employee does not perceive the new culture to match his/her beliefs or (2) the company does not design processes aligned to the new culture. These processes support employees as they begin to exhibit behaviors that are aligned with the new culture.

There are many different strategies an organization can follow when implementing change. Whatever the strategy, it would be invaluable to organizations if they could model and simulate the dynamics of change in a safe, time-compressed environment. That is what TalentSim can do for our clients. A client team can create scenarios and quickly see how their decisions will affect the workforce without any of the real-world consequences, such as lower performance, turnover, low organizational commitment, etc.

We will now discuss the components of the model underlying TalentSim. The current paper will only illustrate parts of the model. Details can be found elsewhere [Nagendra Prasad99].

We use a two-dimensional vector space representation — S_A versus S_B — as the core for our exercise. S_A and S_B represent suitability for type A and type B environments. Work comes in as an entity chosen from a distribution (at this point it has two values, one for S_A and the other for S_B, both drawn from two independent distributions). In a CI environment, the work event has a distribution with S_B biased to be larger than S_A (say, drawn from a Gaussian-like distribution with the mean of S_B being larger than the mean of S_A) and in the case of OE, vice versa. Values range from 0 to 1. When we go beyond the prototype, our research will, of course, lead us to a "rich" multi-dimensional vector with important relevant attributes. The two values of the work event can be looked at as a vector in 2-D space. This vector indicates to us how much of skills of type A or type B are needed to perform effectively on the work event.

Different employees have different personal attributes profiles. They are represented in the same vector space as the work event. As you will soon see, this will address the question of "fit of an employee to a work event" and a number of other related questions. An employee is represented by a 2-D vector. An employee attribute vector is represented by $[S_A, S_B]$. This indicates to us the skill level of an employee in type A and type B jobs.

So what is the fit of an employee to a work event? It is the projection of the employee profile vector E_V on the work event vector W_V . We will represent this as $E_V \mapsto W_V$.

TalentSim models a number of domain aspects, some of which are as follows:

Performance: Performance is the basic premise of the model. The model focuses on the quality of performance and looks at the factors that influence it. In the model, performance is influenced by employee fit, motivation, relevant experience and workload.

Employee fit: This aspect of the domain is determined by the match between the employee's profile (skills, abilities, and competencies) and the job profile (skills, abilities, and competencies required for the job). An employee who has a good fit with his/her job will perform at a higher performance level.

Motivation, experience and workload of an employee.

Perception of culture (CCp): Individuals develop their perception of an organization's culture based on their own background. When an organization changes its culture, the employees' perception of the new culture and its relevance to their work and how they perceive their "fit" to their work affects their motivation.

Communication of vision and enablements: An organization will facilitate an individual's alignment to the new culture through its communication and "enablements." Enablements are processes that are aligned to the new vision of the organization. They reinforce behaviors that are aligned with the new culture. When a message (in this case, the vision) is communicated by an organization to an individual, that person will seek to determine the message's relevance. Relevance is determined by how the individual perceives the "fit" between the vision and his/her job, role or task. An individual's organizational commitment will be affected by this perceived relevance.

Nature of work: The type of work and the employees' perception of the fit of the work to their ability and skill profiles plays an important role in the motivation of the employees. If the employees are consistently saddled with work they are not suited for or trained to perform well, their motivation is likely going to decline over time. Support for this exists in a number of sources, including [Gubman98] and [EIU99].

Peer pressure: Belonging to a good team makes people want to work together effectively. A highly cohesive team has an increased commitment level toward a project or organization. In the domain we are modeling, peer pressure has an effect on an employee to change and adjust his/her profile. This process of adjustment depends on the flexibility of an employee. The more flexible an employee is, more likely he/she will be affected by peer pressure to conform.

Susceptibility: Susceptibility refers to an employee's openness to a job offer from another. According to [EIU99], individuals who have job satisfaction are still susceptible to job offers if they have low organizational commitment. In our modeling effort, susceptibility is affected by organizational commitment and turnover in the acquaintance network (leading to destruction of trust networks).

The assumptions of the model are grounded in theories of human capital management and real-world experiences. The model is a very early prototype and needs further effort to enhance its fidelity. However, it demonstrates the value of such an exercise and provides a solid base to build upon.

Agent-based modeling is an "interaction-oriented" modeling paradigm, where the focus of the knowledge acquisition effort concentrates on defining the behaviors of the entities and the interactions among them. For our modeling effort, we define three types of interactions. Notice how the interactions are defined akin to the more behavior-oriented agent-based modeling methods.

Employee-to-Work Events

An organization attracts "work events" that are defined by duration and certain requirements. Examples of work events are projects, customer phone calls, transactions, and so on. Work events can be defined along certain dimensions of requirements — like personal interaction involved, listening, problem-solving, process control, teamwork, financial understanding, attention to detail, etc. Each work event has values along these dimensions. The relative values determine the type of skills needed for its execution. In our initial prototype we simplified these dimensions and started with just two features S_A and S_B.

An employee has a profile comprising some of the same dimensions as work (or at least functionally mapped abstractions thereof — for example, creativity could be mapped to listening, relationship-building and rapid problem-solving into it). The level of performance regarding a given work event is determined by the employee's profile and the fit of the work event to that profile. Performance is also determined by workload, experience, company culture, motivation, and knowledge management efforts. It is a function of all these factors.

Employee-to-Employee and Employee-to-Company Relationships

An employee's motivation is reduced by mismatches between his/her profile and the company culture. The employee's profile is altered by his/her perception of the "people culture" in the organization. A person's flexibility also a plays a role in the amount altered. Consistently large mismatches between the profiles of employees a person meets and the work events a person deals with lead to lower motivation and higher turnover.

Externalities also lead to turnover. Externalities represent "other greener pastures," such as job offers from the competition or opportunities that are attractive elsewhere. An externality becomes more attractive based on a person's experience, and a person does not leave the company until he/she has been with it for a certain duration. Training has the effect of increasing the knowledge of work processes and skill in performing them. Experience also plays a role in the effectiveness of training. However, the effects of experience are different in different kinds of environments. For OE, experience is not as important a player as in CI environments. Communication and strategic enablement affect the profiles of employees and their effectiveness in particular kinds of jobs (OE versus CI jobs). Higher performance leads to higher satisfaction. Satisfaction has an effect on the turnover.

Entities in the Simulation

Environment

In this paper, we use a "tick" to represent a day. A tick is a unit of time in the simulator and can be mapped to the model in a domain-specific way. S_A is used interchangeably with OE, and S_B is used interchangeably with CI.

The environment selects **Work Event** X at random from a distribution of S_A – S_B, hours, and the number of people needed to execute it.

The organization attracts work events based on the difference in performance of the organization and that of the competitor. As the company performance gets better, it attracts an increasing fraction of the work available in the marketplace.

People with different skills approach the company to be hired. There is a rate at which **potential employees approach the company**.

Company

Individuals are hired according to a **hiring policy**. In our prototype, we used a **replacement-level policy**. When ever there is attrition, hiring goes on until the replacement level is reached. A company hires a person if he is within the threshold requirement. This determination is affected by the pressure on hiring (which varies with the gap between the required versus actual number of employees) and the company's effectiveness at finding people with appropriate profiles.

Individual

Performance of an employee on the work assigned to him/her is given by the following form (we will explain each of the components in detail over the course of the next few pages):

Performance = (k_4 * effect of company culture & enablement + k_5
* effect of employee skill set) * motivation_factor
* workload_factor * experience_factor)

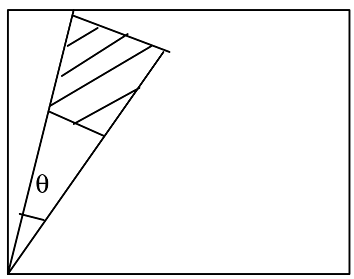
where $k_4 + k_5 = 1.0$

Effect of employee skill set = $(E_V \mapsto W_V)/|W_V|$ when $E_V \bullet W_V < |W|^2$
Else $|W_V|/|W_V| = 1.0$

Notice that this looks akin to how system-dynamics-based methods may model the effect of performance. Below, we look at the specifics of some of the elements in the model.

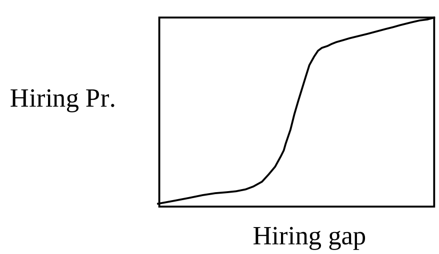
Details for Some Aspects of the Models

How does the hiring actually happen in the model? The company has an "ideal" range.

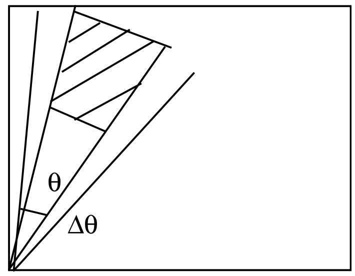


This means that the company is looking for people within the shaded range of skills. Let us say θ is the angle for the vectors bounding the shaded range and m and n the length of the vectors over the shaded regions. Note that this range should also match up with the work event distribution at least approximately. Otherwise, the hiring policies will be misaligned with what is needed.

We want to hire people only in the shaded region, but achieving this goal gets constrained by the amount of hiring that needs to be done and the effectiveness of the company's hiring practices. Hiring pressure is a nonlinear function with the number of people that need to be recruited.



The effectiveness of the HR processes determines how many of the nondesired skill sets get hired, or $\Delta\theta = k_2$ * hiring_pr.



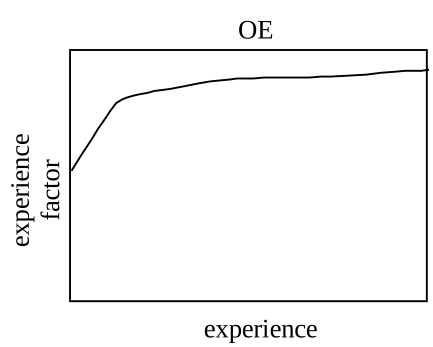
So, anyone with a profile falling into the $\theta+\Delta\theta$ range above and of the right strength (length between m and n) gets into the company.

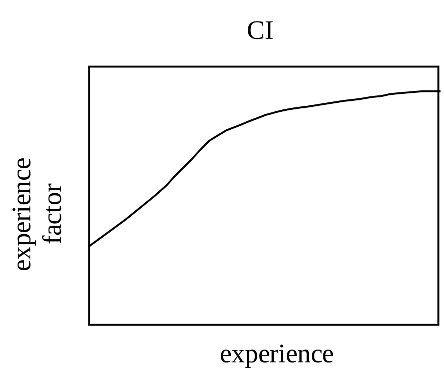
Let us see how **experience** affects performance. Employees build experience with every tick. Let us also look at the notion of experience profile from which experience is calculated. It is again a 2-D vector along [S_A, S_B]. When an employee does a particular job, the job just gets added to the experience profile, *scaled by performance*. We can just make this the exponentially weighted sum along each of the two dimensions. Experience with respect to a new job W_V is calculated as the projection: Experience vector $\mapsto W_V$. In addition to an experience profile, an employee also has "experience time" — the time he spends building it:

$$\text{Total\_experience}_v = \text{performance (on this task)} * W_V + \gamma * \text{total\_experience}_v \text{ (before this work event)},$$

where γ is a constant close to but less than 1.0 (something like .999). It is called the discount factor.

The **experience factor** is calculated from experience. It is a nonlinear function of experience, as shown below. The experience factor graphs for OE and CI are different. In OE, zero experience still gives a large value for the experience factor. This means that even without any experience the person can deliver quite some performance. And the graph rises a little. So experience doesn't make too much difference and it rises quickly. In the CI graph, experience is a great teacher and it takes time to learn it all.





The **Organizational Commitment** variable ranges from [0 1.0] and is affected as follows:

- Δ Organizational Commitment$_X$ = k_8 * $|({}^{A}CC_{PX} - {}^{A}E_X)|$ * Organizational Commitment$_X$
  when the mismatch is greater than a threshold, i.e., $({}^{A}CC_{PX} - {}^{A}E_X) > 0.5$,
  where k_8 is a small negative fraction,
  ${}^{A}E_X$ is the component of employee skill profile along A-dimension, and
  ${}^{A}CC_{PX}$ is the A-component of X's perception of organizational culture.

  We do a component-wise check to get the net loss of Organizational Commitment. This check can be done every 30 ticks once.

  k_8 = - 0.03 or it takes about 36 months (about 1/0.03) for a person to lose all commitment when there is complete mismatch.

- Δ Organizational Commitment$_X$ = k_9 * $|({}^{A}W_V - {}^{A}E_X)|$ * Organizational Commitment$_X$
  when the mismatch is greater than a threshold, i.e., $({}^{A}W_V - {}^{A}E_X) > 0.5$,
  where k_9 is a small negative fraction.

  k_9 = -0.01 or it takes about 20 changes of 30 ticks in length — about 600 ticks or about 2 years — for a person to lose all Organizational Commitment in the case of extreme mismatch between his profile and the work requirements.

- Δ Organizational Commitment$_X$ = k_{10} * performance * (1 - Organizational Commitment$_X$),
  where k_{10} is a small positive fraction and the performance of a work event achieved performance greater than a threshold, performance > 0.6.

  k_{10} = 0.01 for reasons similar to the above.

Motivation is affected by Organizational Commitment (and perhaps a host of other factors that have not been modeled here). For the present, Motivation is mathematically the same as Organizational Commitment:

Motivation = k_{11} * Organizational_Commitment, where k_{11} = 1.0.

Above, we discussed just a few aspects of the model — just enough to illustrate some of our claims. The model has been "qualitatively" verified by the subject matter experts. However, we have not yet done more specific data gathering in the context of the health maintenance organization that was used for the modeling purpose. Efforts are afoot to get this done.

LESSONS IN KNOWLEDGE ACQUISITION FOR BOTTOM-UP SIMULATION MODELS

This section will only briefly discuss the knowledge acquisition methods that we used for behavior-oriented modeling. We are developing a separate document detailing our lessons and insights.

System-dynamics techniques [Forrester61, Richardson81] have been extensively researched, and there is a vast body of work on knowledge acquisition processes involved in building system dynamics models. Agent-based techniques are newer, and there is less collective experience using these techniques. We found it useful to borrow extensively from the vast body of insights that the system dynamics community developed over the last 35 years and adapt them to the agent-based modeling problems where appropriate. While it is true that agent-based systems are not restricted to designing differential equations for levels and flows, the heuristics developed for designing functional forms in system dynamics could easily be applicable in the agent-based systems. In fact, those readers who are aware of system-dynamics modeling can easily see the spirit of these techniques running freely through the above modeling exercise.

Perhaps the single most important thing to do as a first step is to identify the purpose of the modeling exercise. A crisply defined purpose identifies the boundaries of the model. What belongs to the model and what is outside the model are very intimately tied to the purpose of the simulation. For example, early on in our modeling exercise, we wanted to model "managing human performance in an organization." A loosely and broadly defined purpose led to flailing about, until we revisited the purpose and established it as "dynamics of workforce transformation from an operationally oriented organization to a customer-intimate organization." Moreover, we identified a case — a large health management organization that is trying to move its strategic direction from an OE company to CI company — as our baseline. In summary,

- Explicitly define a purpose. It helps prevent the modelers "from having to think about everything in order to think about something" [Richardson81]. Some of the questions that we used include "What do you want to model and why?" and "Suppose we have a great modeling exercise and we have the simulator ready. How would you use it? What do you want to show with it?"

- Always have a real-life case in mind as the baseline. This plays an important role in defining the purpose and also the actual modeling exercise.

- Focus should be on the problem domain rather than on a system [Richardson81].

We found it useful to let the early sessions be more in the "free talking" mode. This was helpful along a number of dimensions:

- It familiarizes us to the client's domain. It is also recommended that where possible, the modelers should back these sessions up with some of their reading about the domain. We asked for one or two pieces of literature to introduce us to the main thoughts in the workforce transformation domain. We were recommended the book by Gubman [Gubman98].

- These sessions also establish the client's vocabulary. It is very important to speak the language of the domain experts to be able to get at their thought processes.

- These sessions also help establish other intangibles like our willingness to listen the domain experts. Trying too early to establish the rigor needed for modeling right from the beginning does not seem to go down well in terms of developing a rapport with the subject matter experts. Once this trust has been established, if the modelers can clearly and rationally explain in business terms their approach to modeling the domain, the domain experts will concede the mathematics to the modeler.

Behavior-oriented modeling paradigm is close to our activity-oriented view of the real world. This should make it relatively easy to identify the major actors or agents in the system and their behaviors. However, in a manner akin to object-oriented design methods, identifying the noun and verb phrases and collaborations (The Wirfs-Brock, Wilkerson, and Wiener Technique) can aid this process. The documentation of the "free talking" sessions usually leads to an informal user specification document. This document can guide the modelers towards identifying the agents or social entities in the system. Looking for the major noun phrases allows modelers to home in on the agents. Verb phrases indicate behaviors. Agents are assigned behaviors and attributes and at all times these assignments are made to correlate to the real world. It is important to note that no modeling effort proceeds inexorably forward to completion. It is an iterative process that needs multiple cycles of conceptualize, build, run, explain discrepancy, and refine. Our choice of defining the system components can be guided by such questions as

- What are the different types of entities that play relevant roles in the system?
- What are the behaviors exhibited by these entities and attributes relevant to them?
- What are the interactions between these entities and their behaviors?
- What is the time horizon over which the problem plays out? This question is important because it determines the granularity at which we model different attributes, behaviors, and their interactions.
- What are the reference modes? The graphs over time for some of the important variables can help focus the modeling exercise and help in the later stages of validation [Richardson81].

System conceptualization is driven by identifying the key entities, attributes, behaviors, and interactions in the domain.

Attention to terminology is very important because the names one chooses for the entities and behaviors biases the view of the SMEs and the modelers, leading to subtle impacts on the evolution of the model. It is also important that all parties involved understand these names and their definitions. Being lax in the names can be very distracting and can create disjunction in the team, dragging and diverting the modeling process repeatedly.

Since simulators serve as laboratories where users can experiment and gain an understanding of complex systems, flexible user interface design is a very important aspect. Good interfaces for simulators go beyond eye candy. They are an implicit requirement of the purpose of the simulators, i.e., to facilitate exploration. Visualization of the dynamics of a complex system is one of the primary value adds of a simulator.

It is important to have a up-front buy-in from the parties for whom the simulator is being designed. In the best of all possible worlds, all the users of the simulator should be represented during the process of modeling. The process is as important as (or may be even more important than) the end product in the form of an explicit model. The process can lead to insights that are valuable. It also imposes a rigor and focus that many of the SMEs never had to deal with. Consequently, it brings a whole new facet to their domains. However, it is not always (in fact most of the time) possible to represent all the users at every step of the process. In our work, we had to make do with check-pointing every so often — whenever we made some progress.

Another important but subtle factor comes into play when there is an up-front buy-in for the model. One of the primary benefits of a simulator is in the insights it triggers in the users. Benefits derived from such an exercise are only maximized when both the parties involved — the modelers and the users — are committed to generating them. Early buy-in motivates the users and focuses them on working towards generating insights rather than distracting them with questions about validity of the models.

It is not advisable to spend enormous resources on data collection for all the variables and their effects even before we build the models. We can initially build a simulator with approximate functional forms for the variables and their effects. A subsequent sensitivity analysis reveals the most important effects. Depending on the availability of resources, one can invest in increasing the fidelity of these effects. Another subtle point to note is that a simulator needs to just have sufficient fidelity to generate good insights. Getting sucked into a phase of enormous and accurate data-gathering may not be of much value. As long as the patterns of behavior are captured right, the generated dynamics can serve to instigate useful insights in the user.

REFERENCES

[Bierman98] H. Scott Bierman and Luis Fernandez, *Game Theory with Economic Applications*, Reading, Massachusetts: Addison-Wesley, 1998.

[Dubois80] Didier Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, New York: Academic Press, 1980.

[EIU98] Economist Intelligence Unit, *Knowledge Workers Revealed: New Challenges for Asia*, Hong Kong: The Economist Intelligence Unit, December 1998.

[Epstein96] Joshua M. Epstein and Robert L. Axtell, *Growing Artificial Societies: Social Science from the Bottom Up,* Cambridge: MIT Press, 1996.

[Farrell98] Winslow Farrell, *How Hits Happen*, New York: HarperCollins, 1998.

[Forrester61] Jay W. Forrester, *Industrial Dynamics,* Cambridge: MIT Press, 1961.

[Gubman98] Edward L. Gubman, *The Talent Solution*, New York: McGraw-Hill, 1998.

[Miller93] Byron Miller, *Object-Oriented Design Made Easy*, Minneapolis, Minnesota: Impatiens Publications, 1993.

[Mindrum97] Craig Mindrum and Dennis Bowers, “The Human Performance Framework: A Holistic Approach to Linking Human Performance to Business Performance,” Andersen Consulting Internal Document, 1997.

[Richardson81] George P. Richardson and Alexander L. Pugh III, *Introduction to Systems Dynamics Modeling with DYNAMO*, Cambridge: MIT Press, 1981.

[Treacy97] Michael Treacy and Frederik D. Wiersema, The Discipline of Market Leaders: Choose Your Customers, Narrow Your Focus, Dominate Your Market, 1997.

[Nagendra Prasad99] M.V. Nagendra Prasad and Donald A. Chartier, Agent-Based Workforce Transformation Simulator and Behavior-Oriented Modeling, forthcoming technical report, 1999.

DISCUSSION:

COMPUTATIONAL MODELS OF SOCIAL AND ECONOMIC PROCESSES

Ian Lustick (to Jing Yang): I wanted to know where you got the rational expectation price. How did that get calculated?

Jing Yang: That's just a mathematical solution for the rational expectation. That's a very regular rational expectation model. As long as you know the dividend process, then you can calculate the rational expectation.

Lustick: Okay. So you used the values posted for the stock and it produces what ought to be the [price].

Yang: Yes.

David Sallach (discussant): I'll just make a couple of remarks, and then we'll open it up to more general discussion. As I said at the outset, I think there's a lot of breadth in these talks. I think they interact in some interesting kinds of ways and raise interesting kinds of issues. This last talk [by Nagendra Prasad] I think raises some issues that, as an embryonic community, we'll need to address sooner or later. And one of them has to do with the relationship between more traditional methods of simulation, such as systems dynamics, and agent-based simulation. I think that one of the questions there is the extent to which they can be married in an effective way. One thing that I didn't see you specifically mention, but that I think would be of interest, is a kind of foreground/background focus — the extent to which it's possible to take those factors that are not focal, but that do need variability and variation, and develop a systems dynamics model for them that sets a broader context for the agent-based simulation and, to some extent, automates your sensitivity analysis. I think that's an important kind of possibility.

A second issue that's raised, which I think comes out more strongly in the paper than it does in the presentation [by Nagendra Prasad], is the whole question of system design and analysis. And I think that, in the first place, what this does is to locate the development of agent simulation within a more traditional software development process, which is, "How do you arrive at the goals that you're trying to represent?" In your case you're driven by that process, because you have a client base, and you actually have to go out and find that answer. But I think in another sense there's a dialogue that can potentially go on between more applied kinds of projects like this and the ones in the forthcoming session and more academic projects. There can be a dialogue about the kinds of goals that are possible — the application-oriented research can bring in practical insights and things that are needed to answer specific questions, whereas the academic research can actually provide insights into how those models might be built.

I thought that Scott [Page's] points were very constructive and interesting and could be useful in both of the other topics [modeling stock markets and organizations] in terms of how to diversify the agents and assess the kinds of agents that are relevant. Regarding Scott's presentation, one question that came to mind is what larger issues does it raise? I guess this a computer science or an information systems kind of point. What issues does it raise for knowledge representation?

To me, one of the horizons that the agent simulation community wants to be thinking about is precisely the question of knowledge representation. I mean, we do face complex entities, we do face complex interactions. A kind of flat attribute representation of the way in which agents are distinguished from each other is going to, I think, have its definite limits.

I think that what this point really does is to reinvoke Rob [Axtell's] point earlier about computational social theory, or maybe it's *artificial* social theory — I'm kidding! By that I mean the kinds of knowledge representations that we use to represent social agents and social processes. Just as a specific example, think about the way in which social processes are layered and interpermeate each other, so that you have individuals who are themselves social creations operating in organizations that operate within institutions, and they're all interleaved. This, I think, could be posed as a knowledge representation problem and — insofar as it's effectively addressed as a knowledge representation problem — can potentially feed back into both the research and the application orientations.

On the question of modeling the stock market, it seems to me that Scott [Page's] point about the diversity of agents, the diversity of motives — the diversity of focus, I mean — and especially diversity of time horizons is relevant. I mean, there's a step toward it by having two types of traders, but it would be nice to have that far more diverse, and I would think that the robustness of results would be strengthened by having a diversification of agent types.

So those are just a few reactions that I had to these papers, very different from each other, but each making its own distinctive kind of contribution.

So let's open up the floor more generally.

John Padgett: This is quite a specific question I should have asked Scott [Page] earlier; it's addressed to you. The take-home message of your talk, namely that diversity helps, is very simple to comprehend. What's a little less simple, at least for me, is the domain conditions that that sentence is supposed to apply to. And I was just wondering if you could say something about what is common across the diversity of your examples that makes your conclusion hold.

Scott Page: That's a great question. I think that what we've found is that if we assume diversity of language and diversity of knowledge representation, then these results don't hold. One of the things that I'm implicitly assuming in the diversity of problem solvers case is that everybody's representing reality in the same way. If you allow people to have different representations — miscommunication costs — then it turns out that a lot of the benefits of diversity can go away.

And I think one of the big questions — and this is the reason why it's worthwhile to create simple models and explore them — I think a big question is what sort of social ones — you know, let me give a simple critical example. Suppose I give a majority rule — first past the post, plurality rule. Well, you could make an argument that that's going to lead to maybe one or two [parties]. There's Duverger's Law that says [this condition will lead] to two parties. We've seen that, although that's not true at the aggregate [level], at the local level it is true. Tim Feddersen has a paper saying that.

So you could argue that that leads to two ways of thinking about politics, at least locally, that if you go to proportional representation, you could get 20 ways or 10 ways — or in Italy an infinite number of ways! And there's the question of under what circumstances is that diversity good and under what circumstances is that diversity bad — it could depend entirely on the type of problem.

The simple things we've found out so far are that if people are encoding it the same way and can communicate, then it seems to always be good. But if they encode it in different ways, then it isn't. This idea is also reflected in the chain store thing: if people are living in different environments, where there's a different pattern of stores, it's likely that the ideas they generate are going to be different, right? So diversity will lead to more people trying new things, whereas homogeneity leads to fewer people concentrating on specific things. As to which one of those is better, some of the landscape people here who've hit Santa Fe can tell you, "Well, that depends on what the problem is."

So I think one of the really interesting institutional questions is whether these additions help us sort out problems — the problems where we should have diversity and problems where we shouldn't.

M.V. Nagendra Prasad: I heard the boundary issue raised at least a couple of times during the talks, and I have a different take on the boundary issue, which is that if you find that you have a system that's very sensitive to boundary conditions, you're actually lucky. And the reason for that is that you really have a nonlinear system out there, for which humans without a model will have umpteen amount of trouble to really comprehend.

Take, for example, TalentSim. Say you make some assumptions about peer pressure — I don't know, some graphical assumptions of functional forms for the effect of peer pressure on performance — and then do a sensitivity analysis at the end of the model and by changing the peer pressure forms. You may suddenly find that if peer pressure changes its form, it's having a drastically different effect on the performance of the whole change management process. I mean, humans most likely don't really have functional forms for these. They're just trying to intuit their way through organizations. That means you are actually going to give much more value to the humans when the model is very sensitive. So sensitive boundary conditions are great. Jump for joy when you get them, actually.

I had actually one more comment about David [Sallach's] comment about interplay between analog and discrete. Yes, it's a tough point. Right now the way I do it is pretty ad hoc. I do some thresholding measures. But there is one area of computer science that actually does this routinely, and that is fuzzy set theory. Fuzzy sets use continuous membership representations and convert to discrete rules. We could look to that to try to get some insights into how to go between analog and discrete effects. I've started to look into it, but don't hold me to it! [*Laughter from audience.*] I think there is something there.

Modeling Electrical Networks

LEARNING WITH SIMULATION: UNDERSTANDING THE STRATEGIC IMPLICATIONS OF DEREGULATION AND COMPETITION IN THE ELECTRICITY INDUSTRY

A. LOMI, University of Bologna
E.R. LARSEN, City University of London*

ABSTRACT

As deregulation of the electricity industry continues to gain momentum around the world, electricity companies face unprecedented challenges. Competitive complexity and intensity will increase substantially as deregulated companies find themselves competing in new industries, with new rules, against unfamiliar competitors — and without any history to learn from. We describe the different kinds of strategic issues that newly deregulated utility companies are facing and the risks that these strategic issues imply. We identify a number of problems induced by experiential learning under conditions of "competence-destroying" changes, and we illustrate ways in which companies can activate history-independent learning processes. We suggest that Microworlds — a new generation of computer-based learning environments made possible by conceptual and technological progress in the fields of system dynamics and systems thinking — are particularly appropriate tools to accelerate and enhance organizational and managerial learning under conditions of increased competitive complexity.

INTRODUCTION

Electric utility industries around the world are shaken by an unprecedented wave of change, and uncertainty exists about how this radical transformation will affect utility companies. Some analysts predict that the number of utility companies will dramatically decrease following a steep rise in the number of mergers and acquisitions, while other industry specialists predict the emergence of a new generation of aggressive, specialized niche players that will take away market share from established utility companies (Navarro, 1996; Weiner et al., 1997). All agree that competitive complexity in the energy industries will increase substantially. Utilities of tomorrow will not only compete with other utilities, but with municipalities, brokers, non-profit consumer groups and cooperatives, and perhaps with giant multinational chemical and oil corporations. How long and how difficult the transition toward this new competitive world will be depends on a number of crucial issues. How far will deregulation at the national and supranational level go? What is the minimum number of producers that must be present for an energy market to be competitive? What degree of vertical integration is acceptable? How diversified will utility companies become? What will be the impact of globalization? Will generators be allowed to wheel energy directly to small consumers? How will technological change affect entry barriers? How will transition toward private ownership be accomplished?

---

* *Corresponding author address*: Erik R. Larsen, Department of Management Systems and Information, City University of London Business School, Northampton Square, London EC1V 0HB, UK; phone: +44 171 477 8374; fax: +44 171 477 8579; e-mail: e.r.larsen@city.ac.uk.

What form of governance will be more appropriate? How will the financial market affect the technological choices of the companies? Who will pay for past mistakes?

At least three implications are already quite clear for utility companies in the midst of these dramatic uncertainties. First, the organizational structure of utility companies of tomorrow will look and be very different from that of their monopolistic, government-owned (or controlled), vertically integrated grandparents that have dominated — *and* helped to build — many national economies. Second, electric utility companies will differ in their ability to learn how to live in harmony with their new environments — and profit from them. Third, as new competitive forces sweep through organizations, markets and governments the mindsets of utility executives also must undergo radical change (Navarro, 1996). The electricity industry of tomorrow provides the perfect context in which the ability to learn faster than competitors is the only real source of sustainable competitive advantage (de Geus, 1988; Hamel and Prahalad, 1994). If this is indeed the case, then it is hard to imagine more important questions for electricity companies than the following: How can individual learning be accelerated? What individual learning mechanisms should be activated to facilitate the construction of a shared vision of the future? In this paper we attempt to articulate possible answers to these questions.

In our attempt, we focus on organizational learning — a frequently invoked but ill-defined and rather abstract concept. For this reason we found it useful to anchor some of our claims about how organizational learning happens (or fails to happen) to concrete examples taken from the electric utility industry — an industry to which we have been devoting some of our research efforts during the recent past (Bunn and Larsen, 1997). In our discussion we concentrate on electricity companies because we believe that the problem of learning is particularly salient for these companies, but most of our arguments extend naturally to organizations operating under conditions of what might be called "competence-destroying" institutional change, i.e., discontinuous change in a company's environments which renders obsolete the expertise, competencies, and capabilities that facilitated its adaptation and improved its performance under the previous regime (Tushman and Anderson, 1986). While we do not deny that many of the elements that we discuss are specific to utility companies, we believe that administrators and executives of telecommunication companies, hospitals and health service organizations, airports service companies, gas and water companies, museums and art galleries, football teams, national railway companies, national television stations, and — perhaps — educational institutions can all relate to our arguments, at least to some extent. We believe that in all these organizations the value of experience will soon become problematic. Consequently, we predict that in this kind of organization, the debate about how exactly the past might (or, as the case may be, might not) be connected to multiple possible futures will move to center stage of the policy debate. In simple words, the solution to this central problem in the management of change depends on the ability of organizations to imagine and implement new ways to learn without experience. How this unusual type of learning happens and how it can be induced is the central issue that we address in this paper.

Our discussion is organized as follows. In the next section, we identify the main drivers of change in the international electricity industry. This discussion provides the empirical context for our subsequent conceptualization effort. Then we identify a number of problems associated with experiential learning that are specific to companies witnessing fundamental transformations in their competitive and institutional environments. In the fourth section, we identify two history-independent learning mechanisms that top management teams can leverage to accelerate organizational change. In the fifth section, we discuss the role of Microworlds — a new generation of computer-based learning environments for managers — as one possible solution to the problem of learning without experience. We conclude the paper with a brief discussion of the main implications of our current work for organizational learning and learners.

MAIN DRIVERS OF CHANGE IN UTILITY COMPANIES

Following Weiner et al. (1997), it is useful to identify three main types of large-scale change shaping electric utility companies and industries in most developed countries: market change, regulatory change, and technological change. While it is unclear exactly how these different changes will play themselves out over time, it is easy to predict that our current understanding of what a utility company is and does will be permanently altered.

Market Change

Under monopolistic conditions, the mechanisms of price formation are relatively well understood, customers are captive and are not considered by companies as sources of relevant information, tariffs are negotiated with/imposed by a regulatory body, and information about the industry is generally available and used in centralized planning exercises (Larsen and Bunn, 1998). Competition introduces consumer choice, price and product differentiation strategies, asymmetric information between companies and regulators, and new entrants fighting aggressively for market share. The effects of these changes are already appreciable in many national utility industries, for example:

- In the first five months of retail competition in the U.K. gas market, almost 20% of domestic customers have switched suppliers.

- The combined market share of the dominant generators in the U.K. declined from 74% in 1991 to 56% in 1995. The market share of independent power producers during the same period went from zero to 10% by output.

Regulatory Change

In most western European countries, utility regulation has been relatively light during the era of national monopolies. Also, European-level coordination and regulation were virtually non-existent until the mid-’90s (Matlàry, 1997; Larsen, 1998). Within individual countries, there was no obvious conflict of interest between the regulatory body and the regulated utility companies, as both were seen as trying to achieve the same goal of secure (and, at least to a certain extent, efficient) delivery of what was typically perceived by companies, consumers, and labor unions as a public good (Gilbert and Kahn, 1996).

However, as national energy industries become more competitive and integrated, the objectives of the regulator and those of the companies will begin to drift apart. The relationship between regulators (who are now seen as “watchdogs”) and regulated (who are now seen as profit-maximizing investor-owned corporations) will tend to become less cooperative, with more occasions for strategic behavior and opportunism. Management will be under increased pressure to concentrate on profitability and shareholder value, which might not be viewed by regulatory bodies as necessarily coinciding with consumers’ best interests — at least in the short term.

Competition typically creates the context for information asymmetries and opportunistic behavior. Companies are likely to have information about themselves and their competitors that is more accurate than information collected by regulators. Regulators will find themselves in the uncomfortable position of having to design and enforce rules under conditions of partial ignorance about how these rules actually affect companies' profitability and, ultimately, industry structure. Consequently, a new kind of flexible regulation will emerge, characterized by a continuous conversation and by an almost real-time activity of mutual adjustment and negotiation between companies and regulatory bodies.

For example, in the U.K., the regulator and the regional electricity distribution companies (RECs) had agreed on the price increase that these companies were allowed to impose over the next four-year period (the domestic supply business was still a monopoly at the time). In the six months following this agreement, the press systematically drew the attention of the public to the high pay increases that senior managers of these companies were awarded (in some cases much more than 100%). The general opinion that the terms of the price increase that had been negotiated were too favorable to the companies was confirmed when one of the RECs somehow managed to "find" £500 million to pay back to the shareholders should a hostile takeover bid be rejected — financial resources that the regulator could just not find in the official accounts that the company presented six months earlier for audit. After this unexpected "incident," the pressure coming from public opinion increased until the regulator was forced to renegotiate the price increase that the RECs were allowed to impose. The renegotiation of price increases was necessary only nine months after the original agreement, instead of the four years originally planned.

How fast companies learn and adjust to new constraints becomes the key issue in a world where it is not the big company that eats the small, but the fast company that eats the slow. Companies are trading off size for speed in a way that was unthinkable only five years ago, when the discussion was all about market share and installed capacity.

Companies might be able to enjoy a considerable competitive advantage when they choose to embrace the new competition early on in the process, and position themselves to take full advantage of deregulation. A company that actively looks for opportunities to compete not only reduces the exposure to regulatory action (Currie, 1997), but also improves its ability to learn faster than more inert — or regulated — competitors. In this sense deregulation becomes a crucial opportunity for building a decisive competence-based competitive advantage. A simple example taken from the South American experience might help to clarify this point.

The first country to deregulate in South America was Chile, which started the deregulation process in the early '80s (Bitran and Serra, 1997). During the last decade, the majority of other South American countries have deregulated and privatized their domestic electricity industries, including Argentina, Columbia, and partly Brazil. Today Chilean electricity companies operate or own capacity in all the South American countries that have opened their national electricity industries to competition. However, there are no companies from other South American countries operating in Chile. U.S. and Spanish companies are the only foreign companies operating in Chile today.

Technological Change

Technological innovation and change have been important enablers of competition in the electricity industry worldwide. The impact of technological change is clear when one considers its implications in terms of economies of scale and entry barriers. Up to the ’80s, the efficient economic size of electricity generation plants kept increasing to reach an estimated 1,000 MW in 1980, after which the trend reversed because of the availability of natural gas and the rapid development of gas turbine technology. A combined-cycle gas turbine (CCGT) reaches maximum economic efficiency at a much smaller scale, estimated around 400 MW (Energie Verwertunsagentur, 1996; Energy Information Administration, 1996). Over the last decade, electricity companies around the world redesigned their technological portfolios, and this resulted in large-scale structural change at the industry level. For example,

- In the U.K. all the new capacity since deregulation has been CCGT, i.e., gas-fired capacity. This has threatened the future of coal production in the U.K., and the government has put a moratorium on the building of new gas generation plants.

- In Denmark, there has been a rapid growth of decentralized, small-scale combined heating and power (CHP) capacity. Many villages now have their own CHP plant, selling the electricity to the local distribution company at a favorable rate. The growth has been encouraged by the government as part of promoting a better use of energy and lowering emissions.

The change from a monopolistic to a competitive electricity market will strongly affect the choice of generation technology also because of construction time considerations. Monopolists typically can afford to take a reasonably long-term approach to the composition of their technological portfolios. This is possibly the main reason behind the construction of nuclear or large-scale hydro-electric generation plants — projects that typically require at least 10 years of work for obtaining permissions, planning construction, and actual execution. When the investment horizon is 40-50 years (as is usually the case for large-scale hydro plants), these technological choice can be considered as economically viable and perhaps even socially desirable. But private companies typically have much shorter time horizons, because it is hard to think that stockholders will judge favourably investment plans that imply waiting for 10 years without any returns. Also, in a rapidly changing industry in which supply and demand conditions are evolving toward a permanently higher degree of competitive uncertainty, it is unlikely that private investors will feel comfortable with investments whose financial implications extend so far ahead in time. On the other hand, the new gas-fired plants (CCGT) can be built in less than three years, provide more flexibility for the generation company (because the actual size can be adjusted to short-term market fluctuations), and are considerably less capital-intensive. For example, in 1994 the operating and maintenance cost for fossil-fuel generation in the main investor-owned electric utilities in the U.S. was estimated to be between 2.2 and 3.2 cents per kilowatt-hour (Energy Information Administration, 1996). For the same year, Linden (1995) reports that the total cost (i.e., including operating, maintenance, and capital costs) for the new CCGT plants was about 3 cents per kilowatt-hour. In other words, for many companies considering the possibility of entering the industry, it will frequently be less expensive to build new capacity than to operate some of the more expensive capacity in existence (Energy Information Administration, 1996).

LEARNING WITHOUT EXPERIENCE AND THE REWIRING OF MENTAL MODELS

Simultaneous changes in market structure, regulatory regime and production technology interact to create a competitively complex environment — not only for utility companies**.** Business strategists and planners agree that in periods of rapid change and uncertainty, learning becomes the only sustainable competitive advantage (de Geus, 1988; Hamel and Prahalad, 1994). From this perspective, the leaders' new work involves creating a learning organization — i.e., an organization that encourages decentralized experimentation and that helps its members to reflect on, and improve, their own decision processes (Senge, 1990b). More specifically, under conditions of rapid and fundamental transformation, the main focus within the company becomes the creation of conditions for "institutional learning," or "the process whereby management teams change their shared mental models of their company, their market, and their competitors" (de Geus, 1988: 62). Unfortunately learning is no magic word. In fact, organizational learning is more often likely to be a problem that executives have to solve, than a solution that they can readily adopt. Organizational learning is problematic essentially for three reasons: the paucity of experience relative to possibilities, the dangers of experience traps, and the ambiguity of vicarious learning.

The Paucity of Experience Relative to Possibilities

Learning typically requires the existence of repeated experiences. However, no electric utility company in Europe or the U.S. can count on a sufficiently large sample of experiences about surviving in a globally competitive energy market. Consider the following examples based on our observations of the behavior of many European energy companies.

- The CEO of a nationalized utility company recently told managers in the generation division that in the near future their performance would be evaluated in terms of their ability to maximize shareholders value. But no one in the electricity generation business ever had to respond to a similar structure of incentives.

- As new competitive power markets emerge, utilities, investment houses and other organizations will increasingly be involved in commodity trading and purchasing. But most electricity companies never had to develop risk-management competencies to deliver hedges, options, and other futures. There are several ways of building these new competencies — for example, they could be imported from another industry — but it is not clear what resource accumulation strategy is likely to work best for electricity companies.

In all these cases it is obvious that experience cannot be a useful basis for action. As March, Sproull, and Tamuz observe (1989), small samples of experiences trigger processes of interpretation, i.e., addition of elements to allow the accumulation of knowledge. The diversity of these interpretations is necessarily restricted by the diversity of mental models available within the top management team.

Experience Traps

Even when enough observations are available to derive accurate inferences about underlying structures or processes, there is no guarantee that learning will lead to the right solution. The main issue in this case is the self-reinforcing nature of mental models, which tend to produce adaptation to past experience, whereby, as Lounaama and March put it, “false lessons are learned as rapidly as true lessons” (1987: 122). Research in organizational decision making shows that when outcome feedback is delayed or ambiguous, organizations are likely to repeat decisions because they have made them in the past. In other words, decision processes become a basis for learning independent of decision outcomes. The effects of this “experience trap” are evident in the following real-life examples.

- Managers of a large European utility company that has enjoyed (and still enjoys) a protected positional advantage recently told us, “If we are here after such a long time we must be doing something right,” and “While everything can be improved, we see no immediate need for change because we are already very efficient. After all, when was the last time that you were left without power?”

- After the vertical separation of the generation, transmission, and distribution components of the business, electricity companies in Colombia face for the first time a significant market risk. Local distribution companies now buy electricity at market prices but have to sell electricity to a captive market at a price fixed by government authorities. This situation implies a significant market risk for distribution companies, which now have to learn how to use financial instruments to cover their contracts. After the recent unexpected increase in prices because of “El Nino,” for the first time a number of Colombian electricity distribution companies went bankrupt and were taken over by other companies.

We believe that these simple examples illustrate clearly the dangers of experience-based learning under conditions of structural change. Because organizational structures respond only with delay to pressures for change, companies continue to adjust to a past set of contingencies that are no longer relevant and evoke routine solutions for solving new problems.

Vicarious Learning

Learning through the experience of others by imitating or avoiding what others do is frequently portrayed as a powerful source of organizational structuring (DiMaggio and Powell, 1983), and as an efficient strategy for reducing uncertainty (Grant and Baden-Fuller, 1995). Learning from others is greatly facilitated when the “others” are perceived as similar and when the new knowledge to be assimilated is somehow comparable to existing knowledge (Cohen and Levinthal, 1990). However, the value of vicarious learning is ambiguous when (a) there exist profound technological, economic, and historical differences among apparently similar companies and (b) the new competencies to be absorbed are not sufficiently similar to the existing competencies. For example,

- The regulatory principles underlying the market for power in Colombia and Great Britain are very similar, partly because of the role played by consulting companies in designing electric power markets in South America. However, because of significant differences in technology (less than 5% of the electricity generation in Britain is hydro, while more than 75% of the electricity produced in Colombia comes from hydroelectric plants), the dynamics of price formation in these two apparently similar markets are completely different. In spite of the surprising institutional similarities, it is unclear how much a Colombian company could learn from its British counterpart about survival in a new competitive market for electricity.

- When top managers in a large, state-owned electricity company were asked to reflect on the strategic and competitive consequences of unbundling — the separation into independent corporate entities of the electricity generation, transmission, and distribution businesses — they refused to acknowledge unbundling as a viable strategic option ("We would be far too small to compete in the global marketplace") and indicated EDF, the giant French state-owned electric utility, as the organizational model to imitate ("Why can't we do like the French? They obviously know what they are doing").

The first example illustrates the problems and dangers of learning through the experiences of others facing apparently similar constraints. The second example illustrates how an organization might tend to establish its aspiration levels and performance criteria on the basis of the experience of other organizations to which it compares itself (Cyert and March, 1963). Frequently, these "similar others" are selected in a way that makes inaction appear fully justified, and in fact rational. In other words, the search for new solutions is restricted to the firm's immediate neighborhood defined in terms of perceived strategic similarity (Odorici and Lomi, 1999). The main consequence of the strictly local character of this search is that the *status quo* is maintained.

As all of our stylized facts and examples illustrate more or less directly, what makes a company well adjusted to its environment also makes experiential learning problematic for companies experiencing fundamental transformation. Learning from experience is almost by definition a backward-looking activity based on self reinforcing processes (Lomi, Larsen, and Ginsberg, 1997). Organizational learning is encoded in routines, and organizations learn and remember by doing (Nelson and Winter, 1982), but the likelihood that a specific routine will be evoked depends on its association with *past* success (Cyert and March, 1963). This is one of the most uncontroversial results of research in organizational decision making because, as March and Simon put it (1958: 140), "When a stimulus is of a kind that has been experienced repeatedly in the past, the response will be highly routinized. The stimulus will evoke — with a minimum of problem solving or other computational activity — a well-structured definition of the situation."

For utility companies that are getting ready to navigate through the storm of change, the real question is: How can we improve on our backward-looking strategic decision processes? How can we generate possible futures and learn without the possibility of experiencing these futures directly? How can we encourage managers to reorient their attention away from the replication of well-defined solutions that have worked in the past and toward the exploration of alternative time-paths into the future? To these questions we turn our attention next.

SIMULATED EXPERIENCES: LEARNING FROM HYPOTHETICAL HISTORIES AND HISTORICAL NON-EVENTS

Experiential learning is necessarily influenced by historical events, i.e., by what was actually observed to happen in a specific business context. As a company learns from its experience, organizational structures are progressively modified to accommodate and reinforce the dominant interpretation of historical events. This is essentially what Peter Senge identified as "adaptive learning" (Senge, 1990b), which results in a continuous activity of improvement of organizational structures and processes — until something breaks. Given the unexpected and frequently dysfunctional consequences of experience-based learning under conditions of increased competitive complexity, how are we supposed to make sense of "organizational learning"? Clearly, the challenge for organizational learning theorists, organizational designers and planners is to imagine learning mechanisms that are as much as possible history-independent, but that at the same time may be used to trigger the reframing and re-perception of taken-for-granted "facts," "events," "situations," and "positions." Once these history-independent learning mechanisms are identified and controlled, the rewiring of individual mental models — and therefore large-scale processes of organizational change — can be greatly facilitated. Below we discuss two of these history-independent learning mechanisms: simulating hypothetical histories, and learning from historical non-events.

As organization theorists March, Sproull, and Tamuz (1991) recently pointed out, organizations also learn by constantly simulating hypothetical histories and by making sense of historical non-events. Hypothetical histories and historical non-events provide crucial opportunities for organizations to learn without experience. They force managers to (a) experiment with novel solutions to non-routine problems and (b) come to terms with the fragility of their preferred strategies and policies. If properly managed, the systematic reflection on "what could have been" and on what "almost was" can be most conducive to a creative problem-solving activity aimed at constructing and sharing a new definition of the situation.

In the words of March, Sproull, and Tamuz (1991: 5), "Hypothetical histories play a role in organizational learning similar to that of mental models or simulations in studies of individual learning." Exploring hypothetical histories requires the generation of scenarios, i.e., unrealized but plausible stories about how the future may unfold. Arie de Geus provides a particularly vivid example of how scenarios are used by organizations to learn from hypothetical histories. Reflecting on his own experience as the Head of Planning for the Royal Dutch/Shell group, he recalls, "In 1984 we had a scenario that talked about a $15-a-barrel oil… (Bear in mind that in 1984 the price of a barrel of oil was $28 and $15 was the end of the world to oil people). We thought it important that — as early in 1985 as possible — senior managers throughout Shell start learning about a world of $15 oil" (de Geus, 1988: 72-73). He continues,

> Following this inspiring example, we tried to get senior managers and engineers in a large electric utility company to start thinking about selected implications for their organization of fully competitive energy markets. So we asked them how they would handle a situation in which supermarkets (instead of energy companies) became the main distributors of electricity to small consumers and households. Would they try to differentiate their products? How would they build brand equity? Would they compete with supermarkets for distribution? How would they try to identify segments of the market? Would they cut prices? How would they compete for shelf-space with other — possibly foreign — generators of electricity? Would they open or buy supermarkets? The first reaction to our questions was one of disbelief and almost dismissal. The discussion concentrated on why this situation was not realistic, on why these problems were not relevant to the company at the moment, and on how national regulatory authorities would never let this happen. Then we showed excerpts of interviews with the CEOs of some among the fastest growing utility companies in the world, and this helped to

> overcome the initial resistance to engage in the discussion. For example, the tone of the discussion changed appreciably after we passed around an excerpt of an interview with Richard Green, the CEO of Utilicorp in which he predicts that "Soon, you'll buy your energy in a box off-the-shelf at Wal-Mart." Despite the large amount of "energy" absorbed by this exercise, in the course of the discussion it became clear that the really important issue was not trying to predict "What will happen" but rather reflect on "What we would do if it happens." This shift resulted in a rather dramatic change of perspective on a business typically perceived as "mature" and represented as a "natural monopoly."

Historical non-events provide similarly important opportunities for organizations to learn without experience. Historical non-events are "incidents" or "decision nodes" that under slightly different circumstances could have produced dramatic differences in the current situation. Using a concept from organization theory, historical non-events can be thought of as "occasions for structuring" — brief local opportunities to re-perceive and re-discuss the entire causal structure of decision environments (Barley, 1986). When systematically collected and interpreted, information on historical non-events may help decision makers to understand the causal structure behind their own policy decisions.

Organizations frequently simulate hypothetical histories and try actively to learn from historical non-events. However, organizations rarely do so systematically and intentionally. Rather, they must be helped by establishing and promoting structured ways to experience the future (Meadows, 1984, 1989). Similarly, individual managers are rarely given the opportunity of de-biasing their judgement by reflecting systematically on the consequences of their decisions in the light of "what could have happened" ("hypothetical histories") and of "what did not happen" (historical non-events). Managers and — in more general terms — decision makers within organizations also need tools and opportunities to explore alternative time-paths into the future and test the robustness of their complex understanding of the "real world." In other words, they need opportunities to rewire their mental models. And this is exactly what Microworlds are for.

FROM THE "REAL WORLD" TO WORLDS OF POSSIBILITY: THE ROLE OF MICROWORLDS

Developing the skills needed for converting unsystematic and intangible history-independent learning processes into specific competitive advantages requires new tools and new approaches to the engineering of strategic choice. Microworlds are among the most promising new technologies specifically designed for improving and accelerating organizational learning.

First we must answer the question: What is a Microworld? The expression comes from computer scientist Seymour Papert, who in his book *Mindstorms* defines a microworld as a kind of learning environment emerging from a specific interaction between learners, transitional objects, and learning processes (Papert, 1980). The more operational, if less profound, definition offered in the web page of the NASA productive conversation training program serves as a useful point of departure for our present discussion: "A Microworld is essentially a computer simulation of an organizational process or system. It is used to apply the practice of systems thinking and mental models to a more realistic business case in which quantitative as well as qualitative assessments can be made. The ability to experiment also makes it an excellent tool for strategy development." The value of Microworlds for strategic planning is becoming apparent as corporate planning processes become more open (i.e., more transparent to individuals in different parts of the organization and even outside the organization), decentralized (i.e., closer to the experience of individual decisionmakers), and inclusive (i.e., based on the input of the largest possible number of people and institutions that could be affected by the outcomes) (Ginsberg, 1997).

Microworlds are technically possible thanks to (a) the recent development of software used to map and model the structure of organizational systems; (b) the availability of a new generation of object-oriented computer languages that allow the design and implementation of innovative graphical interfaces, and (c) important progress achieved within the systems thinking/system dynamics (Morecroft, 1988, 1992) and the Soft OR communities (Lane, 1994) on developing a conceptual relationship between mental models and computer simulation models.

Defining Microworlds as "computer simulations" is useful to fix ideas, but perhaps too restrictive for our current purposes. In a Microworld, users interact with a computer model through a user-friendly graphical interface that simulates a realistic, information-rich learning environment. The main objectives of computer-based learning environment that a Microworld embodies are to provide

- An opportunity to managers for discovering the causal structure of their decision environments,
- An opportunity to explore and create alternative futures by linking decisions to their intended and unintended consequences,
- A risk-free environment in which managers can actually try to "fly" their company, and
- A structured arena for dialogue and exchange within which players can build a shared understanding of the business situation that makes their decisions interdependent.

Given these rather specific learning objectives, models behind Microworlds are typically small, but capable of producing a high level of dynamic, as opposed to detailed, complexity (Senge, 1990a). The computer model that is at the heart of a Microworld should not be judged on the basis of its predictive accuracy (and/or validity), but rather on the basis of its

- Attitude to stimulate novel thinking about future business opportunities,
- Capacity to facilitate the sharing of mental models of a specific business situation,
- Potential for enhancing the participants' introspection about their own decision processes and routines,
- Ability to create a new language and new concepts to re-perceive events of common experience, and
- Usefulness for unveiling hidden assumptions behind collective interpretations typically presented as "taken for granted" or "obvious."

While these performance criteria would be daunting for a model of any kind, we feel that this is what it takes if Microworlds are to fulfill their promise of becoming “the technology of the learning organization” (Senge, 1990a).

Perhaps the most distinctive feature of the system dynamics model behind a Microworld is the model-building process itself. The model behind a successful Microworld is firmly rooted in the understanding and knowledge of participants in a specific decision situation. Participants are the real “owners” of the model, while model-builders simply facilitate the translation of mental models into a computer model. For this reason, it is essential to ensure an intense participation of senior managers early in the model conceptualization stage. The final product of the model-building process hinges crucially on the ability of the management team to recognize the hidden assumptions that undergrid their shared understanding of the business.

First-generation Microworlds now feature prominently in reputable international MBA and executive education programs. One example of a first-generation Microworld is the People Express Management Flight Simulator (PEX MFS), developed by John Sterman at MIT (Sterman, 1988) on the basis of the homonymous HBS case study (Whitestone, 1983). Participants are put in the role of People Express top management against the background of the U.S. air-travel market and competitive environment in the early ’80s. The main task is to balance the conflicting demands posed by managing operations, human resources, organizational structure, and pricing as the company experiences rapid growth. Since its introduction, many universities including London Business School, Harvard Business School, Stanford Law School, and the IMD of Lausanne have adopted the PEX MFS. Graham, Morecroft, Senge, and Sterman (1992) provide additional information on the Microworlds that have been developed following the successful introduction of the PEX MFS.

Companies obtain the maximum benefit from Microworlds developed to address specific problems that are perceived by the top management team as directly relevant, or to support a specific set of decision making processes. For example, the Oil Producers Microworld (OPM) grew out of a project at Royal Dutch/Shell to develop a simulation model of global oil markets to explore the strategic implications of changes in the structure of the energy industry. The model has been used extensively within the company as a way to generate new insight on the dynamics of oil prices and the investment opportunities of non-OPEC producers (Morecroft and van der Heijden, 1992) and as support for more comprehensive scenario planning exercises within Royal Dutch/Shell (Morecroft and Marsh, 1997).

As the technology for building computerized learning environments becomes more easily available and widely accessible and as the cost of computer resources declines, an increasing number of organizations are recognizing the value of engaging in the process of developing a computer-based learning environment, for both internal training as well as planning purposes. Consequently, Microworlds available on the market have increased in number and quality.

One example is the Beefeater Restaurant Microworld (BRM), based on the real-life history of the Beefeater restaurant chain, which was started by the U.K. brewer Whitbred PLC around 1980 and by the 1990s grew to dominate the U.K. market for mid-priced family restaurants (Warren and Langley, 1996). In the BRM, players take on the role of divisional managers and have to strike a delicate balance between the conflicting pressures simultaneously coming from the customers (who form expectations about quality and prices) and the corporate headquarters (who demand return on capital in order to grant further resources for growth). The BRM also provides the option of interacting with predefined scenarios (called management challenges) that force the players to reflect on the powerful effect of history on organizational growth and change. In their article, Larsen, van Ackere, and Warren (1997) discuss the main features of the BRM in the more general context of system dynamics models for enhancing organizational learning.

Encouraged and inspired by the success of these and other computer-based learning laboratories developed in close contacts with companies, we recently started to work with one of the largest European utility companies, whose management will soon have to face the disruptive consequences of deregulation in its quasi captive domestic market. Our main goal is to build a computer-based learning environment to help the management to prepare for change by interacting with alternative futures generated by different hypotheses about how the national and international market for energy might evolve. The Microworld project is providing an important opportunity to engage key decision makers in a disciplined conversation about how the company's profitability might be affected by the forthcoming wave of economic, institutional, and organizational changes that will be transforming the structures, strategies, and identities of utility companies around the world. The main consequence of this conversation is to help managers and planners to bring back to the present the information produced by the interaction with alternative futures and design courses of action that would be sufficiently robust across the different worlds of possibility that can be reasonably imagined. This work is firmly rooted in our belief that no strategy could be more effective in helping the company to build generative — and not simply adaptive — learning capabilities (Senge, 1990b) and to create a shared vision of its future. The appendix outlines key elements of the underlying model used in this Microworld.

DISCUSSION AND CONCLUSIONS

The most pressing issues that electric utility companies of today have to face are related to the inherent difficulties of adapting to unfamiliar and uncertain environments swept by competence-destroying changes and disruptive competitive selection. Clearly, electric utilities are not the only kind of companies experiencing these tectonic shifts in their competitive environments, and in this paper we have offered our reflections on what other companies would find it easy to relate to our discussion of organizational learning, its problems, and its solutions.

Building directly on recent advances in the fields of strategic management and organizational learning, we started from the claim that under conditions of increased competitive complexity the ability to learn faster than competitors is the only real source of sustainable competitive advantage. But organizational learning is more like a problem to be solved than a solution ready to be adopted. We suggested that what makes a company well adjusted to its competitive environment also makes experiential learning problematic under conditions of structural change because learning from experience is, almost by definition, a backward-looking activity based on self-reinforcing processes (Lomi, Larsen, and Ginsberg, 1997). Hence, we argued that in order to help companies to generate and explore alternative paths into the future, we need to identify and understand organizational and individual learning processes that are, at least in part, history-independent. Building on recent results in the field of organizational learning theory (March, Sproull, and Tamuz, 1991), we discussed two such processes related to the simulation of hypothetical histories and the interpretation of historical non-events. The simulation of hypothetical histories takes the form of rapid excursions into possible futures based on scenarios, which are unrealized but plausible stories about how the future may unfold. Through historical non-events (which March and collaborators call "near histories") the boundaries of "parallel worlds of possibility" become tangent to each other, and decision makers are presented with the unique opportunity to mentally simulate or, as it were, "experience," the consequences of what "was not" or what "nearly was." These history-independent learning processes result in the organization-level analogue of what neuroscientist D. Ingvar (1985) has called "memory of the future" — the process by which decision makers bring back to the present fragments of information they obtained during their journey through possible futures. An illuminating discussion of the implications of "memories of the future" for strategic planning and organizational learning can be found in Arie de Geus' thought-provoking new book *The Living Company* (1997).

These processes are not as artificial or infrequent as they may sound. We suggested that organizations do in fact recurrently learn without experience by simulating hypothetical histories and by making sense of information generated by historical non-events. As Peter Senge correctly observed (1990a), when companies are seen as complex social systems, every company is in fact a "learning organization." The problem is to understand the subtle, idiosyncratic, and often counterintuitive ways in which specific organizations do learn and refine their competencies. While companies in modern business environments have been forced to evolve systematic and highly structured ways to keep track of their past and provide rational justifications for their decisions, until very recently no evolutionary pressure has forced companies to rely systematically on history-independent learning processes for testing and modifying established mental models and improving the quality of organizational decision processes. Hence, the problem becomes how to intervene in an ongoing sophisticated organizational learning system with a long evolutionary history by providing a context for imaginative yet disciplined exploration of alternative paths into the future. We proposed that Microworlds — computer-based learning environments built in close contact with members of top management teams and corporate planners — can be seen as an innovative technology for triggering and supporting history-independent learning processes. We outlined some of the criteria that Microworlds have to satisfy in order to be credibly considered the "new technology" for the learning organization.

Obviously, it would be misleading to suggest that Microworlds should be thought of as all-purpose, ready-made solutions to the complex and delicate problems of organizational learning, and in fact our experience has been that companies differ greatly in the extent to which they are able to benefit from the intellectual investment that building a successful computer-based learning environment requires. However, we take the several examples that we reported as evidence that companies are beginning to realize the strategic value and potential of investing in sophisticated learning technologies. We believe that this kind of investment is crucial for companies as they prepare to compete for the future.

APPENDIX:

A QUICK GUIDE TO THE MODEL UNDERLYING THE *CAPACITY INVESTMENT MANAGEMENT MICROWORLD*

This appendix shows an example of the underlying model of a Microworld recently developed for a major electricity company. We will not provide the detailed model, but rather give an overview of some of the most important parts of the model. In this appendix the client company for which the microworld was developed will be simply called the "Company."

The model underlying the CIMM is a system dynamics, or feedback, model. This systems perspective builds on the interaction of negative (or balancing) loops and positive (or reinforcing) loops. The building blocks of system dynamics models are stocks and flows. Stocks are variables that accumulate activities and resources over time. Flows are variables that regulate the rate at which these activities and resources are accumulated or, as the case may be, are lost. A "model" is a network of interconnected stocks and flows. The model in the CIMM was originally built in Powersim™ and was later translated into Visual C++™, the language that we also used to develop the interface that allows users to interact with the model, even if they know little about its structure and internal mechanics. The model consists of 285 equations that can be clustered in a number of different interacting macro-sectors corresponding to broadly defined policy areas. In the reminder of this appendix we describe the main structural elements of the model in the form of stock-and-flow diagrams.

Capacity Sector

The engine of the model is the capacity sector. Each company represented in the model has two types of capacity: (1) CCGT (which stands for combined-cycle gas turbine) and (2) the sum of any other capacity (e.g., oil, coal, wind, hydro, etc.). The generic structure of each type of technology for each company is as shown in Figure 1.

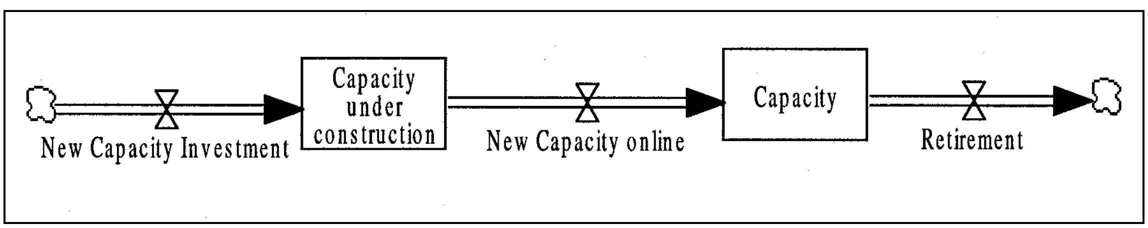


FIGURE 1 Generic structure of capacity flow.

As investments in new capacity are decided upon, they are progressively converted into a stock called Capacity under construction. After an appropriate amount of time (the construction time) the capacity will come online and become working capacity. Working capacity might eventually be retired according to some schedule. As is the case in the CIMM, the amount of capacity to be retired is decided by individual players (or teams). The total amount of capacity is defined as the sum of all the "capacity stocks" across different technologies. Likewise, the "total retirement" is calculated as the sum of the capacity retired across all companies and all technologies.

The total capacity under construction is the sum of all the capacity under construction across all technologies and all companies. However, in the CIMM this might not be represented by the exact number that appears in the graphs and reports because it depends on the specific value of the information exchange parameter. If the Company (the player) sets this parameter to (say) 0.5, only half of the level of the combined stock of the Company's capacity across technologies will be actually be reported to competitors. Similarly, if the value of the competitors' information exchange parameter is set at 1.5, competitors will announce to the market that they are building 50% more capacity than they are actually doing.

Market Sector

The market sector bounds the area of the model in which electricity demand and supply interact to determine the electricity price. We assume that demand grows at a steady yearly rate, as implied by the stock and flow diagram shown in Figure 2.

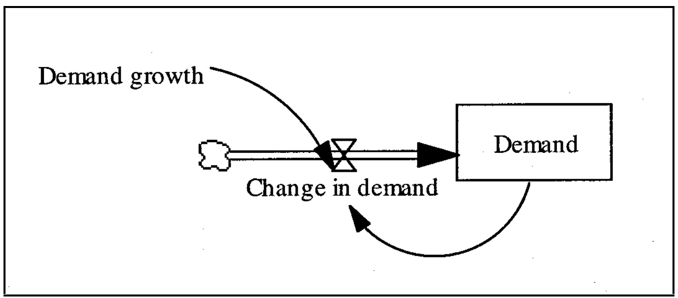


FIGURE 2 The formulation of demand.

The price-setting process is controlled by a relatively slow price-adjustment mechanism based on the aggregate supply-demand balance that emerges as a result of the interaction between the player and the model. Because the model is based on annual decisions, it is not necessary to model the exact mechanism underlying price formation (e.g., the hourly price based on a pool mechanism or the process of pricing contracts). At the time at which the model was formulated, people in the Company's planning department agreed there was just too much uncertainty about which market institution would ultimately regulate price formation in the national electricity market — and the idea of a single European electricity market just seemed too far-fetched. The variety of institutional experiments going on in Europe at the moment in which the model was built (first quarter of 1999) was such that we just could not find a non-ideological way of modeling the price-setting mechanism. For this reason the modeling team and the management team preferred to avoid the specification of *one* particular market institution for regulating electricity prices. Rather, we worked under the assumption that *whatever* type of market institution that will be implemented in the end should reflect the supply and demand balance, at least at the aggregate level (e.g., yearly average). This condition should hold true almost independent of the details of the pricing mechanism.

The model exhibits a certain amount of inertia, or "stickiness," in prices, a feature that is likely to be prominent in the "early phase" of deregulation (although this early phase might last 10 years or more in the case of electricity). No electricity system in the world would run at 100% of its available capacity. There is always a certain reserve margin built into the system. In the case of CIMM, this reserve margin is 20%, i.e., we consider a system in which an 80% utilization reflects an almost “ideal state.” Figure 3 shows the generic structure behind the price-formation mechanism that we adopted in the CIMM.

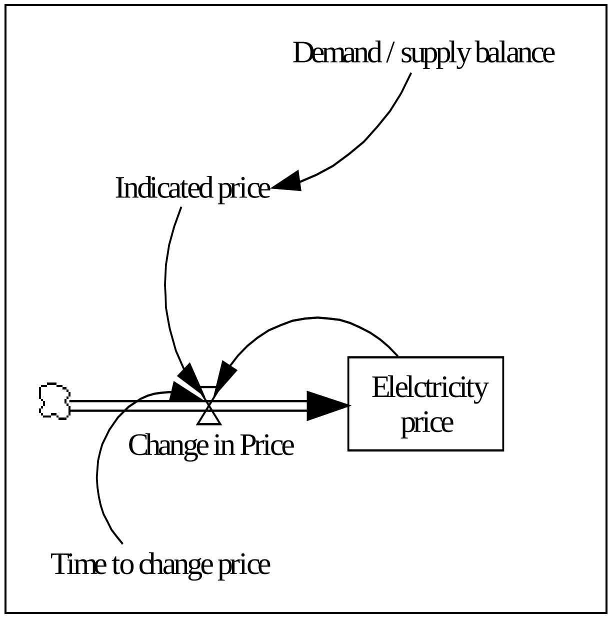


FIGURE 3 Generic structure of price formation in the model.

Company Sectors and Capacity Investment Policies

The two remaining sectors of the model represent two generalized types of competitors to the Company, “Independent Power Producer” (IPP) and “Market Share Builders” (MSB). Although there can be many companies in each sector, the model clusters them into two homogeneous classes of competitors that the Company might face. Competitive interaction among competitors determines the actual level of investment in new capacity in the industry. The only difference between the two types of competitors is really the final stage of the new capacity investment decision, as described in Step 4 below. Because their goal is to consolidate their market presence in an attractive foreign market, MSB may apply a lower rate of return on investment in their capacity-approval decision. The capacity investment decision process (or *policy*) in the model unfolds as follows:

1. Competitors start by forecasting aggregate demand in 3 years’ time. Using a standard formulation for calculating the trend in demand, the model allows for what we can call “management bias,” which reflects the opinion of management about possible changes in demand growth patterns. If the “management bias parameter” is 1, then the current trend in demand is just extrapolated: management thinks that demand in the near future will closely resemble demand in the recent past. A value of the “management bias parameter” less than 1 reflects a situation in which management believes there might be a slowdown in the demand growth rate relative to the past. A value of the “management bias parameter” greater than 1 defines a situation in which management believes that the future rate of growth will be greater than what was observed in the past. Figure 4 shows the generic structure behind this simple expectation-formation mechanism.

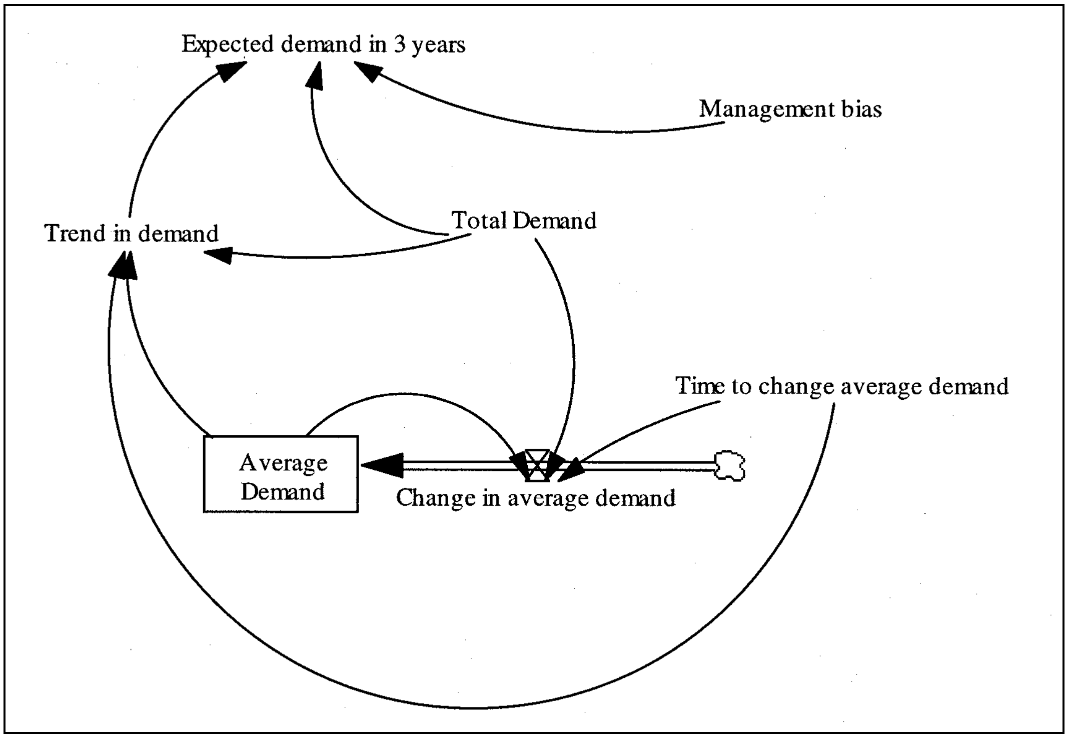


FIGURE 4 The structure of expectation formation.

2. Competitors now form an opinion on how much capacity they believe is going to be available in 3 years from the present time. This is done by adding to the total current capacity available the amount of capacity known to be under construction at present, and subtracting expected capacity to be retired over the next 3 years. This simple accounting mechanism is shown in Figure 5.

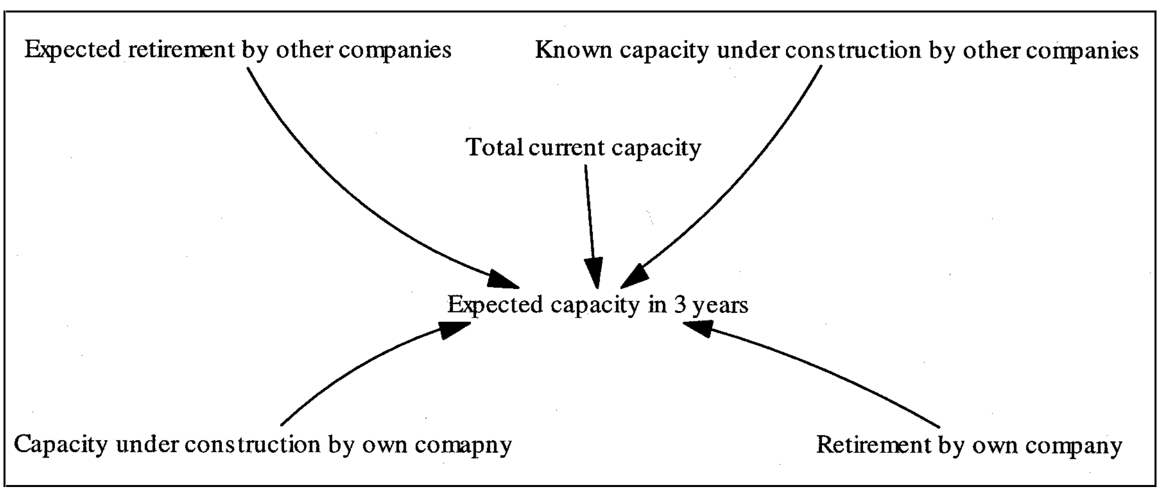


FIGURE 5 The formulation of expected capacity in the model.

3. After estimating the expected demand/supply balance over a 3-year time horizon (Steps 1 and 2), competitors (both IPP and MSB) now evaluate the effect of new investments on the expected electricity price over the same 3-year time horizon. The model provides the companies with an opportunity to choose among four different investments differentiated by size (500 MW, 1,000 MW, 1,500 MW, and 2,000 MW). A company is supposed to calculate the expected change in price as a function of the amount of generation capacity to be built, i.e., if the company made a 500 MW investment (assuming nobody else in the sector made any investments), a 1,000 MW investment, and so on. In this way the companies generate alternative forecasts of the electricity price in 3 years' time under four different investment scenarios (as well as the no-investment scenario calculated in Steps 1 and 2). The generic structure of this process that leads to the formation of an expected market price for electricity is shown in Figure 6.

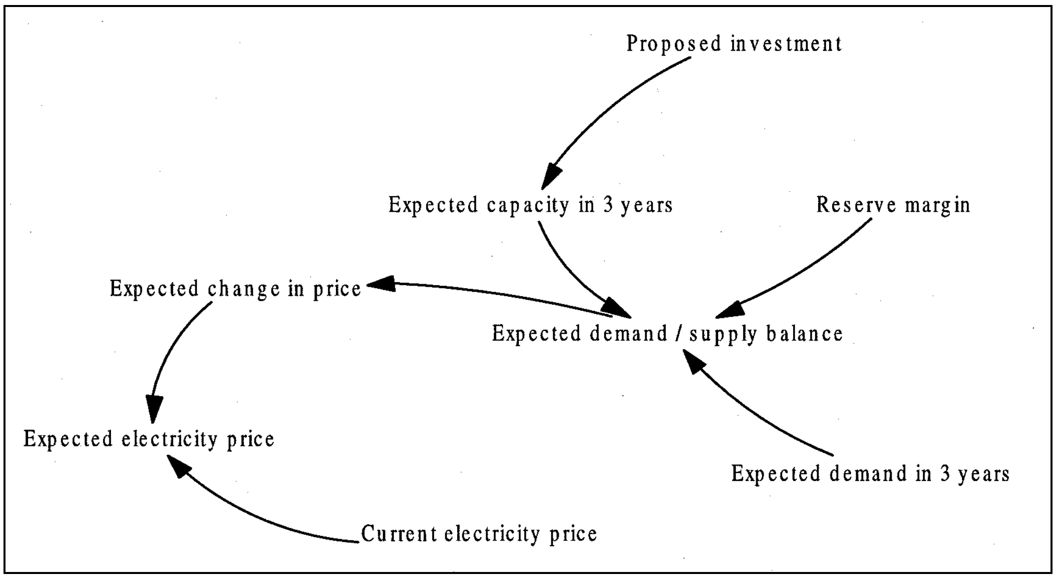


FIGURE 6 The generic structure of calculating the expected electricity price given a new investment.

4. Each of the two types of competitors will then calculate the minimum electricity price needed to obtain an acceptable rate of return on their investments in new capacity. This calculation is based on the cost of new capacity, the expected economic life of new capacity, and the economic rate of return required by the company before it is willing to make an investment.

   There is an important point of difference in investment behavior between independent power producers (IPPs), who have a fixed rate of return throughout the whole period, and the market share builders (MSB). MSB companies are willing to trade off a lower return on their investment for the possibility of building up a stronger presence in a market that they consider attractive (for example, because they expect further liberalization or because they see opportunities for growth that domestic firms are not well equipped to seize). When MSBs reach their desired market, they will start to behave like other IPPs and invest only when it is strictly economically rational to do so in order to maintain their market share.

The rationale for including in the model these two different types of competitive behavior is based on both historically observable facts and assumptions about expected patterns of competition that might emerge in a single European market for electricity. In the first half of the ’90s, U.K. large buyers — mainly regional electricity companies (RECs) — encouraged new IPPs to enter the market in a variety of different ways. One was providing long-term contracts to new entrants (up to 10-12 years), thus removing the risk for the IPP of not being able to sell electricity at a profitable rate. These incentives implied some degree of risk-sharing between RECs and new IPPs and translated into a lower expected rate of return for the latter. Another way in which RECs encouraged new entries was by making direct investments in new IPPs. Above and beyond these historical facts, it is also reasonable to expect that companies in other European countries might be willing to “pay a price” (in terms of lower returns) if they can see a long-term strategic benefit of being in a given market. Recent examples of this competitive behavior include the $1.45 billion tender offer by Endesa of Spain to control Enersis in Chile and build a solid platform for expansion in South America; Reliant Energy’s $2.4 billion acquisition of Energieproduktiebedrijf UNA (the largest power company in the Netherlands), and the $3.2 billion acquisition of London Electricity by Electricite de France.

5. The last step involves the comparison between the required rate of return for investments to be made and the expected return calculated in Step 3. Each type of company will look at its expected rate of return on new investments computed in Step 3 and compare the result with the required rate of return computed in Step 4. It is assumed that a company will like to invest as much as possible given that the expected return meets the required rate of return. In other words, each company will choose the largest possible investment that meets the requirements. First, the possibility of a 2,000-MW investment will be considered, then, if the rate of return is not met, a 1,500-MW investment will be considered, and so on. If none of the possible investments meets the required rate of return criterion, the company will make no investment that year. When the investment decision is in fact made, the amount of new capacity approved will appear as capacity under construction the following year.

REFERENCES

Barley, S. (1986) Technology as an occasion for structuring: Evidence from observations of CT scanners and the social order of radiology departments. *Administrative Science Quarterly*, **31**(1), 78-108.

Bitran, E. and P. Serra (1997) Regulation of privatised utilities: Lessons from the Chilean experience. In: *Privatisation of utilities and infrastructure: Methods and constraints*. OECD Proceedings, 25-50.

Bunn, D.W. (1996) Electricity restructuring and market-based pricing in the U.K. electricity industry during 1990-1995. In: *The British privatisation experiment: 5 years later*. Electric Power Research Institute, TR-106528, Palo Alto, California.

Bunn, D.W. and E.R. Larsen. eds. (1997) *Systems Modelling for Energy Policy*. Wiley, London.

Central European Economic Review (1998) Tractbel’s Kazak Gambit. **VI**(9), 18-20.

Cohen, W.M. and D.A. Levinthal (1990) Absorptive capacity: a new perspective on learning and innovation. *Administrative Science Quarterly*, **35**, 128-152.

Currie, D. (1997) Challenges and opportunities in European utility markets. *Business Strategy Review*, **8**, 1-6.

Cyert, R.M. and J.G. March (1963) *A behavioral theory of the firm*. Prentice-Hall, Englewood Cliffs, New Jersey.

de Geus, A. (1988) Planning as learning. *Harvard Business Review*, March-April, 70-74.

de Geus, A. (1997) *The living company*. Harvard Business School Press, Boston, Massachusetts.

DiMaggio, P.J. and W.W. Powell (1983) The iron case revisited: Institutional isomorphism and collective rationality in organizational fields. *American Sociological Review*, **48**, 147-160.

Diehl, E.W. (1992) Participatory simulation software for managers: The design philosophy behind MicroWorld Creator. *European Journal of Operational Research*, **59**(1), 210-215.

Energie Verwertungsagentur (1996) Organisation und regulierung “netzgebundener” branchen am beispiel der elektrizitätswirtschaft. Bundesministerium für Wissenschaft, Verkehr und Kunst, Wien, Austria.

Energy Information Administration (1996) *The changing structure of the electric power industry: An update*. Office of Coal, Nuclear, Electric and Alternate Fuels, U.S. Department of Energy, Washington, DC.

Gilbert, R.J. and E.P. Kahn, eds. (1996) *International comparisons of electricity regulation.* Cambridge University Press, Cambridge.

Ginsberg, A. (1997) “New Age” strategic planning: Bridging theory and practice. *Long Range Planning*, **30**(1), 125-128.

Graham, A.K., J.D.W. Morecroft, P.M. Senge and J.D. Sterman (1992) Model-supported case studies for management education. *European Journal of Operational Research*, **59**(1), 151-166.

Grant, R.M. and C. Baden-Fuller (1995). A knowledge-based theory of inter-organizational collaboration. *Academy of Management Best Paper Proceedings*, 17-21.

Hamel, G. and C.K. Prahalad (1994) *Competing for the future.* Harvard Business School Press, Boston, Massachusetts.

Hunt, S. and G. Shuttleworth (1996) *Competition and choice in electricity*. Wiley, London.

Ingvar, D. (1985) Memory of the future: An essay on the temporal organization of conscious awareness. *Human Neurobiology*, **4**, 127-136.

Lane, D.C. (1993) With a little help from our friends: How system dynamics and soft OR can learn from each other. *System Dynamics Review*, **10**(2-3), 101-134.

Larsen, E.R. (1998) *From regulation through integration to ?: Three scenarios for the future of electricity in Europe*. Working paper. Department of Management Systems and Information, City University, London.

Larsen, E.R. and D.W. Bunn (1999) Deregulation in the electricity sector: Understanding strategic and regulatory risk. *Journal of the Operational Research Society*, **50**, 337-344.

Larsen, E.R., D.W. Bunn and C. Haxholdt (1997) *Integration and regulation in the energy sector: A Danish and European perspective*. Report, Decision Technology Center, London Business School, London.

Larsen, E.R., van A. Ackere and K. Warren (1997) The Growth of Service and the Service of Growth, *Decision Support Systems*, **19**(4), 271-287.

Linden, H.R. (1995) The revolution continues. *The Electricity Journal*. December.

Lomi, A., E.R. Larsen, and A. Ginsberg (1997) Adaptive learning in organizations: A system dynamics-based exploration. *Journal of Management*, **23**(4), 561-582.

Lounaama, P.H. and J.G. March (1987) Adaptive coordination of learning teams. *Management Science*, **33,** 107-123.

March, J.G. and H. Simon (1958) *Organizations*. Wiley, New York.

March, J.G., L.S. Spraull, and M. Tamuz (1991) Learning from samples of one or fewer. *Organizational Science*, **2**(1), 1-13.

Matlàry, J.H. (1997) *Energy policy in the European Union*. Macmillan Press, London.

Meadows, D. (1984) *Strategem I, a management training exercise on long-term economic-environment interaction*. Available from Institute for Policy and Social Science Research, Hood House, University of New Hampshire, Durham, New Hampshire 03824, U.S.A.

Meadows, D. (1989) Gaming to implement system dynamics models. *ICSD 1989*, 635-640.

Morecroft, J.D.W. (1988) System dynamics and microworlds for policymakers. *European Journal of Operational Research*, **35**(3), 301-320.

Morecroft, J.D.W. (1992) Executive knowledge, models and learning. *European Journal of Operational Research,* **59**(1), 9-27.

Morecroft, J.D.W. and K.A.J.M. van der Heijden (1992) Modelling the oil producers — capturing oil industry knowledge in a behavioural simulation model. *European Journal of Operational Research*, **59**(1), 102-122.

Morecroft, J.D.W. and B. Marsh (1997) Exploring oil market dynamics: A system dynamics model and microworld of the oil producers. In: *Systems modelling for energy policy*, D.W. Bunn and E.R. Larsen, eds., Wiley, London.

Navarro, P. (1996) Electric utilities: The argument for radical deregulation. *Harvard Business Review*, January-February, 112-125.

Nelson, R.R. and S.G. Winter (1982) *An evolutionary theory of economic change*. Belknap Press, Cambridge, Massachusetts.

Odorici, V. and A. Lomi (Forthcoming) Classifying the competition: An empirical study on the cognitive social structure of strategic groups. In: *Managerial and Organizational Cognition*, Z. Shapira and T. Lant, eds., Lawrence Erlbaum Associates, New Jersey.

Papert, S. (1980) *Mindstorms*. Basic Books, New York.

Senge, P.M. (1990a) *The fifth discipline: The art and practice of the learning organization*. Doubleday-Currency, New York.

Senge, P.M. (1990b) The leader's new work: Building learning organizations. *Sloan Management Review*, Fall, 7-23.

Senge, P.M. and J.D. Sterman (1992) Systems thinking and organizational learning: Acting locally and thinking globally in the organization of the future. *European Journal of Operational Research*, **59**(1), 137-150.

The Wall Street Journal Europe (1998) National power plant sale fuels concern. November 26, 3.

Tushman, M.L. and P. Anderson (1986) Technological discontinuities and organizational environments. *Administrative Science Quarterly*, **31**, 439-465.

Warren, K. and P. Langley (1996) The Beefeater Restaurant Microworld. SMSim Ltd.

Weiner, M., N. Nohria, A. Hichman, and H. Smith (1997) Value networks — the future of the U.S. electric utility industry. *Sloan Management Review*, **38**(4), 21-34.

MODELING OF ELECTRIC POWER MARKETS BY ADAPTIVE AGENTS

G.B. SHEBLÉ, Iowa State University*

ABSTRACT

The electric power system is unique in that the commodity travels almost at the speed of light and responds to changes faster than any command and control system can direct. The number of products is very large to manage an electric system. California markets include ten to fourteen products depending on the type of service provided to customers. Additionally, the forced outage rate of most generators is high, typically twenty to thirty percent. The resultant market prices show very abnormal behavior as plant production is curtailed or cancelled. The modeling of these effects and the adaptive agent capabilities to respond to such complex markets is correspondingly complex. This paper will discuss ways to decompose the problem to identify experiments that can be accomplished with the same data as used by real-time traders.

INTRODUCTION

Due to recent deregulation intended to bring about competition, the U.S. electrical industry is in the midst of some major operational changes. Although the details of the deregulated marketplace for each region of the country are not yet fully defined, they are being more clearly defined as time passes. Many legislators, researchers, and electric customers and suppliers are convinced that electricity will be traded in a manner similar to that of other commodities at exchanges around the country.

Configuration of the transmission system and the fact that electricity flow is subject to the laws of physics have some speculating that we will see the formation of regional commodity exchanges that would be oligopolistic in nature (having a limited numbers of sellers). Others postulate that the number of sellers will be sufficient to have near-perfect competition. Regardless of the actual level of the resulting competition, companies wishing to survive in the deregulated marketplace must change the way they do business and will need to develop bidding strategies for trading electricity via an exchange.

Economists have developed theoretical results of how markets are supposed to behave under varying numbers of sellers or buyers with varying degrees of competition. Often the economic results pertain only when aggregating across an entire industry and require assumptions that may not be realistic. These results, while considered sound in a macroscopic sense, may be less than helpful to a particular company not fitting the industry profile that is trying to develop a strategy that will allow it to remain competitive.

---

* *Corresponding author address:* Gerald B. Sheblé, Professor of Electrical and Computer Engineering, Director of Complex Adaptive Systems, 1129 Coover Hall, Iowa State University, Ames, IA 50011; e-mail: gsheble@iastate.edu.

Generation companies (GENCOs) and energy service companies (ESCOs) that participate in an energy commodity exchange must learn to place effective bids in order to win energy contracts.

Microeconomic theory states that in the long term, a hypothetical firm selling in a competitive market should price its product at its marginal cost of production. The theory is normally based on several assumptions (e.g., all market players will behave rationally, all market players have perfect information) that may tend to be true industry-wide, but might not be true for a particular region or a particular firm. One of the goals of this research [29] is to determine the impact of market power, market share, and market rules.

THE MARKETPLACE

The basic framework for the research described in this paper is adopted from Sheblé [14, 15, and 20], which is an extension to the framework being proposed in California. Sheblé [21] described the different types of commodity markets and their operation. He outlined how each could be applied in the evolved electric energy marketplace. Under this framework (shown in Figure 1, which was presented in Sheblé et al. [24]) companies presently having generation, transmission, and distribution facilities would be divided into separate profit and loss centers. Power would be generated by GENCOs and transported via transmission companies (TRANSCOs). ESCOs would purchase the power from the generator for the customer. It has been proposed that NERC would set the reliability and security standards. It is predicted that we'll see ESCOs replacing the current distribution utilities as the main customer representatives. An Independent Contract Administrator (ICA) will review the power transactions to ensure that system security and integrity are maintained. Distribution companies would own and maintain the distribution facilities. Companies providing energy mercantile associations (EMAs) have emerged in this new framework.

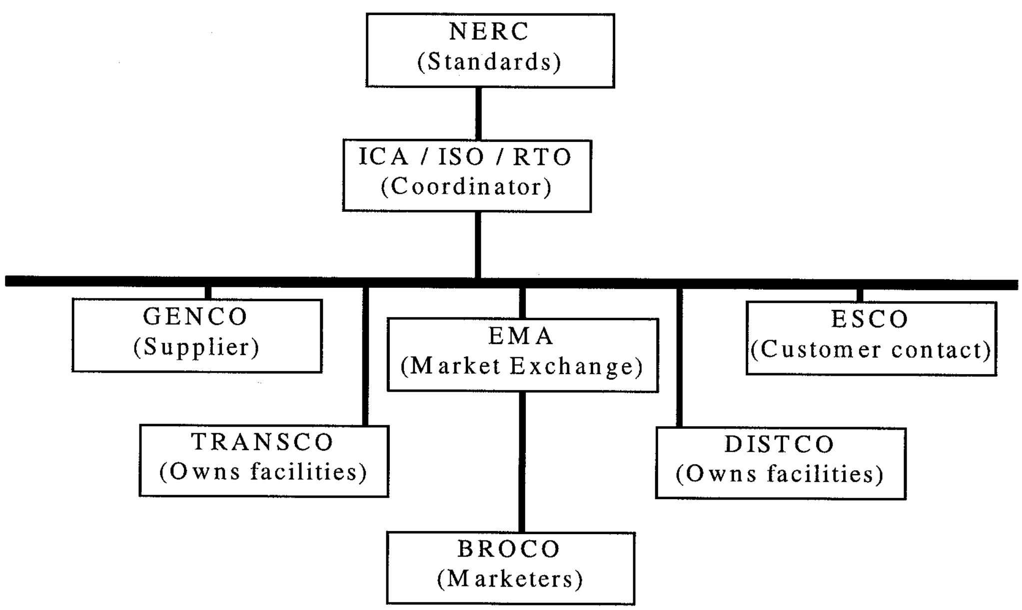


FIGURE 1 Brokerage system model.

In the double auction used for this research, the bids and offers are sorted into descending and ascending order, respectively, similar to the Florida Coordination Group approach described by Wood and Wollenberg [27]. If the buy bid is higher than the sell offer that is to be matched, this is a potential valid match. The ICA must determine whether the transaction would endanger system security and whether transmission capacity exists. Specifically, the contract approval is subject to meeting requirements for maintaining sufficient spinning reserve, ready reserve, reactive support, and area network control (contract-based AGC). If the ICA does approve, the valid offers and bids are matched, and the difference in the bids ($/megawatt) is split to determine the final price, termed the *equilibrium price.* This is similar to the power pool split-savings approach that many regions have been using for years.

If there are an insufficient number of valid matches, then *price discovery* has not occurred. The auctioneer reports the results of the auction to the market participants. If all bids and offers are collected, and the valid bids and offers are found to be insufficient, the auction has gone through one cycle. The auctioneer then reports that price discovery did not occur and asks for bids and offers again. The auctioneer requests that the buyers and sellers adjust their bids and offers. To aid in eventually finding a feasible solution, during subsequent cycles within a round, buyers may not decrease their bids, and sellers may not increase their offers. The cycles continue until price discovery occurs, or until the auctioneer decides to bind whatever valid matches exist and continue to the next round or hour of bidding.

After price discovery, those buyers and sellers whose bids were bound potentially have a contract. This contract is subject to the approval of the ICA, which verifies that none of the security criteria have been violated. Following the completion of one round of bidding, the auctioneer asks if another round of bidding is requested. If the market participants have more power to sell or buy, they request another round. Allowing multiple rounds of bidding each hour (versus one-shot bidding) allows the participants the opportunity to use the latest pricing information in forming their present bid. This process is continued until no more requests are received or until the auctioneer decides that enough rounds have taken place. See Figure 2 for a block diagram of the auction process.

EVOLVING BIDDING STRATEGIES WITH GENETIC ALGORITHMS

A genetic algorithm (GA) is an algorithm that allows evolution of the contents of a data structure. GAs were developed by John Holland and are loosely based on the biological notion of evolution. The data structure being evolved contains a solution to the problem being studied. A population of syntactically valid solutions is initialized randomly during the first step of the algorithm. Each of the solutions is assigned a fitness based on its suitability for solving the particular problem being studied. If these solutions are initialized randomly, their chances of being highly *fit* during the first *generation* are not very high. At each generation, the GA randomly chooses members of the population to be “parents” favoring the highly fit members. The parents then produce offspring via the *crossover* and *mutation* processes. Crossover is the means by which two parents produce two offspring and involves combining parts of each parent to produce each child. Mutation can be thought of as copying errors introduced into the children due to background noise. The newly produced offspring replace the members of the population that have low fitness. As the generations progress, there is a tendency for the contents of the data structures to adapt so that they become more suited to solving the problem. See Goldberg [12] for a more complete description of genetic algorithms.

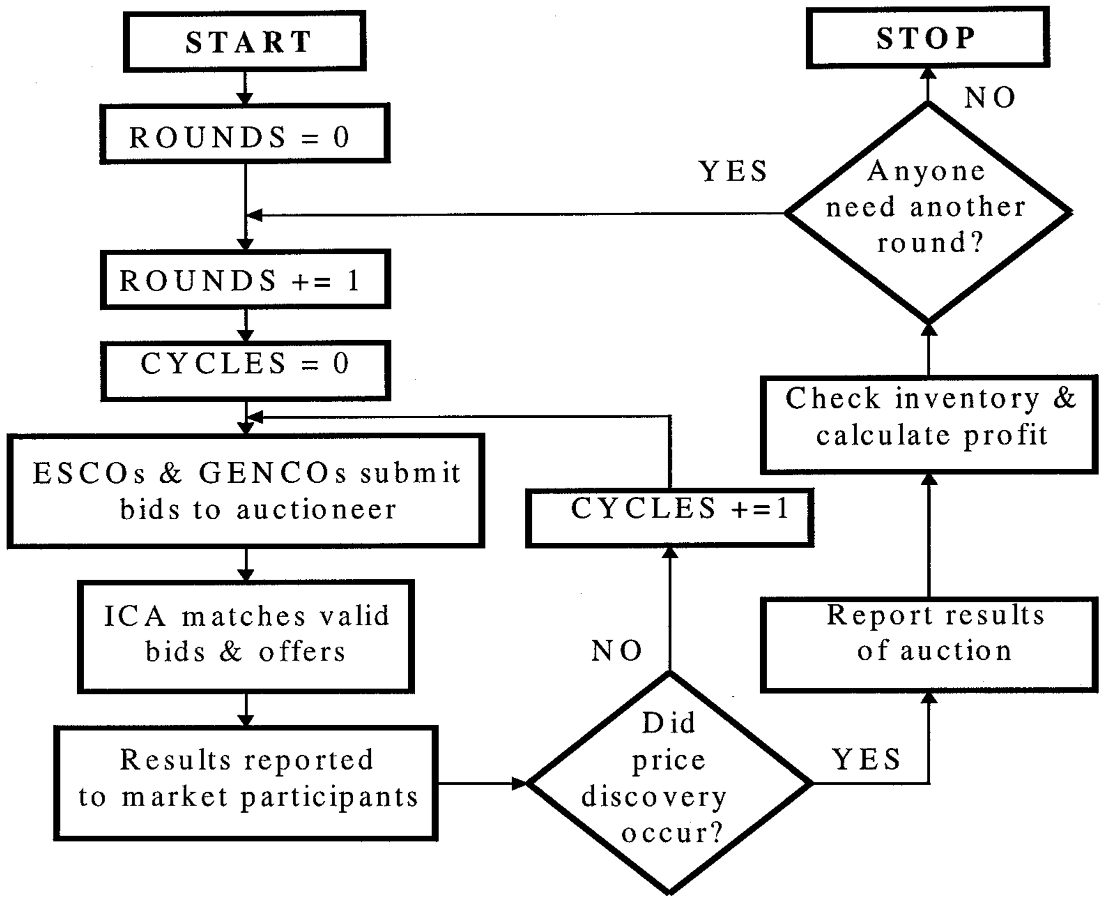


FIGURE 2 Block diagram of the auction process.

In Richter and Sheblé [5], the authors use a GA to evolve a structure containing bid multipliers. Others have used GAs for computational economics [18, 25]. The bidding strategies that come from the evolved structures (shown in Table 1) are fairly simple. The bid multipliers multiply the expected price of the electricity (obtained via some prediction scheme), and the result is used as the bid for that round of bidding. In addition to the bid multipliers, the number of MWs to offer for sale at each round of bidding and the choice of price prediction techniques are also evolved.

The results presented in [5] are promising. As the GA progresses, the bidding strategies become better and yield more profit, indicating that "intelligent agents" are learning. However, the strategies are somewhat limited because they do not make use of inputs that are available during a particular round of bidding. Evolving bidding strategies as in [5] is like learning the steps of a dance or memorizing a list of things to do mechanically in order to make a successful bid for a particular set of circumstances. Using the approach in [5] means that the evolved rules are not very adaptive, i.e., they don't react to the environment. Each set of rules is evolved to be used only for a specific set of circumstances. If the circumstances vary from that, the set of rules may yield disappointing results. We could attempt to create scenarios in which we are interested, but we would find that the number of credible scenarios is so large that we could not possibly

TABLE 1 Data Structure Used in Previous Research

Agent N	Rounds of Bidding ------------------------>				
MWs each round	12	4	20		14
Quantity each round	01011	01101	10101	. . .	00101
Prediction Technique	Moving average, exponential, autoregressive, artificial neural network, log regression				

hope to cover them all. So the question becomes how can we develop adaptive bidding strategies that take advantage of currently available information?

BIDDING

Lotfi Zadeh brought the use of "fuzzy logic" forward during the 1960s. Fuzzy logic provides a methodical means of dealing with uncertainty and ambiguity. It allows its users to code problem solutions with a natural language syntax with which people are comfortable. In fact, many of us regularly use fuzzy terms to describe things or events. For instance, if we were asked to describe a person, we might use terms like "pretty tall," with a "big nose" and "somewhat overweight." These terms can be defined differently by different people. There is a certain amount of ambiguity or uncertainty associated with any description involving natural language terms such as these. Most of the things we deal with daily in this universe are ambiguous and uncertain. "The only subsets of the universe that are not in principle fuzzy are the constructs of classical mathematics." [28]

Fuzzy logic allows us to represent the ambiguous or uncertain with membership functions. The membership functions map the natural language descriptions onto a numerical value. Membership to a particular description or class is then a matter of degree. For instance, if we define a person's height as described in Figure 3, we can see that a person that is 6 feet in height is tall with a membership value of one. This membership value is also known as a truth-value. From the same figure, we can see that a person who is 5 feet 9 inches is tall to a lesser degree and but, at the same time, he/she is also short to a certain degree.

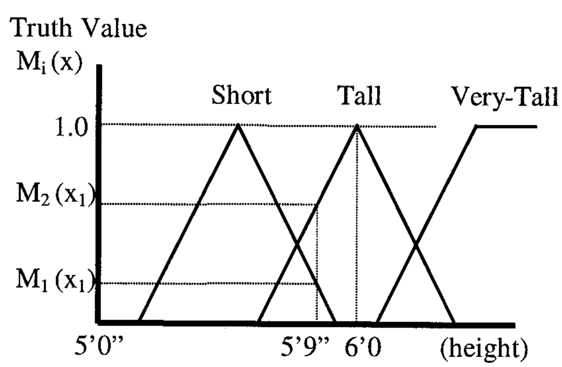


FIGURE 3 Fuzzy membership functions.

Using similar reasoning, we might say that electrical demand is high in a region if it goes above 100 MW and normal if it is between 50 MW and 75 MW. What if the demand is 90 MW? Using traditional logic, we would classify it as neither high nor normal. However, using fuzzy logic, we might find that this demand is actually both high and normal, each to a certain degree (based on its membership function). Similarly we could have fuzzy membership functions for other inputs like fuel costs, risk aversion, level of competition, etc.

Once defined, these inputs can then be used in a set of fuzzy rules. For instance, a simple rule might be as follows:

IF demand is HIGH, then bid should be HIGH

Where a “high” bid would be defined using another membership function. Multiple input conditions can be considered by combining rules with the “and’ and “or” functions. For example a rule might be as follows:

IF (demand is LOW) AND (risk aversion is HIGH) THEN (bid should be LOW)

Although it may not be necessary, we could have an output for all combinations of inputs. A three-input fuzzy rule system, in which each input is broken into five classifications, might be represented as in Figure 4. The small squares each contain the output of a rule on how to bid relative to cost. Because some conditions might be very unlikely to occur, some of these squares may not have an output. In addition, a particular input maybe classified in more than one square at a given instant. In the figure, the letters V, L, H, C, and N stand for very, low, high, cost, and normal respectively. The output of the rule states how to bid with respect to generation cost. We could have more or fewer inputs, and we could use different classifications.

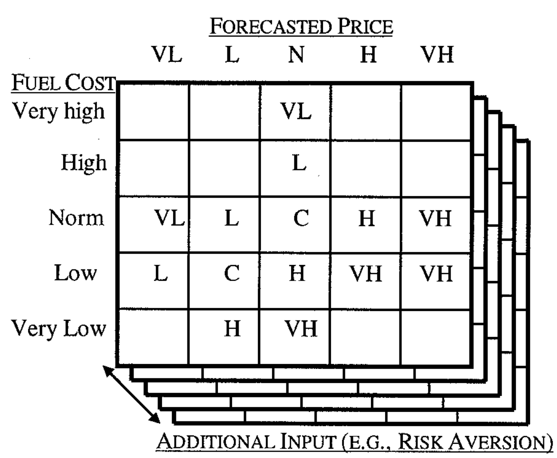


FIGURE 4 Three input fuzzy rule set.

The architecture of the bidding process is essentially a knowledge-based system (KBS). The inputs are fed into the rule base. The output (i.e., the bid values in the example) of each rule can be classified by a fuzzy membership function in the same manner as the inputs. The output of each rule may be assigned a certain weight depending on how important we determine that rule or corresponding inputs to be. We can then sum the weighted output of the rules and determine an overall fuzzy output. However, when the time comes to place the bid, we can’t just say, “bid high.” We need a way to convert the fuzzy output to a single number. This is called the defuzzification process.

According to Kosko [28], defuzzification formally means to round off a fuzzy set from some point in a unit hypercube to the nearest bit-vector vertex. Practically, defuzzification has been done by using the mode of the distribution of outputs as the crisp output, or by the more popular method of calculating the centroid or center of mass of the outputs and using that as the crisp output.

COMPARING BIDDING STRATEGIES

This section provides a comparison of approaches that we are taking in developing bidding strategies. First, we will be generating fuzzy bidding rules manually using expert knowledge. Secondly, we will search for good rule-set parameters from a limited search space. With a small number of inputs and a limited number of weighting, we can do an exhaustive search of all rules and determine the best possible rule. (The best rule is the one whose use results in the largest amount of profit for its user.) Thirdly, we note that if we increase the number of fuzzy inputs, increase the number of membership functions describing the inputs, and allow more flexibility with the weighting, it may become desirable to use a genetic algorithm to search for the “optimal” rule parameters, rather than do an exhaustive search. Finally, we will attempt the use of a technique developed in [7] to extract, from a historical database containing the bidding details of an auction, the rules that were used by others in developing their bids.

The research described here builds on the techniques used by the author and described in [5]. To measure the performance of the bidding rules created in each of the methods described below, a group of GENCOs will compete to serve the electrical demands of the ESCOs. Transmission constraints are not being considered directly here, but can be accounted for after the fact if desired. The essential problem to be solved is what bid should be made given the history of the other players, of other markets, and of perceived market trends.

Generating the Rule Sets Manually

If we consider only a limited number of fuzzy economical inputs, (e.g., expected price, risk aversion, and generating costs), then it is possible to generate rules manually with expert knowledge from power traders. We can transform the rules of thumb used by experienced power traders into a fuzzy rule base. We may also use theoretical economics to influence the rule sets that we construct. If we have 3 fuzzy inputs, each divided into 5 classifications, we could have need for as many as 125 rules in each rule set (one for each little square in Figure 5). Each of the rules can be weighted according to its importance; if any weighting is allowed, we have infinite possibilities.

Search for the “Optimal” Rule Set

To reduce the amount of time spent tuning the rule sets, we can predefine a structure and allow a computer program to search through the possibilities to find the optimal rule set. If we predefine each of the three inputs by five fixed ranges, and only allow discrete rule weightings (e.g., 0.0, 0.1, 0.2, 1.0), then there are a finite number of permutations to investigate. A possible indication of optimality would be obtained by having an agent use each of the possible rule sets while engaging in a fixed set of trial auctions competing with a set of agents that had evolved to play the market described in [5]. To ensure that the rules aren't market specific, the set of agents against which the rule will be competing can be taken from different populations and from various stages of evolution. This increases the certainty that the tested rule will be profitable against a diverse set of agents and circumstances.

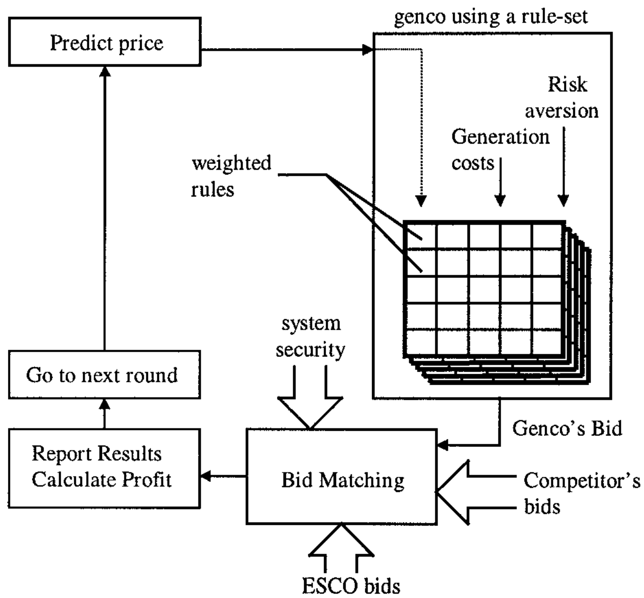


FIGURE 5 Using the rule set.

Using a GA to Evolve Rules for Bidding

If we relax the requirement that each rule have discrete weighting, we can see that the size of the search space becomes quite large. If we also increase the number of inputs to consider, the search space grows even larger. The exhaustive search no longer remains feasible. In addition, if we do not wisely chose the set of agents against which our rules will be tested, we would be left with rules that are not extremely robust. Therefore, the authors plan to use a GA to evolve rule sets in a fashion similar to [5], but with slightly modified data structures.

Each of the GENCOs will have its own evolving data structure consisting of a fuzzy set of rules and weights associated with each of those rules. The weights will allow some rules to have more importance than others. In previous work, the authors allowed each of the individual GENCOs to have their choice of price forecasting techniques. This created a lot of overhead, and for simplicity, current research will have each GENCO receiving globally forecasted data. In addition, the contract size (i.e., number of megawatts to offer) at each round of bidding would be fixed rather than evolvable to reduce the search space.

Using a GA to Extract Expert-System Bidding Rules from a Historical Database

The authors have investigated the use of GAs and other so-called artificial intelligence techniques to search through large databases in order to learn the expert system rules that can be used to reproduce the historical results. Presently this technique is being used to develop standardized treatment methods for hospital patients receiving medical care. Based on extensive records, the software is able to determine what the doctor did based on patient conditions. Similarly, a database of trading data could be fed into the software (which would require tuning and some restructuring) to estimate what bidding rules the traders were using. Determining the rules that other electricity traders and brokers are using could be of great benefit to those who wish to gain a competitive edge when participating in the deregulated market.

ALTERNATIVE AUCTION MECHANISMS

Auctions are considered to be a good pricing mechanism for competitive markets. There have been various auction structures proposed for electric power markets. The major types of auctions for electric power can be classified into centralized daily commitment auctions (CDCAs) and single-period commodity auctions (SPCAs). An example of a CDCA is the power pool auction as implemented in the United Kingdom and some portions of the United States [30]. An example of an SPCA is the electric power auctions used in New Zealand [31]. The more detailed characteristics of CDCAs and SPCAs are described in section three.

The methods for matching bids in auctions are based on optimization techniques. Various auction structures may be implemented properly and efficiently with different optimization techniques. The power pool auction (CDCA) in the United Kingdom is implemented by LaGrangian relaxation (LR) while the SPCA in New Zealand is implemented by the advanced dual simplex and interior-point methods. LR, interior-point linear programming (IPLP), and upperbounded linear programming (UBLP) have been programmed to implement various types of auctions [32] for this research. Supportive services such as loss coverage, spinning reserve, voltage control, frequency control, and load following control have been included. Ongoing research is including contingent contracts and strictness of guarantee for reliability services.

The theme of this work is to illustrate various mathematical formulations for auctions when different auction structures having various characteristics are desired. The focus of this work is on SPCAs and thus the formulations illustrated are for SPCAs. However, the formulations illustrated can be enhanced easily to be used with CDCAs. The formulations of a few cases have been presented in previous research [33-36]. However, this work presents formulations for many more comprehensive cases. In addition, this work clearly separates the cases based on several criteria and thus the formulations can be easily modified for other cases that are not presented in this work.

CDCAs and SPCAs

Each of the major types of auctions for electric power — CDCAs and SPCAs — can be subclassified by various criteria. The first sub-classification divides the auctions into single-sided and double-sided cases. Single-sided auctions allow only GENCOs to bid, while double-sided auctions allow both GENCOs and ESCOs to submit bids. A second sub-classification is that of uniform and discriminating pricing. Uniform pricing means every seller gets paid the same price and every buyer pays the same price; discriminating pricing means each seller gets paid and each buyer pays corresponding to their bids. GENCOs and ESCOs can be sellers or buyers in double-sided auctions. For single-sided auctions, GENCOs are sellers. Several other criteria can be used to classify auctions. For example, trading via bilateral contracts or exchanges, with or without reservation prices, and homogeneity or heterogeneity of electric power can each be used to classify the type of auction. These criteria will be explained in more detail in the next section.

CDCAs

An example of a CDCA is the power pool auction as implemented in the United Kingdom and some portions of the United States. Because the bids are submitted to the authority controlling the power system, this work assumes that they are submitted to the ICA (i.e., ISO in California). For single-sided auctions, GENCOs submit their generation cost models to an ICA, and ESCOs submit their hourly loads to an ICA. Then the ICA performs a unit commitment (UC) analysis using LR for the system for a specified period (e.g., 24 or 168 hours). After the ICA finds the optimal solution, the optimal schedule is reported to each GENCO, and the optimal cost is reported to each ESCO. In a double-sided auction, ESCOs are also allowed to bid for power by submitting a set of pseudo-unit parameters to the ICA. This work focuses on single-sided CDCA. The detail of the double-sided CDCA can be seen in [32]. Indeed, even though ESCOs do not represent generating units explicitly, they would gain advantage by proper manipulation of pseudo-unit parameters in the CDCA. An ESCO's pseudo-unit parameters are found by constructing an equivalent unit to achieve a specified revenue function. The details of the revenue functions of ESCOs are described in [32].

SPCAs

The SPCA is an auction commonly used for commodity exchange. The auction is to provide power for a single period, which can be any length of time (e.g., 15 minutes, half an hour, or one hour). However, the SPCA can be enhanced for use with auctions for multiple time periods (e.g., one day). In this research, the SPCA is used for electric power transactions. For the single-sided SPCA, GENCOs submit offers and ESCOs submit their incremental loads to the ICA. The bid is basically composed of a price and an amount for energy. The bid may have other components (e.g., spinning reserve, ready reserve) if these components are bundled. The ICA finds the optimal match of the bids by minimizing the total cost. For double-sided SPCA, ESCOs are allowed to submit offers or bids for selling or buying power. This work focuses on double-sided SPCAs. Details of the single-sided SPCA can be seen in [32]. Although ESCOs do not have explicit cost functions to calculate bid parameters like GENCOs, ESCOs do build revenue models and use them for calculating bid parameters. Even though they do not represent generating units explicitly, ESCOs would gain advantage by proper manipulation bid parameters in the SPCA.

AUCTIONS AS AN ASSIGNMENT PROBLEM

An auction can be viewed as the assignment of products from sellers to buyers. This is why it is more appropriate to treat an auction as an assignment problem. The term "assignment problem" used here is in the context of assigning products from sellers to buyers. The term "assignment problem" is different from that used in the context of the minimal cost network flow problem in most textbooks [33, 37, and 38]. In the context of most textbooks, the assignment problem is a special class of the minimal cost network flow problem, since the solution procedure and not the type of application is explained. As a solution procedure, the assignment problem has a particular structure for a special method to solve a unique topographic tableau. The minimal cost network flow problem is a special type of linear programming problem that has unique network structures that severely modify the application of the optimization rules. Such assignment problems are a special type of the transportation problem, which is a special type of the minimal cost network flow problem. The minimal cost network flow, the transportation, and the assignment problems are referred to in this section because they have a problem structure similar to that of auction problems. The transportation problem and the assignment problem are separated from the minimal cost network flow problem according to their special structures so that special methods can be applied to solve the problems. The network simplex method has been applied to solve the minimal cost network flow problem, and the transportation simplex method is applied to solve the transportation problem [37, 38]. The Hungarian algorithm has been applied to solve the assignment problem [38]. The network simplex method, the transportation simplex method, and the Hungarian algorithm are special versions of the simplex method. For this work, it is not appropriate to focus on these special methods to solve the auction problem. The general auction problem is formulated without regard to special equation structures and, thus, it can be solved by general simplex method.

Assignment Problems

Two major types of products are considered: heterogeneous products and homogeneous products. Homogeneous products are indistinguishable from each other, while quality or characteristics can distinguish heterogeneous products. Because heterogeneous products are distinguishable, they have different value to each seller and each buyer, while homogeneous products have the same value to each seller and each buyer. Products can be traded through an exchange or traded by individuals via bilateral contracts. Trading through an exchange is more convenient for traders because the exchange gathers different types of products together, which means that traders do not waste time finding the products they desire. In addition, an exchange provides insurance to protect parties from sellers or buyers who default. For example, sellers who do not supply products according to the contracts will be fined through the exchange. Then, the exchange can distribute the money to participants or use the fine to provide products from other sellers to the buyers for compensation. Regardless of whether products are traded through the exchange, the parties to the transactions remain identifiable.

In trading heterogeneous or homogeneous products as a bilateral contract, transaction x_{ij} is defined as from seller i to buyer j. For heterogeneous products, products from different sellers have different unit values to each customer: i.e., x_{1j}, x_{2j}, x_{3j}, ..., x_{mj} have different unit value to buyer j. This is the case when buyers can distinguish products of different sellers. For example, electricity produced from one GENCO has higher power quality than electricity from other GENCOs. Another example involves a buyer who is concerned about environment values; the electricity produced from clean energy has more value than the electricity produced from the energy that pollutes the environment. Another property of heterogeneous products is that products may vary from seller to seller. The more general case when products from a seller are different is considered. Specifically, each buyer has a different unit price from each seller. An interesting example of this case is price discrimination. One prevalent example of price discrimination in electricity is when a seller prices electricity corresponding to a guaranteed level of reliability. For homogeneous products, products from any seller have the same unit value to each customer. In other words, x;j is the same for all i from 1 to m and all j from 1 to n.

In trading homogeneous products through an exchange, sellers sell products to the exchange and buyers buy products from the exchange. In trading heterogeneous products through two exchanges, there are two classes of exchanges, Exchange a and Exchange b, which are classified according to types of products. Many classes of exchanges can be added. In addition, when classes are provided for pairs of every seller and buyer (the number of classes is $m*n$), trading through the exchange is equivalent to trading through bilateral contracts. The difference is one of convenience, as explained above. However, each class of exchanges is usually provided for a group of products, so the number of classes is usually less than $m*n$. In addition, when the properties used to separate classes of exchanges are continuous quantities, they are usually discretized. For example, reliability is discretized when used to separate classes of exchanges, since reliability is measured in continuous quantities; e.g., six classes of exchanges are provided, which have reliabilities of 0.7, 0.75, 0.80, 0.85, 0.90, 0.95. Three classes of exchanges may be provided instead, which have reliabilities of 0.7, 0.8, 0.9. This results when the reliability levels 0.7, 0.75 are grouped together, and so are 0.8, 0.85, and 0.9, 0.95.

Formulation of Assignment Problems

This section formulates the assignment problems for heterogeneous and homogeneous products. Both primal and dual problems are shown. The dual problems are useful for the analysis in many aspects; especially that they show the relationship between the dual prices and bid prices. The formulations are for finding the partial equilibrium. For heterogeneous products, formulations for trading without or with the exchange are different except when the number of exchange classes is equal to $m*n$. For homogeneous products, formulations for trading without or with the exchange are the same. The formulations are classified in different cases. One criterion used for classification is based on the parties who specify prices. This criterion breaks the assignment problems into three cases: (a) sellers specify prices, (b) only buyers specify prices, and (c) both sellers and buyers specify prices. Sellers and buyers only know their own prices of other sellers and buyers. This is a sealed bid auction.

The effects of reservation prices are also considered. The reservation price of a seller is the lowest price at which the seller is willing to sell, and the reservation price of a buyer is the highest price at which the buyer is willing to buy. Reservation prices are considered when only sellers or buyers specify prices to ensure that parties who do not specify prices get the products at acceptable prices. Buyers include reservation prices when only sellers specify prices, and sellers include reservation prices when only buyers specify prices. When both sellers and buyers specify prices, reservation prices are not needed because both parties can specify the prices according to their willingness. Not only are the reservation prices considered to ensure that parties get products at acceptable prices, but they are also useful for avoiding the degeneracy problem, which will be explained in the next section. Formulations for all cases are classified in Table 2. Cases 1 to 5 belong to heterogeneous products when they are traded via bilateral contracts. Cases 5 to 10 belong to homogeneous products, and the formulations are applicable to when products are traded through either bilateral contracts or an exchange. Case 11 is when heterogeneous products are traded via exchanges.

Presently, in many electric power auctions, electric power is treated as a homogeneous product and is auctioned in the exchange. The ICA performs bid matching. In assignment problems, cases where only sellers specify prices are similar to single-sided auctions in which only GENCOs submit the bids. Cases where both sellers and buyers specify prices are similar to double-sided auctions in which both GENCOs and ESCOs submit the bids. The formulations of assignment problems can be applied to electric power auctions. All the formulations shown in Table 2 can be applied to electric power auctions when different auction frameworks are needed. The complete formulation for each auction structure can be acquired by adding additional constraints to the formulations. Examples of additional constraints are power flow constraints and transmission line flow limits.

In the present case — with electric power treated as a homogeneous product — formulations in cases 6 to 10 can be applied to when electric power is traded via either bilateral contracts or auctions. Cases 6 and 7 are one-sided in which prices are specified by sellers, and cases 8 and 9 are one-sided in which prices are specified by buyers. Case 10 is double-sided. If electric power is considered as a heterogeneous product, formulations in cases 1 to 5 and in case 11 can be applied. Cases 1 to 5 are when electric power is traded via bilateral contracts, and case 11 is when electric power is traded via auctions. Note that this work considers electricity as a heterogeneous product in the case when electric power has different reliability levels. This work does not consider other cases of heterogeneous electric power because of complexity in transmission. The auction formulations of 11 cases are summarized and differentiated (according to the criteria mentioned) in Table 2. It should be noted that the number of markets implemented depends on the services offered. Reliability requires that multiple markets be emulated, one per level of reliability required by ESCOs, to assemble a portfolio of contracts to achieve the strictness of guarantee required by contracts with customers. The GENCOs have to bid on the various markets to sell their product based on the price of each market as well as the availability of the generation equipment. Speculators can play the various markets to maximize profit based on the mismatch between the reliability required by the ESCOs and the availability of the GENCOs. The result is that there are three types of players, each maximizing a portfolio subject to different optimal criterion. The most recent work is to determine whether players can manipulate such complex markets to exercise market power or simply to disrupt the markets, causing economic chaos.

TABLE 2 Summary of All Cases

	Products	Bilateral/ Exchange	Price By	Reservation Price
1	Heterogeneous	Bilateral	Sellers	Without
2	Heterogeneous	Bilateral	Sellers	With
3	Heterogeneous	Bilateral	Buyers	Without
4	Heterogeneous	Bilateral	Buyers	With
5	Heterogeneous	Bilateral	Both	Without
6	Homogeneous	Either one	Sellers	Without
7	Homogeneous	Either one	Sellers	With
8	Homogeneous	Either one	Buyers	Without
9	Homogeneous	Either one	Buyers	With
10	Homogeneous	Either one	Both	Without
11	Heterogeneous	Exchange	Both	Without

The notation of symbols used in the formulation is:

c_{sij} price specified by seller *i* to buyer *j* for a heterogeneous product
c_{bij} price specified by buyer *j* to seller *i* for a heterogeneous product
c_{si} price specified by seller *i* for a homogeneous product
c_{bj} price specified by buyer *j* for a homogeneous product
$c_{si,h}$ price specified by seller *i* for a heterogeneous product sold in exchange *h*
$c_{bj,h}$ price specified by buyer *j* for a heterogeneous product bought in exchange *h*
π_{si} reservation price specified by seller *i*
π_{bj} reservation price specified by buyer *j*
x_{ij} amount of a heterogeneous product sold from seller *i* to buyer *j*
x_{si} amount of a homogeneous product sold by seller *i*
x_{bj} amount of a homogeneous product bought from buyer *j*
$x_{si,h}$ amount of a heterogeneous product sold by seller *i* in exchange *h*
$x_{bj,h}$ amount of a heterogeneous product bought from buyer *j* in exchange *h*
y_{si} amount sold back of seller *i*
y_{bj} amount bought back of buyer *j*
S_i supply capacity of seller *i*
D_j potential demand of buyer *j*
u_i dual variable associated with supply constraint of seller *i*
v_j dual variable associated with demand constraint of buyer *j*
w dual variable associated with the constraint balancing supply and demand of an homogeneous product
w_h dual variable associated with the constraint balancing supply and demand of an heterogeneous product traded in exchange *h*
m number of sellers
n number of buyers
l number of exchanges

Only the formulation for case 1 is shown. The other formulations may be found in Dekrajangpetch [39].

Case l: Heterogeneous products, trading as bilateral contracts, prices specified by sellers, without reservation prices

Primal problem

$$\min_{x_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{sij} x_{ij}$$

s.t.

$$\sum_{j=1}^{n} x_{ij} \le S_i \qquad i = 1,2,3,\ldots,m$$

$$\sum_{i=1}^{m} x_{ij} \ge D_j \qquad j = 1,2,3,\ldots,n$$

$$x_{ij} \ge 0 \qquad i=1,2,3,\ldots,m \qquad j=1,2,3,\ldots,n$$

Dual problem

$$\max_{u_i, v_j} \sum_{i=1}^{m} S_i u_i + \sum_{j=1}^{n} D_j v_j$$

s.t.

$$u_i + v_j \le c_{sij} \qquad i=1,2,3,\ldots,m \qquad j=1,2,3,\ldots,n$$

$$u_i \le 0 \quad v_j \ge 0 \qquad i=1,2,3,\ldots,m \qquad j=1,2,3,\ldots,n$$

The other cases are formed similarly. The objective of altering the market formulation is to determine if the agents can adapt to the different rules now being implemented in the various power exchanges. The market models also include the various supportive (ancillary) services. There are 14 total markets in the case of the California exchange, where everything is unbundled. The Eastern PJM exchange bundles most supportive services with the energy sales. We are presently investigating the benefits of bundled or unbundled services.

SUMMARY OF MARKET RESEARCH

Building good bidding strategies for electricity traders as they move into the deregulated marketplace will continue to be important for those companies wishing to remain profitable. The author’s students have performed extensive research in this area, and this paper describes directions in which they are currently investigating in order to build more robust adaptive bidding strategies. The deregulated market structure that I assume will become standard throughout the U.S. has been defined and is incorporated into our auction simulator. The bidding rule sets or strategies obtained from each method described in this paper have been tested in auction simulations. They are presently being compared via profitability to each other and to the method of using the bid multipliers rules developed in previous work by the author. Future work extends the above to include concurrent processing by human players as well as computer-based players. The major new thrust is to include three types of players, each maximizing a portfolio subject to different optimal criterion. Multiple markets are emulated to represent the various levels of electrical delivery reliability. Financial markets, as well as fuel markets, are also being included.

The most recent work is to determine if players can manipulate such complex markets to exercise market power or simply to disrupt the markets, causing economic chaos.

REFERENCES

1. Brian R. Binger and Elizabeth Hoffman, *Microeconomics with Calculus*. Harper Collins, 1988.

2. Yusuf M. Mansur, Fuzzy Sets and Economics, Edward Elger Publishing Limited: Vermont, 1995.

3. Koji Shimojima, Yasuhisa Hasegawa, and Toshio Fukuda, "Unsupervised/Supervised Learning for RBF-Fuzzy System," Lecture Notes in Computer Science, vol. 1011, pp. 127-147, Springer-Verlag: New York, 1995.

4. Suran Goonatilake, John A. Campbell, and Nesar Ahmad, "Genetic-Fuzzy Systems for Financial Decision Making," Lecture Notes in Computer Science, vol. 1011, pp. 202-223, Springer-Verlag: New York, 1995.

5. Charles W. Richter, Jr., and Gerald B. Sheblé, "Genetic Algorithm Evolution of Utility Bidding Strategies for the Competitive Marketplace," Proceedings of the 1997 IEEE/PES Summer Meeting, 1997.

6. Charles William Richter, Jr., *Developing bidding strategies for electric utilities in a competitive environment*, Master's thesis, Iowa State University, Ames, IA, 1996.

7. Charles W. Richter, Jr., Tim T. Maifeld, and Gerald B. Sheblé. "Genetic Algorithm Development of a Healthcare Expert System," Proceedings of the 4th Annual Midwest Electro-Technology Conference, pp. 35-38, Ames, IA, 1995.

8. Dan Ashlock, "GP-automata for Dividing the Dollar," Mathematics Department, Iowa State University, Ames, IA, 1995.

9. B. Barkovich and D. Hawk, "Charting a new course in California," *IEEE Spectrum*, Vol. 33, July 1996, p. 26.

10. G. Fahd and G. Sheblé, "Optimal power flow emulation of interchange brokerage systems using linear programming." *IEEE Transactions on Power Systems*, Vol. 7, No. 2, pp. 497-504, 1992.

11. D. Finlay, *Optimal Bidding Strategies in Competitive Electric Power Pools*, Masters thesis, University of Illinois, Urbana-Champaign, IL, 1995.

12. D. Goldberg, *Genetic Algorithms in Search, Optimization & Machine Learning*, Reading, MA: Addison-Wesley Publishing Company, Inc., 1989.

13. Y. Ioannides, "Evolution of Trading Structures," Department of Economics, Tufts University, Medford, MA 02155, 1995.

14. J. Kumar and G. Sheblé, “Auction Game in Electric Power Market Place.” *Proceedings of the 58th American Power Conference,* 1996, pp. 356-364.

15. J. Kumar and G.B. Sheblé, “Framework for Energy Brokerage System with Reserve Margins and Transmission Losses,” *IEEE Trans. on Power Systems,* vol. 11, no. 4, November 1996, pp. 1763-1769.

16. B. LeBaron, “Experiments in Evolutionary Finance,” University of Wisconsin - Madison, Madison, Wisconsin, 1995.

17. P. Milgrom, *Auctions and Bidding: A Primer,* Journal of Economic Perspectives, Vol. 3, No. 3, Summer 1989, pp 3-22, 1989.

18. A. Papalexopoulos, S. Hao, and T. Peng, “An implementation of a neural network based load forecasting model for the EMS,” Presented at the 1994 IEEE/PES Winter Meeting, 94 WM 209 7 PWRS, New York: IEEE, 1994.

19. D. Post, *Electric power interchange transaction analysis and selection,* Master’s thesis, Iowa State University, Ames, IA, 1994.

20. G. Sheblé, “Priced based operation in an auction market structure,” *IEEE Trans. on Power Systems,* vol. 11, no. 4, November 1996, pp. 1770-1777.

21. G. Sheblé, “Electric energy in a fully evolved marketplace,” Presented at the 1994 North American Power Symposium, Kansas State University, KS, 1994.

22. G. Sheblé and J. McCalley, “Discrete auction systems for power system management,” Presented at the 1994 National Science Foundation Workshop, Pullman, WA, 1994.

23. G. Sheblé, “Simulation of discrete auction systems for power system risk management,” Frontiers of Power, Oklahoma, 1994.

24. G. Sheblé, M. Ilic, B. Wollenberg, and F. Wu, Lecture Notes from: *Engineering Strategies for Open Access Transmission Systems: A Two-Day Short Course*, presented Dec. 5 and 6, 1996, in San Francisco, CA.

25. L. Tesfatsion, “A Trade Network Game with Endogenous Partner Selection,” *Economic Report Series,* Department of Economics, Iowa State University, Ames, IA, 1995.

26. A. Vojdani, C. Imparto, N. Saini, B. Wollenberg, and H. Happ, “Transmission Access Issues,” Presented at the 1995 IEEE/PES Winter Meeting, 95 WM 121-4 PWRS, New York: IEEE, 1994.

27. A. Wood and B. Wollenberg, *Power Generation, Operation, and Control,* New York, NY: John Wiley & Sons, 1984.

28. Bart Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence,* Englewood Cliffs, NJ: Prentice Hall, 1992.

29. G. B. Sheblé, *Computational Auction Mechanism for Restructured Electric Power Industry Operation,* Kluwer Academic Press, Norwell, Massachusetts, 1999.

30. M. A. Einhorn, *From Regulation to Competition: New Frontiers in Electricity Markets*, Kluwer Academic Publishers, Boston, 1994.

31. T. Alvey, D. Goodwin, X. Ma, D. Streiffert, and D. Sun, "A security-constrained bid-clearing system for the New Zealand wholesale electricity market," presented at the 1997 IEEE PES Summer Power Meeting, Berlin, Germany, in press.

32. S. Dekrajangpetch, *Auction Implementations Using LaGrangian Relaxation, Interior-Point Linear Programming, and Upper-Bound Linear Programming*, Master's Thesis, Iowa State University, Ames, IA, 1997.

33. G. L. Thompson, and S. Thore, *Computational Economics: Economic Modeling with Optimization Software*, Boyd and Fraser Publishing Company, Denver, 1992.

34. D. Post, S. Coppinger, and G. Sheblé, "Application of auctions as a pricing mechanism for the interchange of electric power," *IEEE Trans. on Power Systems,* vol. 10, no. 3, August 1995, pp. 1580-1584.

35. G. Fahd, D. Richards, and G. B. Sheblé, "The implementation of an energy brokerage system using linear programming," *IEEE Trans. on Power Systems,* T-PWRS, vol. 7, no. 1, pp. 90-96, February 1992.

36. G. B. Sheblé, M. Ilic, B. F. Wollenberg, and F. Wu, Lecture Notes from: *Engineering Strategies for Open Access Transmission Systems: A Two-Day Short Course*, presented, Dec. 5 and 6, 1996, in San Francisco, CA.

37. M. Bazaraa, J. Jarvis, and H. Sherali, *Linear Programming and Network Flows,* John Wiley and Sons, New York, 2nd Edition, 1990.

38. F. Hillier and G. Lieberman, *Introduction to Operations Research*, McGraw-Hill, New York, 6th edition, 1995.

39. S. Dekrajangpetch, *Auction Implementations Using LaGrangian Relaxation, Interior-Point Linear Programming, and Upper-Bound Linear Programming*, Master's thesis, Iowa State University, Ames, IA, 1997.

ANTICIPATORY AGENTS FOR THE DEREGULATED ELECTRIC POWER SYSTEM*

L.H. TSOUKALAS, Purdue University**
Ö. ULUYOL, Purdue University

ABSTRACT

An agent-based anticipatory approach for the protection of the electric power grid is presented. Agents modeling the consumption behaviors of electricity customers are used to predict demand and anticipate contingencies within a subset of the power system called a Local Area Grid (LAG). Analysis of the temporal characteristics of major customers suggests that, under certain conditions, their behavior can be reliably predicted and therefore taken into account in anticipatory strategies for regulation and security. Each LAG is expected to have a sufficient mixture of residential, commercial, and industrial customers as well as some auxiliary generating capacity so that it can be effectively defended through anticipatory strategies for demand-side management or the dispatch of small generators. The connectivity of the grid permits LAGs to be defined in functional, not geographic, terms. Agents dedicated to electricity customers can be thought of as intelligent software generalizations of the familiar electric meter. They keep track of aggregate consumption and, in addition, are endowed with computing, communication, and modeling capabilities that allow them to identify patterns, predict demand, share knowledge of contingencies, receive price information, etc. Out of the interactions of such electricity agents a more robust deregulated power system is expected to emerge, capable of anticipating future demand and meeting supply and security constraints in ways that ensure a stable, efficient, and healthy power infrastructure.

INTRODUCTION

The electric power industry in the United States is undergoing deep and profound reforms through the process of *deregulation*. With the number of U.S. households projected to rise by 1.0% a year between 1996 and 2020, residential demand for electricity is projected to grow by 1.5% annually. Residential electricity demand changes as a function of the time of day, week, or year. During summer, residential demand peaks in the late afternoon and evening, when household cooling and lighting needs are highest. This periodicity increases the peak-to-average load ratio for utilities. Although many regions currently have surplus baseload capacity, strong growth in the residential sector will result in a need for more “peaking” capacity. Yet many experts suggest that deregulated electricity systems will result in decreased peaking capacity, with likely margins of less than 10% being a possibility. It is thus of utmost significance to prevent local problems on the grid from cascading into global failures.

---

* Research supported by EPRI/DOD/ARO grant no. W08333-02.

** *Corresponding author address*: Lefteri H. Tsoukalas, Consortium for the Intelligent Management of the Electric Power Grid (CIMEG), Purdue University, West Lafayette, IN 47907-1290; e-mail: tsoukala@ecn.purdue.edu.

The power grid is a multifaceted, multiscale complex system involving a variety of entities, including hardware components (such as generators, buses, transmission lines, relays and transformers, meters, and loads) as well as human operators, marketers of electricity, management teams, maintenance and operations engineers, and industrial and residential customers. Power grid entities are continuously engaged in collaborative/competitive interactions where the underlying network provides the means and physical constraints for the transactions involved. The time scales for these interactions vary from microseconds for lightning-caused overvoltages, to milliseconds for fault protection, to 10-20 seconds for load shedding, to minutes for dispatching small generators and hours for large units, to weeks and months for planning, and to decades for construction of major units. These time scales provide a natural criterion for establishing a hierarchy of actions and strategies that can be automated to facilitate anticipatory management control (Tsoukalas, 1997).

Our research draws heavily on a number of novel technologies including, but not limited to, *neurofuzzy systems, agent-based complex adaptive systems*, and *high-performance* computing to achieve its overall aims (Amin, 2000). In this paper we will limit ourselves to the modeling of electric agents that hold consumption information and predictive models of individual customers. The rest of the paper is organized in the following manner. Section 2 provides an overview of the structure and features of the local area grid. Section 3 gives some examples of individual agents and the predictability of electricity consumption patterns. Section 4 summarizes some of the important issues and discusses future efforts.

OVERVIEW OF THE ELECTRICITY/INFORMATION STRUCTURE OF LAGs

Conventional engineering models of the power grid focus on controlling electric power generation by estimating the state of the system through transmission/distribution measurements while considering loads, or customers, as exogenous disturbances (because of their unpredictable nature). The approach of the Consortium for the Intelligent Management of the Electric Power Grid (CIMEG)[1] pays unique attention to the customer side of the grid as the driver of the entire electric power system. The analysis of the temporal characteristics of major loads suggests that, under certain conditions, customer behavior can be reliably predicted and therefore taken into account in anticipatory strategies. Important to this perspective is the assumption that the grid can be segmented into subsets, called Local Area Grids (LAGs), with each LAG having a sufficient mixture of commercial, industrial, and residential loads and some auxiliary (standby) generating capacity. Individual LAGs are to be defended through anticipatory strategies for demand-side management and the dispatch of small generators. The connectivity of the grid permits LAGs to be defined in functional, not geographic, terms (for example, in terms of the number and types of loads and/or small generators). The overall aim is to enable the grid to protect itself from cascading failures and/or recover from unforeseen upset events by employing anticipatory strategies that safely and reliably

- Dispatch small, independent (standby) generators, and
- Engage in effective demand-side management strategies.

[1] The Consortium for the Intelligent Management of the Electric Power Grid (CIMEG) is part of the Complex Interactive Networks and Self-Healing Infrastructures Research Initiative sponsored by the Electric Power Research Institute and the U.S. Department of Defense, Army Research Office.

In anticipatory systems, the goal is to synthesize engineering systems as analogues of biological systems, which are capable of modifying their present state on the basis of anticipated future states (Rosen, 1985). Fuzzy logic provides an important idiom for describing anticipatory control and decision-making strategies, taking as input predictions obtained from neural models (Tsoukalas, 1998). Research on developing a comprehensive theory that verifies and validates the performance and stability of neurofuzzy algorithms under selected conditions is underway. The noise-tolerant predictive capabilities of neural networks can be exploited as a means of creating intelligent agents that estimate and predict the values of variables used in anticipatory decision making and control in a timely and reliable fashion.

Multiagent systems in which agents interact with each other are now being used as a solution to problems involved in complex energy infrastructures (Wildberger, 1997). For such systems to work properly, it is necessary that agents learn from their environment and adapt their behavior accordingly. In this project, we develop primarily customer agents, which can be thought of as a software generalization of electric meters. Other agents, modeling generation and transmission as well as corporate agents as described by Wildberger (1998) using a combination of neurofuzzy learning and static adaptation also need to be developed. There is a compelling need to study the modalities and mechanisms used by individual agents in interacting with each other and in acting proactively with respect to changes in their environment (Davidsson, 1994; Ekdahl, 1995). Since the environment is affected by the activities of multiple autonomous agents, it seems evident that an interaction strategy that adapts to the changing circumstances would be better than a static, nonadaptive one. It also seems intuitive that the adaptive behavior would occur as a product of *learning*.

Grid managers and marketers of electricity urgently need to be able to meet accurate short-term predictions (minutes to hours) of the demand from their major customers. Yet, they cannot do this by instrumenting a customer’s facility. Observations of the history of use and contextual information must suffice. With the ability to predict local activities, it appears feasible to automate and build into the system certain sequences of protective actions or strategies whose purpose would be to prevent local problems from cascading to global failures.

The main possibilities for such action are to engage in effective demand-side management (including load shedding) and/or dispatch small generating units. Since the time scales involved are those of seconds or at most a few minutes, it is desirable to automate these actions (while maintaining crucial human presence in the loop) and make them part of a local defensive structure for protecting the integrity of the grid as a whole. The situation is graphically illustrated in Figure 1. In all LAGs, some protective measures can be taken automatically if the local dispatcher has reasonable indication that a threatening sequence of events may be incipient or possibly unfolding. As seen in Figure 1, each LAG has its own computer, that is, local computational power. The local computer is networked with central dispatch but also has the computational tools to identify local problems and exercise anticipatory local management, in other words, to function as a local dispatcher. Thus, each local grid has its own customized defensive structure. Prior arrangements will need to have been worked out with the proprietors of the generators and major customers whose load may be locally managed. Economic and marketing agreements will be involved so that local grid actions may be preauthorized.

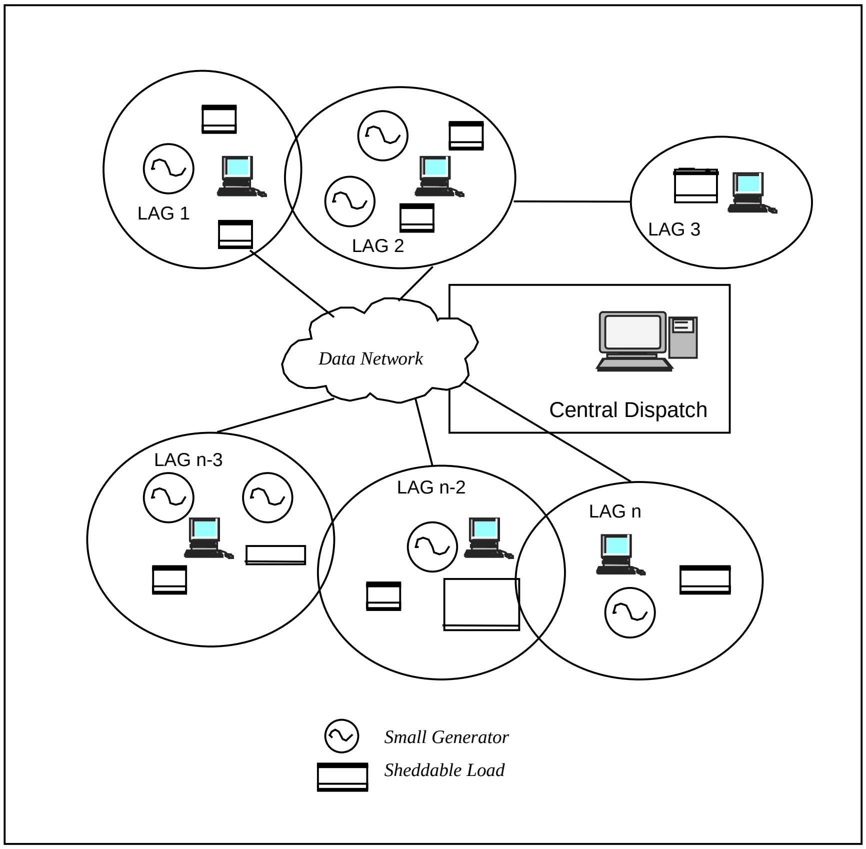


FIGURE 1 An overall architecture of the electric power system (grid) segmented in local area grids (LAGs). Both anticipatory strategies for demand-side management and the dispatch of small generating units can be locally employed.

Figure 2 shows schematically a more detailed view of a LAG. The grid *per se* has three major levels. The transmission level is comprised of pylons, wires, and hardware associated with high and very high voltage (typically 100 kV to 700 kV); the subtransmission level is an intermediate between transmission and distribution; the distribution level is the lower voltage part of the system (ranging anywhere between 69 kV to the 110 V of residential usage). At each level there may be some generating capacity, and one may find customers at all levels, although the largest number of customers by far is to be found at the distribution level.

Such LAGs have a variety of structures, designs, geographical features, voltage levels, load patterns, equipment characteristics, sizes, topological configurations, organizations, policies, and operating rules. Yet, as indicated in Figure 2, the foundation of the entire system is the customer side. This side comprises the agent info-space, and it includes data about customer behavior, patterns of consumption, features of customer idiosyncrasies, prices, and a variety of models. Customers drive the production and dissemination of electric power, and to the extent that their behavior can be reliably predicted, we can control and protect each LAG by anticipating customer behavior. Planning the dispatch of generators as well as network security monitoring are important LAG activities that can be coordinated on the basis of anticipated demand. Network security requires analysis of present and planned operating states (after the execution of

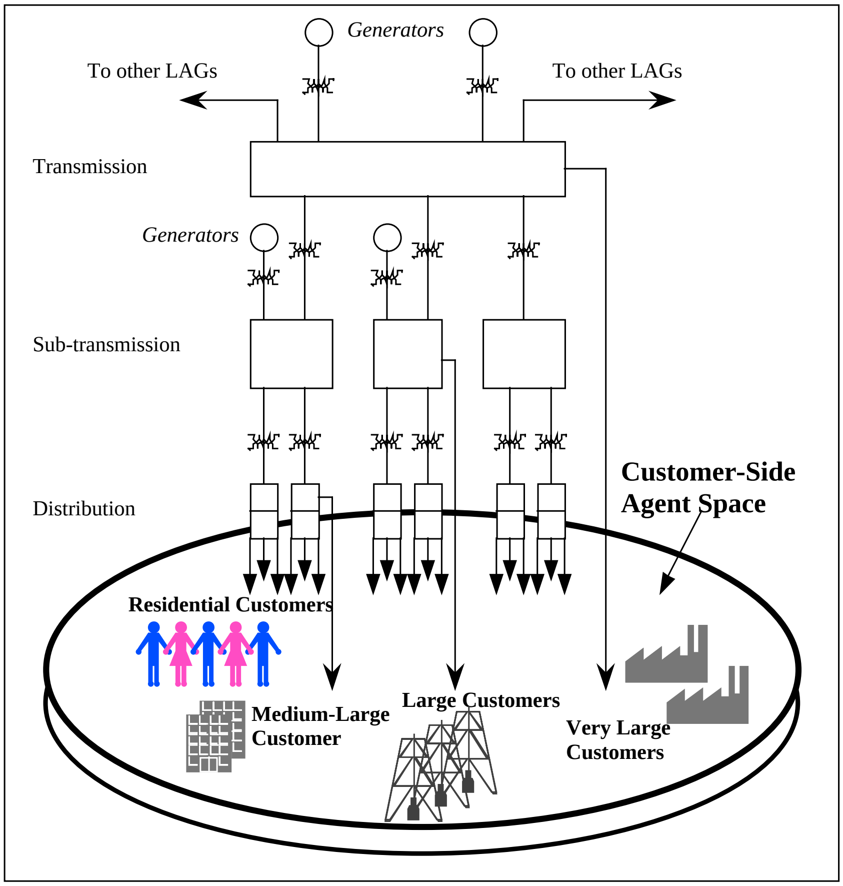


FIGURE 2 Schematic of a local area grid modeled as a customer-driven system.

switching operations) to obtain a complete security assessment emerging out of individual LAG assessments. Short circuit programs, contingency analysis, and stability programs are tools that support the assessment and determine the choice of the best preventive or corrective measures.

AGENT ISSUES FOR ELECTRICITY CUSTOMERS

As discussed in the previous section, whereas the conventional engineering perspective of the power grid focuses on electric power generation and transmission/distribution, our perspective is that the electric power system is a customer-driven system. The crucial question is, of course, how do we go from electric meters to software agents? Our research indicates that an important part in answering this question has to do with the *predictability* of electricity usage patterns. Predictability plays a very important role in making agents maintain some autonomy on a variety of tasks such as security, modeling, decision-making, tuning predictive models, and last, but not least, using information about prices and costs. Predictability is what is required for effective anticipatory strategies, that is, sequences of actions that modify the current state on the basis of anticipated future states (Tsoukalas, 1998). Local anticipatory control to protect the grid crucially depends on our ability to model the temporal behavior of individual loads and/or classes of loads that have significant local impact.

Consider a local area grid with a number of customers totaling about 100 MW of electricity consumption. Suppose we are trying to make agents for a typical residential customer, a small commercial industrial customer, and a major industrial customer. Our starting point is historical patterns of consumption. A few years of hourly consumption data provide interesting insights about the nature of the customers. Figures 3, 4 and 5 show the load profiles of three customers using hourly data for one year (May 1997 to April 1998). For each of these we have hourly electricity consumption data plotted for a year (the y-axis is energy in kilowatts consumed in one hour). It should be noted that some of these are statistical customers (Figures 3 to 5), while Figure 6 shows an actual customer.

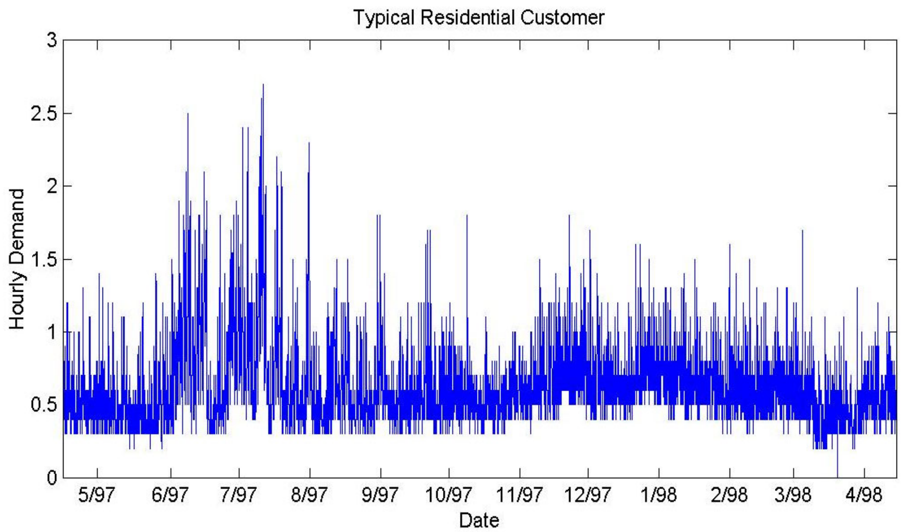


FIGURE 3 Hourly electricity demand by a typical residential customer.

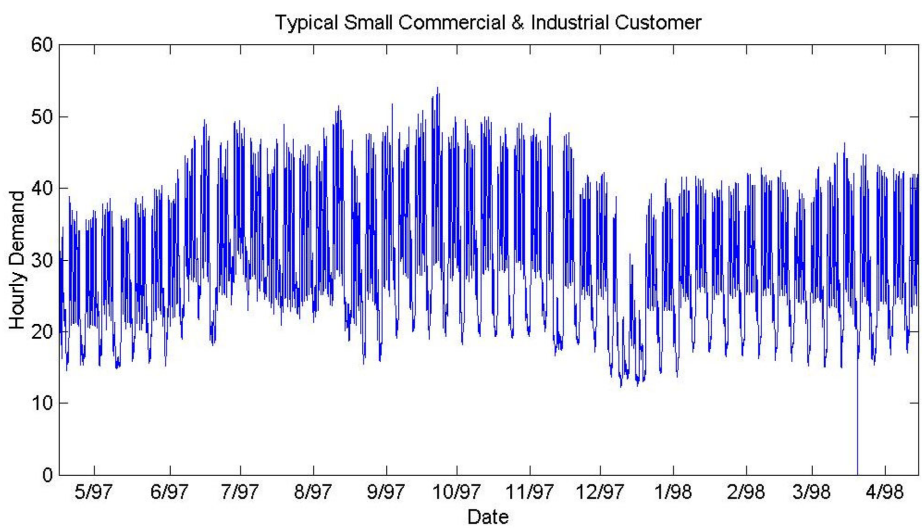


FIGURE 4 Hourly electricity demand by a typical small commercial customer.

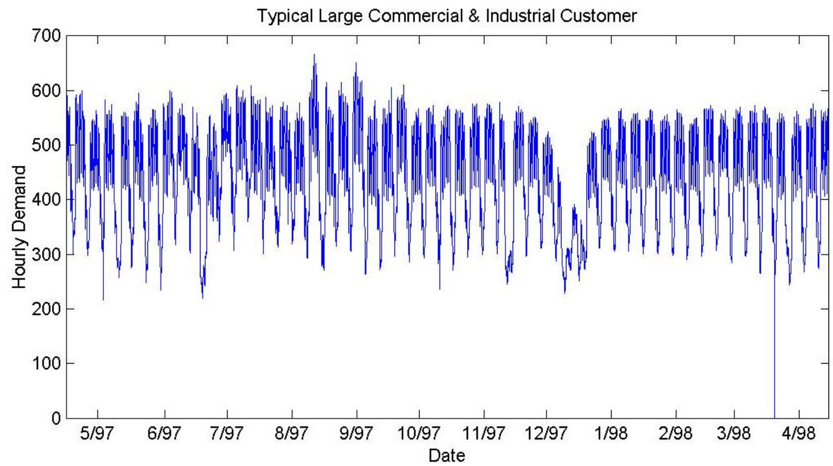


FIGURE 5 Hourly electricity demand by a typical large customer.

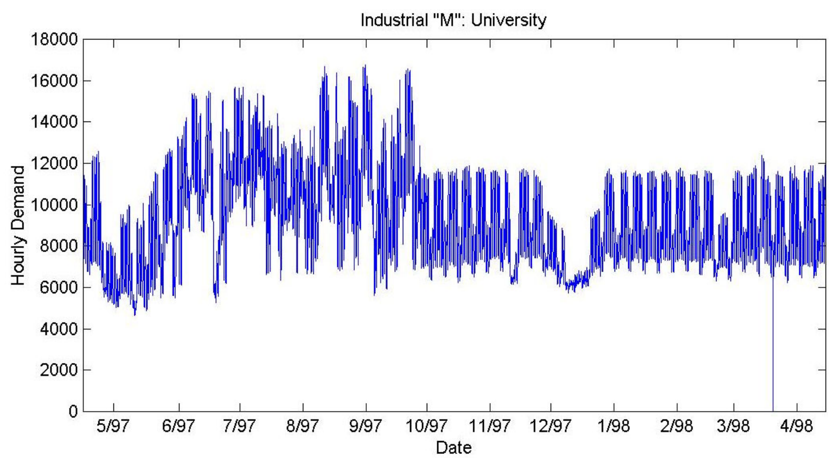


FIGURE 6 Hourly demand by an actual large customer.

A careful examination of the profile of the actual customer (Figure 6) reveals that electricity demand by this large customer, actually a local university, has a very regular operation during the fall and spring semesters, but during the summer season its electricity demand has far greater variability. The big variance may indicate that the university offers short-term programs (possibly lasting a few days) during the summer. This is an inference that the agent needs to obtain on each own (although it could be information given as input by users).

The sudden changes that affect the statistical properties of the profiles and hence make the data series non-stationary can be dealt with by using fuzzy logic techniques. We are developing a switching or a weighting mechanism that facilitates an appropriate amount of contribution from each approach to the final forecast. This requires a certain metric, which will assess to what degree the most recent data belongs to the current state, to be defined (Amin, 2000).

Although the electricity demand data is one-dimensional, and the load is usually measured in terms of kW/h every hour, it is crucial that the data be analyzed at various granularities. Through this analysis many interesting features can be discovered. Consider, for example, phase portraits of the data (also known as *directed scatter diagrams*). Phase portraits reveal some temporal idiosyncrasies of the loads. Consider Figures 7, 8 and 9. In Figure 7 we have a phase portrait of a typical residential customer (from the data in Figure 3), with a phase shift of six hours. It is evident that there is a square-like feature in the residential consumption of electricity, where the side of the square is about six hours. In Figure 8 we see the phase portrait of the typical small commercial customer (the annual profile of which is shown in Figure 4). Here we see a P-shaped form that uniquely identifies the consumption pattern of this particular type of customer.

On the other hand, Figure 9 shows a totally different portrait, which is unique to the actual large customer (that is, the university). It should be noted that both Figures 8 and 9 show features of residential-type usage of electricity in the form of the fuzzy squares with six-hour sides. In all three plots, a cyclical phenomenon analogous to that of a limit cycle is discernible.

SUMMARY

The phenomenal advancements in information technologies and the concept of agents offer attractive possibilities for the intelligent management of the vast deregulated electric power system. Essentially, it is now possible to develop agent generalizations of that familiar household item, the electric meter. Every significant customer of electricity (including the typical residential user) will have its own agent and the interactions of all agents within a local area grid will provide significant guidance for anticipatory self-regulation. Crucial to all this is the

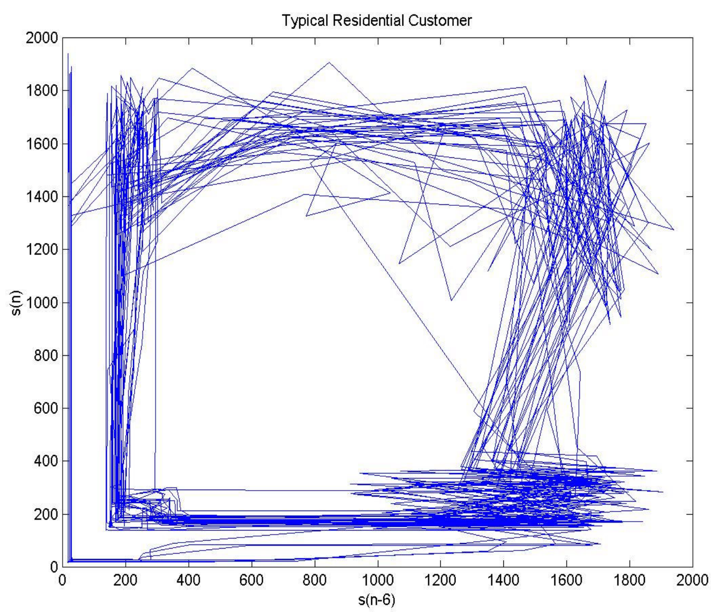


FIGURE 7 Phase portrait of electricity consumption for a typical residential customer.

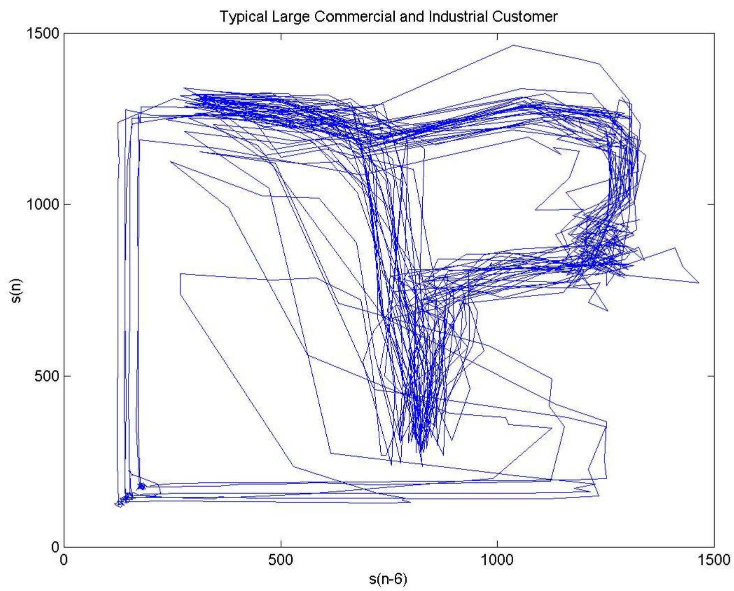


FIGURE 8 Phase portrait of electricity consumption for a large/commercial customer.

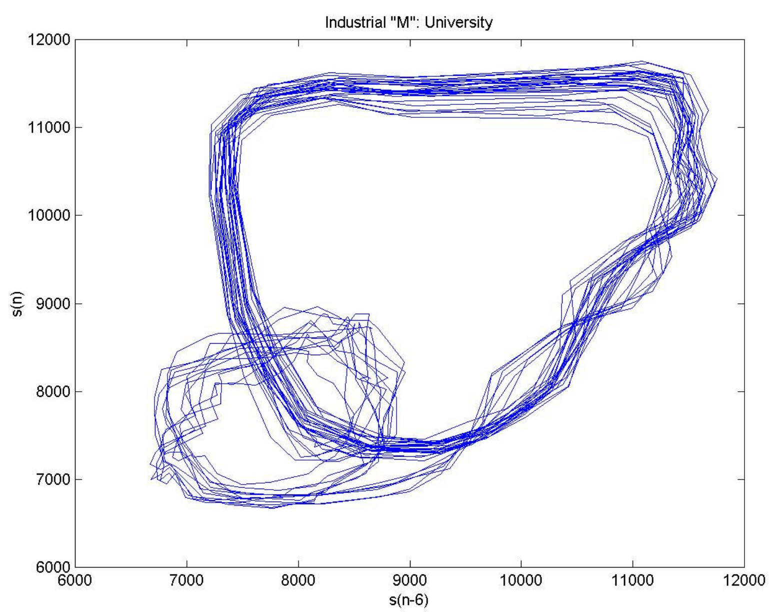


FIGURE 9 Phase portrait of electricity consumption for a specific large customer.

predictability of customer behavior. We have seen that the phase diagrams for a university, a typical residential customer; and a typical large commercial/industrial customer clearly reveal that each customer has unique features and, hence, predictability is possible. Considerable effort is underway to endow each individual agent with its own predictive neural network using local memory neurons (Amin, 2000). A variety of simple yet effective algorithms for communication and anticipatory decision making are to be added to the electric agents. CIMEG plans to demonstrate the developed methodologies through a prototype called TELOS (Transmission Entities with Learning-capabilities and On-line Self-healing), which is expected to operate off-line by the end of 2001 (for selected LAGs).

REFERENCES

Amin, S.M. Intelligent Management of the Power Grid: An Anticipatory, Multi-Agent, High-Performance Computing Approach, EPRI Technical Report, 2000 (in press).

Davidsson, P., E. Astor, and B. Ekdahl. “A Framework for Autonomous Agents Based on the Concept of Anticipatory Systems,” in R. Trappl, ed., *Cybernetics and Systems ’94*, 1427-1434, World Scientific, 1994.

Ekdahl, B., E. Astor, and P. Davidsson. “Toward Anticipatory Agents,” in M. Woolridge and N.R. Jennings, eds., *Intelligent Agents — Theories, Architectures, and Languages*, Lecture Notes in Artificial Intelligence 890, 191-202, Springer Verlag, 1995.

Rosen, R. *Anticipatory Systems*, Pergamon Press, Oxford, 1985.

Tsoukalas, L.H. and R.E. Uhrig, *Fuzzy and Neural Approaches in Engineering*, Wiley, New York, 1997.

Tsoukalas, L.H. “Neurofuzzy Approaches to Anticipation: A New Paradigm for Intelligent Systems,” *IEEE Transactions on Systems, Man and Cybernetics*, *28*(4), 573-582, 1998.

Wildberger, A.M. “Autonomous Adaptive Agents for Distributed Control of the Electric Power Grid in a Competitive Electric Power Industry,” Keynote Presentation at KES’97, Adelaide, Australia, 1997.

Wildberger, A.M. “Modeling the Infrastructure Industries as Complex Adaptive Systems,” *Proceedings of the Advanced Simulation Technology Conference (ASTC),* April 1998.

DISCUSSION:

MODELING ELECTRICAL NETWORKS

M.V. Nagendra Prasad (to Erik Larsen): Let me make a comment and see if you agree with this or not. The point about soft variables: system dynamics does it pretty often. But there's nothing specific to system dynamics that says that it cannot be done in agent-based systems. We can just borrow it very liberally. And that's what we should be doing, I think.

Erik Larsen: Oh, yes. I mean, there's nothing secret about soft variables, and as you said, people ignore them.

Nagendra Prasad: Yes, not necessarily in the agent-based community, but in a lot of the operations research, they just ignore them because they can't model them.

Charles Macal: Could you comment on the relative homogeneity or heterogeneity of the organizations or agents — well, if they're regarded as agents — within the systems dynamics framework that you've been applying?

Larsen: You can do almost whatever you want to do. I mean, I think what you try to do is try to look at the specific situation. Let's say that you have different companies within your model. What you have then is the focal company, which you can maybe describe as an agent. It's a different kind of agent. And then you say, "This is where we have most of the details." Then we might have all kind of competitors in this market. Depending on what we are modeling, we'd probably model them so that they would have different logic, different kinds of information they're looking at, different ways of making decisions. What you typically do is try to find out if there are any stereotypes. So, you often have the focal company and, say, two or three representatives of other, different ways of looking at a company's logic.

So you could have that company's competitor, who wants a market share but doesn't care so much about return on equity, at least to start with. You could also have a group of companies that are just going for economic value. Then, if you've calculated economic value, they're going to build. However, you could have companies that have some other logic. And you typically find that it depends on what the company team you're working with believes. It's their beliefs you're trying to capture.

Macal: It seems that in the systems dynamics framework, you're not modeling as much the interactions that occur among the agents so much as the flows of information or physical quantities that could be transferred among them.

Larsen: I think I would argue that that isn't action. It's information flowing from the company to its owners and to the market. You see the signals in the market. You take that information, and you do something about it. So in a sense you are modeling the action resulting from information flow. What you don't do, you don't have 2,500 of them. You normally have very few agents or companies or whatever you call them — entities. So I think that's a big difference, that you have relatively few. You don't have a big population of companies.

I think that's also why you can make each company much more detailed, in a sense; if you had 3,000 of them, but they were detailed, it would take forever to run the model. So what you do is trade off this variety in companies against something that is closer to reality and so actually provide some kind of real insight into the problem.

Scott Page: Could you give me a better sense of how you model the evolution of the regulatory framework in these settings?

Larsen: Yes, I can. What we do is look at what triggered the regulator action. So you could have a simple thing, say, market share. You cannot have more than a 25% market share. Well, that would be simple. If you want to represent the regulator in a more profound way, you could say, "What is it that a regulator learns over time? What is the change in how he/she views the world?"

Page: Does that mean you bring into the idea political capture, as we would talk about it here?

Larsen: You could do that, yes. You can model the political influence on the regulator. They say that profits grow the companies. The public gets uncertain, they put pressure on the politicians, and the politicians put pressure on the regulator. You can actually model that process, yes.

Page: And that's what you've actually done here?

Larsen: In those models, no. I have other models, yes.

[*Gerald Scheblé and Lefteri Tsoukalas presented their papers at this point.*]

Gale Boyd [to Tsoukalas]: You said something earlier about how [the electrical industry is] traditionally modeled in terms of the grid and the generation. And then for this particular project you turned it around and looked at the loads and the grid. To some extent, the focus of deregulation, and the entities that are created by deregulation, is also on the customer creating the load. It seems interesting to me that by focusing on this and modeling the system just in terms of the demand and then on how it's all connected, you may have some insight into how the generation is then distributed, which generates price implications.

Lefteri Tsoukalas: Yes, indeed. In fact, I'm somewhat disingenuous when I say that we don't look at the generators, but, for the engineers in the audience, this is intended to be a little bit of a stimulus for extra thinking, because the canonical model of the grid is of generators, wires, and then these exogenous variables that pertain to demand. In fact, we do look at the generating side. For example, our second set of anticipatory strategies has to do with small generators. Utilities are looking into this idea — having 50-kilowatt or 100-kilowatt small generators available without owning them. Universities and hospitals already have these small generators on standby. With the right sort of incentive — say I get a better price from Commonwealth Edison — I give them the right to dispatch my generators somewhere else 10 times a year. So the model does have to do with generators, with using small, local, ancillary or auxiliary generating capacity for averting problems that could propagate into a brownout or blackout.

Of course, what makes this possible is having an equivalent model of the grid in terms of how the demand will behave, how the customers will behave, and the wires, and of course, the hardware. It's this information infrastructure on which the thing is based. And in that sense, it's not unlike other customer-centered or customer-focused models.

Of course, the crucial question is whether the customer is predictable. The standard engineering assumption, actually from the 1920s and '30s, was that the electricity customer is not predictable. But there is a statistical customer — there are actual customers, the bigger ones, that are predictable under certain conditions.

Richard Cirillo (discussant): Do we have any questions on the general area of the application of agent-based simulation to the electric power grid?

Kathy Lee Simunich: I had actually two questions, and they're more properly directed toward Gerald [Scheblé]. Everybody knows that weather really affects load demand, and I was wondering how you took into account the effect of weather on load when modeling the electric market in general. And my follow-on question was — I've been reading about these weather derivatives, which are a stock market option or whatever on ...

Gerald Scheblé: Cold degree days and hot degree days.

Simunich: Yes, exactly, on how to maybe offset losses due to extreme weather events. And I was wondering if the power companies are using these tools, tricks, whatever they are, and if so, do we have to bring this extra complexity into the modeling of the overall electric market?

Sheblé: Yes, you do. Those derivatives come from electric power utilities — Southern Company in particular, and a new company, well, it's an old company really, under Utility Corp., called Aquila Energy Corp. They also came up with the cold degree days. Southern Company came up with the hot degree days contracts. And what they want to do is, the demand on the electric utility varies greatly depending upon weather temperatures, especially in the summer, and especially in the South, where you like to have air conditioning.

And so what they've found is there's a number of industries that had the offsetting problem when it got too hot. So they decided that these derivatives would allow them to share the risk of that forecast of the number of days of heat with these other companies. So what they're doing is just like the example I gave where the one utility went to the automobile auction industry — that's what these utilities are doing. They're looking into other industries and looking at how they can share their risk with those other companies.

Simunich: Is that common nowadays, then?

Sheblé: Very common. These companies are into anything that will allow them to have a stable cash flow, because their number one risk today is regulatory risk. There are some people in the federal government who feel that FERC [Federal Energy Regulatory Commission] commissioners should remove all the price caps from these various markets. Right now, FERC has put a price cap on all the different electric products in every state. What if all of a sudden you have three bidders in California and they remove the price caps, what will the price go to? But the price caps were hit almost all the time last summer in California.

Simunich: How would you model this concept?

Sheblé: What you have to do is, you have to do multiple markets. You have to look at the company — the firm as a whole — and what markets they are playing in. If you're a generation company — you burn coal, for example — you have to look at coal future contracts, and you have to look at the spot market price for coal. You should probably also hedge with natural gas, because your competitors are using natural gas, and right now they're cheaper than you, so you'd better play their markets a little bit so you get some of the financial advantage there. You have to look to your customers, because what you want to do is you want to be able to shift their load or shed their load when demand gets too high and the price goes up. You want to be able to take care of the availability of your units. Most power plants, by the way, are only available about 80% of the time, if they're really maintained well, which is not that great a reliability, because we normally like to see 99.99% reliability. It's hard to get 80% without a lot of diversification, a lot of contingent contracts. Utilities sign contracts with each other right now that they will pick up the contract if this other unit goes down.

So it becomes intertwined, almost incestuous, in terms of all the contracts and how to keep track of them. The City of Los Angeles last year did a count of the number of contracts they are maintaining on an hourly basis. During their peak periods, they were maintaining records on over 50,000 contracts. You know, for three individuals to do this, this is a nice computer database to get into in terms of making it efficient.

Simunich: Well, in order to model these, do you have to model the weather as well, to see how it works?

Sheblé: Yes, definitely. The weather component of the demand is extreme, especially with the demand of air conditioning. But actually that's your best resource. If you go to, say, Woodfield Mall out here, as they've done in the South — what the energy service companies will do is they'll go in and instead of selling electricity, they'll say, "We'll sell you cooling." And they'll say, "What do you mean?" "Well, we'll cut your bill in half if you'll allow us to put this extra gear in your basement." Now, the extra gear happens to be 10,000 pounds of salt water, because they will cool that down at night so that they can shift the air-conditioning demand for electricity from during the daytime, the peak pricing period, to midnight to 3:00 a.m. And by doing so, they achieve a factor of almost 15:1, in the South, of cost.

So, you know, that's where the new ideas are coming up from; that is, what does the customer really want? Okay, we don't manufacture electrons, even though a lot of engineers will say that. At least not to my knowledge, we're not doing that, except maybe at some experimental physics facilities! The electrons have always been there. What we do is we just keep pushing them back and forth, and we transfer the energy that way. So what you've got to think about is, people need energy. It's not the electrons we're selling; it's the product they need — whether it be air conditioning or heat, or just keeping their computers up.

Cirillo: I would like to pose a question, I guess to the audience, in the context of what Rob Axtell mentioned this morning in his opening address with regard to the application of agent-based modeling to policy making and policy decisions. I believe his comment was something to the effect that the tools are not quite ready yet to make policy decisions. I think what we've seen here is a potentially widespread application for agent-based simulation, where it's beginning to make inroads into operating the electric system and dealing with the electric system in new ways. And I would ask the question, are the tools ready? Is the technology there to apply to this situation? Comments?

Kathleen Carley: This is just purely by way of comment. Actually, they are being used to make policy already. There's an agent-based modeling group that's actually helping the government in the European Common Market make decisions about pricing. And there's another group that's been helping the U.S. government set foreign policy. And there are other ones helping various companies already, so they're already being used as support systems to do what-if analysis on various policies — to try them out and decide which one might be the most effective.

Cirillo: So you would not concur with the statement that they're not ready for policymaking?

Carley: Absolutely not.

Ian Lustick: I'm not sure what the details are of those programs. I've had a little bit of contact with the intelligence community, and they are fascinated by the potential, but yet don't really see how they can exploit it. They actually have people working in this area, and on the kinds of problems that I'm interested in — you can imagine — ethnic mobilization and so on. And I'm interested in learning more about it, but I'm not prepared to say that my work or any other work that I've seen goes beyond what Rob [Axtell] suggested.

Now, this is a different kind of problem in a way, and perhaps, oddly enough, more tractable, but I don't yet see point prediction. It seems to me that the intelligence community would be interested in point predictions of one sort or another. And the best I'm able to do or can imagine doing for the foreseeable future is shifting distributions of possibilities, which would be a very difficult thing to rely on if you had the option of [applying] human intelligence to the specific case in mind.

Sheblé: I think most of the tools are there for analyzing how to conduct the firm under a given set of rules. But when you're changing the market rules on a monthly basis, how do you predict what's going to happen next month? I think that's where the tools cannot do anything. And I don't think any tool can do that. I don't think anybody can predict how much money ENRON's going to throw to a Congressman to be elected to stop or start something. And I'm not sure — I'm sure it'd be a good problem to solve, but, you know, I think the tools are there. I know the GA unit commitment tool I have is being used by some utilities, and the energy service companies are using some of my tools to look at contract terms. Okay, so they're working in that context. But when you change the market on a monthly basis, how do you track that? How do you predict what will happen as a consensus?

And one other question I have for the community here as a whole, being a professor: in the old world we used to publish a lot. What do you do when these companies approach you and they will fund your research with nice sums of money, but you cannot publish it. The doctoral dissertation cannot be published for at least a year after graduation, if not two. And nobody else is to see it for at least three years, because the amount of money we're talking about — if I can save Commonwealth Edison 1% of their fuel budget, that's roughly $2 billion a year. What happens when you come up with that kind of money?

Catherine Dibble: I've been thinking about that question a lot. To me, the answer for how academics who are doing things that are useful for the real world can collaborate with private sector companies and still get publishing done is the old NASA issue of spin-off. That is, if being funded to do a particular project for a private company gets you better tools to do your research with, you can then use those tools for other things that you can publish. If it gets you thinking about problems in a new way or gives you insights that again lead to publishable research that is separate from their project, then both can gain.

Tsoukalas: If I may say something about the previous question, the maturity of agents — I think that this question comes with a kind of periodicity. Every 10 or 15 years there's a new tool; it used to be expert systems. Back in the '40s and '50s, it was electronic brains. There's something peculiar to this latter part of the 20th century, and that is our preoccupation with artifacts that have to do with the way we relate to other human beings and with other kinds of artifacts that are really ways of relating with other humans. We relate to people in a personal way or via the artifacts that people build. And as society has become more complex and sophisticated, we relate more and more via artifacts rather than in person. I mean, the caveman didn't have much around; it was all personal relations. Think about how the 19th century was preoccupied with debates about mechanics and energy: out of that we got terms such as "work" and "power," which now are standard technical terms — everybody knows about "work" as force times displacement. But that's not how people thought early in the 19th century. You know, work was going to the field, sweating — you know, *doing*.

So in the same way, our fields bring in heavily these anthropomorphic terms, which cause a lot of excitement and sometimes a lot of disappointment. And I think when it comes to the agent technologies, there is a metaphorical way of speaking about them that is probably somewhat exaggerated. It's not that there are entities that will make decisions — "Now we are going to work," "Now we will shut down the power drill." But they're extremely promising and extremely important entities for enabling us to communicate with each other through time. They can enable engineers, let's say, who had nothing to do with the design of substations or the maintenance of transformers and substations, to communicate with the people who were there 5, 10, or 50 years before, because there was some essential aspect of their knowledge or their relation with this artifact that could be captured or reflected in what the agents carried to these later human beings.

So in that sense I think that the potential is tremendous. It works a little better in this context because it is more technically constrained. You don't have too much; the goal is more clear. But I think there is tremendous potential for software entities that have some degrees of autonomy and have some kind of "socialization," again in anthropomorphic terms, right, but not like us. It couldn't be intelligent in any sense that we really are intelligent. They would have to live the lives of human beings. It's not that the computer can be intelligent, but we use this term because we are saying something about our own intelligence. Even though it's not a personal sort of thing, these tools do enable us to go and hold a kind of dialogue with people who may not even be around anymore or are in faraway places.

Richard Burton: This is kind of a red-letter day for me because the mere fact that we might be dealing with something that would save somebody $2 billion is kind of exciting. But [industry sponsorship] is not a new problem in the university. We have traditionally been in the public-goods business and doing that through publication and not worrying too much about the other side of it. But universities throughout the world are struggling with this at the moment, particularly with respect to biotech developments. There's lots of work being done on this, both by universities and by people who are interested in the relationship between the universities and private industry. And the general answer is that universities have been very reluctant to take on restricted clauses about publication and making things available.

Agent Toolkits:
Strategies and Tradeoffs

INTEGRATING SIMULATION TECHNOLOGIES WITH SWARM

M. DANIELS, Swarm Development Group*

ABSTRACT

The Swarm simulator is a set of portable libraries that can be used in a variety of environments. This paper surveys tradeoffs between language usability, clarity, expressiveness, and flexibility. Two new language layers for Swarm (Scheme and XML) are demonstrated.**

1 INTRODUCTION

This paper discusses several ways the Swarm simulator has been integrated with other software systems. If you are familiar with Swarm and Swarm's interfaces, please skip ahead to Section 2.

1.1 Background

Swarm is a set of libraries that facilitate implementation of agent-based models. Swarm's inspiration comes from the field of artificial life. Artificial life is an approach to studying biological systems that attempts to infer mechanism from biological phenomena, using the elaboration, refinement, and generalization of these mechanisms to identify unifying dynamical properties of biological systems.

Two strategies characteristic of this approach have proven to be useful to researchers across fields. The first is empirical evaluation of dynamics. The combination of autonomous entities in a shared environment is typically a mutually recursive process that is analytically intractable. In many systems, the only way to know what global dynamics will occur is to run the numbers and find out.

The second idea is synthesis. Synthetic chemistry "invented" the buckyball (C_{60}) from the theoretical notion that it should be possible to build. Of course, it turns out it was possible. Similarly, artificial life seeks to extrapolate biological knowledge in order to suggest new experiments for things that "ought" to work. Swarm serves this scientific goal by providing a means to do this extrapolation via computer simulation.

At the time of Swarm's inception, researchers in the field of complex systems were finding that ad hoc programming was not a sufficiently powerful, reliable, or economical way to ask the kinds of questions that needed to be asked. The design and implementation of the computing infrastructure to manage and measure autonomous entities in a precise and reproducible way is a serious engineering task beyond the resources of most scientists.

---

* *Corresponding author address*: Marcus Daniels, Swarm Development Group, 624 Agua Fria, Suite 2, Santa Fe, NM 87501; e-mail: mgd@swarm.org.

** This document is available on the web (http://www.santafe.edu/~mgd/anl/anlchicago.html).

To help fill this need, Chris Langton initiated the Swarm project in 1994 at the Santa Fe Institute. The first version was available by 1996, and since then it has evolved to serve researchers not only in biology, but also in anthropology, computer science, defense, ecology, economics, geography, industry, and political science.

1.2 What Does Swarm Do?

The primary feature of Swarm is the virtual machine. The virtual machine allows the researcher to describe agent behaviors one by one, agent by agent, context by context, all while keeping an exact notion of time and concurrency in the world. Swarm also makes it possible to compose or decompose hierarchies of agents. We call this attribute *composability*.

This notion of composability is useful because it often isn't clear where to begin a modeling effort. For example, in modeling a large organization, it may be the case that the subjective understandings of individuals' or departments' roles and responsibilities differ widely, and that this variance includes poor performance in some cases and outstanding performance in other cases. Rather than seeking denotation on how the organization should work and looking for deviations, one can build independent model components from many perspectives and then combine them (mirroring abstractions of people for real people). This *bottom-up* approach has the advantage of documenting both the unexpected bad and good things in the organization, as well as contextual sensitivities.

Figure 1 shows an example of how the structure of the United States military might map to Swarm. The notation is as follows: the boxes are Swarms. A *Swarm* is a temporal container and physical home for a set of agents in the system. Swarms can contain other Swarms. The diamonds are agents. In Swarm, an agent has no built-in semantics. The modeler provides the semantics by defining behaviors using methods on an object (in the object-oriented programming sense) and by describing the transactions among agents and events using Swarm's scheduling machinery. Nothing related to scheduling is pictured below, just the hierarchies of localized activity.

A *Schedule* is an agent's to-do list. There are different kinds of to-do lists and different attributes that *Action* items on the to-do list can have. An Action is something that happens in the world, e.g., a solider shoots a gun, a captain orders an engagement, or the president declares war. In Swarm, Schedules and Actions are typically closely associated with an agent or model component. Agents may have their own Schedules (perhaps several) and a repertoire of Actions they know how to perform. (Of course, in this diagram many of the items on their to-do lists are *orders*.)

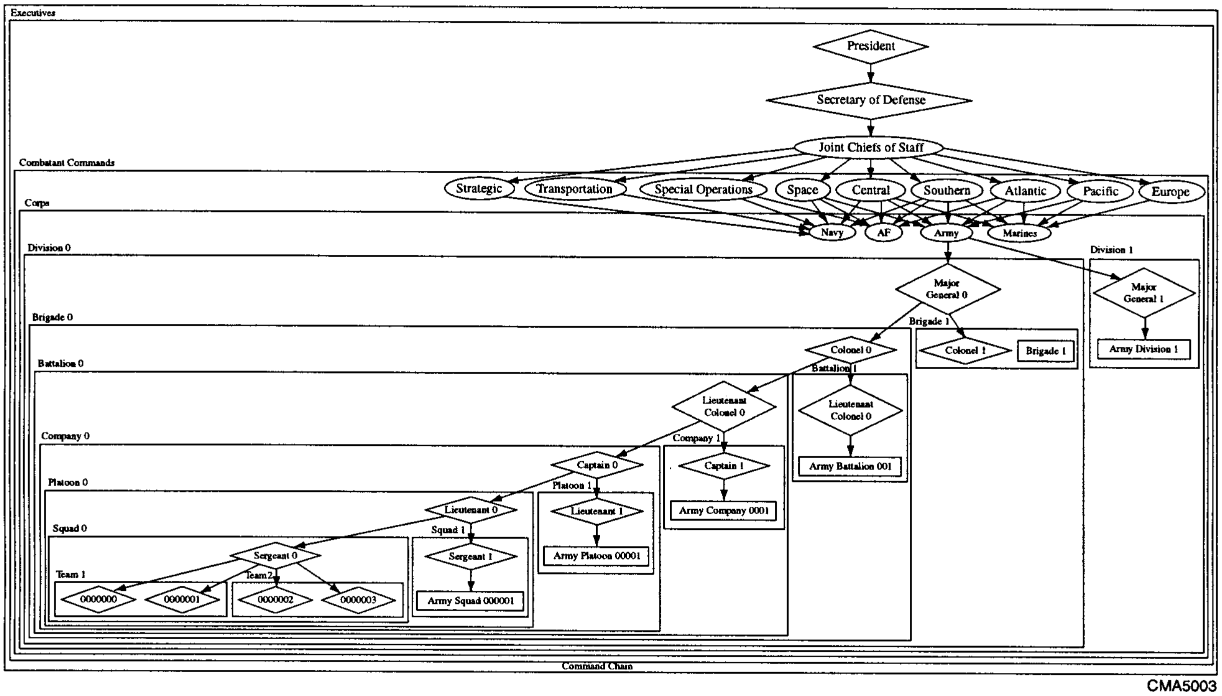


FIGURE 1 U.S. military mapped to Swarm.

2 HYDRA

Swarm is not a single application that is "turned on." Swarm is a set of libraries that you can pick and choose features from (the primary feature being the virtual machine). In order to use the Swarm libraries, it is necessary to create or use code that calls Swarm features.

A supplement to this paper containing example programs is available for downloading (ftp://ftp.santafe.edu/pub/swarm/src/users-contrib/anarchy/anl-0.0.tar.gz).

2.1 Dynamic vs. Static Languages

2.1.1 Objective C

Until recently, there was one way to use Swarm features: write and compile a program in Objective C. This is a flexible way to write a model using Swarm. Objective C models tend to have good performance because they are compiled by a native code optimizing compiler, namely GCC. Objective C is what is called a *dynamic* object-oriented language.

Dynamic languages have the attribute that they put information about typing in instances of objects, not in variables that hold the objects. This is convenient for agent-based modeling because it is analogous to the idea of working from the bottom up: the modeler describes the subjective experience of agents, not committing to roles for agents across the board. As the agents are glued together and data structures grow, opportunities for encapsulation present themselves. In this way, dynamic languages are useful for prototyping because the program itself serves as a sort of extension of the mental workspace.

While dynamic languages are theoretically pleasing for agent-based modeling, they have the significant practical disadvantage that the compiler is poorly informed about typing and thus cannot inform users of many kinds of coding errors (nor can it do as good a job of optimizing the code). Objective C is especially problematic for new users because there is no interactive runtime environment to catch errors. In most dynamically typed language implementations there is a interpreter that will report exceptions with a backtrace and show the context of a type misuse. The interpreter makes it easy to look at the environment and query objects in the vicinity.

In Objective C, it is possible to approximate these features by using a debugger, but since there are pointers in Objective C, any object can be corrupted by a coding error. This also means typing is mutable. (For the experienced user this is actually a feature of Objective C — you can construct new classes and objects from bits.)

A common problem in debugging Objective C Swarm models is that the stack will get corrupted via a bad pointer, and so there will be no direct information about the context of a failure. This is very frustrating for new users of Swarm.

To combat this problem, we have done two things: the first is to investigate integration with languages that preserve dynamic typing while restricting dangerous constructs like pointers.

2.1.2 Scheme

One such language is Scheme (http://www.swiss.ai.mit.edu/projects/scheme). Scheme has the additional feature that it can treat data as code and that the data and code are represented using the same syntax. Consider the case of modeling how DNA specifies how a protein is created. On one end you have a data vector and on the other end a thing that does something (runnable code). Implementing these mappings works in an obvious, efficient way in Scheme. For example, note how these two commands result in the same output:

```
#|kawa:1|# (display "hello world\n")
hello world
#|kawa:2|# (eval (list 'display "hello world\n"))
hello world
```

The first expression is a simple function call “display” with the argument “hello world.” The second expression *constructs* that same expression and then evaluates it. Since expressions are lists, and lists have simple syntax, it is mechanically easy to make new behaviors and mutate old ones. This is not feasible in Objective C because methods are compiled, and compilation is a very heavy and expensive operation. Of course, it is possible to implement an interpreter in Objective C, but that takes time and results in an ad hoc component not integrated with the language.

An example of a simple use of the Swarm scheduling machinery in Scheme is available in the "scm/" subdirectory in the supplement to this paper (ftp://ftp.santafe.edu/pub/swarm/src/users-contrib/anarchy/anl-0.0.tar.gz). Below is an example of how a model would look ("model.scm" in the supplement):

```
(load "swarm.scm")
(initSwarm)

(define (print-current-time)
  (display (string-append "t: " (number->string (getCurrentTime))))
  (newline))

(define (make-model swarm)
    (let* ((schedule (make-repeating-schedule swarm 5))
           (stopSchedule (make-schedule swarm))
           (target (object ()
                           ((step) <object> (print-current-time) #!null)
                           ((stop) <object> (terminate) #!null)))
           (stopSelector (selector target 'stop))
           (selector (selector target 'step)))
      (scheduleActionTo schedule 1 target selector)
      (scheduleActionTo schedule 3 target selector)
      (scheduleActionTo stopSchedule 10 target stopSelector)
      (let ((activity (swarmActivateIn swarm #!null)))
        (scheduleActivateIn schedule swarm)
        (scheduleActivateIn stopSchedule swarm)
        activity)))

(define (run-model)
    (run (make-model (make <swarm.objectbase.SwarmImpl> *globalZone*))))

(run-model)
```

The file "swarm.scm" in the supplement is the glue between Swarm and Kawa (http://www.gnu.org/software/kawa) the Scheme system. To test the example, first install Swarm 2.0.1 (http://www.santafe.edu/projects/swarm/release.html), then issue the following command:

```
$ CLASSPATH=./kawa-1.6.60-compiled.zip javaswarm kawa.repl -f model.scm
t: 1
t: 3
t: 6
t: 8
```

2.1.3 Java

In contrast, *static* languages like Java have the user confront typing early on and think about how components fit together. The current release of Swarm (2.0.1) supports Java. For new users of Swarm, writing models in Java is considerably harder to get wrong. Java is also a more attractive language for new users to learn since it is a popular language that has benefits outside of Swarm modeling.

The purpose of the Java layer of Swarm (actually it is a system extensible to other languages) is to mirror the protocols of the Swarm libraries as Java *interfaces*. The fact that Java is a more statically typed language is handled by introducing new types such as *Selector*. The Selector loads up type information so that the Objective C virtual machine can talk to Java on Java's terms.

Swarm doesn't assume that things that happen in the world are associated with types. For example, if you're walking down the street and you get hit by a car, you don't say to yourself, "Oh, now it is time to run my predefined get-hit-by-a-car routine"; you just get hit and something happens.

2.2 Declarative vs. Imperative

Objective C and Java are called imperative programming languages because they can be used to specify mechanism: *"do exactly as I say."* This is what most people probably think of when they hear the word "programming."

In one sense, this direct kind of control is useful and necessary. Much of human communication is oriented toward stories having a beginning, middle, and end. But in another sense, imperative programming is fragile, ad hoc, and obscures objective, rigorous pursuit of modeling. Agent-based modeling should aim to complement, not replace, traditional statistical and analytical techniques. Large bodies of coded imperative description may "work." but it is an unsatisfying situation if these descriptions must be treated opaquely, merely "watching" them. Ideally, the descriptions should be decomposable and have clear parameterizations.

Declarative programming languages aim to eliminate programming in the traditional sense. The user of a declarative language communicates in terms of goals, facts, and relations. Roughly, the idea is that the *system* critiques the description and then does whatever is necessary to find an answer or set of answers. There is no direct specification of mechanism.

There are three potential advantages to a declarative interface to Swarm.

- The first is practical: Swarm modelers are not, by in large, programmers. Software development infrastructure can have a large learning curve and isn't necessarily appropriate for representing model abstractions. Reviewing the code in freely available Swarm models (implemented in Objective C), makes a compelling case that Swarm modelers don't generally spend much time rationalizing their peers' code or, for that matter, their own code. Unless the actual code in models is treated as a work product by modelers, there will always be a danger that the intended mechanism differs from real mechanism. This assumes that modelers get as far as drawing interesting conclusions from their models. For example, there are instances in the Swarm community of modelers compromising their experiments because they've run into memory leaks they can't solve. Debugging complex mechanisms is a skill that takes time to acquire.

- Besides issues of internal correctness (e.g., that what the professor believes is true is the same as the grad-student programmer intends, and is in fact implemented that way), there is the matter of conceptual clarity. It's useful if ten things that are basically the same *are* the same plus or minus some clearly defined parameters. It's also useful to be aware of appropriate and inappropriate use of modeling constructs. In an imperative programming language, it is all too easy to blur boundaries between components that are conceptually distinct, because it is easy to glue incommensurate entities with a little extra mechanism. Further, it is useful to expose the exact semantics of model components and have it be clear what they mean without digging through a paper appendix or code archive of a model. As much as a possible, agent-based model descriptions should be one and the same with the descriptions submitted for peer review.

- Finally, there are technical advantages to having the model abstractions that modelers use be clearly defined and parameterized: it is easier to write useful visual programming tools for well-behaved, high-level components with bounded semantics than it is to write completely general-purpose visual programming tools. It is also easier to integrate these abstractions with other tools (e.g., constraint satisfaction, statistical packages, CASE tools, symbolic math packages).

2.3 Language Space

Figure 2 shows a slice of programming language space on these dimensions of declarativeness vs. dynamism. While there are arguments that some of these languages are particularly strong (or that there exist better implementations for some languages than others), the intent here is rather to suggest that what we have are tradeoffs: on one extreme you can have correct, pure, and theoretically satisfying abstractions that don't actually run or do anything interesting, and on another extreme there is the possibility of cool and realistic behavior

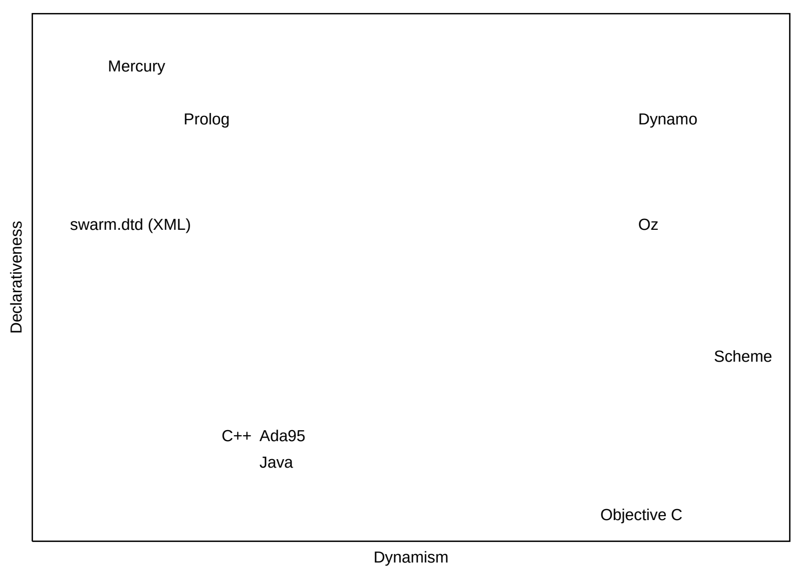


FIGURE 2 Programming languages arranged by style of object representation.

represented with incomprehensible, analytically intractable spaghetti code. Depending on the nature of a model (or components within a model), different tradeoffs may be desirable.

2.4 A Declarative Interface to Swarm

So far, we have described

- Swarm libraries and virtual machine;
- Scheme as a safer, yet more dynamic interface to Swarm;
- Java as a more strongly typed interface to Swarm; and
- The benefits of declarative description.

Before moving on how to realize a system for declarative description, let's take a look at how this fits together. This diagram shows how some of the declarative tools fit together, for reference during the discussion that follows.

In the diagram, solid lines and node-outlines indicate that the construct or connection is a done deal; it works now. Dashed lines and node-outlines indicate that the construct or connection has not yet been fully investigated, and there may be problems (but probably not many). Bold lines and node-outlines indicate that the construct or connection is known to be hard or doesn't yet exist. Ellipses are user applications or tools. Boxes are representation schemes (including code). Diamonds are engines or transformation processes. Parallelograms are servers.

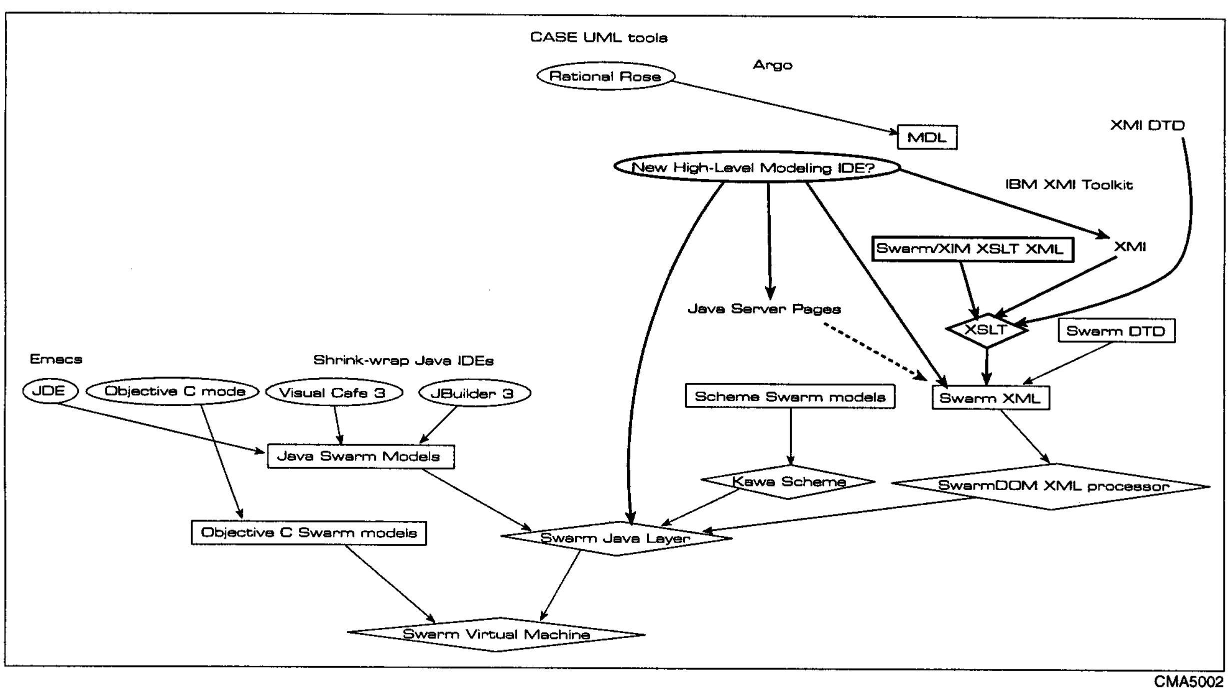


FIGURE 3 Interrelationship of declarative tools.

2.4.1 Representation

An obvious place to start in designing a declarative interface to Swarm is by documenting, compressing, and formalizing current practice in the Swarm community. We have done this to a small extent by looking at the simple idioms of a well-known Swarm application: heatbugs.

Heatbugs is a demonstration simulation in which a population of heatbugs, each with a preferred temperature and output heat, share a toroidal heat-diffusing space. When the heatbugs are comfortable, they stay put; when they aren't, they roam. The idea is that communities of heatbugs group together to maintain the heat they prefer, and of course these groups are impacted by other groups of heatbugs in the area.

Heatbugs is a flat simulation. Heatbugs have no deeper structure or internal mechanisms besides that which I just described. The model merely consists of agents and a space with some observation features for the experimenter.

The first question which arises is how to represent these idioms and features in Swarm. Since one thing we're trying to do is make Swarm accessible to a broader scientific audience, it's useful if the representation scheme is easy to understand. Adopting a declarative logic programming language means that users would need to install this package and learn some things about it. Since the immediate goal is to formalize current practice, and that practice doesn't come from a declarative background, it seems premature to adopt such a language.

On the other hand, since we are trying to provide an alternative to ad hoc imperative coded model components, it is inconsistent to our goal to go in the direction of a locally invented description language.

Luckily, a technically strong, standard, popular, and language-neutral data representation scheme is available: Extensible Markup Language, or XML (http://www.w3.org/XML).

2.4.1.1 Standard

XML is subset of SGML, an international standard since 1986. XML is a World Wide Web consortium (http://www.w3.org) recommendation and has been adopted by Microsoft in many of its products.

2.4.1.2 Familiar

XML looks like HTML but is intended to encode information, not just data for display in a web browser. In the spirit of dynamic typing, XML files can written first and then augmented with structural information for validation. Or they can be validated from the start. To validate an XML document, another file called a Document Type Definition (DTD) is used that describes the valid contexts for the pieces of the XML document.

2.4.1.3 Supported

XML is supported by a number of applications. Internet Explorer 5 for Windows has excellent support for browsing XML documents. There are freely available DTD editors (http://www.alphaworks.ibm.com/tech/) and translators and a number of commercial authoring tools (http://www.xml.com/pub/pt/Authoring). Essentially, these tools make it easy to build correct modeling grammars and models without remembering syntax or using command-line tools.

2.4.1.4 Web-Compatible

XML also is used in Java Server Pages (http://java.sun.com/products/jsp/index.html), a new standard from Sun for building servlets (Java programs running on a web server) from extensible XML tag libraries. In other words, a Swarm model could be represented as a XML document on a web server, where parameters in a model could be set via a web browser or by other JSP simulation web servers.

2.4.1.5 Development Infrastructure

XML is related to many useful technologies. Document Object Model, or DOM (http://www.w3.org/DOM), specifies an programming interface to XML documents and makes it possible to easily read, write, and modify XML. There are freely available XML parsers (http://www.alphaworks.ibm.com/tech/xml4j) that create DOM data structures.

2.4.1.6 Foundation for Other Standards

Other standards are built on XML. The ones shown in Figure 3 are Extensible Stylesheet Language [Transformation], or XSL[T] (http://www.w3.org/Style/XSL), and XML Metadata Interchange Format, or XMI (http://www-4.ibm.com/software/ad/features/xmi.html). XSL is interesting for two reasons: (1) it is a declarative way to describe the transformation of one model into another, e.g., *model docking*, and (2) XSL has a library (in fact implemented by IE5) for visualizing these transformations. XMI is a way to represent the Unified Modeling Language, or UML (http://www.omg.org/uml), in XML. UML is interesting because there are powerful CASE tools based on UML for software design (in our case, models). Examples of these are Argo (http://www.ArgoUML.org) (free) and Rational Rose (http://www.rational.com/products/rose) (commercial). In the supplement, the file “swarm.mdl” is an example of how the declarative Swarm interface discussed below can be precisely represented in Rational Rose. It’s also possible to use Rational Rose to do design and implementation of imperative software, but we have not tested that.

2.4.2 Heatbugs in XML

Figure 4 shows what Heatbugs looks like using the “swarm.dtd” document type definition grammar and vocabulary (in the supplement). The first four lines are a standard XML header. The ENTITY lines name constants (worldWidth and worldHeight) that are reused in several places in the model. The underlining and other type features are explained in the next section.

swarmexcel.xml.color **1/1**
/opt/src/mgd/src/Swarm/anl/doc/ Oct 24 1999

```xml
<?xml version="1.0"?>
<!DOCTYPE swarmModel SYSTEM "swarm.dtd" [
<!ENTITY worldWidth "80">
<!ENTITY worldHeight "80">
]>
<swarmModel>
  <GUISwarm id="HeatbugObserverSwarm">
    <Swarm id="HeatbugModelSwarm">
      <environment>
        <Grid2d id="world"
                width="&worldWidth;"
                height="&worldHeight;"/>
        <HeatSpace id="heatspace"
                   width="&worldwidth;"
                   height="&worldHeight;"
                   diffuseConstant="1.0"
                   evaporationRate="0.99"/>
      </environment>
      <classes>
        <class name="heatbugs.Heatbug">
          <uniformRandom name="idealTemperature" min="17000" max="31000"/>
          <uniformRandom name="outputHeat" min="3000" max="10000"/>
          <byteConstant name="bugColor" value="64"/>
          <doubleConstant name="randomMoveProbability" value="0.0"/>
        </class>
      </classes>
      <agents id="heatbugs" count="100" class="heatbugs.Heatbug"
              populate="world"/>
      <Schedule repeatInterval="1">
        <ActionGroup time="0">
          <ActionTo target="heatspace" message="stepRule"/>
          <ActionForEach target="heatbugs" message="step"/>
          <ActionTo target="heatspace" message="updateLattice"/>
        </ActionGroup>
      </Schedule>
    </Swarm>
    <gui-environment>
      <HeatbugsColormap id="globalColormap"/>
      <ZoomRaster id="worldRaster"
                  width="&worldWidth;"
                  height="&worldHeight;"
                  colormap="globalColormap"
                  zoomFactor="4"/>
      <Value2dDisplay id="heatDisplay"
                      raster="worldRaster"
                      colormap="globalColormap"
                      space="heatspace"
                      factor="512"
                      colorOffset="0"/>
      <Object2dDisplay id="heatbugDisplay"
                       collection="heatbugs"
                       raster="worldRaster"
                       space="world"
                       message="drawSelfOn"/>
      <excel id="excel"/>
    </gui-environment>
    <Schedule repeatInterval="1">
      <ActionGroup time"0">
        <ActionTo target="heatDisplay" message="display"/>
        <ActionTo target="heatbugDisplay" message="display"/>
        <ActionTo target="worldRaster" message="drawSelf"/>
        <ActionTo target="excel" message="addHeatbugUnhappinessColumn"/>
      </ActionGroup>
    </Schedule>
  </GUISwarm>
</swarmModel>
```

FIGURE 4 Heatbugs implemented in Swarm.

2.4.2.1 Notation

- Elements in capital letters (e.g. "Swarm") are declarative counterparts to Swarm library classes.

- Elements in lower case are new declarative constructs.

- Elements in bold indicate facilities related to scheduling.

- Elements in italic and underlined are classes; structural descriptions of new agent types (the vocabulary provides many that don't need to be re-described).

- Elements in italic are instances of classes (either built-in or the extension blue classes).

- Elements underlined only are considered to be stable parts of the environment, as contrasted with objects that are created as a function of the simulation, like declaratively defined agents.

2.4.2.2 Vocabulary

New declarative constructs

- classes

  This section is where agents are described. There are several subelements that can be used to describe agent variables.

  - uniformRandom

    In Objective C Swarm models, when agents are initialized per some distribution, it is typically the job of a *ModelSwarm* class to install those values. This is undesirable for two reasons: (1) the distribution can be parameterized in a clear way, and having mechanism to do it is ad hoc, and (2) the distribution is a property of the class of agents, not a property of the model environment. This element makes it possible to define an agent variable that has a default initialization selected from a distribution.

  - byteConstant, doubleConstant

    Agent variables can also be initialized using constants. In Objective C Swarm models, this initialization is typically done by adding code in a +createBegin: method.

- agents

  This section is where classes of agents are instantiated and related to other agents or environment objects. Besides the class and count of agents, this element has the *populate* attribute that takes a Grid2d instance. This directive takes the set of agents that are instantiated (which implicitly are getting randomly selected ideal temperature and output heat values as they are instantiated) and randomly places them within the provided Grid2d.

- environment

  This section is where stable environmental objects in the model are parameterized for instantiation.

- gui-environment

  This section is where stable environmental objects for observation of a model are parameterized for instantiation. For example, it doesn't make sense to have a graphical ProbeDisplay or ZoomRaster in a logical model — that is strictly for observational purposes. The grammar specifies these constraints.

Instances of formalized idioms

- HeatbugsColormap

  In Heatbugs, the temperature of a heatbug is indicated by the intensity of red. In the Objective C *HeatbugsObserverSwarm*, there is ad hoc code to initialize a colormap with such a range. Here, we're forced to encapsulate and parameterize this behavior.

- HeatSpace

  A parameterized version of the HeatSpace class in the heatbugs app.

Features not found in Swarm

- excel

  This declarative interface to Swarm is based on a Java XML DOM library from IBM. On Windows, using the Microsoft Java virtual machine provides integration to Component Object Model, or COM (http://www.microsoft.com/com). COM is a way that Windows applications can communicate with one another and the outside world. Since Excel provides a COM (a.k.a. automation) interface, Java programs can be used to control Excel. The "excel" element provides the ability for declarative Swarm actions to send data to Excel. This facility is used instead of Swarm's EZGraph to draw an unhappiness graph as an Excel chart.

Swarm library declarative counterparts

- Swarm

  A temporal context and physical vicinity for agents.

- GUISwarm

  A kind of Swarm intended as a context for components that observe a model in action. It also provides a panel for controlling a simulation.

- Grid2d

  A simple grid where agents can live.

- Schedule

  The means by which a model or an agent plans behavior.

- ActionGroup

  A way to group behaviors that happen concurrently.

- ActionTo

  An Action intended for a particular recipient.

- ActionForEach

  An Action intended for a set of recipients.

- ZoomRaster

  A feature in Swarm for visualizing two-dimensional data.

- Value2dDisplay, Object2dDisplay

  Classes for representing values and objects on a two-dimensional grid. Notice how these reference instances of agents, Grid2d, ZoomRaster, and HeatbugColormap. (In XML, these references are checked for correctness.)

2.4.2.3 DOM object hierarchies of XML

As mentioned earlier, XML can be loaded by a DOM library. The representation that is built in memory is a tree of model component specifications. If it is validated against a DTD (in our case it is), the DOM user knows what the structure of the tree will be. This makes it easy to interpret the DOM tree: to convert the declarative model representation into a running model, it is a simple matter of recursively expanding the tree using nodes or subtrees to instantiate objects. For example, in Objective C, suppose you have a message sequence like

```
[[[[Agent createBegin: aZone]
    setSubcomponent1: subcomponent1]
   setSubcomponent2: subcomponent2]
  createEnd];
```

This would be represented as an Agent node with two child nodes; the code that dealt with an Agent element would be set up to look at the child subtree rather than expanding it.

Figure 5 shows how DOM would “see” the above XML between <Swarm> and </Swarm> tags. Notation: the type conventions are as given above. Ellipses are DOM element objects, and the triangles are DOM attribute objects. The arrows are captured in DOM by “id” and “idref” attributes. So, again, to make this diagram “go,” it’s just matter of iterating through all the objects and connections and instantiating the Swarm counterparts. The grammar and semantics of Schedule, Swarm, and GUISwarm ensure that schedule activation and model invocation are done in the right sequence.

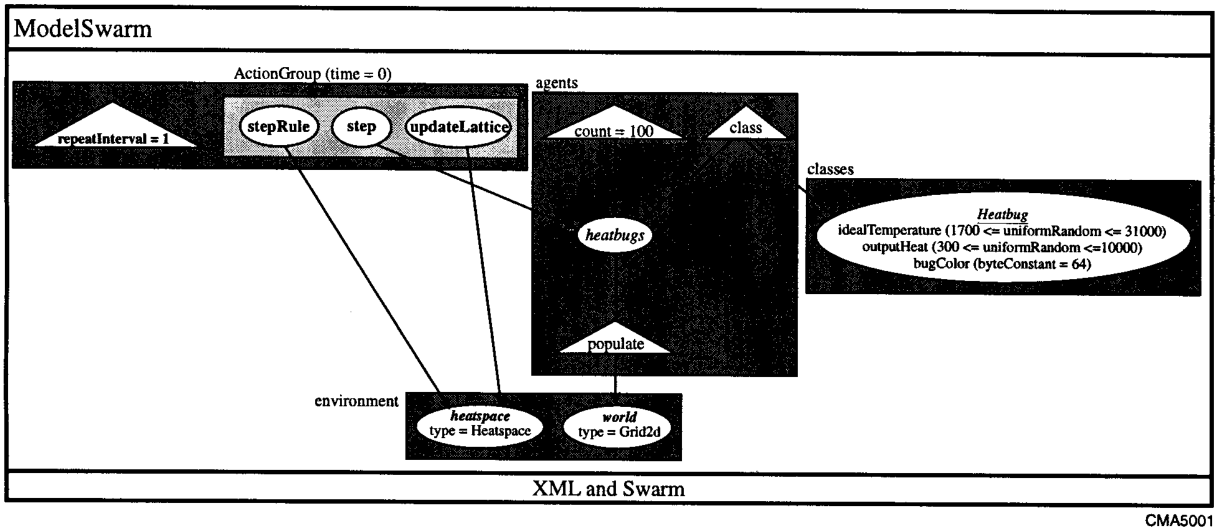


FIGURE 5 DOM representation of heatbugs model.

2.4.2.4 Running XML heatbugs

There are two demos that can be run: a simple heatbugs and an extended heatbugs that draws an unhappiness graph using Excel via COM.

1. Get and install Swarm 2.0.1 (http://www.santafe.edu/projects/swarm/release.html).

2. Unpack the anl-0.0.tar.gz distribution (ftp://ftp.santafe.edu/pub/swarm/src/users-contrib/
anarchy/anl-0.0.tar.gz) in some directory, then run:

```
$ cd anl-0.0/xml
```

3. Get and install xml4j.jar (http://www.alphaworks.ibm.com/tech/xml4j) from Alphaworks (http://www.alphaworks.ibm.com) in the (current) xml/subdirectory.

4. If you are using Windows and have Office 2000 and want to see the Java/DOM/COM/Excel demo, get the current Microsoft Java virtual machine (http://www.microsoft.com/java). For the Excel demo, you'll also need to get updated Swarm DLLs (ftp://ftp.santafe.edu/pub/ swarm/2.0.1-fixes). Install them in Swarm-2.0.1\bin.

   Then, from a Swarm "terminal":

   ```
   $ ./runjactivexforexcel
   ```

   Tweak the paths in this script to your install locations.

5. To compile the Excel-equipped demo, run:

   ```
   $ ./mscompile
   ```

   To compile the generic demo, run:

   ```
   $ ./compile
   ```

   Again, tweak the paths as appropriate to your system.

   To run the Excel-equipped demo, run:

   ```
   $ ./runexcel
   ```

   To run the generic demo, run:

   ```
   $ ./run
   ```

ASCAPE: AN AGENT BASED MODELING FRAMEWORK IN JAVA*

M.T. PARKER, The Brookings Institution**

ABSTRACT

Ascape is a toolkit created at Brookings to support the design, analysis, and distribution of agent-based models. Its principal design goals include abstraction and generalization of key agent modeling concepts, ease of use and configurability, best attainable performance, and deployment anywhere. Ascape was developed primarily to support our models of social and economic systems, which typically comprise agents with rules of behavior interacting in networks (e.g., regular lattices, random graphs, soups), but the framework may be adaptable to other model types. In addition to demonstrating Ascape design features and capabilities, we'll build a simple model in Ascape, talk about future goals, discuss the use of Java for agent based modeling, and invite questions about Ascape and modeling design issues.

** *Corresponding author address*: Miles T. Parker, The Brookings Institution, 1775 Massachusetts Ave. NW, Washington, DC 20036; e-mail: mparker@brook.edu.

Ascape: An Agent-Based Modeling Framework in Java

Presented at Agent Simulation:
Applications, Models and Tools October 16, 1999
Miles T. Parker
Software Engineer
Center for Social and Economic Dynamics
Http://www.brook.edu/es/dynamics/models/ascape
mparker@brook.edu
The Brookings Institution
1775 Mass. Ave. NW Washington, DC 20036
voice 202.797.6136
fax 202.797.6319

Ascape: An Agent Based Modeling Framework in Java

Ascape is a toolkit created at Brookings to support the design, analysis and distribution of agent based models. Its principle design goals include abstraction and generalization of key agent modeling concepts, ease of use and configurability, best attainable performance, and deployment anywhere. Ascape was developed primarily to support our models of social and economic systems, which typically comprise agents with rules of behavior interacting in networks (e.g., regular lattices, random graphs, soups) but the framework may be adaptable to other model types. In addition to demonstrating Ascape design features and capabilities, we'll build (look at?) a simple model in Ascape, talk about future goals, discuss the use of Java for agent based modeling, and invite questions about Ascape and modeling design issues.

Design Goals

- Generalized
- Abstract
- Portable
- Easy to Use
- Expressive
- Robust
- Fast

Generalization

- Obviously, applicable to many problem domains
- As many common features as possible
  - Charting
  - Model views
  - Parameter management tools
  - etc., etc.
- Large libraries of common structures and behaviors

Abstraction

- Should be possible to make significant changes in one aspect of model without affecting others
  - Dimensions, topology
  - Rules
  - Structure
  - Rule execution order
- Promote exploration and experimentation
- Allows easy mixing and matching of model design and tools

Portability

- Greatly facilitates sharing of projects, methods, and results with colleagues and the general public
- Does not lock you into specific hardware or technology choices, except...
- Java
  - Cross-platform really works
  - Web *very* important
  - All or nothing: code must be 100% pure Java, no native code in core
  - Other solutions possible

Ease of Use

- Basic assumption: Small population of "experts," large population of potential "smart users"
- Avoid frustration barriers

Ease for Non-Coders

- User configurable and modifiable
  - Of course, complete control of model parameters at runtime
  - No programming necessary for creating graphs, customizing views
  - Soon, should be no programming for changing basic rules and possibly structure
- Long term goal: complete model development without coding
  - Not as hard as it might seem
  - But would require significant development resources

Ease for Coders

- Relatively easy and straightforward to develop
  - Should be possible for people with basic skills to build simple models from "off-the-shelf" parts and progress from there
  - The most complex models should be reasonably straightforward, and the code should remain easy to work with and understand
  - As much functionality as possible should be provided "automatically"

Expressiveness

- Should be able to specify and develop a model with the simplest possible high-level description without obscuring important details
- Careful design can provide power and control
- Possible problem: important details can be glossed over
- Helps in mapping to more general descriptions (XML!)

Robust

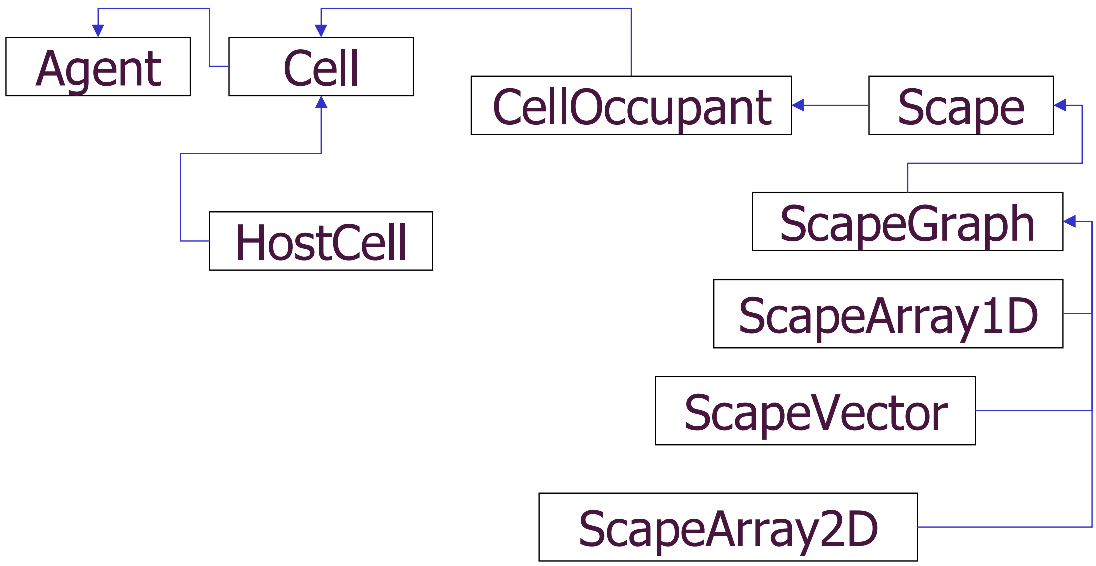
- Tolerant of different styles and methods of use
- Reports when broken
- Breaks at compile time, not runtime

Performance

- For many models, performance is not a big issue....But we always seem to want more speed
- Java performance good
  - Speed now comparable to C++ in many applications
  - Significant improvements in Java environments
  - Perception vs. reality
  - Graphics still need work
- Design often much more important than platform
- Supporting pluggable views also helps

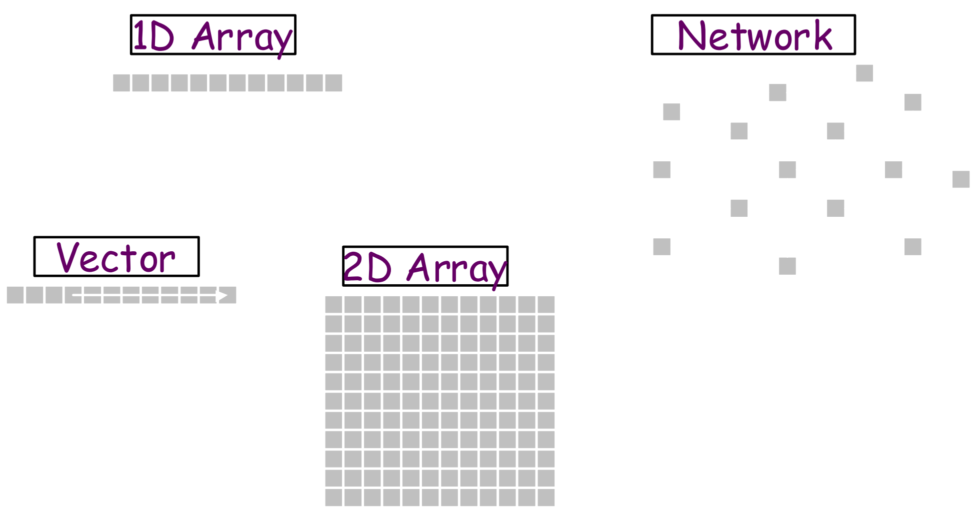
Structure

- Scapes are Agents



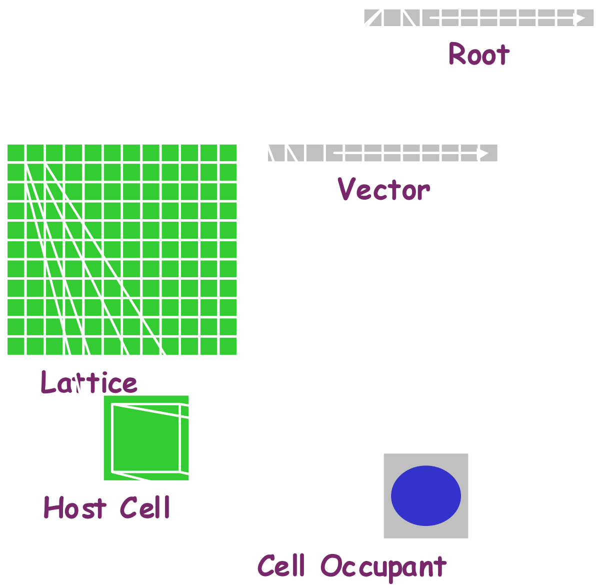
Structure

- There can be many different kinds of Scapes



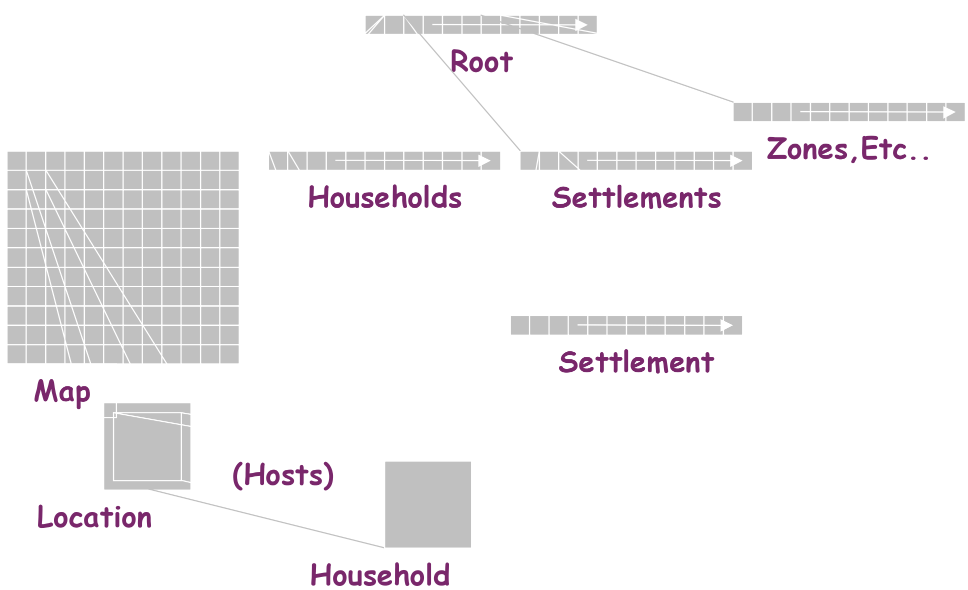
Structure

- Example: Prisoner's Dilemma



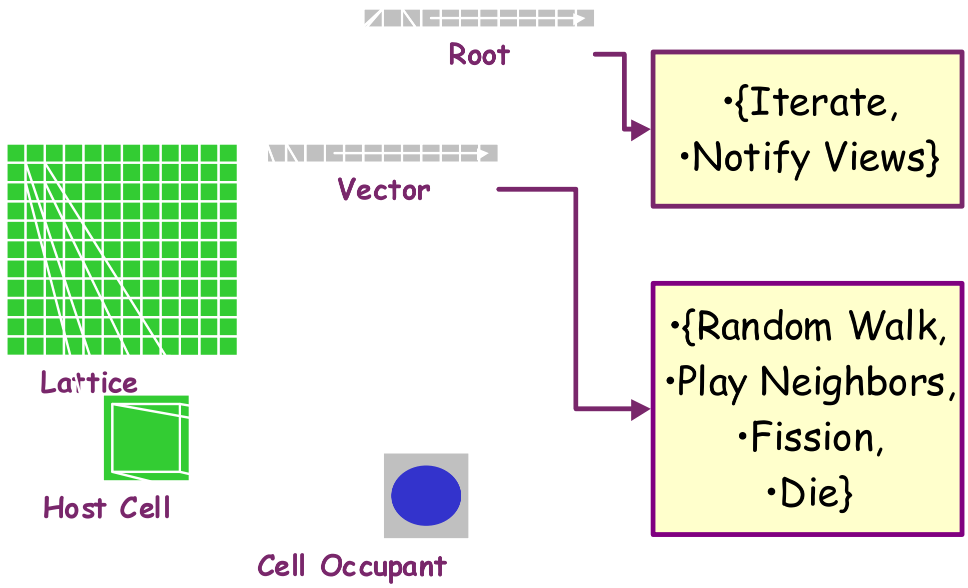
Structure

- Example: Long House Valley



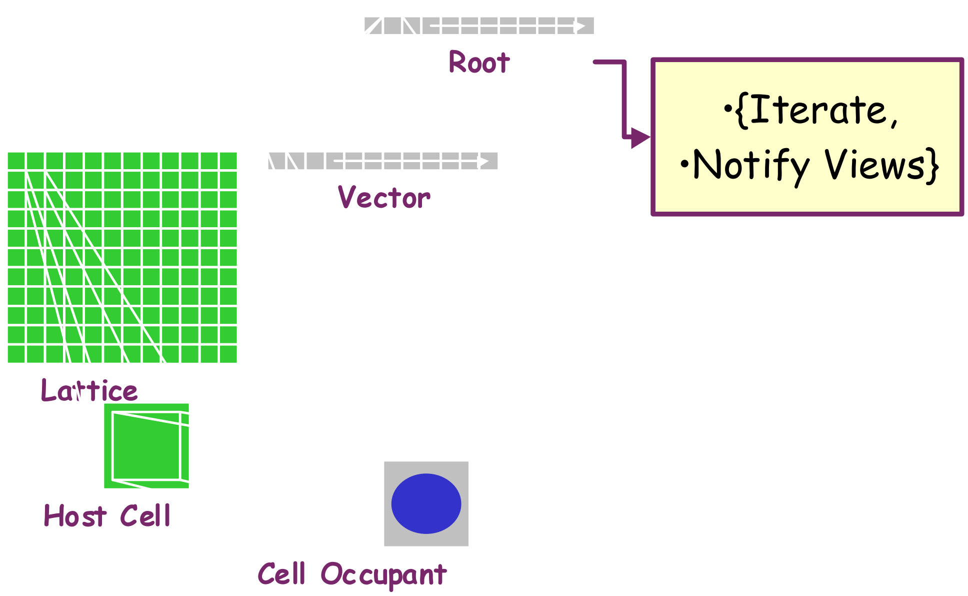
Behavior

- Rules belong to Scapes



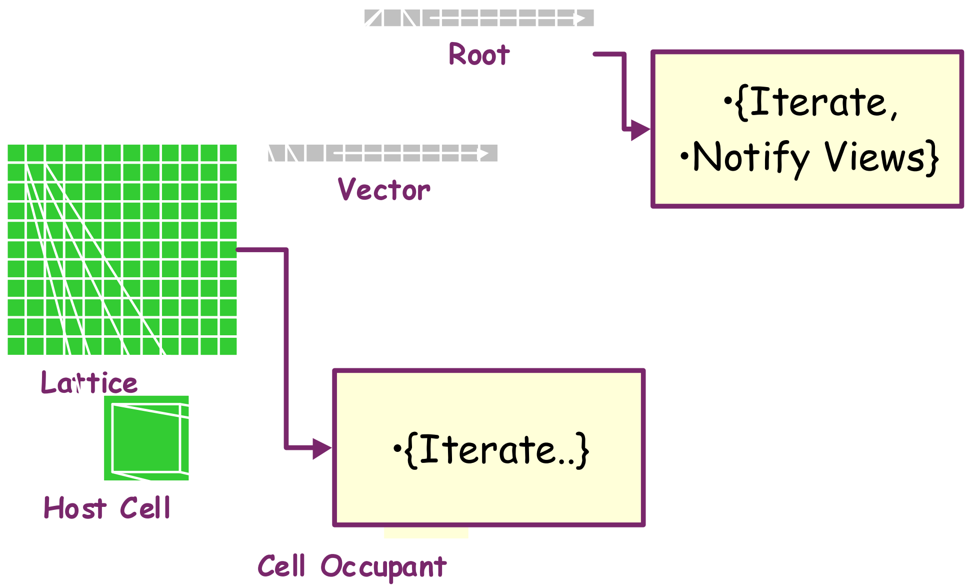
Behavior

- Eat your own dog food



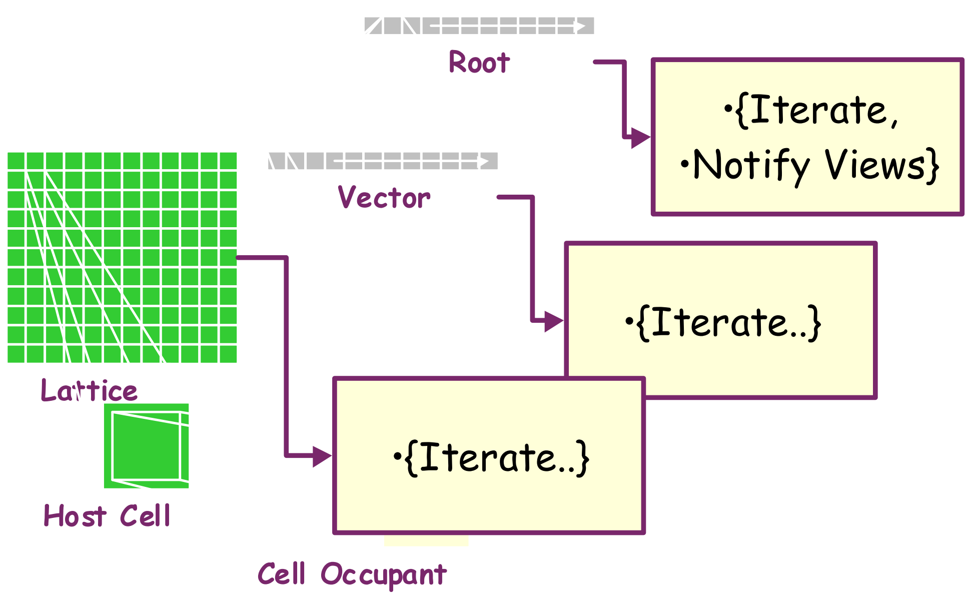
Behavior

- Rules can propagate



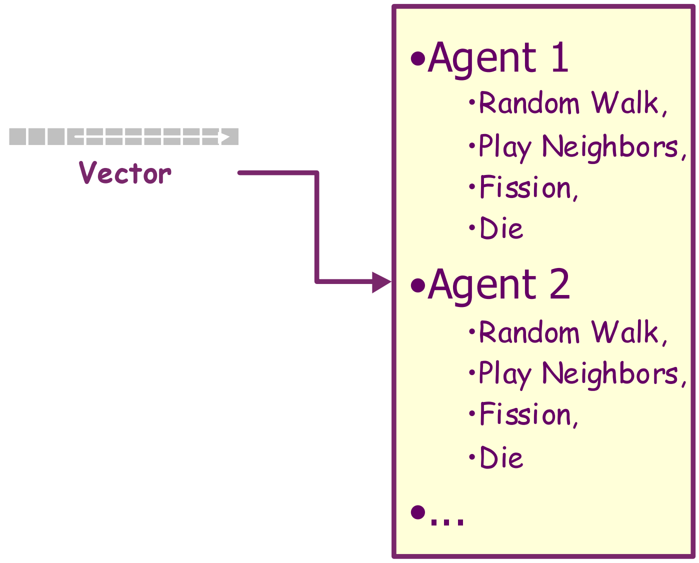
Behavior

- Rules can propagate



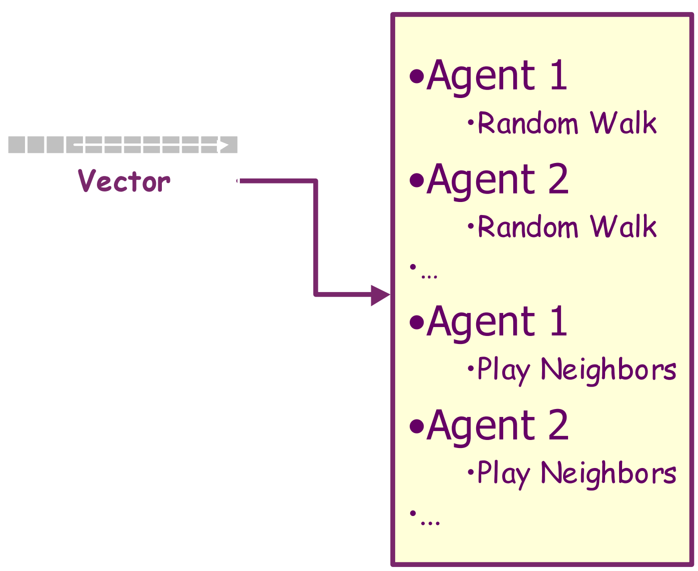
Behavior

- Rules can be executed in series



Behavior

- Or rules can be executed in parallel

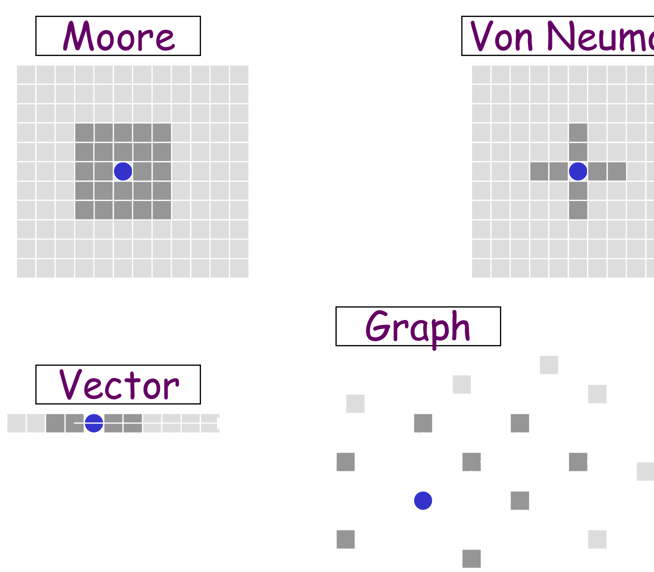


Behavior

- Rules can provide information about themselves
  - Is random
  - Can cause removal
- Execute and Update
  - Example: Diffusion

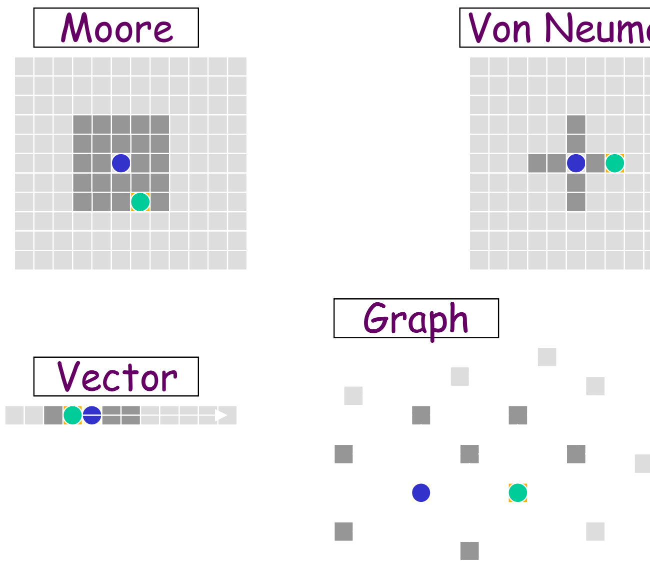
Generalization/Abstraction

- myCell.getCellsNear(2);



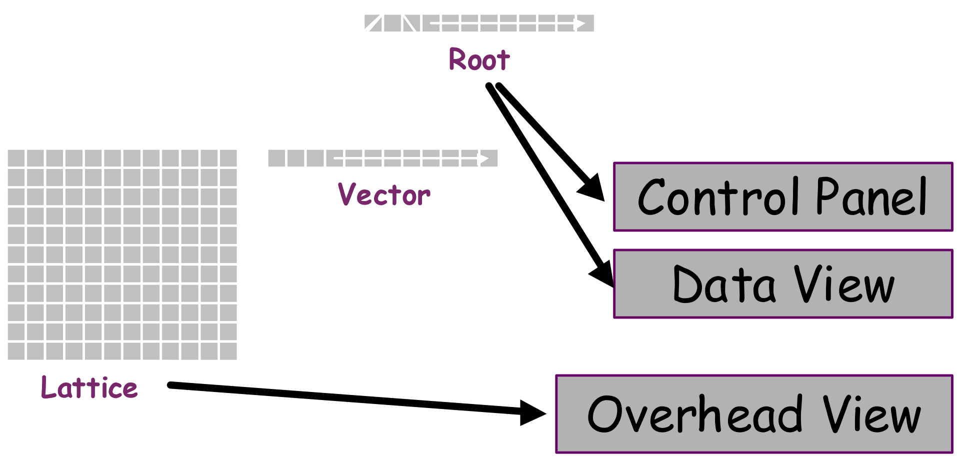
Generalization/Abstraction

- myCell.findMaximumWithin(FOOD, 2);



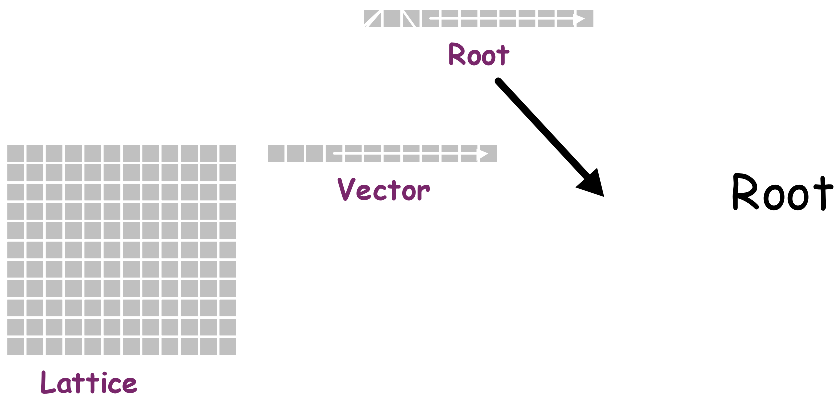
Observation / Control

- Views belong to Scapes



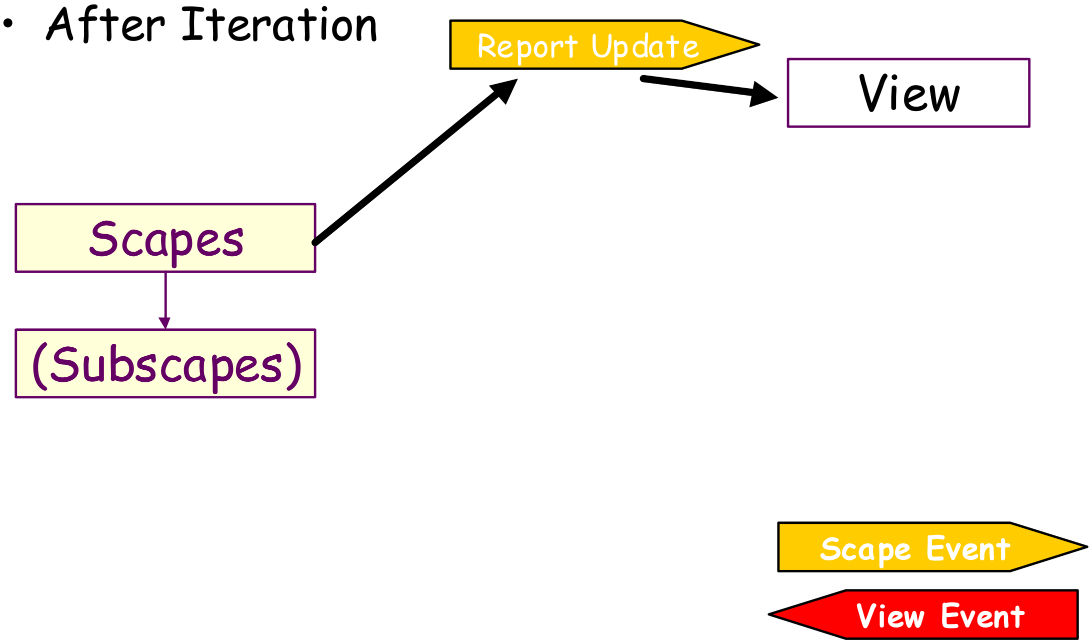
Observation / Control

- Scapes can be their own views



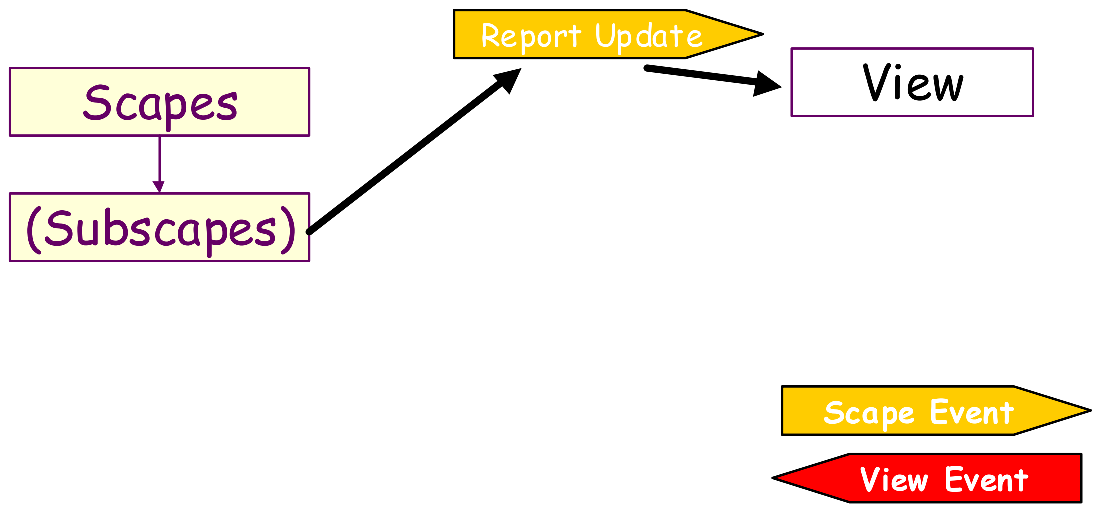
Observation / Control

- After Iteration



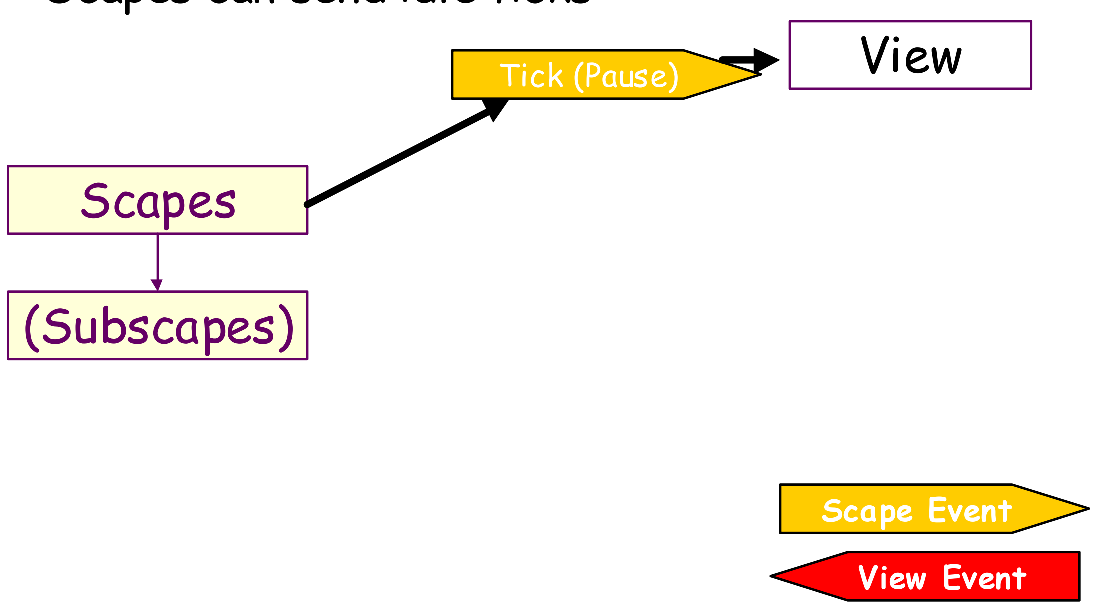
Observation / Control

- Subscapes report to their own views



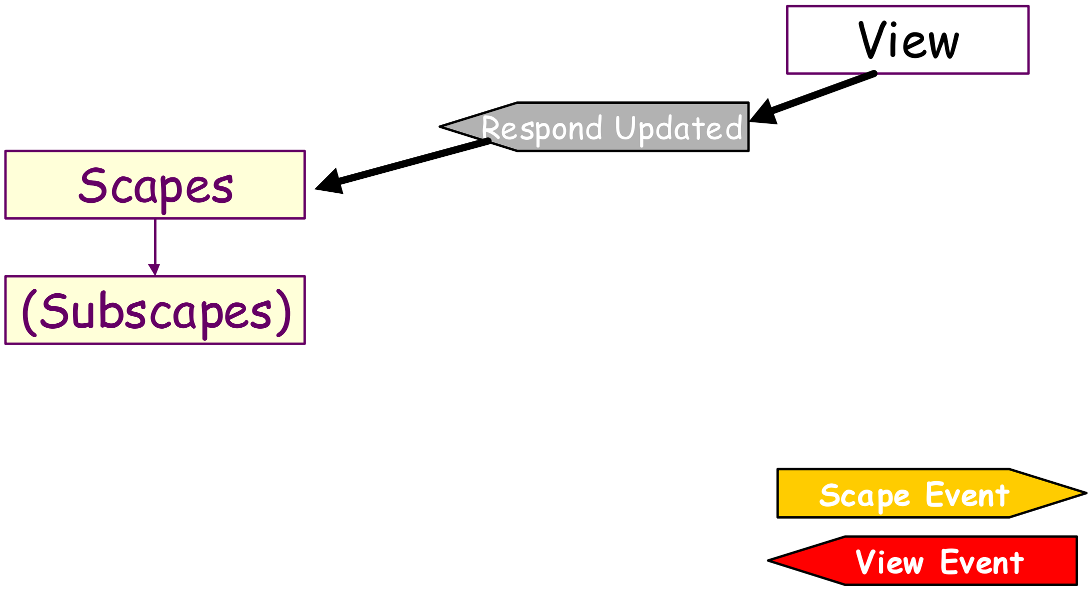
Observation / Control

- Scapes can send idle ticks



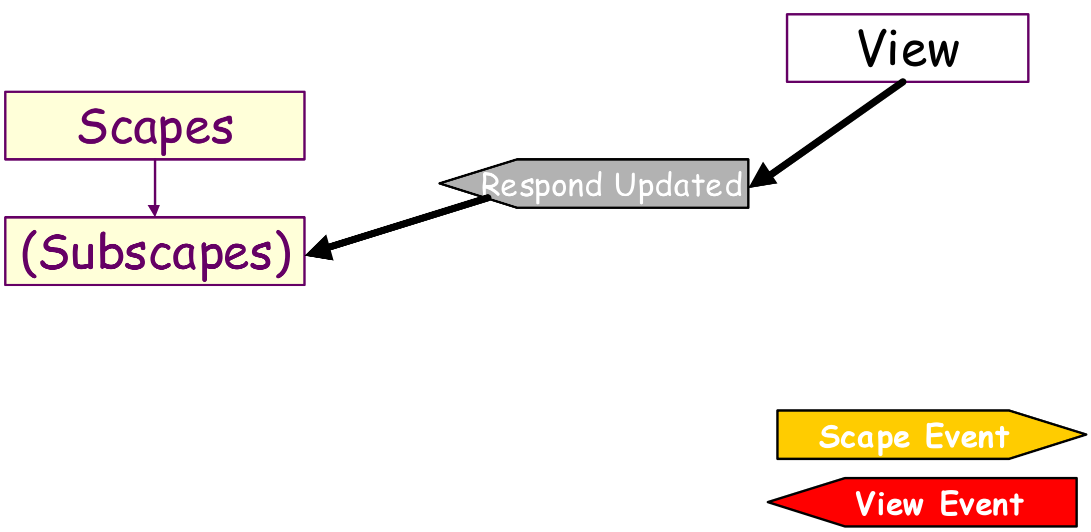
Observation / Control

- When views have finished updating...



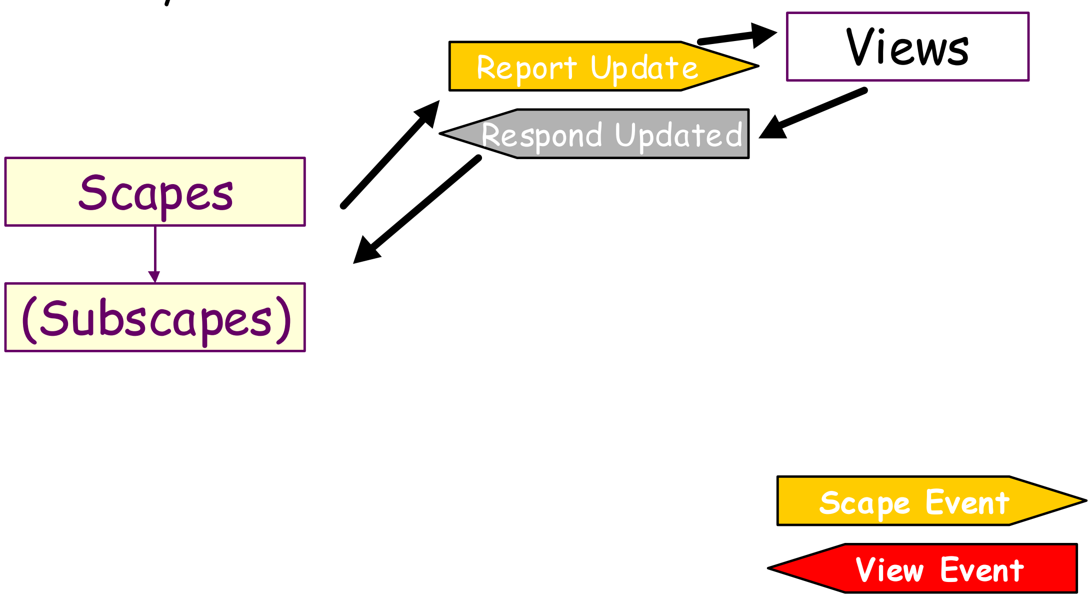
Observation / Control

- ...And all subscape views have responded to their scapes



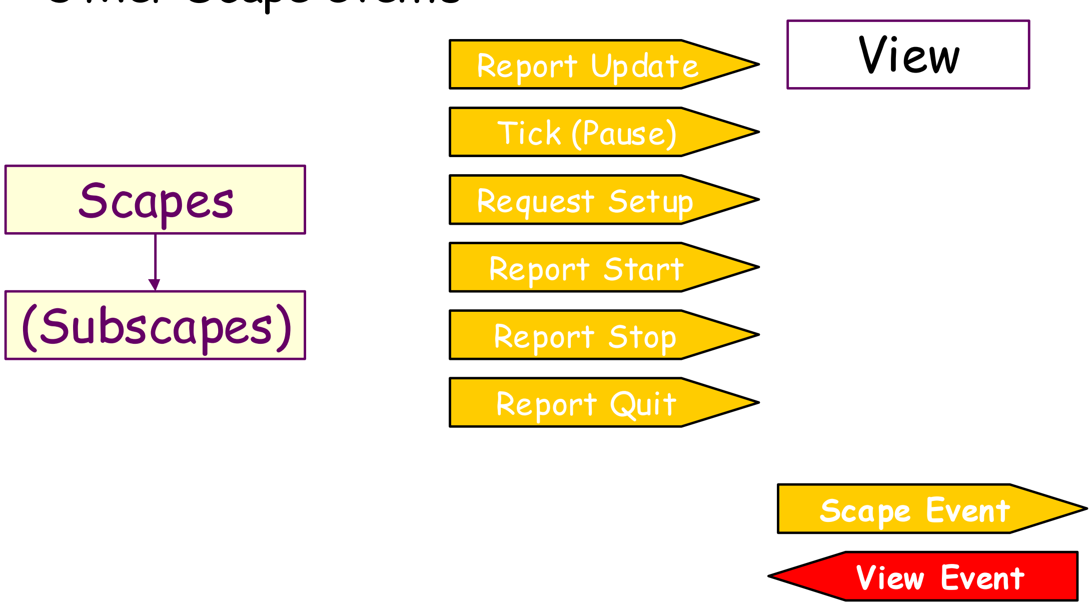
Observation / Control

- The cycle continues



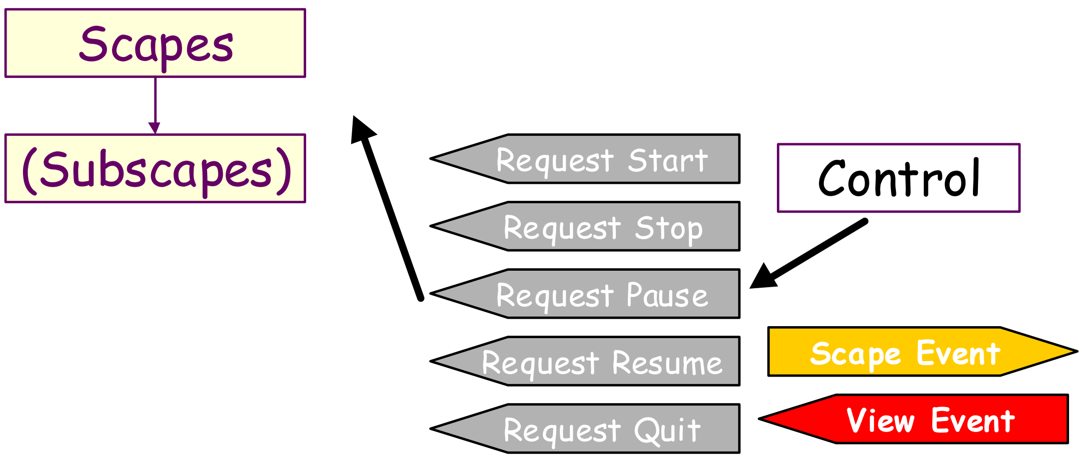
Observation / Control

- Other Scape events



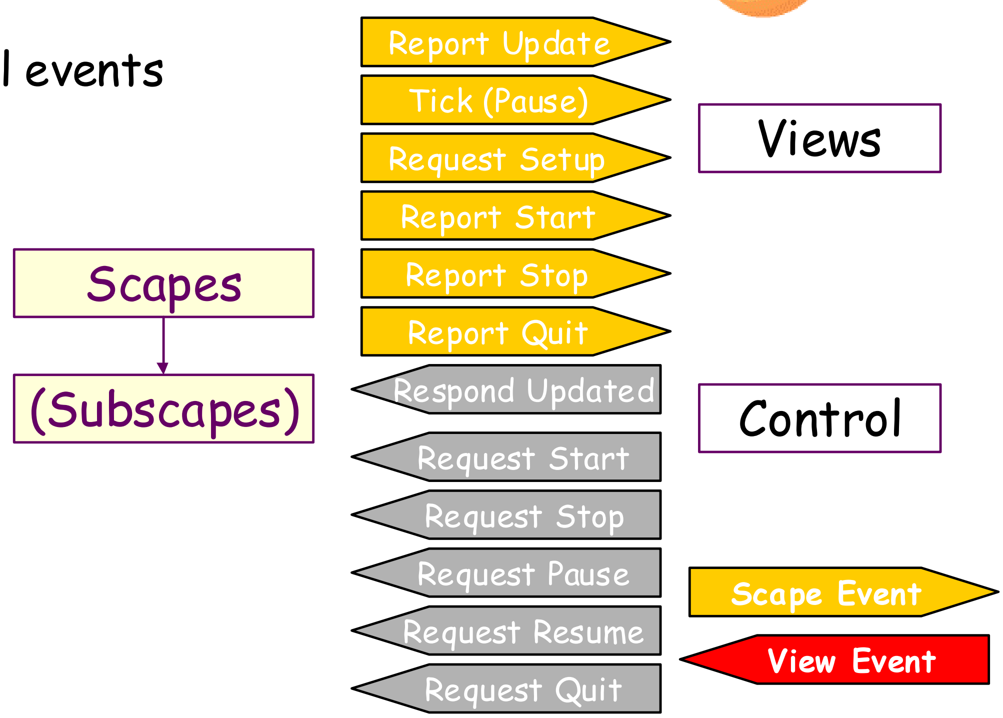
Observation / Control

- Control events



Observation / Control

- All events

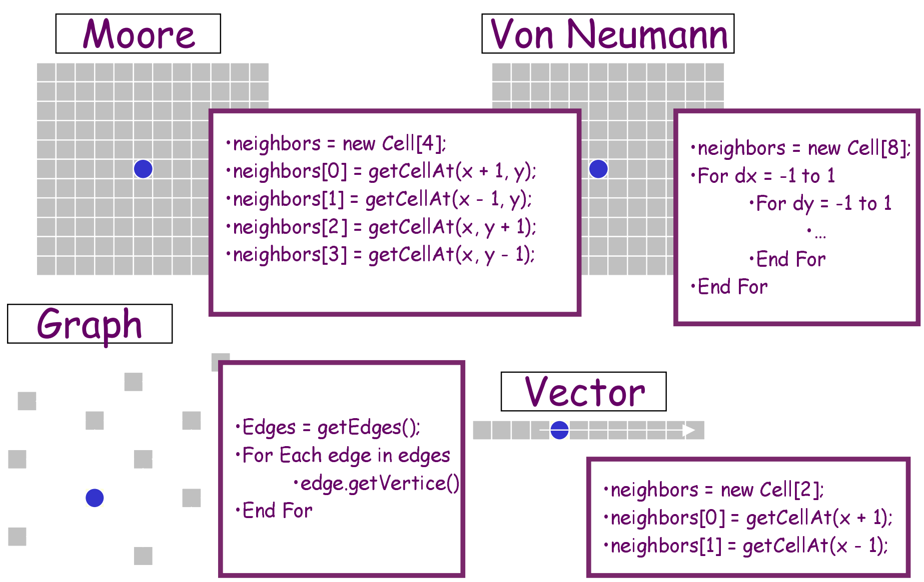


Performance Optimization

- The best optimizations are natural outgrowths of abstract design
  - Example: storing neighbors

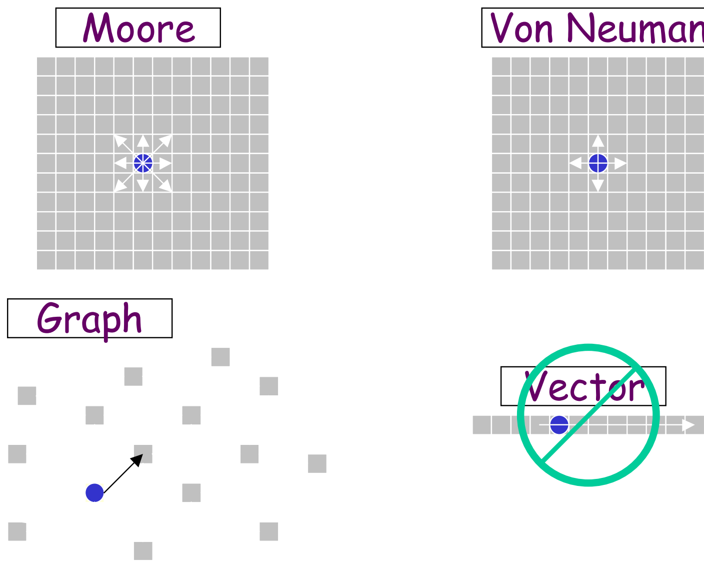
Performance Optimization

- Example: storing neighbors



Performance Optimization

- Example: storing neighbors

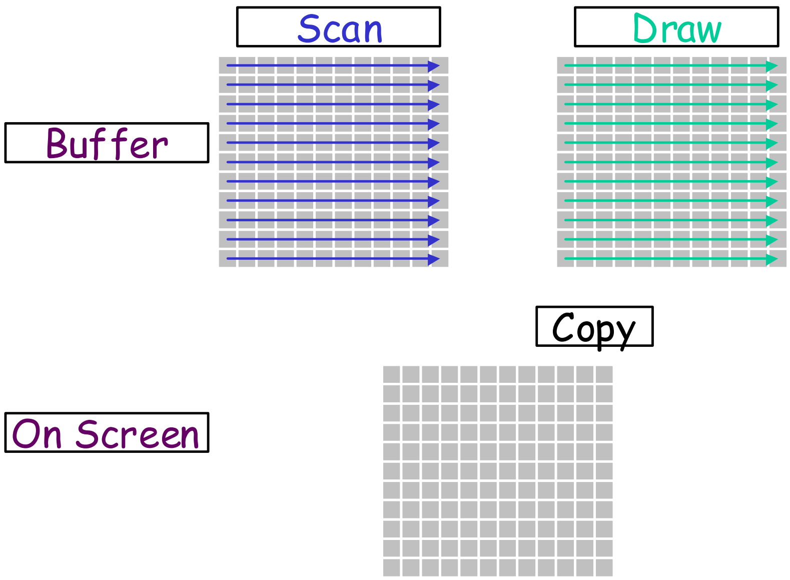


Performance Optimization

- But others are not so obvious
  - Profiling can help
  - Example: Factor of a hundred improvement by eliminating unnecessary object instantiation
- In general, shouldn't be model developers job: frameworks allow optimizations to be transparent to users
- Still some areas where developer's intervention is needed, but make it optional
  - Example: Draw Updates

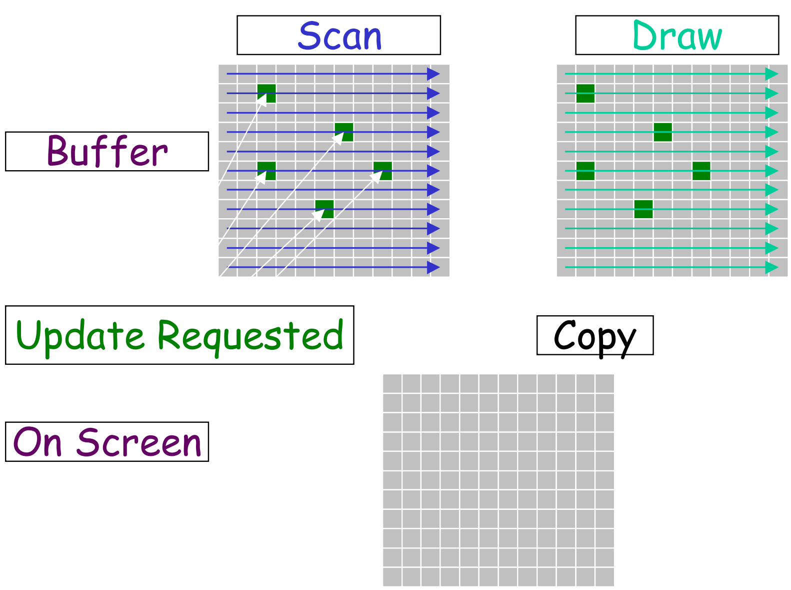
Performance Optimization

- Example: Draw Updates



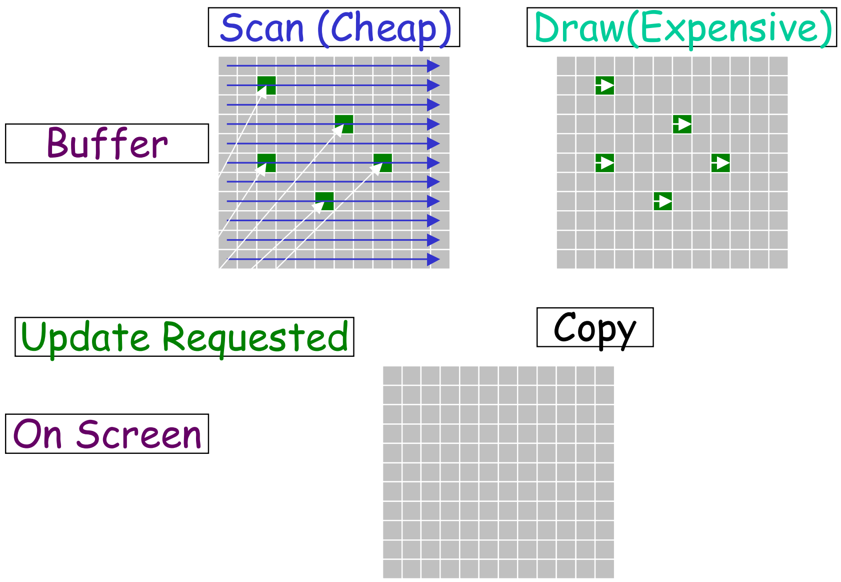
Performance Optimization

- Example: Draw Updates



Performance Optimization

- Example: Draw Updates



Demonstrations

- Prisoner's Dilemma
- Norms
- Firms
- Long House Valley

Short-term Goals

- Structure and rule browsing
  - Hierarchical (Explorer) style views
  - Ability to change any model state on the fly
  - Ability to easily add, remove and reorder rules
- Support for graphs
  - (Just waiting for a good model)
- Complete abstractions for all common graphs
- More rules and searching methods
- Overhead view enhancements

Mid-term Goals

- Mapping to higher level descriptions (XML)
- Switch to enumeration model throughout algorithms
  - Cleaner
  - More amenable to abstraction
  - May be significant performance advantages
  - A bit more of a conceptual load
- New view styles
  - Image view already exists
  - 3D views, zooming views etc.

Long-term Goals

- Non-discrete time and space
- Different time and space resolution
- Parallelization / SIMD
  - All in the engine

Long-term Goals

- Scalability
  - Many, though not all, of our models have fairly small populations
  - Those with very large populations need special case anyway
    - Trade models with over a million agents interacting
  - Rely on technology to innovate or grow us out of problems
  - If that doesn't work…

Long-term Goals, cont.

- Scheduling sophistication
  - Our models typically use static, homogeneous rules that are iterated against whole populations

StarLogo: BUILDING A MODELING CONSTRUCTION KIT FOR KIDS

A. BEGEL, Massachusetts Institute of Technology*

ABSTRACT

StarLogo is a programmable modeling environment intended for educational use by kids ages 12 and up. Because of its student-friendly orientation, StarLogo is particularly suitable for researchers and modelers who are novice programmers and nonexpert computer users.

This talk discusses the design methodology that we used to create the different versions of StarLogo. This includes the evaluation of several alternative models of parallelism (and the emulation of parallelism on single-processor computers), three methods for parallel communication between agents, and a few parallel debugging techniques. Each of these subjects is illustrated with some examples. We then talk a bit about our most recent StarLogo workshop, held this past summer at the Santa Fe Institute. Finally, we describe our current Java implementation of StarLogo and where we are taking this toolkit in the future.

* *Corresponding author address*: Andrew Begel, now at the University of California, Berkeley, Computer Science Division, 387 Soda Hall #1776, Berkeley, CA 94720-1776; e-mail: abegel@cs.berkeley.edu.

StarLogo

Building a Modeling Construction Kit for Kids

The StarLogo Team at MIT:
Prof. Mitchel Resnick
Brian Silverman
Andrew Begel
Bill Thies
Vanessa Colella

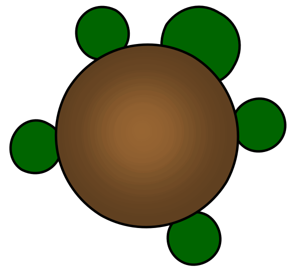
Andrew Begel
University of California, Berkeley
Agent Simulation Workshop
October 16, 1999

Big Ideas

- StarLogo: a programmable modeling environment
- Intended for nonexpert users and non-programmers
  - Great for kids, great for researchers!
- Emphasis on decentralized behaviors with local interactions

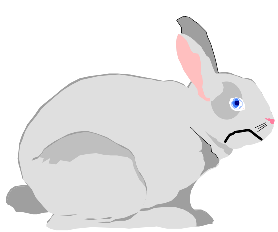
Talk Outline

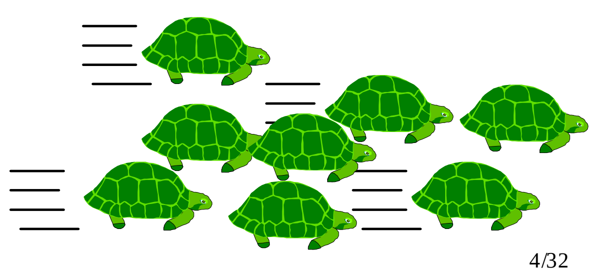
- History of StarLogo
- Models of Parallelism
- Parallel Communication
- Parallel Debugging
- StarLogo Workshop
- StarLogo for Java



History

- 1990's: *Logo on the Connection Machine 2 (a massively parallel computer)
- 1994: MacStarLogo on 68K and PPC Macs
- 1999: StarLogo in Java



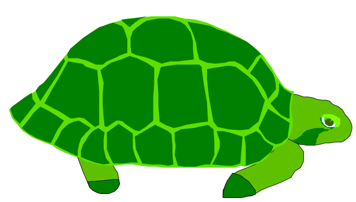


Logo

- Developed by Feurzeig and Papert in 60’s
- Based on Lisp
  - Simpler syntax
  - Incorporates elements of natural language
- Interactive programming environment

Turtle Logo

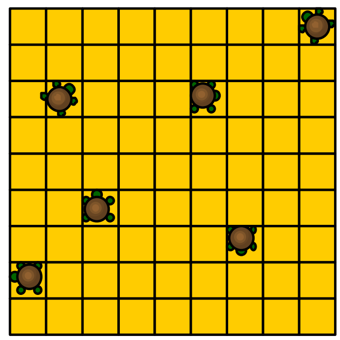
- Turtle can move around a grid-based world
- The turtle is an “object to think with”
  - Body syntonics
- Example code:



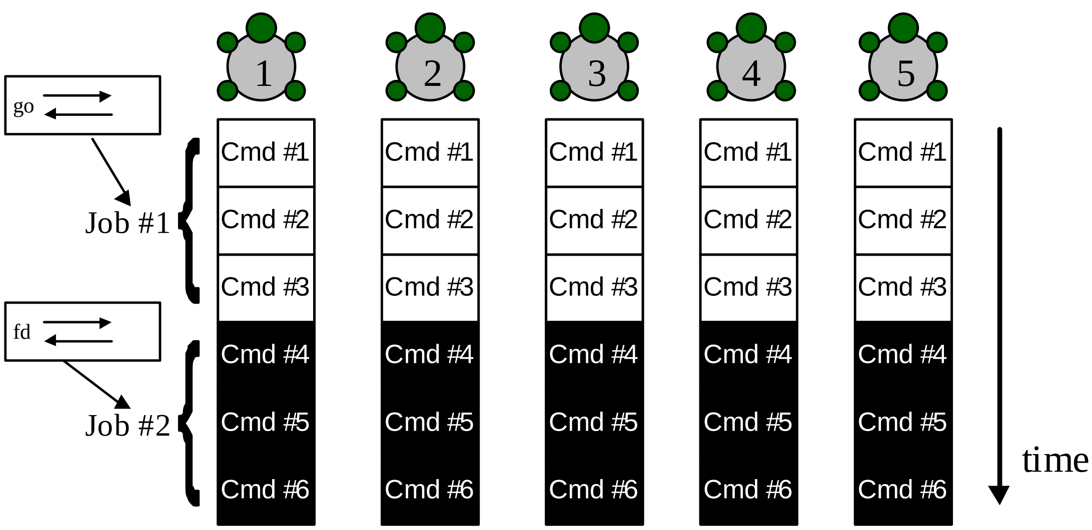
```
to square
pendown
repeat 4 [forward 10 right 90]
end
```

StarLogo

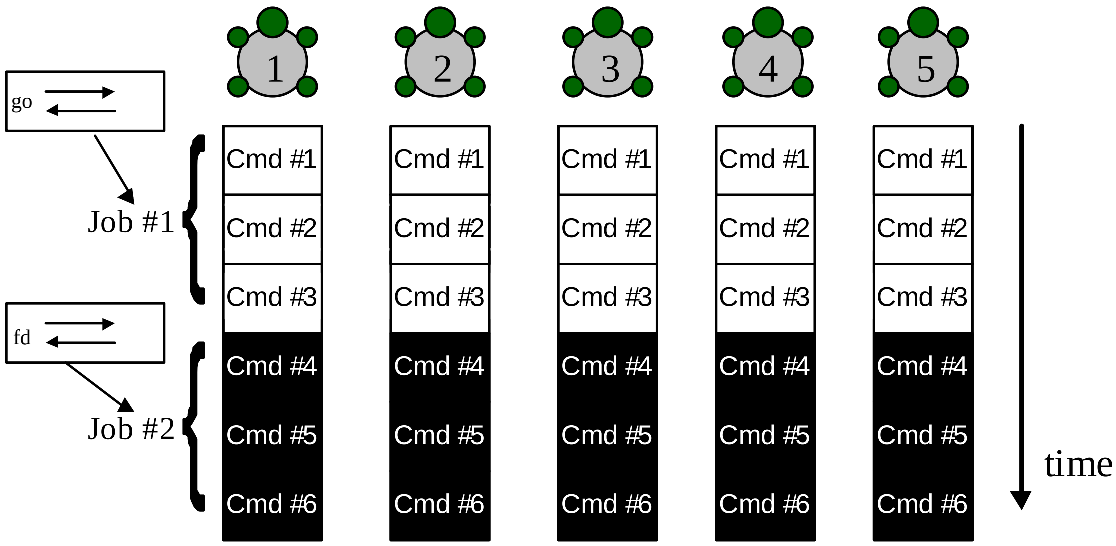
- Thousands of turtles instead of just one (can be organized in groups called *breeds*)
- Background grid of *patches* can run Logo code
- The user is the *observer* and can discover and modify global characteristics of the model



StarLogo Parallelism



CM2 *Logo Parallelism (SIMD)

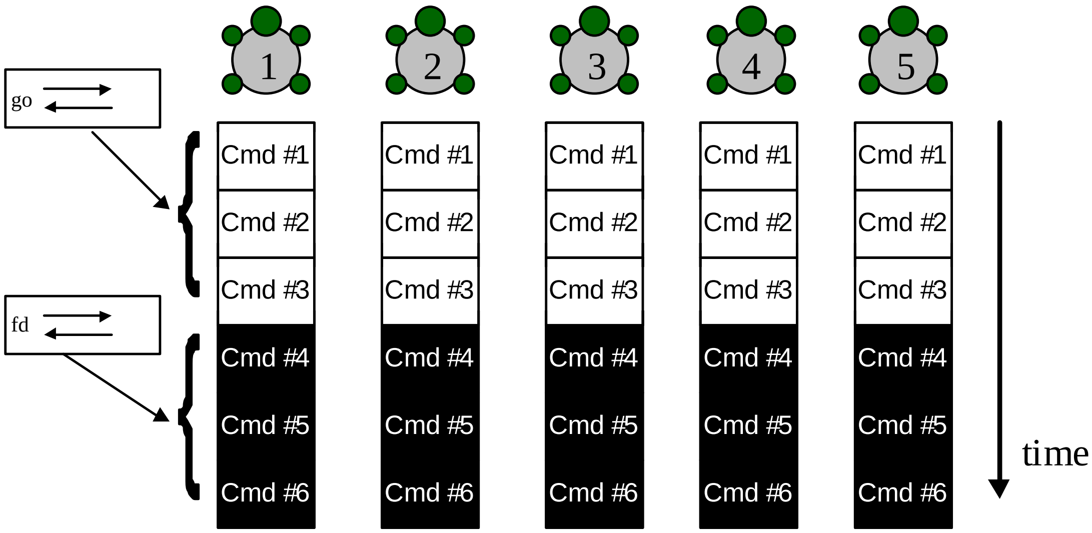


Turtles run commands in lockstep
Each job executes in series

Simulating Parallelism

- How do you simulate parallelism on a computer with one processor?
- Our goal is realistic-looking parallelism
  - Preemptive multi-threading
    - Switch threads every *n* milliseconds
  - Cooperative multi-threading
    - Switch threads at carefully chosen program points
    - Fine-granularity vs. coarse granularity
    - We context-switch after each command, but not each reporter

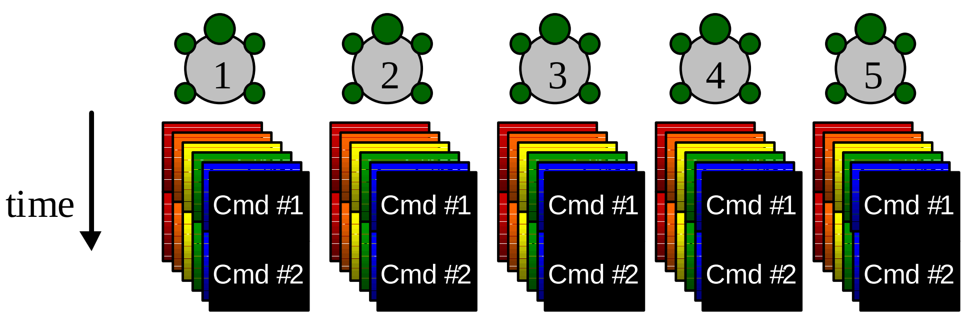
MacStarLogo Parallelism



Each job executes in series
Turtles are switched one after another
Turtles may get out of sync

11/32

StarLogo for Java Parallelism



All jobs are scheduled in parallel
Commands are switched one after another
Jobs may get out of sync

Patch Parallelism

- CM2:
  - All patches execute the same code in lockstep
- Mac:
  - Each patch runs through the code one by one
  - Context-switch after each patch has finished
- Java:
  - Patches may no longer run code

Observer Execution

- There's only one observer
- It's like a lifeguard sitting in a high chair at (0, 0)
- May view and modify global characteristics of the model
  - Create turtles
  - Gather statistics about turtles and patches
- Performs various auxiliary functions:
  - Plotting, movies, file I/O, data collection

Putting It All Together

- In MacStarLogo, how do we run the turtles, patches and observer?

Forever buttons:

In a loop,

– Run turtles as many times as you can for 1/60th of a second

– Run patches once

– Run one observer forever button

Command Center and Buttons:

• Observer code interrupts loop

• Turtle or patch commands are run after forever button code has finished running once

• Only one command center function may be running at any time

Putting It All Together (2)

- In StarLogo for Java:
  - All jobs are scheduled in a round-robin queue
  - Each job has equal priority
  - Forever buttons are the same as normal buttons, but the code has a **`loop [ button-code ]`** around it
  - Monitors spawn jobs, too
    - While anything is running, monitors are run in a loop with a **`wait`** ***`delay`*** at the end
    - When everything stops, monitors are run once more to show current values

Model Timing

- How do you relate "real time" (in seconds, minutes, hours, days or years) to "model time" (in observer/turtle commands)?
- Answer: It's not easy.
  - StarLogo is *qualitative*, not *quantitative*
  - One idea: Use the observer to time how long the turtles take to finish one cycle

Parallel Communication

- Goal: Turtles must communicate with each other
  - Message passing
  - Action at a distance
- How can we do it?

1. Set a global variable
2. Set a patch variable
3. Set a turtle variable

Communicating Through Globals

- Visible from anywhere in the world
- There's only one copy, so it better not change quickly (else only monotonically) in order that all turtles have a chance to see it
- Example (next slide)

```
globals [season [fall winter spring summer]]

to go
every 10 [change-seasons]
end

to change-seasons
case season
  [fall   [set season winter]
   winter [set season spring]
   spring [set season summer]
   summer [set season fall]]
end

to grow-grass
case season
  [spring [repeat 100 [plant-grass]]
   summer [repeat 85  [plant-grass]]
   fall   [turn-all-grass-brown]
   winter [kill-all-grass]]
end
```

Communicating Through Patches

- Only visible on that patch
- Useful for communicating information to all turtles on that location (i.e., infection)
- Example (next slide)


```
patches-own [sick-here?]
turtles-own [sick?]

to infect
  ifelse sick?
    [set sick-here? true]   ;; I'm sick.
    [if sick-here? [set sick? true]] ;; healthy
  wiggle
end

to wiggle
  right random 100
  left random 100
  if sick? and count-turtles-here-with [sick?] = 1
    [set sick-here? false]
  forward 1
end
```

Communication via Turtles

- Directly read and modify other turtles' variables.

```
turtles-own [dead?]

to kill :turtle-id
  set dead?-of :turtle-id true
end

to check-if-dead
  if dead? [die]
end
```


Turtle-Turtle Communication Issues

- Must be able to find a turtle to talk to it
  - `one-of-turtles-here, one-of-frogs, one-of-turtles-with [color = red]`
- Must remember its name to talk to it more than once
  - i.e., `one-of-turtles-here` changes over time
  - Other turtles *never* stop moving
- Communication is asymmetric
  - Just because turtle #1 talks to turtle #2 doesn't mean that turtle #2 talks to turtle #1

Example: Turtle Mating

- Buggy MacStarLogo code:

```
breeds [girls guys]
turtles-own [father-color mygene child-gene]

to procreate
ask-girls
    [if count-guys-here > 1
       [setfather-color color-of one-of-guys-here
        setchild-gene
          combine mygene mygene-of one-of-guys-here
        hatch [ifelse (random 2) = 0
                   [setbreed guys]
                   [setbreed girls]
                setmygene child-gene
                setcolor father-color]]]
end
```


Example: Turtle Mating (2)

- Correct MacStarLogo code:

```
breeds [girls guys]
turtles-own [partner father-color mygene child-gene]

to procreate
ask-girls
    [if count-guys-here > 1
       [set partner one-of-guys-here
        setfather-color color-of partner
        setchild-gene
          combine mygene mygene-of partner
        hatch [ifelse (random 2) = 0
                   [setbreed guys]
                   [setbreed girls]
                setmygene child-gene
                setcolor father-color]]]
end
```

Grab!

- StarLogo for Java

```
breeds [girls guys]
turtles-own [mygene child-gene father-color]

to procreate
if breed = girls
   [grab one-of-guys-here
        [set father-color color-of partner
         set child-gene
           combine mygene mygene-of partner
         hatch [ifelse (random 2) = 0
                  [set breed guys]
                  [set breed girls]
               set mygene child-gene
               set color father-color]]]
end
```


Parallel Debugging

- In MacStarLogo, with 2000 turtles, how do you figure out if something went wrong?
  - Stack overflow (too many nested functions) and divide by zero in turtles and patches are ignored
  - Unexpected behaviors due to not knowing how the compiler interpreted your code
  - Look at turtle or patch state:
    - Oops, no `print` capability for turtles or patches
    - Use turtle monitors to view all variables for a turtle
    - Use command center to ask turtles or patches to set observer variables (or set turtle variables that are visible from the turtle monitor)

Parallel Debugging (2)

- Java StarLogo
  - Simpler programming model (separate turtle and observer procedures) to eliminate certain kinds of programming bugs
  - Turtles and observer can use **`print`** (output shows up in the appropriate command center)
  - Runtime errors in turtles and observer pop up in a dialog box
    - (What happens if all 2000 turtles have the error? 2000 dialog boxes?)
  - Much better compiler error messages. They even report the line number of the error!
  - Turtle monitors and patch monitors will be added soon

29/32

StarLogo for Java: New Features

- Works on PC, Mac and Unix!
- Rectangular (non-square) patch grid
- Turtles and observer can play sounds
- (**`count, one-of, list-of`**)-(**`turtles, breeds`**)-(**`here, at, towards`**) reporters
- 64-bit double math
- Unlimited number of turtles and number of variables
- All math and list operations work for both turtles and observer
- New primitives: **`case, let, loop, wait-until, random-gaussian, pick, kill, nmin, nmin4, nmax, nmax4, diffuse4`**

30/32

Workshops

- Teacher and student workshops held at Santa Fe Institute in Summer ‘99
  – *Learning through Adaptive Agent Computer Models* (Pictures: *http://www.taumoda.com/web/sfi99/*)
  – Run by Vanessa Colella, Eric Klopfer and Monica Linden from MIT; Larry Latour from U. Maine; and Nigel Snoad from SFI
  – Project building (StarLogo Workbook Challenges)
  – Group activities (StarPeople)
  – Predator/prey badge activity

31/32

What’s next?

- January 2000: Finish StarLogo for Java 1.0
  – Plotting, shapes, paint tools, turtle and patch monitors, output and information windows
  – StarLogo Project Web Player
  – GIS support
- Finish StarLogo Workbook

- For more information:

http://www.media.mit.edu/starlogo

32/32

USING A RULE-BASED PROGRAMMING LANGUAGE TO MODEL RULE-BASED BEHAVIOR IN AGENT SIMULATION MODELS

R.J. GAYLORD, University of Illinois*

ABSTRACT

The basic building blocks of agent-based modeling are individual agents, typically characterized by a list of their attributes, behaving according to simple rules or heuristics. Mathematica contains a programming language that is rule-based, provides list manipulation capabilities, and is therefore well-suited for writing agent-based simulation programs. A detailed explanation of how the Mathematica language works will be given and its use in agent-based simulation will be illustrated by various models.

INTRODUCTION

My intent is to make the case that Mathematica is a productive product for the development of agent-based simulations. Mathematica should not be regarded as a toolkit, or an environment: it's a programming language. In *Simulating Society* (1998), we looked at all social models of interest and simulated them in Mathematica as cellular automata.

In general, dedicated software environments for agent-based modeling are inflexible. They set up a system and you have to work within it; change is either difficult or impossible. Dedicated agent toolkits, for example, tend to have a strong spatial component that is present whether you use it or not.

ISSUES

Schelling Tipping Model

The Schelling (1978) tipping model is considered a prototype of agent simulation. It's constructed from 2-D identity nodes, not cellular automata. You move to the nearest location where you are satisfied with the company of your neighbors. However, even if the tipping criterion is minimal, that you need only one similar neighbor, the result is still segregation.

This is considered an exemplar of agent-based modeling, but, actually, it is less convincing than it might seem. In particular, there are two indications of a problem: (1) there is little experimental evidence indicating that the results are true, and (2) generating "an interesting result that we didn't expect" does not speak to the verisimilitude of the model. The source of these problems may lie in the spatial nature of the model. Spatial patterns have the potential to obscure important social network effects that are unrepresented in the model.

* *Corresponding author address*: Richard J. Gaylord, University of Illinois at Urbana-Champaign, Department of Materials Science and Engineering, 1304 W. Green St., Urbana, IL 61801; e-mail: gaylord@uiuc.edu.

Issues for Spatial Models

In a spatial model, whether in math or agent-based models, boundary conditions and initial conditions must be well defined and well understood. For example, what does a wraparound boundary (e.g., in a cylindrical topology) mean? The boundary is supposed to be infinite, but it clearly isn't. Agents actually wrap around into a counterposed quadrant that will have its own density and dynamics.

Spatial models would be improved by borrowing two rules from physics:

- Two objects cannot be in the same place at the same time.
- Properties of the space may result in objects not having equal mobility.

Empirical research indicates that social neighbors may be as significant as spatial neighbors. This raises issues that contemporary approaches to agent-based simulation must address.

Agent-Based Modeling as Computational Science?

I agree with Rob Axtell (Axtell 2000) that it is important to distinguish agent-based modeling from computational science. Agent-based modeling is implemented through the programming of numerous autonomous agents. But computational social science, like computational physics, might include numerical solutions, diffusion equations, and many other forms of mathematical modeling. Thus, there is a difference: computational social science will be much broader than agent simulation.

Psychology in Agent-Based Models

A model by Alan Kirman (1993), an economist in France, that replicates sudden shifts in opinion and action in a group of individuals has attractive properties. The model illustrates how the "objects" in social science models differ from "objects" in physical models: they are not only influenced by neighbors but also can make decisions, change identity, etc., in reaction to forces from within. Kirman shows how this principle applies to the evolution of fads in areas as different as fashion and the stock market.

The stock market is not usually viewed in terms of people wanting to fit in, but it can be seen from the same perspective as fashion: "fundamentalist" investors decide on their own, while "chartists" track moving averages (which are essentially a record of what others are doing) and follow the crowd. Thus stock market bubbles are identical in nature to the rise and fall of the hula hoop.

Social science has to be very careful about when to make psychological assumptions, just as a chemist must in deciding how to incorporate physics.

MATHEMATICA SHELLS FOR AGENT-BASED MODELING

We have developed three types of shells in Mathematica for use in agent-based modeling: (1) cellular automata, (2) social networks, and (3) binary interactions (e.g., recruitment). As an example, I will describe the third type of shell.

Shell for Binary Interactions (#3)

Define a social system based on a list of people, each with two characteristics.

1. First characteristic: a person can have one of two beliefs, represented by a 0 or 1.

2. Second characteristic: a person makes decisions in one of two ways, independently (0) or in response to persuasion (1).

3. Agents are randomly paired.

4. Rules of interaction are as follows.

There are three types of possible change. Change of an earlier type forestalls the activation of change of a subsequent type.

- If in this time step the person has made an independent decision, then in this time step the person cannot be influenced.

- Spontaneous change:
  a. If belief is 1, some probability of spontaneous change to 0.
  b. If belief is 0, some probability of spontaneous change to 1.
  c. If neither a nor b happens, belief stays the same.

- Influenced change:
  a. If other person's belief is different, some probability of adopting that belief.
  b. If the other belief differs, there is a calculable probability that the belief will change.
  c. If the two beliefs are the same, the belief will remain unchanged.

What these rules ensure is that everybody has the opportunity to change and that, in a pairing, there is an opportunity to influence the partner. Various updating schemes can be used.

Frequently, the result is a U-shaped distribution in which a great deal of time is spent in one state before reverting to the other. This pattern is analogous to geometric phase changes in physics, or spontaneous traffic jams.

General Remarks about the Shells

- Cellular automata. This shell operates in a spatial neighborhood. In a "Gaylord neighborhood," (a) no collisions are allowed and (b) simultaneous updating occurs: look at value of site, value of neighbor, and created neighborhood *before* the update. A spontaneous update is possible: e.g., random number at each site; if 1, OK to update.

- Social networks. Feed in a list of friends and specify the nature of their relations. Are they bilateral? Are their ties strong or weak?

- Random couplings. The only task is to program the rules. Undergraduate students, for example, are programming traffic flows based on random couplings.

At one point, Von Neumann said that it's unlikely that a mere repetition of the tricks that served in physics will serve in social science.

There is only one difference between agent-based modeling and theoretical modeling:

- In theoretical modeling, one develops the model, then solves the math.

- In agent-based modeling, one develops the model, then creates a computer implementation.

Mathematica is a language that works well for agent-based models. While there is some loss of speed relative to dedicated toolkits, the simulation model can be developed rapidly. There is a caveat: a prototype written in Mathematica doesn't easily translate into Objective C and, thus, to Swarm. However, the combination of development productivity and expressiveness makes Mathematica a valuable language.

REFERENCES

Axtell, Robert (2000) "On the Varieties of Agent Models in the Social Sciences," *Proceedings of the Workshop on Agent Simulation: Applications, Models, and Tools*. Argonne, Illinois: Argonne National Laboratory.

Gaylord, Richard J. and Louis J. D'Andria (1998) *Simulating Society: A Mathematica Toolkit for Modeling Socioeconomic Behavior.* New York: Springer-Verlag.

Kirman, Alan (1993) Ants, rationality and recruitment. *Quarterly Journal of Economics*, 108 (February): 137-156.

Schelling, Thomas C. (1978) *Micromotives and Macrobehavior.* New York: W.W. Norton.

DISCUSSION:

AGENT TOOLKITS: STRATEGIES AND TRADEOFFS

[*Presentation given by Marcus Daniels.*]

Michael North: I'm wondering, you talked about having a COM, which is Microsoft's linking technology, to connect things like Excel and Word together. And you said that that's inside of Swarm. How much access do the users have to that? Can I add in Word or some other thing as well?

Marcus Daniels: The way that works is via the ActiveX layer to Java, and so you just run a command called Java Active X, and it builds a bunch of stuff for the trusted librarian. Microsoft's virtual machine takes care of loading that all in and providing the COM interface. It's really not anything that we added to Swarm. There's no COM infrastructure per se. But by dynamically loading these classes that are generated via this Active X program, you get COM for free for any Microsoft application, yes.

Nicholson Collier (discussant), to Daniels: Is part of the point of the XML here to provide a general model description kind of language?

Daniels: Yes.

Collier: Okay, I think that's fantastic, because it seems like a lot of these models are made even by the same people. And as you said, there's sort of no connection between the different models. I think Miles [Parker] had some ideas about this at the last SwarmFest, about design patterns of model description language so we can translate in one model from the other — part of getting rid of this private language.

Daniels: Exactly. And UML has a lot to do with that.

[*Presentation given by Miles Parker.*]

Randal Picker: You emphasized the portability of the Java. Can you talk about performance on different platforms? Here's what it is, if I run it on the 266, if I run on a dual-processor NT workstation, if I run on Unix?

Miles Parker: In general, I've found performance to be best on NT or Wintel, even faster than some lower-end Sun workstations I've used. Linux is getting better. IBM's releasing a JDK for Linux, and their JDK for Wintel is fantastic, so it'll probably be pretty good. On the Mac, the performance is getting quite good, especially on the G3 machines.

Picker: Well, so have you run actual performance tests to say, "Here's the different speeds on different machines"?

Parker: Casually. But you know, it's all definitely within a factor of two, and often, with the faster Macs, say, it's probably pretty indistinguishable from a faster Pentium machine, let's say.

North: One of the things that I've found is that with Java the graphics are very slow, as you're talking about. Can you shut down the graphical display so that you can just run the simulation?

Parker: Yes, yes. That's a good idea. As you can see, this isn't as dramatic as, say, if we ran the Longhouse Valley code. Just as an example, this code was originally in C++ on the Mac, and because there was no way to pull out views or anything, you were sort of stuck with the run that you had. And it maybe took six or seven minutes on a relatively fast Mac. If you take out all the views and put the code in Java, it takes — best case — we're getting about 20 seconds. Another issue with Java is that initialization time isn't so great.

Charles Franklin: One of the things that came up yesterday was the issue of experimental design, and I think that's something that both talks didn't get into very much. Naturally, specification of the model is a little important. But it seems to me that for production use by an applied social scientist, the issue is not the dynamic graphs and being able to watch them populate and move. It's fun, but it's not what the core is. The core is establishing the experimental design for your analysis that varies over parameter space, that collects the data and then analyzes the data. So I think Swarm being able to put out to R is one example of that. Would you say a little bit more about that? Is there anything that lets me specify an experimental design, run and collect that entire design, and then stick that out directly into objects that R or S+ or something can read?

Parker: Well, there are two issues there. First of all, the stack collection mechanisms do allow you to do variance and standard deviation on these measures in real time, and you can actually do measures across the data set, too. But the other thing is, of course, you can write out results to a file. And the data model control is just another view, and of course a view doesn't have to be graphical. Right? So you create a view that essentially manages the model, and that's quite straightforward. Does that answer it?

Franklin: Experimental design? I guess my point is that you have to manually run through the entire parameter space that you want to explore. There's no automated mechanism, you're saying.

Parker: Oh, I see what you mean. No, we don't — I mean, it's certainly very straightforward to just do a sweep. This is the idea of leverage, and there are certainly other toolkits in development that are out there that would allow you to do that kind of parameter exploration. And the whole idea is to build tools that are open enough that they could fit in. We obviously don't have the resources to build a really nice parameter exploration harness, for instance. But it is an important issue.

Robert Axtell: Maybe you could describe the Longhouse Valley tool or the RAND tool.

Parker: Yes, right. We're working with a guy who was at RAND out in California, and he's doing some very sophisticated parameter exploration stuff. It's all sort of in development now and not commercially available, I don't think. But the point is that I work closely with him in building just the very few classes that you would need [for Ascape] to interact with that [software]. You build a sort of socket; it's very straightforward.

[*Presentation given by Andrew Begel.*]

David Sallach: I just wondered if there are any plans for making StarLogo open source.

Andrew Begel: Yes, the Java version will be open source by the time it's released.

North: It seems that StarLogo is very useful for education, but you also mentioned that you can use it for some more general types of modeling, for instance, starting with prototyping. And you also mentioned that you don't have to be a rocket scientist to understand that. I'm sure the rocket scientists feel threatened by that.

Begel: Well, some people feel threatened if they can't use C. We find this among MIT undergraduates. They feel definitely oppressed being forced to use a simpler language than they're really used to, but what you get out of it is that the projects end up being developed much more quickly and are much easier to modify.

North: Have you put together any sorts of educational programs or documents that we could use to put together simple courses to teach people the basics of agent modeling?

Begel: We've done stuff specific to StarLogo. We have the StarLogo workbook. We've done a lot of workshops in schools around the area and also at Santa Fe. There are a lot of professors who are teaching undergraduates and also high school teachers who are teaching their students to use StarLogo in their modeling of biological systems, in physics and chemistry, and all sorts of different areas. We don't have a curriculum-developing team at this point. But we do have documentation, like *Getting Started with StarLogo*. There's this quick guide, which is essentially a list of short little things you can do in StarLogo in five minutes, and each line is like a whole new program by itself. It's a fun thing to play with, and that goes on for several pages, so that provides endless hours of fun and joy. [*Laughter from audience*.] And then beyond that there's reference manuals and stuff, and then there's always e-mail lists where people ask lots of questions. If you're not used to programming in StarLogo, I'd say to start with the Mac version, if you have a Mac. The documentation is all updated for the Macintosh version, where it's not yet for the Java version.

Catherine Dibble: I have a question about trying to do more serious science with StarLogo. I understand that it has very strong advantages for teaching, and for examples and for quick prototyping. I think one of the things that might be a little bit dangerous in terms of trying to go beyond that to the next step of doing more serious modeling is the difficulty with scheduling and synchronization and coordination among the agents — you know, the idea that they walk away when you're trying to do things, so it's hard to schedule. Very often, the coordination of schedules can have profound effects on the behavior of a model, and StarLogo makes it difficult to randomize, for example, to control that at all. And so you have the danger of getting very misleading model behaviors. People become comfortable with it from the teaching end of things, but then try to extend that to doing something more serious. That doesn't mean not to use it for teaching. You just have to migrate eventually.

Begel: Definitely, yes. People run into that, I think, a lot when they start. You can do qualitative models. We had these kids doing this ant model, where ants were foraging for food. And originally they had a queen ant who knew where all the food was telling all the ants where to go. So the ants made a beeline for the food and then came back. Then somebody suggested that maybe some ants are unaware of where the food is and just walk around randomly until they find the food. Then they happened upon the idea of how do you get the other ants to find the food once you've found it? Then they happened on the idea of leaving a trail behind them back to the nest that showed the other ants, if they found the trail, where the food was.

So while you don't necessarily get the right model, what you do get is the ability to try out possible models and see which ones sort of look right, because there's no real way to validate a model. Well, we haven't talked about that yet. I guess that's later today. But things like validating models for kids is probably not so easy, but what you can do is just look at the model as it's running and sort of say, "Well, does that kind of look right? If we have ants in an ant farm, does that look like how they're running?" So, yes, for educational purposes, kids aren't necessarily getting that far, but for adult researchers — we have some researchers who actually are using this for some quantitative models, but they quickly get annoyed.

Dibble: And they need to know to be careful of this, that these are artifacts that can arise, you know.

Begel: Yes. But if you are adding GIS [geographical information system] support, obviously that's not something you do with school kids.

Dibble: And that was part of my concern, too. Because that implies that people might try to use this for more serious modeling and then synchronization [becomes an issue]. And the GIS folks don't necessarily know about that. The modelers won't necessarily be aware of that [concern].

Axtell: What's the relationship between this version of StarLogo and the Tufts version, or the University of Maine version?

Begel: University of Maine extended the Mac StarLogo with lots of extra commands for doing I guess more quantitative types of modeling.

Axtell: It was to correct the things that Catherine mentioned.

Begel: Yes, with better randomness, a lot more plotting tools, and a lot more help functionality. He's much more interested in mathematical models than physics simulations.

Dibble: Is that still being developed?

Begel: I think they're at their last version of Mac StarLogo. I don't think they're going to be developing it much further.

Axtell: But they had a PC version coming out?

Begel: There was a PC version in development from the University of Maine, but that was abandoned about a year ago. And we were working the Java version in parallel, and it turns out that our version was about four or five times faster than the version that they had been building natively for the PC. So it was probably a good thing.

[*Presentation given by Richard Gaylord.*]

Ian Lustick: This was a very provocative talk. I have more questions than you can probably deal with, but I want to list them real quick. One is that I do suggest that on the Schelling model, I think there is evidence out there. In fact, Doug Massey, who used to teach here at the University of Chicago and is now at Penn, wrote an entire book, a very powerful book, *American Apartheid*, that shows how the maintenance and even deepening of segregation levels in American northern cities over the last 50 years helped support that model.

About the traffic flow example — I was interested that you do it in your class. You talk about this arising spontaneously. What I found in my work, and I reported it a bit yesterday, is that it's not quite spontaneous. There are regularities as to when these nonlinear shifts of state occur, and the introduction of slight asymmetries into the environment is a kind of trigger at a much lower level of analysis. And I think your students would find that, as we know in studies of traffic flow, it is the little things that cause rubbernecking. People think it's because everybody's watching. It's just because *one* person watched at one point, or something like that.

Richard Gaylord: Let me just say that physicists have basically invaded traffic modeling, and I thank God I never drive on a highway designed by physicists. It's hard enough to drive on highways that were designed by politicians. But it's a very fascinating field, and I would say this: If you do work with someone [on a traffic model], you should try to make sure that they drive on the same side of the road in their country as yours, because we lost about a month and a half working with the Japanese, because it turned out that we weren't following the same traffic rules.

Lustick: The last question I have is, I think, the most important in a way, and that is the point you raised at the end about what's theoretical work and what's experimental work. And I've had some of these conversations with other political scientists, some rational choice folks — talk about feeling threatened! — who feel threatened by the idea that you could do formal work that's also experimental. And I think it would be worthwhile to look at some of the really good epistemological stuff on social science methods that's part of natural science methodology, because it turns out that there is no "fact" that's freestanding from the theories we use. If all facts are somehow a function of the theories behind the measurement tools, then it's very hard to identify the other exemplar; that is, the place where the data is not partly an artifact of the theory. So I'm not prepared at all to separate the notion of experimentation in a virtual field from experimentation in a nonvirtual field.

Gaylord: And physics has never really solved the problem. And in fact the statement that I would make is what Niels Bohr said when he came over to this country in 1932 and lectured at Columbia University on quantum mechanics. He said, "What I like best about you Americans is that in Europe we spend all our time trying to understand what quantum mechanics means. You Americans couldn't care less what it means as long as you can do something with it." [*Laughter from audience*.] So you know, you're doing an experiment if you're applying to an agency that wants experiments done, and you're doing theory if somebody wants theory. So it really doesn't matter, and I suggest that those things only be thought about in the presence of alcohol.

Tom Baines: I enjoyed your presentation. Those who work with me know that I've never let my ignorance or confusion affect either my ego or my curiosity. So my question is based in my ignorance and confusion. In this model, you make changing opinions a cost-free kind of thing. If there is a cost to changing your opinion, would it show up in the social net function, or how would it be reflected?

Gaylord: Well, of course, this is not my model ...

Baines: Yes, I understand.

Gaylord: ... and we're working on it. One of the things that I'm looking at is a real basic phenomenon — I think it's controversial — and that is consumer lock-in. And I raise this because that's the model that basically says that the more people do things, the more likely you are to do the same thing that they do. And there are historical questions as to whether that's even true, because it was not true with Beta and VHS. One of the things I wanted to put in is, well, what about people who, when everyone starts doing something, they do something different? I don't have an answer, but it is very easy to put that into this program and look at it.

Baines: Yes, I think it would be interesting, because the definition of "friend" could be "do they do things like me?" Or do I want the kind of friends who do nothing like me?

Gaylord: Absolutely. And you can basically show that in code. What I get the biggest kick out of in my class is students who say, "What about this?" I say, "Go ahead and put it in."

Baines: Well, just out of curiosity, if you wanted to make changing opinion have some cost, would you reflect it in the social network function? Is that the way it would come up?

Gaylord: I think at that point you need to talk to the social network specialists, who might say that's where it belongs. And then other people will say that isn't where it belongs. Then run both models.

David Sallach: I have a question that runs a little contrary to your inclinations, which is to solve it all in Mathematica, but you heard the earlier sessions today, and you know that in the toolkits that are out there, there's a tendency to evolve toward Java, with greater or lesser reluctance —the platform independence and other features being desirable. Now, I know that Mathematica has a math link that essentially provides an API with Java. And I was wondering, if you take into context the presentations that you saw, the things that are being done in these toolkits, do you see a natural division of labor between the two in which you might do some parts of the model in a toolkit and then call Mathematica functions or have some shared communication with Mathematica in evolving, say, certain types of complex models?

Gaylord: Okay. Technically speaking, I'm not aware of this. There's a developer's conference on Mathematica next week where this is going to be discussed. As far as I'm concerned, the division of labor is that I write in Mathematica and then find someone else who knows how to do it in Java, and they make the link, but I've been told that it is incredibly simple. Mathematica has the ability to link into C, Fortran. In fact, it links into Excel.

One of the problems in Mathematica is that I have to work with a friend at Wolfram and have him create those nice little graphics that you see up on the Web, for example, at Brookings, where you push a button or you enter an input value and then you just click and it runs. I don't know how to do that. In fact, in the old days in Mathematica, you had to run the entire program, then create all the graphics in Postscript, and then run the graphics, which is really awful, because your simulations get really boring. If everybody dies in the middle of your simulation, you'd sort of like to be able to stop. [*Laughter from audience*.] And now you can actually run it and watch it run and just go in and, as someone said, hit Pause or hit Stop.

But I guess I should mention another technical issue. When people think, "Wow, I could really speed this up if I did it in C and then just exported or imported the result back into Mathematica," the problem that you have is the bottleneck. It is exactly the same as research. Our bottleneck isn't writing programs; our bottleneck is getting all the data that sits on our desk and trying to figure out what it means. And it's same thing when you go into another language, and then try to take the results back: it can take a long time to get all those C results into Mathematica. But we're working on that. We're all trying to give ourselves speed while getting or retaining conceptual simplicity. So we'll see.

Collier: Okay, it doesn't look like there's any more questions for Richard. I'll just give a few brief closing comments and then open the floor for any general comments or questions about toolkits.

First of all, I'd like to thank all the presenters for coming, and I know I got a bunch of good ideas for my own work, and I hope other people did as well. Given what was said yesterday morning about coding models in diverse toolkits and diverse languages, these presentations are very interesting. I realize most researchers probably don't have time to actually code something in more than one way, but it's nice to know that if you did have time, it is possible.

Lastly, I'd like, again, to support what Marcus [Daniels] brought up — this notion of a common language to talk about models. I think this helps in a few ways: it helps modelers talk about models to themselves and then to each other, and, more importantly, it helps people like me talk to modelers. You know, "What do you want? What is this model all about?" instead of "Okay, it's traffic and these are the cars, blah, blah, blah." Talking in more abstract terms will make it easier to communicate.

I know when I first got here, which was a little over a year ago, and David [Sallach] started throwing simulation stuff at me, it was a lot to take in at once, and if I'd had some sort of dictionary or grammar that defined a common language of agent-based simulation, that would have been a big help. And I think for other people getting started, it would be a big help as well. Then there might be something from the computer side about making the translation from this sort of modeling language to software. I'm not sure how to get such a thing started, but it would be nice to have this common language.

Now I'd like to open the floor. If there's any comments about toolkits as a whole, go ahead.

Sallach: I would just like to say, regarding all the earlier presentations, I really appreciate the common thread of driving toward leveraging standards — leveraging UML, leveraging XML, leveraging Java. I think that's really important. I think that what we're going to see is a great proliferation of models as people follow Ian [Lustick's] example, and others, and begin focusing in on the domain issues, and therefore begin needing specialized models for that purpose. So I think that we're faced with the broad question of how to leverage each other's work. I think this leveraging of standards is the first step. But I think that there are other things that we can do, other things that we should be talking about: for example, setting up communication structures that support this idea.

There are two things I'd like to say in that regard. One is that I hope we can begin dialoguing about how to create an open-source community, whereby there's dialogue going on about, you know, "We're building a specialized model in this area, but we're hooking it off of some generally shared framework." It's not going to be easy to spontaneously evolve plug-and-play, but if we have it as a goal, that's a first step. And so I think that setting up the communication structures — even just, say, a common social agent mailing list that transcends toolkits and languages and that kind of thing — might be extremely useful.

And then the second thing is, the point that's come out in the discussions in the last two days about theory and experiments regarding the knowledge representations that underlie the toolkits and the libraries — these have an important component of theory in them. And so we also have to find ways to facilitate the communication between software development on the one side and domain-specific research objectives on the other, because when we can get the representations in our toolkits and development environments to be in dialogue with the domain priorities, that's the point where it'll be the most stimulating and we'll make the most advances.

Gaylord: I'd like to sort of tie into that comment. There are two things I would say. One is that in my course, one of the things I emphasize is that models that apply in one domain actually are very good models in other domains when the objects are translated into the domain you want. This has been done with mathematical physics going into the social sciences. I think an advantage actually of agent-based modeling is that it is far easier for me to understand what someone is talking about when I see an algorithm. I have spent more time reading about social norms without knowing what the heck they were talking about. And of course this is in classical sociology, where basically the idea is to talk rapidly and wave your hands. And I do that well. [*Laughter from audience*.] But a program is a program, and I don't care what you say you're doing, if it's there and it's not too long a piece of code, you can look at it.

One time I was talking to someone about this improvement model, and he'd keep saying, "You've got to have an institution in here." That's because he's an accountant. [*Laughter from audience.*] And I said, "Okay." And he kept saying that for a long time. And I said, "I have no idea what you're talking about. Could you write me a simple program where there is an institution? And I will look at the code and I will know what you mean an institution is." So I think that what we really need to do — and I think agent-based modeling, because of its use of programming languages, helps us do this — is to develop at least an agreed-upon definition of the things we're looking at, and when we don't agree, then we'll also know that. I think that's really very important for us to make progress.

Agent Methodologies and Model Validation

GEOGRAPHIC SMALLWORLDS: AGENT MODELS ON GRAPHS

C. DIBBLE, University of California, Santa Barbara*

ABSTRACT

Structured geographical or organizational environments often mediate agent interactions in profound ways. Both existing geographical or organizational systems and the agent-based simulation models that represent them may exhibit path-dependent co-evolution and multiscale feedback effects, which are difficult to examine except under laboratory conditions. Yet richly structured landscapes for agent simulations have been difficult to develop. Many models are still constrained to aspatial soups, isotropic planes, or, at best, to relational networks among individual agents, but we have not yet seen models using relational networks involving landscapes on which heterogeneous mobile agents interact. This talk introduces a new prototype for a general class of network-based landscapes for the Santa Fe Institute's Swarm simulation platform.

GeoGraphic Smallworlds have the advantage that landscapes are represented as formal graphs with realistic structures. While they can represent isotropic planes as regular lattices, they are most useful when the landscapes are most naturally formalized as one or more interlocking parameterized families of irregular graphs. Separate landscape and agent random number seeds allow us to run many agent simulations on any given GeoGraphic structure. Similarly, we can generate many distinct families of GeoGraphic landscapes that differ in their particular structural details yet share common graph characteristics that are relevant to the behavior of the model. Richer simulation landscapes provide controlled environments in which to build and test formal models grounded in explicit spatial structures, diverse distributed mobile agents, and context-specific behavior.

* *Corresponding author address*: Catherine Dibble, Department of Economics, University of California at Santa Barbara, Santa Barbara, CA 93106; e-mail: cath@econ.ucsb.edu.

GeoGraphic Smallworlds: Agent Models on Graphs

Catherine Dibble
Department of Geography
Computational Laboratories Group, Department of Economics
University of California Santa Barbara

cath@econ.ucsb.edu
http://www.econ.ucsb.edu/~cath

Beyond Cellular Automata

- **We are already moving beyond Cellular Automata in many models:**
  - Cellular network images, and "rewiring" cell neighborhoods (GeoAlgebra (Takeyama 1996), GCA (O'Sullivan GeoComp'99))
  - Constraints on populations of cells
  - Heterogeneous rules and populations
  - "Mobile" populations of cell states, albeit not yet mobile agents (objects that encapsulate rules)
- **Objective:** A generalization which allows us to move beyond the increasingly binding constraints of Cellular Automata, yet which would *include* all possible CA capacities as a subset of its capabilities.
- ***Success*:** Chris Langton's agent-based *Swarm* simulation system (since 1995 beta) … small battle with Swarm … I won.

Multi-Agent Models on Irregular Graphs

- **Connections structure local interactions among agents**, where local can now include shortcuts and irregular structures.
- **Exogenous Graph Structure:** How do selected global and local graph characteristics affect the micro and macro evolution of systems of agents?
- **Endogenous Graph Structure:** Co-evolution of agents and graphs, especially with respect to positive feedback and both micro and macro path dependence.
- **Analytically intractable** except for the simplest examples

  => need for a general-purpose graph-based
  GeoComputational Laboratory.

GeoGraphic

Geographical analysis

Geo*Graph*ic graphs G(V,E)

Smallworlds

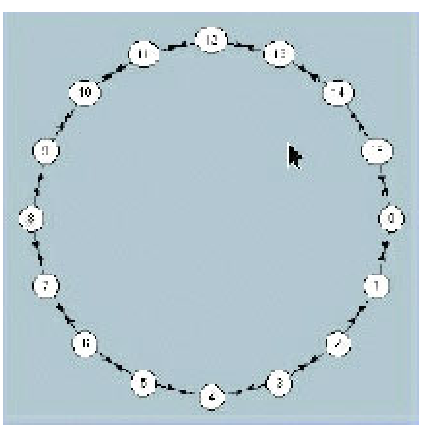
Small words (simulations)

Parameterized families of Smallworld Graphics
(Watts and Strogatz *Nature* 1998)

GeoGraphic Smallworlds in Swarm

- Nodes are places for direct agent interactions
- Links connect nodes to form irregular graphs
  - Disconnected Nodes or a Base Lattice
  - *With selected shortcuts added*
- Graph structure may have nonrandom shortcuts:
  - Exogenous – synthetic or derived from real-world data
  - Endogenous – driven by model behavior
- Agents can be any combination of mobile, heterogeneous, adaptive, or none of the above

Base Nodes as Places for Direct Interaction

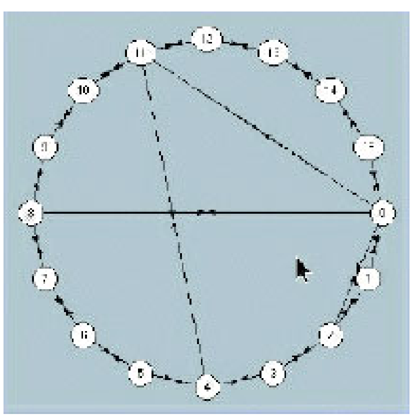


- Up to thousands of nodes for a batch experiment.
- *Any configuration*: line, ring, 2D grid (lattice), random, GIS layer(s).
- May have heterogeneous characteristics.
- May have exogenous or endogenous stocks and flows.
- May be joined by a base grid of links or not.
- Add formal graphs for *structure*. . .

Watts and Strogatz Smallworlds
Nature 4 June 1998

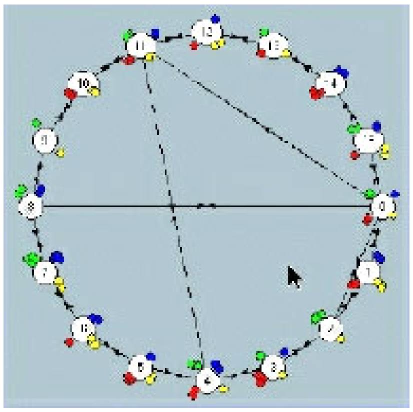
- Locally k-connected rings of 1,000 nodes.
- For each link in the base network, redirect to a random node (=> a shortcut) with probability *p*. All links have unit distance.
- Examine graph characteristics as a function of *p*.
- Characteristic Path Length – average length of the shortest paths between all pairs of nodes (falls precipitously on rings).
- Local Neighborhood saturation – degree to which nodes retain their original local connections (falls very slowly on k-rings).

GeoGraphic Smallworld
Extensions to Watts and Strogatz



- New links may be either additions or substitutions.
- Shortcuts may be random, but may also be biased or determined by:
  - Distance decay
  - Positive feedback
- Links may be assigned any cost (e.g., nonlinear with respect to Euclidean distance).

Initial Agent Populations, Assigned Randomly to Nodes in this Example



- Generic agents know how to:
  - Look around, evaluate nodes wrt agent's objectives (=> context)
  - Move to a new node
  - Leave the old node
  - Make use of GeoGraph links (to see, travel, trade, etc.)
- Agent information may be limited, local, or related to GeoGraph use (e.g., internet).
- Agents may play a locational game on the GeoGraph(s).

GeoComputational Laboratory Advantages

Beginning support for confidence testing and inference:

- *Separable random number seeds* for GeoGraph(s) and Agents allow multiple agent simulations for any given GeoGraph realization.

- *Parameterized families of GeoGraph realizations* can be generated for any given set of GeoGraph parameters.

Support for rich representations – from simple abstract to full GIS:

- *Simulations may incorporate multiple GeoGraphs and/or raster or field layers*. Respective GeoGraphs may intersect at one or more shared nodes.

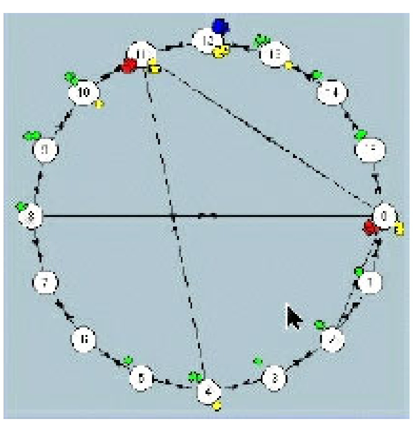
- *Agents, GeoGraphs, and other layers may update at very different rates.*

Toy: Simple Four-Sector GeoGraph Model of Multi-Agent Settlement Patterns

Sectors are defined according to their requirements for spatial interaction (slightly different from the usual):

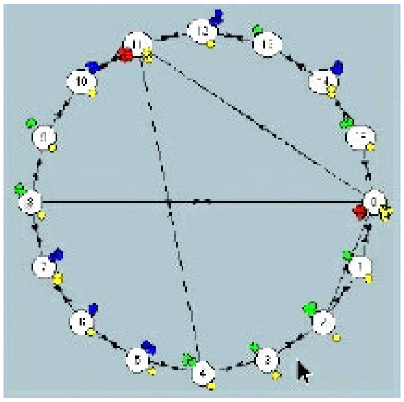
- **GROW** primary sector, harvest resources (want min local population density, but good access to markets)
- **MAKE** secondary sector, processing/manufacturing (want max **MAKE** and **SERV**, especially access to nonlocal **MAKE**)
- **SERV** tertiary sector, direct personal services (want max local ratio of non-**SERV** customers to **SERV** competitors)
- **INFO** quaternary sector, footloose information agents (want to avoid **MAKE** agents and be near **SERV** agents)

Agents Play a (Discrete Choice) Spatial Nash Game, Attempt to Optimize wrt Both Site and Situation



- Begin with random locations.
- 10% Agents move each turn.
- Until they reach a stable configuration (spatial Nash (dNE)).
- Bounded information/rationality can be introduced wrt:
  - % Synchronous versus asynchronous moves
  - Distance bounds or decay
  - Use of GeoGraphs to obtain information
  - Simple agent rules/updates.

Vibrant versus Decimated Rural Regions



Compare the implications for the southwestern region in this variant where **INFO** agents decide that they now also prefer nodes that have low population densities. This is the only change.

Note also that prior decimation may have been a function of GeoGraph structure ***only***.

Future Research

- **Exogenous Graph Structure:** What are the mappings between the characteristics of *a priori* GeoGraph landscapes and characteristics of stable agent distributions (spatial Nash equilibria) and associated measures of macro performance? (Irregular-graph extensions to Dominique Peeters *et al.*, *Geographical Analysis* October 1998).

- **Endogenous Graph Structure:** When GeoGraph landscapes co-evolve with agent distributions, to what degree does path dependence constrain future configurations based on past histories? *Does path dependence in such systems lead to second-best macro outcomes, with relevant lessons for policy?*

- **Per Macmillan (1995a,b 1999a,b):** We have discrete choice (spatial games), but would like to add endogenous *prices* (CGE).

Summary

- Multi-Agent Simulations on a very rich class of landscapes, including multiple network and raster layers, and incorporating Cellular Automata.

- Generation of *Control-Classes of Synthetic Geographies* via generation of parameterized families of GeoGraphic Smallworld networks.
  - With multiple GeoGraphs per family of geographically relevant parameters
  - "Rewinding the tape" (Fontana and Buss 1994) many times for each

THE CHALLENGE OF VALIDATION AND DOCKING

R.M. BURTON, Duke University*
B. OBEL, SDU – Odense University, Denmark

ABSTRACT

Validation of social science models, and particularly agent-based simulation models of organization, is important and difficult. There are many ways to validate a simulation model. Here, we focus on "purpose" as the primary driver to validate a model, not its realism per se. The purpose drives the nature of the model and the experimental design and analysis. Docking, or model alignment, is a new and exciting approach to validation. Docking investigates how similar the simulation models are, how the similarities and differences help us understand the models, and, more importantly, the question and purpose of the models. We then examine some possible extensions of docking and how it can be applied to validate simulation models of organization.

INTRODUCTION

Validation of agent-based simulation models permits us to use the results of the simulations to say something about the real world and the question of study. At the same time, realism itself is not the only issue. The primary issue is the purpose of the model or the question under study and its relation to the model and the experimental design. For the given purpose, we argue that simple models are preferred to very complex realistic models, which are likely to have complicated results to sort out.

Docking, or model alignment, is an approach to validation that can give us greater confidence in both models. The ideas is to compare models in a basic way to see how they are similar and different and, more importantly, to increase our confidence that both models can be used to say something about the question under study.

In the next section, we begin with some social science notions on validation and then argue that the model purpose should be the driver for validation. We then turn to docking. Docking permits us to explore the purpose of the models in greater depth and gain understanding of the question that may not be apparent in either model alone. The docking metaphor is then taken beyond agent-based models and finally related back to classic triangulation notions from social science.

* *Corresponding author address*: Richard M. Burton, Duke University, Fuqua School of Business, Durham, NC, 27708-0120; e-mail: rmb2@mail.duke.edu.

VALIDATION

Validation in social science is a central issue. Cook and Campbell (1976) established criteria that remain the fundamental questions: internal validity, statistical conclusion validity, external validity, and construct validity. These criteria were devised originally for field studies to address particular questions of what we could learn from these studies. The criteria have been applied quite broadly across other types of studies in social science and have become the standard questions for validity. Carley (1996) presents a comprehensive statement on validation for simulation models in the social sciences and an overview of computational organization theory (Carley, 1995).

For computer simulations and modeling, realism has been a major concern. Can a computer model be sufficiently realistic to help us understand the real-world phenomenon of interest? Very early on, Cohen and Cyert (1965) addressed the realism question:

> …even though the assumptions of a model may not literally be exact and complete representation of reality, if they are realistic enough for the purposes of our analysis, we may be able to draw conclusions, which can be shown to apply to the world.

As a single criterion, realism is most closely related to construct validity or external validity and thus to the question of generalizability to the outside world. Realism is clearly an important issue but not the only, nor even the most important issue, for validation.

Burton and Obel (1995) build upon the Cohen and Cyert notion that it is the purpose of the model that should be the main driver for the validity of the model. Further, they argue that the simulation model should be as simple as possible to meet the purpose, or to address the issue or question. Briefly, they argue that complex or realistic models embed all of the difficulties that the real world itself has. Experimental design issues are complex, and experiments are difficult to devise and execute. The analysis of the results similarly requires complicated analyses. One of the advantages of simulation is to construct a manageable and understandable simplicity. They develop a balance model for a valid model for the purpose:

- Purpose. What do we want to accomplish…describe behavior, give advice to management, train decision makers, test a hypothesis, explore to find new relations or theory generation, create alternative explanations?

- Model or computation. What is the model and what should its properties be?

- Experimental design and data analysis. What is the experimental setup, manipulation and how will you analyze the results?

They argue that a simulation should consider these issues simultaneously and prior to building the simulation model. This approach is in contrast to the realism approach where the modeler can embark on an unending quest to create an ever more realistic model.

To illustrate the balance approach, if the purpose is to test a hypothesis and determine whether a given model will confirm a hypothesis, then a parsimonious explanation is appropriate, and even desirable. Cyert and March's (1963) duopoly model is one example. Burton and Obel's (1980) simulation test of the M-form hypothesis is another. Both models are relatively simple and give a minimal set of conditions to yield the predicted outcomes and confirm the hypothesis. But these models are not appropriate to describe actual behavior nor to generate alternative explanations.

To generate new theory or provide alternative explanations through exploration of the results, then more elaborate models are needed. Here, variety of possibilities of what can occur and how it can occur are important. Epstein and Axtell's (1995) Sugarscape model is an example. Similarly, if we are to give advice to managers from a simulation, then the simulation must be "believable" and intuitive to the manager. VDT (Jin and Levitt, 1996) is an example. Realism is important here.

The purpose of the model is then the first issue. The model and the experimental design must follow but must match the purpose and the question.

DOCKING

Axtell, Axelrod, Epstein, and Cohen (1996) developed and tested "docking" or the alignment of simulation modeling. It is a compelling metaphor from space exploration and offers much promise to give simulation modeling greater validity. Docking is straightforward — we want to make two dissimilar models come together to address the same question or problem and to investigate their similarities and their differences, but, most importantly, we want to gain new understanding of the question or issue. Insight is not only gained through parallel approaches to a problem, but also by meshing the two approaches: Do they give equivalent results? Is one a special case of the other? Are there new insights? Do we have a parsimonious approach? Is a third approach and model called for?

Axtell et al. docked the relatively simple Axelrod (1995) Cultural Model (ACM) with Epstein and Axtell's (1995) more complex Sugarscape world. The purpose for the ACM is to study the effects of a simple cultural mechanism and whether cultures will remain diverse or become eventually homogeneous under different circumstances. The transmission mechanism involves neighbors interacting on five attributes, where cultural change is more likely when the neighbors are the same and less likely when different. Sugarscape's purpose is to generate rich "artificial histories." It takes a book to explain the model. These are significant model differences — were they able to dock the models? They tested whether the models were equivalent and would yield the same results. By simplifying Sugarscape to replicate more faithfully the Axelrod processes, they demonstrated equivalency. Similarly, they undertook an agent mobility experiment, which involved a sensitivity analysis. The models can yield equivalent results, and the ACM can be viewed as a special case of Sugarscope, but within limits. We suggest that both models have greater validity through docking than is possible to establish with each model alone. Doing docking is not easy and involves the best of scientific judgement and technique. What do we mean operationally by "equivalent"? How can we simplify a model without changing its essential elements? How do we compare experiments? These difficulties may explain why docking is rare.

Our more usual approach is docking lite. We do a literature survey to show what we are doing is important and germane. We may even do a thought exercise to compare our simulation models with other models and research — a kind of light touching. Rarely do we dock? We tend to use our time in creating and building new and more complex, perhaps realistic models. New and creative models are needed, but as a next-best step we may learn more about the questions we are studying through docking as a way to accumulate our knowledge more rapidly than starting each time from the beginning. Axtell et al. argue that docking is essential to the progress of computational modeling; they make a compelling case.

Here is one possibility. In Prietula, Carley, and Gasser (1998), there are no docking experiments. One possibility is a focus on trust and cooperation. Huberman and Glance investigate cooperation, and Carley and Prietula trust. Defection is a fundamental notion in both. These are very similar organizational concerns. By docking these two models, we could learn more about mechanisms to realize nondefection; that is of great interest to us. There are eight models presented in the book and then 28 possible docking experiments. Not all would be of interest, but a few could be. We suggest that the question, e.g., trust and cooperation, should drive the docking effort, not the model per se. For Axtell et al., both models address the culture assimilation question.

DOCKING EXTENSIONS

Docking experiments are not limited to agent-based models. Burton and Obel (1998) have devised a knowledge-base expert system (Organizational Consultant or OrgCon) for organizational diagnosis and design. The knowledge has been gleaned from the organization theory literature and validated with executives and students (Baligh et al., 1994, 1996). Levitt and his associates (Jin and Levitt, 1994) have developed VDT (Virtual Design Team); an agent-based project organization simulation model that incorporates a wide range of individual behaviors and organizational possibilities in an information processing model. Both models consider concepts of decentralization, formalization, coordination, incentives, etc., and are consistent with contingency notions of organization. Yet they are different: the OrgCon is a macro level model and the VDT is a micro level model. Concepts are operationalized in different ways. There are a number of docking issues: Can we learn more about the validity of each model? Is the macro level OrgCon consistent with the micro VDT in the concepts of decentralization, etc., and can we learn more about these organizational concepts? Do macro level organizational concepts apply to micro level project organizations? Can the macro level OrgCon be used to guide VDT micro level experiments for actual design situations? Here too, the validity of both models could be enhanced through docking.

Docking as a concept is not restricted to simulation or computational models. It seems quite possible to dock a simulation model with a laboratory experiment, where both are investigating the same issue. Similarly, it is possible to dock a simulation model with an ethnography study. In this concept, the ethnography study is just one realization of the model, or possible real world outcomes. Here, it seems that the simulation model could enrich the real-world interpretations of the real-world observations.

We have now extended docking to a more familiar social science notion of triangulation (McGrath et al., 1982: chapter 4). Obviously, triangulation is also a borrowed term from navigation and surveying. In social science, triangulation suggests that we should be investigating a given question from different perspectives and using different methods, i.e., different observation posts. They are some derived rules of thumb. No one study can answer a question definitely. No one method can answer all questions. No one method can answer a question through continued duplication. (Do not drive all the observation posts at the same spot.) Alternatively, we can learn about an issue or question by using more than one approach or method. And simulation is one laboratory in which to experiment and learn.

SUMMARY

Validation is an ever-present issue in simulation modeling or computational approaches for social science. We have argued that

- Purpose should drive the validation approach, not realism of the model per se;
- Simple models have advantages and should be used if they meet the purpose;
- Docking is an exciting, new, and all too rarely used approach to validate simulation models and more generally enhance our understanding of the question or issue that the models address; and
- Docking can be extended beyond agent-based simulation models, and we can develop better understanding of social science issues by docking simulation models with laboratory, field, and ethnographic studies.

REFERENCES

Axelrod, Robert (1995) "The Convergence and Stability of Cultures: Local Convergence and Global Polarities," Santa Fe Institute Working Paper 95-030028.
.

Axtell, Robert, Robert Axelrod, Joshua M. Epstein, and Michael D. Cohen (1996) "Aligning Simulation Models: A Case Study and Results," *Computational and Mathematical Organization Theory*, 1(2):123-141.

Baligh, Helmy H., Richard M. Burton, and Børge Obel (1994) "Validating an Expert System That Designs Organizations," Kathleen M. Carley and Michael J. Prietula. eds., *Computational Organization Theory*, Hillsdale, New Jersey, Lawrence Erlbaum Associates.

Baligh, Helmy H., Richard M. Burton and Børge Obel (1996) "Organizational Consultant: Creating a Useable Theory for Organizational Design," *Management Science*, 42(12):1648-1662.

Burton, Richard (1998) "Validating and Docking: An Overview, Summary and Challenge," M. Prietula, K. Carley, and L. Gasser, eds., *Simulating Societies: Computational Models of Institutions and Groups*, Cambridge, Massachusetts, AAAI/MIT Press.

Burton, Richard M., and Børge Obel (1980) "A Computer Simulation Test of the M-form Hypothesis," *Administrative Science Quarterly*, 25(3):457-466.

Burton, Richard M., and Børge Obel (1995) "The Validity of Computational Models in Organization Science: From Model Realism to Purpose of the Model," *Computational and Mathematical Organization Theory*, 1(1):57-71.

Carley, Kathleen M. (1995) "Computational and Mathematical Organization Theory: Perspective and Directions," *Computational and Mathematical Organization Theory*, 1(1):39-56.

Carley, Kathleen M. (1996) "Validating Computational Models," Carnegie-Mellon University Working Paper.

Cohen, Kalman J., and Richard M. Cyert (1965) "Simulating Organizational Behavior," J.G. March, ed., *Handbook of Organizations*, Chicago, Illinois, Rand-McNally, 305-334.

Cook, T. D., and D. T. Campbell (1976) "The Design and Conduct of Quasi-Experiments and True Experiments in Field Settings," M.D. Dunette, ed., *Handbook of Industrial and Organizational Psychology*, Chicago, Illinois, Rand-McNally.

Cyert, Richard M., and James G. March (1963), *A Behavioral Theory of the Firm*, Englewood Cliffs, New Jersey, Prentice-Hall, Inc.

Epstein, Joshua M., and Robert Axtell (1995) *Growing Artificial Societies: Social Science From The Bottom Up*, Washington, D.C., The Brookings Institution.

Jin, Yan, and Raymond E. Levitt (1996) "The Virtual Design Team: A Computational Model of Project Organizations," *Computational and Mathematical Organization Theory*, 3(3):171-196.

Levitt, Raymond E., Geoffrey P. Cohen, John C. Kunz, Clifford L. Nass, Tore Christiansen, and Yan Jin (1994) "The 'Virtual Design Team': Simulating How Organization Structure and Information Processing Tools Affect Team Performance," K.M. Carley and M.J. Prietula, eds., *Computational Organization Theory*, Hillsdale, New Jersey, Lawrence Erlbaum Associates, 1-18.

McGrath, Joseph E., Joanne Martin, and Richard A. Kulka (1982) *Judgement Calls in Research*, Beverly Hills, California, Sage Publications, Inc.

Prietula, M., K. Carley, and L. Gasser, eds. (1998) *Simulating Societies: Computational Models of Institutions and Groups*, Cambridge, Massachusetts, AAAI/MIT Press.

DISCUSSION:

AGENT METHODOLOGIES AND MODEL VALIDATION

M.V. Nagendra Prasad (to Richard Burton): About the purpose of simulation, the list of purposes that you put up there was pretty domain independent. I think the purpose at the domain-specific level is very important. For example, for something that I was doing, looking at the dynamics of workforce transformation in an organization changing from one type to another type — that defines what is within the boundary of the simulation and what is outside the boundary of the simulation.

So the issue that I have with respect to docking is this: What's the probability that two different people will have similar enough purposes that they'll define the boundaries of their models close enough to make docking sensible at all. Otherwise, docking itself becomes a purpose, and you'll have to strip one model down to match the other model.

Richard Burton: Well, in the case I described, the similarity was that they were trying to understand a bit about this phenomenon of behavioral and cultural diffusion in communities. And so it wasn't so much the models per se; it was more that they found similar questions that they were interested in. And in some sense we do this a lot, in that we do a field study, we do a laboratory study, we do an ethnographic study in social science — all around the same kind of questions. And that is the motivating thing. Rob [Axtell] may want to comment on that latter point.

Robert Axtell [inaudible on tape]: [*Axtell noted that the two models he worked with in the project Burton discussed, the Axelrod Cultural Model (ACM) and Sugarscape, had very different purposes. However, docking was facilitated by the fact that Sugarscape was created for more general purposes and thus could be particularized for this case.*]

Mark Jusko: A year or two ago we had a rather intractable problem that seemed like we *should* be able to solve it analytically, but we were having all sorts of problems. We tried linear programming of several flavors on several different platforms, and the solution times were just on the order of years. We eventually figured out that we could do it in a modified type of dynamic program. We got what we thought were some optimal solutions, and a couple of people who looked at the code said, "Yes, this looks reasonable." But we couldn't prove to ourselves that out of all these kabillions of permutations that we really had optimal values. So we put together a quick and dirty genetic algorithm and let it crank for a couple of days, and it could not find any solutions that were better than what the dynamic program produced. So there we had what I assume were two totally different approaches to a problem. I guess you could call that docking, if I've got that theory right....

Burton: I wouldn't call that docking.

Jusko: Okay. But they confirmed our suspicions that we had optimal values. But I have two questions, two points of confusion on my part. You said the ACM was shown to be a special case of Sugarscape. I think that was one of your statements. If one's a special case of the other, how could you consider them to be two different models, and how could you dock them? It seems that if they're the same model, how could they be docked?

And the other point of confusion was your comment about how realistic do models really need to be. I started getting uncomfortable when you said, "Well, it's not always important in all cases that they be.…" Maybe you can expand on that. If the purpose of modeling is to find new relationships or to try to create new theories, how could the model not be based on reality?

Burton: Let me take an example from a discussion some of us got into yesterday. Someone was talking about the [post-Civil War] Reconstruction. And he created, as I understand it, a model that was docked. The description wasn't a realistic model; it was built to replicate a theoretical construct, which I think is really marvelous. And what he learned was a lot of things about the assumptions and the nature of the theory that he couldn't learn from the real world, because there *was* no real-world laboratory. This was the laboratory, if you like.

I think he was able to test the assumptions of that construct, if you like, better than we were able to do in the real world. I think in some sense Sugarscape has some of the same properties, in terms of theory generation and theory testing. We have an advantage that nobody else has; after all, the world that we live in is still an artificial world in many ways. The claim was that whether it's democracy or whether it's an organization, or whatever else, these are still man-made and created. And so the notion that we ought to experiment with them and understand them better before we *make* them realities seems to be at the heart of what we're trying to do. You almost flip the notion of "what is reality" in some sense. God has this huge simulation machine that's running up there, and "reality" is just one run. [*Laughter from audience*.] For the most part, that's what we have, and that gives us, as social scientists, a horribly limited laboratory. I don't know about as engineers.

Gary An: I'm new to these two models, but just as someone listening to your description of the docking process, I'd like to mirror what Mark just said. It sounds like you're doing some sort of equivalency function between the two models. And wouldn't a successful dock mean that your models both have the same assumptions inherent in agent-based modeling? Therefore, one would be a special case of another at some level, at some point in time.

I think that if you were to use different forms of analysis — for instance, dynamic systems analysis, which, from what was just described, is a completely different approach — and then verify the results in some way, that would be better verification of the validity of those two models, whatever the purpose you intend, as opposed to just some sort of equivalency function between two agent-based models.

Axtell: I don't disagree with what you're saying. So the background was that Axelrod had published a paper on this model [ACM], but he had not published the code. And he had actually a quite counterintuitive result coming out of the model. So the question was, given only his verbal, textual description of the model — there was no pseudocode provided, for example — could we modify the Sugarscape code in a suitable way so that we thought we were reproducing what his words said. And would we get the same counterintuitive result.

Now, he thought that actually his result might be a bug, might be an artifact of the way he programmed it. And so the dockings experiment really was about working from a verbal description. In this experiment, I think we recounted the fact that we actually kept the code separate between the two programming camps. We shared no code; we only shared the verbal description. And so the question with the experiments was simply this: Could we reproduce this counterintuitive outcome from simply a verbal description. In one sense, it may seem quite obvious that that could be done, but looked at a different way, it might not be obvious.

I was speaking to Fredrik [Liljeros] today about a more recent model that I've been working on with firms, and he asked, "Is the updating in the model synchronous or asynchronous?" And it turns out that it makes all the difference in the world to know that. So if you put in a different updating, you get different results out. And so very subtle assumptions, which are sometimes even put in footnotes or omitted altogether, can make a difference as to whether you can dock or not. So feasibility of docking is somewhat in doubt. It's not that you know it will come out right ahead of time.

Charles Macal: What we've been hearing is that there's a tension between having a minimalist view of the level of detail in a model — from time to time I think we all aspire to that — and having a maximalist view of trying to put in all the detail possible that corresponds to what's out in the real world. But it seems to me that one of the major things that agent simulation offers is the fact that there is now a capability to add at least a couple of levels of detail that have always been assumed away in traditional models in virtually all the fields. For example, just take the fact that agents are not assumed to be homogeneous anymore. We can begin to have heterogeneous agents. And relating back to Scott Page's talk, the implications of the result that suddenly having diversity produces qualitatively different things that we haven't been aware of or studied before. So the notion of having a minimalist detail model, as I put it, is highly subjective and offers the danger that we may be assuming away a lot of interesting things. Is there any objective way to address that?

Burton: No, I don't know as there's an *objective* way to do that. It seems to me that if you're trying to explore and, as it were, create these artificial histories, then what you suggest is very, very important. On the other side of it, the argument I was making is that for some sufficiency relationships, you're also interested in what is the *minimal* set of assumptions, the minimal model, that will give you the results you're interested in. In statistical models, you have all this unexplained variance. And so we're in some sense modeling out the unexplained variance. We're saying that these are not the causes of the results that we're interested in. And that's an interesting point, too. We're having two different purposes here, and it seems to me both of them are legitimate. And all I'm suggesting is that I don't think that you want to use really complicated models to, shall we say, test simple propositions if you can do it with simpler models. That'd be a different way to say it.

Macal: I think the one element you added there is the fact that you have some real-world data, or a real-world representation, that at least forms a reference point.

Pamela Sydelko: I'm from a domain of ecosystem management and ecology, which, I'm finding is actually very similar to sociology in the sense that it's one of those "hard" sciences that physics and mathematics have a hard time solving. And one of the things about the docking issue, and something to do with the simplicity of models, too, is that the experience from this domain has been that people will start trying to build models in an area they feel very comfortable in — a simple model, let's say, even a hydrologic model, which may be simple to them. But they realize that there are atmospheric inputs and that there are inputs from land use, and what they'll start doing is building those parts of the model on, because they realize the information needs to be there, even though they really are not domain experts in that. We started seeing that happen quite a bit, and we'd get these very huge, very hard to manage models. So what we started doing in this domain — it's been several years, we're doing a lot of it at Argonne — is this: Can we keep the models fairly simple, and can we then do some docking so that the parameter passing [is easier]. There are still those issues of what happens if the scales are different, temporally and spatially. And we're struggling with those. But that's how we've overcome that, because we realize we can't have everyone building these very complex models, half the time because they weren't the domain experts in the fields that they were going into. That's just a little perspective from another domain that's struggling with the same kind of thing.

Burton: I think that's fairly typical — if you look at the history of science, we have followed this pattern more than we have the other, toward more and more elaborate representations.

Michael North (discussant): As a very rapid summary, it seems that validation is one of the most important questions, because it answers the simple query, "Is there descriptive value in this model?" And docking also has a substantial value, at least in terms of isomorphism. That is, are two models functionally equivalent in some sense? It may not tell us if those models are correct, but it at least may say that they're equally wrong.

The Agent Research Horizon

AGENTS PLUS: NEW DIRECTIONS IN MODELING SOCIAL AND ORGANIZATIONAL SYSTEMS

K.M. CARLEY, Carnegie-Mellon University*

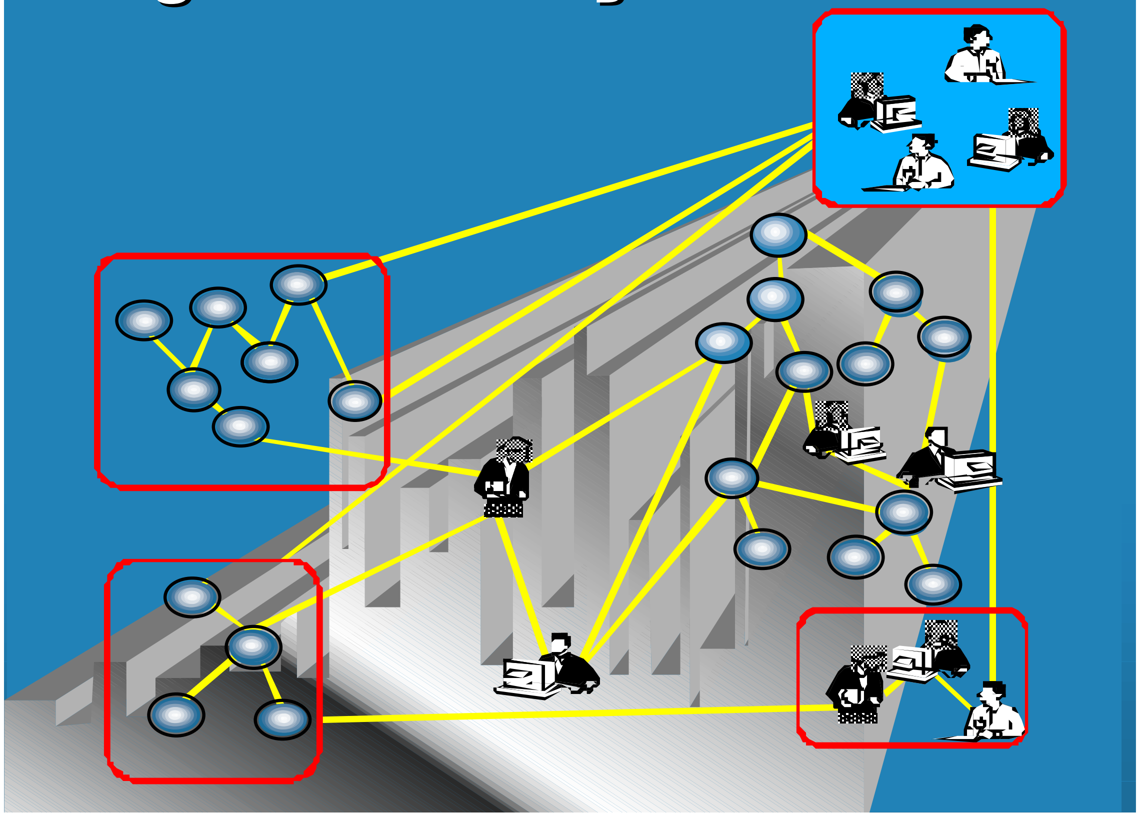
ABSTRACT

Computational analysis is being increasingly used both in academia and in industry to address issues related to group performance, behavior, and the impact of technology. The 1990s witnessed a movement toward agent-based models and, to some extent, a movement away from the system dynamic and expert system types of models. Major rationales for researchers interested in social and organizational systems centered around the need to examine adaptive systems, to take learning into account, to take cognitive constraints into account, and to determine the extent to which social and organizational complexity resulted from variations across the agents that populated the system. Now we have "computational social science." Major rationales for researchers interested in technology and management centered around the need to create better Web-based tools, the need to create an infrastructure to support electronic commerce, and the need to create computer-based agents that could better interact with humans and be more flexible, more robust, and more capable of responding in dynamic conditions. Now we have "social computer science."

These two perspectives on agent-based modeling are opening important research doors. This talk discusses some of the trends in how agent-based models are being used in the social and organizational sciences, what research opportunities are afforded by using such models, and what infrastructure problems we are faced with as social simulators move to use agent-based models. Four trends are particularly examined: the more humanistic agent, collections of multiple types of agents, agent-in-the-loop experimentation, and agent environments.

* *Corresponding author address*: Kathleen M. Carley, Carnegie-Mellon University, Porter Hall 219a, Pittsburg, PA 15213; e-mail: carley+@andrew.cmu.edu.

Agents Plus: New directions in modeling social and organizational systems



Center for Computational Analysis of Social and Organizational Systems

Kathleen M. Carley

Director & °Professor

Carnegie Mellon University

Porter Hall 219a

Pittsburgh, PA 15213

Tel: 1-412-268-3225

Fax: 1-412-268-6938

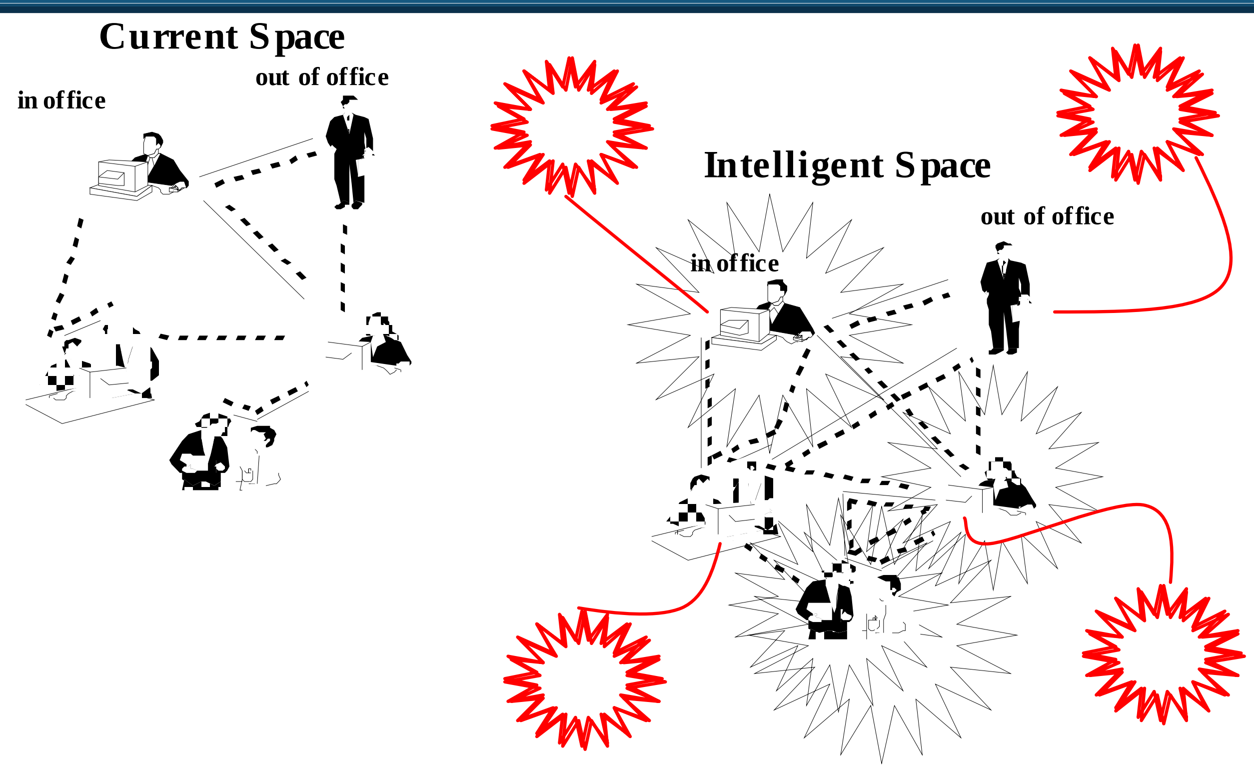
Email: carley+@andrew.cmu.edu

1999

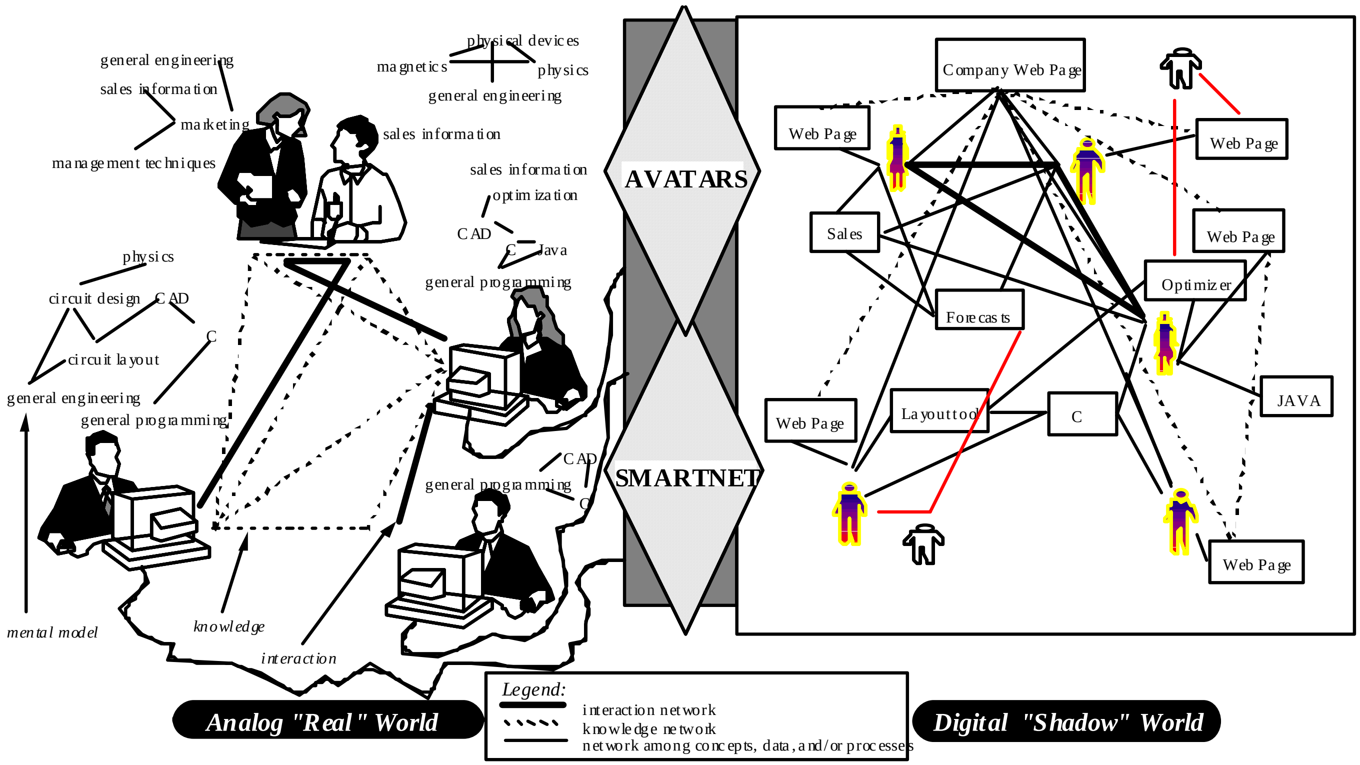
Computational Analysis Approach

- Multi-agent models
  - Agents are humans or artificial
  - Collections of diverse agents
- Socio-Information processing approach
  - Social linkages, constraints
  - Cognitive capabilities, constraints
  - Technological changes, constraints
- Emergent Behavior
  - Individual, group, organization, population level
- Empirically grounded
  - Models based on empirical findings
  - Multi-level validation

Expanding Spheres

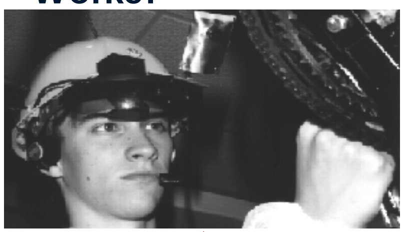


Digital Shadow World



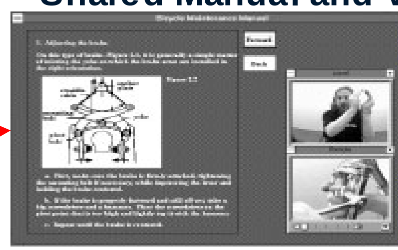
Technology Assessment: COMIT

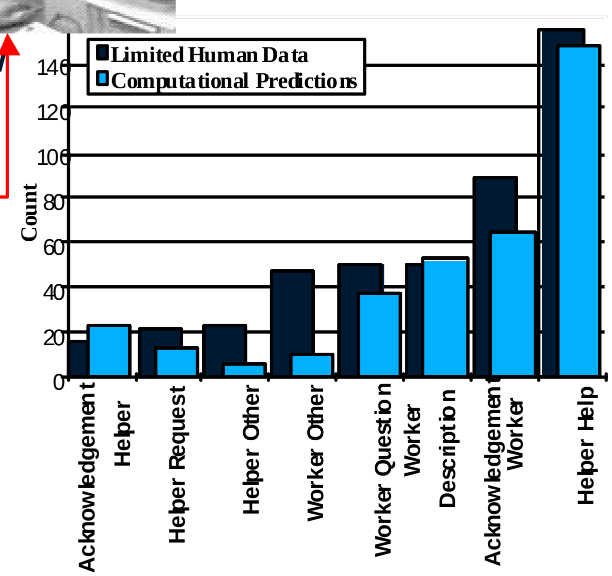
Worker

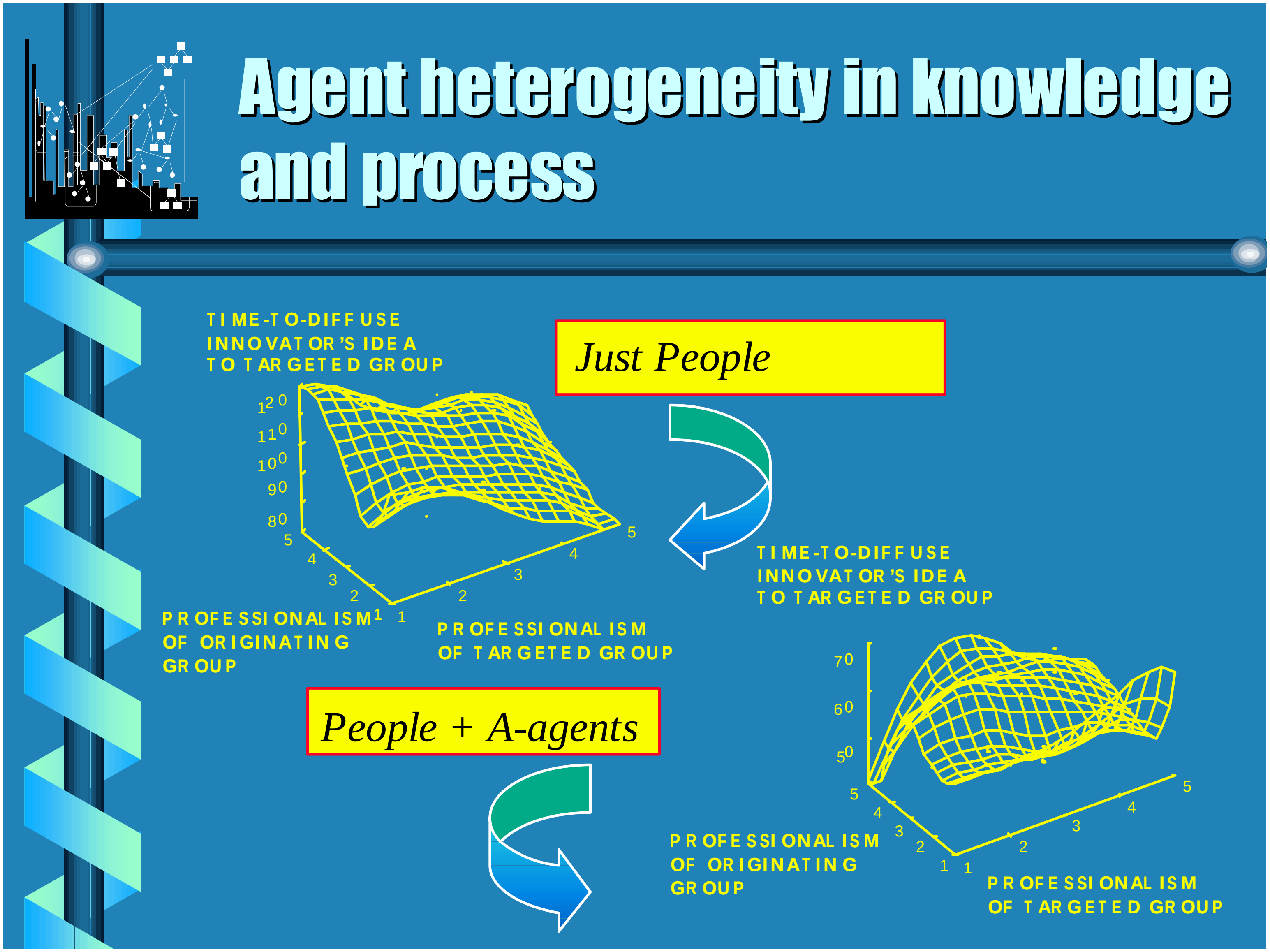


Helper

Shared Manual and View

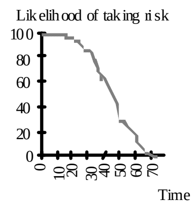





Agent heterogeneity in knowledge and process
TIME-TO-DIFFUSE INNOVATOR'S IDEA TO TARGETED GROUP
120
110
100
90
80
5
4
3
2
1
1
2
3
4
5
PROFESSIONALISM OF ORIGINATING GROUP
PROFESSIONALISM OF TARGETED GROUP
Just People
TIME-TO-DIFFUSE INNOVATOR'S IDEA TO TARGETED GROUP
70
60
50
5
4
3
2
1
1
2
3
4
5
PROFESSIONALISM OF ORIGINATING GROUP
PROFESSIONALISM OF TARGETED GROUP
People + A-agents

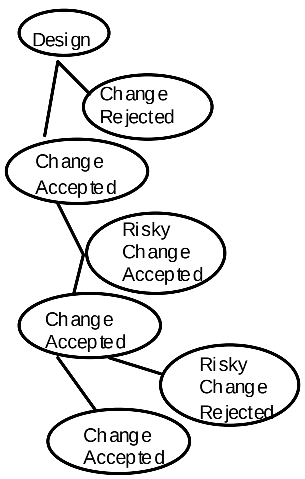
ORGAHEAD: Multi-level agents

Organization — Strategies of change



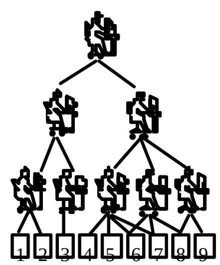
(Simulated Annealer, structural changes, optimize performance, expectation based learning)

strategic agent



Organization

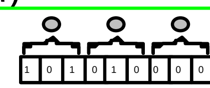
Structure & Agents



Adaptive Worker Agents

(Learn, communicate boundedly rational task oriented)

Task



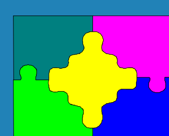
(choice, classification, radar)

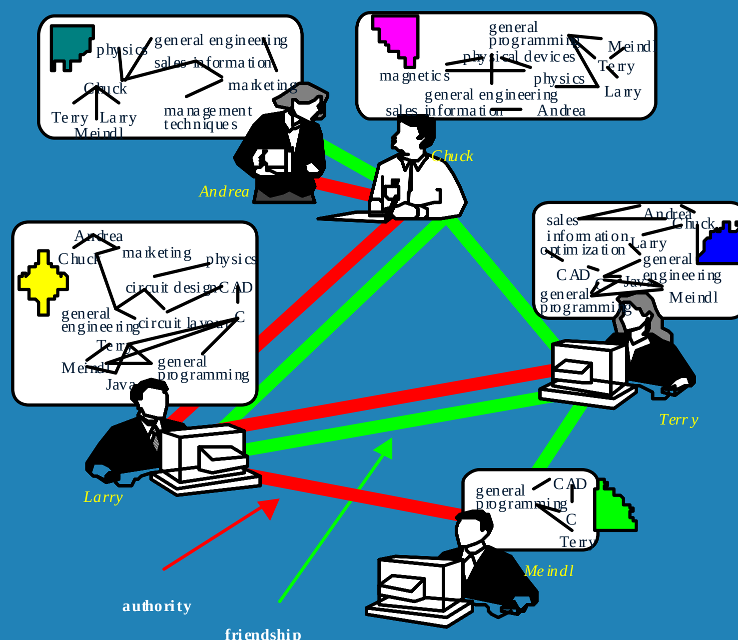
Understand, Manage, Improve Performance of Groups & Organizations

Through Analyzing Networks

- **Communication network**
  - **Who to whom**
- **Information network**
  - **What to what**
- **Knowledge networks**
  - **Who to what**
- **Organizational networks**
  - **Company to company**

networks
& simulation
used to piece
puzzle together





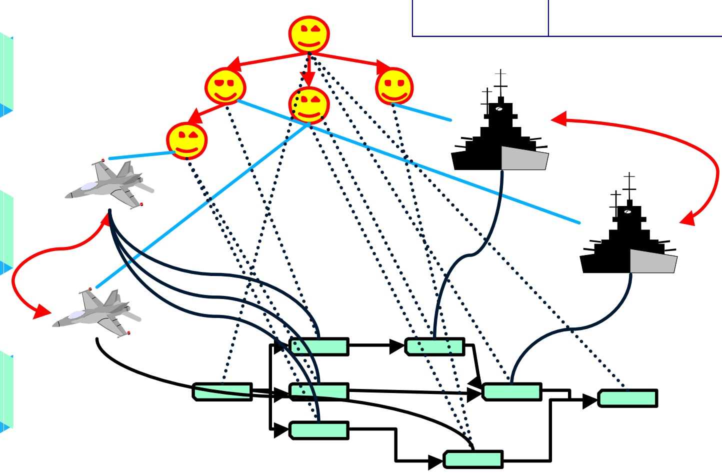
Predicting and analyzing
performance & vulnerability

Matrix of Network Based Measures

	Personnel	Resources	Tasks
Personnel	Networks (e.g., authority and communication) Size Span of control	Capabilities Coverage	Assignments Workload
Resources		Substitutes Uniqueness	Needs Usage
Tasks			Precedence Complexity

Application of Model

	Personnel	Resources	Tasks
Personnel	Networks (e.g., authority and communication) 5 Size 2 Span of control	Capabilities 1 Coverage	Assignments 1.8 Workload
Resources		Substitutes 0 Uniqueness	Needs 1.5 Usage
Tasks			Precedence 0.25 Complexity

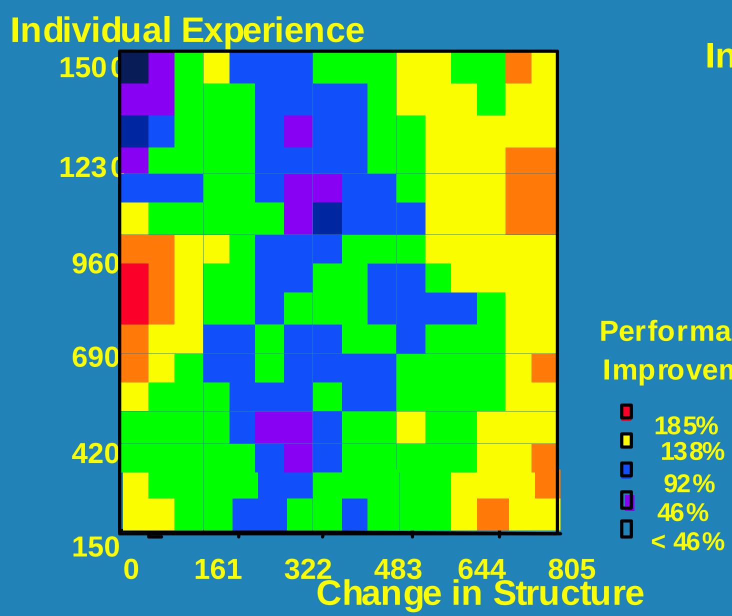


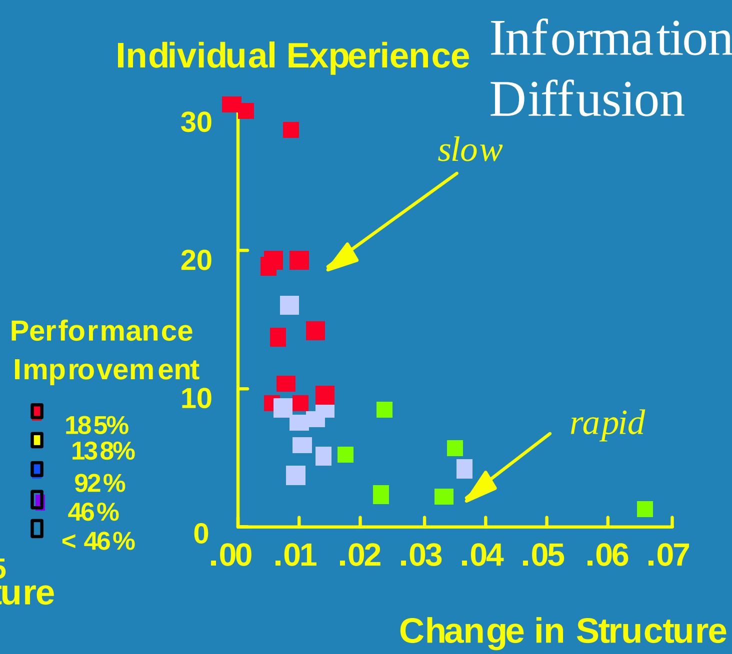
01110 0000 10000011
00001 0010 01000000
00000 0100 00001100
00000 0010 00001000
00000 1000 00110000

------ 0100 01110000
------ 1000 00000100
------ 0001 00001000
------ 0010 00000010

------ ------ 01110000
------ ------ 00001000
------ ------ 00000010
------ ------ 00000100
------ ------ 00000010
------ ------ 00000001
------ ------ 00000001
------ ------ 00000000

Characteristics of Top Performing Organizations





Trends

- Increased computational modeling in Academia & Industry
  - Where
    - Academia and Industry
  - Areas
    - Group behavior
    - Organizational performance
    - Impact of technology
    - Marketing
  - Why
    - Cost of redesign, restructuring
    - Cost of experimentation
    - Technological opportunity
- Increased usage of agent based models

Combining Social Science & Computer Science

- Computational Social Science
  - examine adaptive systems
  - take learning into account
  - take cognitive constraints into account
  - determine the extent to which social and organizational complexity resulted from variations across the agents
- Social Computer Science
  - create better web-based tools
  - create an infrastructure to support electronic commerce
  - create computer-based agents that could better interact with humans
  - more flexible, more robust and more capable of responding in dynamic conditions

Research Opportunities

- Four Opportunity Areas
  - the more humanistic agent
  - collections of multiple types of agents
  - agent in the loop experimentation
  - agent environments
- Application Areas
  - Organizational design and adaptation
  - Diffusion and social change
  - Impact of policy
  - Impact of technology
  - E-commerce

The more humanistic agent

- How human like do web agents need to be to be useful to humans?
  - emotions
- When should structures used in human organizations be used for artificial agents?
  - IBIZA
- What do we need to do to agents to create a good model of the human?
  - Soar
- How human like do agents need to be to get recognizably social behavior and accurate predictions?
  - Plural Soar - vs. - Orgahead

Collections of multiple types of agents

- Agent heterogeneity in knowledge
  - Many systems e.g., all of a-life, construct, etc.
- Agent heterogeneity in process
  - Increasing number of systems e.g., genetic algorithm models
- Multi-level systems
  - Orgahead
- Multi-optimization/learning
  - Joint optimization/learning techniques
  - Market approaches to choose between them
- Multi-authored
  - Gaming situations

Agent in the loop experimentation

- Fine tuning of experiments
  - Agent based models to pre-run experiments
  - Suggest how to alter experiment to maximize observable results
  - E.g., Bonacich exchange theory models
  - E.g., ORGAHEAD and NPS
- Agents take human roles
  - Agents replace some humans in experiment
  - Enables larger experiments
  - E.g., radar experiments

Agent environments

Creation of artificial worlds that agents can populate

- Toolkits for designing agents
- Data collection tools
- Visualization tools
- Communication media
- Physical artifacts
- Time control
- Geographic or physical environment control
- Sample tasks

Early prototypes
- *Swarm*
- *Star*

Issues

- Replication
  - Somewhat higher in the social science
- Linkage to data
  - Higher in parts of the social sciences
- Joint tools for industry and academia
  - Higher in organization science
- Education
  - Programming + statistics + experimental design + modeling + social science/ organization science
  - Modeling as art
  - Unified systems
  - Canonical tasks
  - Standard representations

Infrastructure Needs

- Model repositories
  - Docking
  - Rapid prototyping
- Code repositories
  - Rapid prototyping
- Multi-agent organizational language
  - Education
  - Rapid prototyping
  - Comparison
- Paper repositories
- Data repositories
- Visualization techniques

DISCUSSION:

THE AGENT RESEARCH HORIZON

Robert Axtell: I'll just be real brief here, because I have to go catch a plane. It's a great pleasure to be able to comment on Kathleen [Carley's] presentation. It's an unusual opportunity to be able to comment on your teacher. Although I never took a course from Kathleen when I was in graduate school, I've learned much from her over the years, and so, as I said, it's an unusual opportunity, and I appreciate the chance to do so.

I agree with almost everything Kathleen said about the future of the way things are going to go here, hopefully. I'd like to point out a couple of things, though, that she maybe didn't give a lot of weight to. You may have noticed in her talk there was much discussion of how knowledge representation, including things like Soar, TeamSoar, PluralSoar, and those kinds of things, can be used in agent models. That's an idea that I think is not widely held by people in the social sciences. I'm thinking particularly of economists and that ilk. There's a widely held conception that top-down AI [artificial intelligence] has been a failure and that DAI [distributed artificial intelligence] is the wave of the future. [But this conception] fails to grasp the importance of knowledge representation and these well-described and well-worked-out software tools that we have today for doing that. So I think her emphasis on that shows a middle ground that can be followed. It's not just that DAI and CMOT [computational and mathematical organization theory] are something different, but we can also use these older tools that are well-worn.

Another point she went through quickly was the dichotomy between computational social science and, well, social computer science is the way she put it, I guess. I'll just mention how that's playing out in reality. Some of you may know that for several years now there has been a funding agency within NSF [National Science Foundation], within the computer and information systems engineering directorate, I guess, called Computational Social Science. That group was run by Les Gasser a couple years ago and is now run by Suzie Iacono. It turns out they can only really fund novel computation applied to social systems. So if you want to apply, say, existing techniques to a social science problem, you can't really get funding from them. It turns out there are political reasons why — they're an engineering directorate, etc.

So it turns out that we need to have a way to do what she suggests in her first point, this computational social science. And so there is a proposal within the social and behavioral sciences branch of NSF to have such a directorate, such an orientation. But as of today it's not funded yet. So that's something that is on the horizon. Hopefully, it'll all shake out the way Kathleen has forecasted.

Kathleen quickly mentioned replication, and Richard Burton talked about that earlier. Just to follow up on the issue about our community being held to higher standards than other communities — I've only recently come across the fact that in the mid-'80s there was this so-called replication study done that was actually funded by NSF. For a more-or-less randomly picked issue of an empirical economics journal, *The Journal of Money, Credit and Banking*, which included maybe a dozen empirical papers, could those papers be replicated? This is not just whether the econometrics could be replicated. And the striking answer was that even when the data was provided to the replicators, about half the papers could not be replicated. There were various incredible errors found, like algebraic problems in how equations were manipulated that led to nonreplication, purely statistical problems, and problems of using archaic software. Crazy things were the source of error, like just the fact that the data files were read in incorrectly.

So about half of the papers were not replicatable. Of the remaining five, three of them were not replicat*ed*, meaning that all the techniques were applied properly and everything, but just the conclusions the authors drew were considered incorrect based on the analysis that was performed. So in fact only two of the 11 or 12, or however many there were, were actually replicated. And that was just a random sample. So I think that there is a sense in which empirical work in the social sciences has not traditionally been held to these tight standards that we feel are appropriate today, and it may just be that these tight standards we talk about all the time are in some sense kind of a backlash against our community by the vested interests, as Veblen would have said 100 years ago.

My last point to say is that one very important thing that Kathleen failed to mention, although it was on all her slides, is the existence of CASOS [Computational Analysis of Social and Organizational Systems Center], which is her important new forum for doing this kind of work at Carnegie Mellon [CMU]. And I hope it attracts both future students and good faculty members. I think CMU has jumped out in front on this, and I hope other universities follow; a Ph.D. program dedicated simply to using computational approaches, agent-based modeling in particular, to inform social organizations — it's a prototype for what might be done in many other universities. Actually, though, it's going to be a hard act to follow, given the stellar lineup that they've put together there.

Kathleen Carley: Thanks, Rob.

[*A participant asked for details on CASOS.*]

Carley: CASOS is a new center at Carnegie Mellon that's dedicated to bringing computer scientists and social and organizational scientists together. It's both a research environment with outreach to industry as well as an educational program. As part of this program, we've had funded by the National Science Foundation an entirely new Ph.D. program that will be training students jointly in computer science and in social networks and organization science. We'll also be running a summer workshop starting this summer that will be open to both faculty and students that will give a really fast introduction to how you do formal modeling of organizations. That's where we're starting.

Unidentified speaker: This was touched on a little bit in your talk, but I'm curious about what's being done to account for environmental changes in a model. Because how I think about agent models right now is that they're very behavioristic. The way I program in Java, I have objects that have methods, and in order to have an object do something, it has to have a method in order to deal with things.

Like the Sugarscape model, there's also a "Sugar and Spice" model, where all of a sudden the agents find spice, and they have a built-in rule for dealing with spice and for trading. But what happens if they only know what to do in a sugar environment and all of a sudden someone finds spice and says, "Hey, we can eat this too! This stuff's great!" But if they don't have a rule for that, there's no way to account for that. I see agent modeling as a genetic algorithm; we're simulating a genetic algorithm time step by time step, and the agents are the participants talking back and forth developing new stuff. But where can we have them pass functions and objects and ideas back and forth so they can generate new ways to deal with their environments? Is there any research going on regarding that?

Carley: There's a variety of research going on in that area, both in our lab and in a variety of other ones. Let me describe how that's being done right now within Orgahead. In that model, every single agent has the ability to learn. And they also have the ability to communicate. What they do is that they're given a series of tasks, and as the environment changes, the sequence of tasks changes, both in its nature and in its performance expectations. The agents over time learn, based on feedback from their boss, how well they're doing. Then they change how they respond to the environment as the environment changes. So it's done through learning functions.

The second way it's done is at the strategic level, with the CEO, or the change management team, trying to predict the future based on looking at the past and saying, "How well have we done?" and "Here's our prediction for what we think the future will be." Then they make changes in the organization based on their anticipation of the future. And they can be wrong, because they may not be correct in predicting the future. So there's both anticipatory learning and regular learning based on feedback.

Similar methods are being used in other programs. But there's another answer in some work that will be starting that is actually in the area of global climate change, where the issue is not just a changing task environment, but changing physical and institutional and geographical — or geoclimate or whatever — environments. There, one of the things that people are starting to look at is the idea of not only using agent-based learning and communication, but also coupling that with information on social influence and communication models — models that say people can change their beliefs not only in response to changes in the environment, but also in response to changes in the beliefs of the others they interact with. So there we're using learning communication models where individuals don't have a fixed network, but they literally change their network. It evolves over time — who they talk to — and then their beliefs change in response to that, in response to what they've learned about the outside world. And another answer to your question is that we don't do any of it on von Neumann grids.

Donald Hanson: A lot of the climate change policy operates under the worldview that organizations are not working perfectly in terms of the allocation of resources. And the traditional firm in economics has an output; it's producing something. And it has input; it has all the agents that you're talking about, which we call labor, and if they work with the capital stock, they transform materials and they use energy in that process. And if the organization isn't working perfectly, maybe it's not using these inputs as efficiently as it could to produce the outputs. And there's some evidence that resources aren't allocated in firms very well. If society doesn't care, then that's fine, but in the case of global warming, society cares a lot if we're using a lot more energy than is necessary to do the tasks. And I'm wondering whether you're familiar with research that shows that maybe too much energy's used in organizations.

Carley: No. My short answer is no. I would suggest that Hugh Jones might know something in that area. I can put you in touch with him.

Hanson: [Stephen J.] DeCannio out at [the University of California at] Santa Barbara has done some work we're using. He's shown that basically firms are very reluctant to invest capital.

Carley: Yes, yes. In fact, at a recent economics conference on global climate change, no one talked about that particular issue, but Steve and I talked a little bit off-line about it.

Catherine Dibble: The coauthors from the Stanford team [Christopher Field, Harold Mooney, F.S. Chapin III, and Elizabeth Holland] don't usually make an empirical claim that they've demonstrated that they're producing too much carbon dioxide. The point has been more that if they're not already at the frontier, then changes that might make them more efficient with respect to, say [carbon dioxide emissions?] does not necessarily result in a loss…. But that's not quite the same thing as saying that they're already [using too much energy?].

Hanson: Well, just as a comment, there's a professor at the University of Michigan — he went and visited plants, and he indicated that they were investing up to the point where there was a 33% rate of return, whereas their cost of capital was maybe 10 or 15. So more investment would actually lower costs to the firm. And then they hired a GAO [General Accounting Office] investigation to say, "Well, this can't be right." And GAO went out and interviewed twice as many companies, and they got almost the same number for a hurdle rate.

Closing Panel

CLOSING PANEL: RESULTS AND PROSPECTS

R. PICKER, University of Chicago
C. MACAL, Argonne National Laboratory
J. PADGETT, University of Chicago

Charles Macal: My name is Charles Macal, from Argonne National Laboratory, and I'm a leader of one of the simulation research groups at Argonne. We have a closing panel today to facilitate discussion and review some of the conclusions we've come to. We have John Padgett, who is a professor of political science here at the University of Chicago, and Randal Picker, who is a professor of law at the University of Chicago. I might add that we have been communicating with John and Randal, along with David [Sallach] and people here at the Institute, about forming a group interested in this particular area and applying it to various research and development programs at Argonne as well as at the University.

I think we'll say a few words in turn and then just open things up for questions or comments.

I would like to start out with some thoughts on how we came up with the idea of this program and what it was meant to accomplish. It was basically a joint idea between Argonne and the University. The first part had to do with the idea of applications of agent-based simulations, without necessarily defining what agent-based simulation is and letting people define that for themselves or take it as a given. The applications, of course, were heavily grounded in academic research, which I think is very exciting.

But secondly, we wanted to get a little more deeply into an area that we have an interest in and were particularly knowledgeable about through the work that we're doing at the Laboratory. And that has to do with the electric system and the various restructuring alternatives that could lead to various futures, for which no one really has any methodologies that I'm aware of to predict with any degree of validity or accuracy.

And then we looked at the underlying nature of what everyone is trying to do — these toolkits or, in more general terms, languages or techniques for how these systems could be modeled. And as we saw today, there are issues that in a very technical sense have to be addressed with all the tools or approaches. But there are also commonalities, as David pointed out; the field is seeing something of a convergence on its own toward standards or more commonality in terms of tools.

So, finally, we considered how these types of simulations are perhaps very similar to traditional simulations in some ways, but also very different in the sense that they are, for example, amenable to small changes or perturbations in their computation that end up affecting your final results. That has very strong implications for, of course, using these models in a decision-making or policy context, but, as came out during the workshop, it will be important in many other contexts, such as models being used as a laboratory for experimentation or theory development, etc. That was also very interesting — to see other ways in which these simulations could have a very positive impact in the future.

And then lastly, the session on the agent research horizon was an attempt to try to look at the future beyond what we're individually knowledgeable about or involved in during our daily work. And again we saw a lot of common directions. So, having indicated the rationale for why the program was structured as it was, I'll turn things over to Randy.

Randal Picker: That seemed like a very good summary of why we did this. And we've talked a little bit informally in the hall about this. This workshop has really exceeded our expectations, both with regard to the volume of participation and with regard to the content. And we need to congratulate Dave Sallach — he is really the driving force behind this.

In terms of comments, I've done a certain amount of this modeling — sort of deep legal theory involving bounded rationality and spatial coordination games and the like — and started doing this using StarLogo precisely because it had an easy entry path. And so think about other people who are less geeky than I am who might want to do this — figuring out how we bring them in is going to be very hard, I think. I heard what Kathleen Carley said, that you have to know six different things to do this. Maybe that's right, and so maybe there's no easy way to get in. We call that an entry barrier in antitrust, and maybe that's a good thing for those of us who are in, but socially that can't be the right thing. So I think figuring out ways to lower the entry cost is really important.

Kathleen also talked about canonical tasks and that struck me as very important. I would like to see some sort of simple table that tells me — and this is a question I asked Miles Parker before, and David says this is the only question I ever ask — "How fast does your thing run on different platforms?" And my canonical task at this point, I think, is heatbugs. Everyone seems to have a heatbug, right? And I'd like to see a matrix that says Swarm, StarLogo, Ascape; it says Macintosh; it says Unix. It can tell me how fast, say, 100,000 iterations of heatbugs will run on your machine. I'd really like to know that. And again, think about someone coming in from the outside who sits there and says, "Which platform should I adopt, which hardware should I adopt, which software platform should I adopt?" It's very hard to figure that out. And the last thing you want to do is to have people see the Swarm discussion list, right? That will scare people off immediately: "My god, how do you actually set this up?" And maybe that kind of "tools docking," if I can call it that, to follow Richard [Burton's] term, would be very useful. I'd like to see what these tools will do on certain basic tasks.

In terms of ideas, I thought where Rob Axtell started about bounded rationality dynamics, heterogeneity, and antiequilibrium is obviously right. That's what brings us all here. And I think that's very exciting. I think the things that are not there yet, and Rob and I have corresponded about this, are institutional features. What Rob calls "firms" look like a bunch of clumped-together people who are identical. And that doesn't describe any firm that I know. Firms are organizations in which we have differentiated individuals performing differentiated tasks. And so figuring out how to create that level of heterogeneity, which is what we think [agent-based modeling] is good at, is something we certainly need to do.

In that regard, I confess I was quite heartened by what Ian Lustick did, which struck me as a way of operationalizing this idea of differences that get activated in environments and causing those to look much more like firms. I thought that was an interesting marriage there.

And finally on the policy question, which for me, given where I sit at a law school, is the most important one — at one level, there are things that I wouldn't call policy questions that get denominated that way, for example, the discussion of the use of agents in managing the electricity grid. That struck me as a question of screwdrivers versus hammers, and not a question of policy. I can't imagine any Congressperson looking at that question. It's just not what they do.

I testify before them; it's not what they do. It's possible we can use these tools for "what if." We could "test" institutions that way. Though, even at that it seems to me we are a long, long way from doing that.

Charles [Macal] was talking about the reorganization of the electricity business. Well, FERC [Federal Energy Regulatory Commission] right now is trying to figure out whether to move from independent system operators to these RTOs [regional transmission organizations]. They've got 60 pages in the *Federal Register* that go on and on about this. Could these models help us evaluate that question? Should we switch from one organizational form to a different but related organizational form? I just don't think we're close to doing that. It would be nice if we could, but I think we're a long way from doing that.

And finally, then, there's the question of whether we could see these institutions sort of emerge from the ground up in the context of these models, as opposed to imposing them from on high. I teach bankruptcy for a living, so I have a peculiar sense of how important failure is, but I think there's going to be a lot of failure before we get to success. I hope I'm wrong, but my guess is I'm not. We'll see.

John Padgett: I had a very strong feeling that Randy and I would wind up saying much the same thing. And I'll do it in a slightly different way, which is, first, I want to approach this field a bit from the perspective of Rob Axtell's opening remarks and say, where do we stand with this interesting tool from the perspective of the development of social theory? Obviously, I'm a tremendous enthusiast. But I'm also in a regular political science department, and I see a great deal of skepticism out there toward this sort of approach. And I want to reflect a little bit on the idea that maybe they're not all complete boobs in being skeptical and ask how we can actually use these tools to further Rob's point about developing social theory. And I'll end with some troublemaking remarks, because I think there's a little bit of self-complacency in meetings like this, which inhibits our ability to communicate to a wider audience. I'd like to poke at that a little bit.

My first point, about self-complacency, is a methodological point, and the other one is a more substantive point. The methodological point is this: To further social theory, the goal is not necessarily to come up with a model that mimics something, although that's a lot of what happens in this field. If you think about it from a policy point of view, there may be a good case to be made that [mimicking a system is] a tremendously important objective. But from a social theory point of view, that's not the name of the game. The name of the game is to engage in debate about alternative theories and come up with ways of progressing, elaborating, contradicting, and developing the argument.

And another problem is that our own community is too infrequently in communication and argument with "competing contenders." This is not always true, and where it's not true is, I think, the place where we have the most impact and the most success. Let me give you an example where it's not true. And that's where Epstein, Axtell, and so forth are up there trying to duke it out with the neoclassical economists with imperfect equilibrium theory. There's a very powerful null model there. That's their baseline, their benchmark. They're trying to mimic that and show variation from that. This is a formula for success, I want to say. We really need clear, simple — not that Arrow and Debreu are exactly simple — in any event, it's clear and it's a benchmark. It's that benchmark that makes it very obvious what Brian Arthur et al.'s contribution is. It's not to say, "Isn't heterogeneity great? Isn't dynamics great?" The counter you always get to that is, "Okay, we all know dynamics and heterogeneity, these are all good things.

The question is, when does it matter?" And a classical economist will always come back with the fact that maybe it *doesn't* matter, and you can't just dismiss that idea, because maybe they're right. A lot of times it doesn't matter. They tend to converge to the same thing.

To me, the nice thing about Scott Page's argument is that at least he's confronting that, and he's saying not just that heterogeneity's cool and diversity's cool, but that it *matters*, and it matters for robustness of systems and so forth. Statements like that can only be made if there is a clear baseline against which you are comparing — I don't know, docking's not the right word — but this is a very important enterprise. And I feel that once we leave this one area of general equilibrium theory, all too frequently our community is guilty of abandoning any sort of baseline framework against which we're comparing anything, and that inhibits our ability to claim what's important about these phenomena. So, heterogeneity and dynamics — this is indeed why we're all together. But sooner or later we have to go beyond that and say, "What are the conditions under which this is important?" So that's the method point.

Now, my substantive point mimics very much what Randy [Picker] said, and I knew it would, but I just want to take it one step further. First of all, what's the problem? What was it Randy said? No institutions, no language. You say there's no social networks. Well, that's not true. Obviously, we could put in social networks. But there's no *endogenous* social networks, I don't think. There's exogenous ones that we trace the performance and consequences of, but there aren't actually emergent [networks].

Now, if you're a nonbeliever in this field, this is a problem. And why is it a problem? Well, think back to the great examples in the electricity business. If we're in a world where we're talking about a given electrical network and we're talking about the robustness and self-healing and adaptability of the network, we are in great, great shape. We can really say a lot about that, and we're nowhere near to exploiting that. But what we heard about deregulation, a sort of emergence of electrical companies — these sorts of things are really radical institutional transformations, and it's not clear that we have anything to say about that. And we ought to.

In fact, if you go back to the SFI [Santa Fe Institute] origins of this whole school, the word "emergence" was not mentioned, but when people originally talked about the concept of emergence, they had something more in mind than we have actually delivered on. What have we delivered on? What we've delivered on is agglomeration. And this is to be applauded. It's not just spatial [agglomeration]; it could be temporal — traffic jams, avalanches. There's a lot of different spatial or temporal agglomeration topics that we have delivered on, big time. And we can add networks, add homogeneity, add space; we can say, "Agglomerate this way," or "Agglomerate that way." This is a tremendously important contribution. But you have to bear in mind that a lot of people in the social sciences, at the end of the day, are not going to be impressed with that. *We're* impressed with it, but at the end of the day, they're not going to be impressed with it, because there are no institutions, there's no language, and so on.

And how do I diagnose the problem? Do I think the problem is tools? No, I think we have fantastic tools; they're getting better all the time. I think the problem — I'll just preach for a second — I think the problem is in our heads, in our mind-set. And it all goes down to that word "agent based." This is a sort of the mantra: "agent based." If you think about that, that means we're going to be strong in talking about agents — variation in heuristics, variation in learning. Anything that's going on in the machinery of agents — this is where we're making a big contribution. But what we're going to be weak on, just by the language we use, is any inter-agent, trans-agent sort of thing. And a lot of this question of institutions and language is outside the frame of the agent. It's the larger space in which we meet.

Richard Gaylord said, "Don't give me all this institution talk. I can't program it. You guys are soft in the head." I say that it would be a terrible mistake for us to buy into that. We really have to take this issue very, very seriously, and at the same time respond to questions like what do we mean by institutions? What do we mean by this interpersonal thing? And then how are we, in our framework, going to start talking about these bigger issues and start moving beyond agglomeration into organization. And I think we can; we just have to start thinking about it. You have to start thinking about the multiple meanings of the words "institution," "language," and so forth.

Let me just close with different ways that we in our tradition can respond to this very reasonable critique of ourselves. And we ought to take it as a reasonable critique, and we ought to think about how to respond to it. Here's my laundry list, you might say.

First of all, on the side of networks. This is going to be the easy part. That is to say, a lot of people will say, "It's great that you guys went from fully mixed systems to spatial [agglomeration], but you haven't done networks yet." We've heard this already. They're well on the way to doing it. There's a conference coming up in a couple of weeks organized by the MacArthur Foundation about this sort of thing. All we have to do is put a little bit more emphasis on the endogeneity of networks. So this is not a radical thing to do. This is happening in the community already.

Something that is not happening in the community, though, is a focus on this question of communication and language, the emergence of language. One of the talks made this rather nice distinction between accuracy and precision. Precision means we've got an encoding scheme and we are trying to figure out the parameterization of that encoding scheme. This is what a lot of our things do already. What we don't do is deal with encoding itself. We don't figure out what are the important variables that people are talking within. How is it that people converge on certain, not parameters, but certain variables upon which they're communicating? This is what encoding is all about, and I think this type of thing underlies these industry-change issues.

Now, this moves us into communication, into language, which is hard work, but I don't think it's out of the bounds of what this community could do; I think it's *within* those bounds. It's just that people aren't thinking about encoding. Data representation is a term that came up over and over and over again. We really have to talk about endogenous data representation. How is it that people are converging on different interpersonal communication schemes? So one direction is easy — networks. But encoding — that's terribly important. We're not doing it, but we ought to be able to do it somehow.

The last thing is institutions — that's maybe yes, maybe no. Of course, what does "institutions" mean? It means something to do with the idea of "rules of the game." And that could mean either payoff matrices that [give] the nature of the game that we're playing, or it could mean procedure, protocol, the sequencing that things do. I've never seen anything where that itself is emerging. That's what the programmer does, and then we allow things to operate under a sort of rubric. And I'm not sure that that's going be so easy to crack. It's sort of like asking a language to write itself, and I'm not quite sure whether we're up to that issue, and if so, we're going to have a problem with emergence of institutions, if that's the way we define the emergence of institutions.

There's a more limited way, however, which is very amenable to progress in this field, and that is a more modest, you might say, version of institutions, less ambitious, that has to do with the idea of artifacts. We haven't seen much about the ant simulations, but there's some great work on ants, and I wish we had some of it here to represent that field. But they're talking about building up smelling trails and hills, and about manipulating the artifact space out there, the environment. That environment is actually feeding back into the topology of interaction. This is very nice. Is this what we mean by institutions? Not exactly. It's just artifact distributions. But it's a nice step, and it's a move toward an intersection between the agent interaction and the artifact interaction. That's an area where I think progress is being made and could more easily be made.

So, on the whole, I think we should listen to our critics. We should start taking this institutional, interpersonal level more seriously. We can do it up to a point. I think the first step is to tone down a little bit our American individualism — being so concerned about agents all the time. There are super-agents and there are interpersonal things that don't fit into an agent-based logic, and I think that is holding us back.

DISCUSSION:

RESULTS AND PROSPECTS

Charles Macal: We have a few extra minutes here, if anyone would like to bring up some discussion points or otherwise.

Thomas Baines: I think there's a difference between building tools and using tools. I missed the earlier part of this symposium, so you may have already covered this, but I see a difference in perception between the builders of tools and the users of tools. I don't think everybody has to know how to program and manipulate the models. I don't think everyone *should* know how to do that, because I think one of the advantages of submitting an idea to someone who really knows how to use the model is that they will challenge your assumptions; they will challenge the things that are buried in how you are applying your institutional or your subject-matter expertise to the solution of a problem.

The guy who doesn't understand your domain at all but comes to you with expertise in using the model will ask questions that will make you think about how you are specifying the question, about what kind of questions you are asking the model to solve for you. And I guess my question is whether we are making a mistake if we go in the direction I've heard today of trying to find some way of bringing the two domains together — the tool builders and the tool users — and to make a single person out of those two very separate kinds of outlooks.

Miles Parker: I'll comment on that. I think that it's nice when you have those two things together, but it can be even nicer when you have the situation that you describe. One of our big current projects right now is this artificial Anasazi implementation. And in that project we're working closely with two archeologists and an anthropologist, neither of whom, of course, really has any domain-specific [knowledge] about computer modeling. And of course we have no expertise in archeology. But just building these models and getting together — first of all, it's been fascinating for us because we get to learn all kinds of cool stuff about Southwest archeology, but the other side of it is that it really has done exactly what you described. It's forced them to come to terms with the whole set of hidden assumptions they've had. They've become more involved in looking at microdynamics and micro issues, and in looking more directly at broad anecdotal descriptions that they've had of systems. Gumerman et al. is the recent paper.

He said that people had told him that because we're trying to get this complete science view of these complex societies, we're going to lose something. There's some talk, particularly in, I guess you could say, some parts of the post-modern community that these computational models were taking away the humanity. And he said that actually when he sat down and looked at these models for the first time on the computer screen, it was the first time he thought of these people as people.

And just really briefly on the issue of having these benchmarks for all the different projects, one thing to keep in mind is that the dimension of that question is actually quite large. I mean, you have different kinds of problem sets, different kinds of models, where one particular framework might work much better than another framework, one set of tools might work much better than another set of tools. Somebody could come out with a new virtual machine next month and it could completely change the results. So the best that you could do really is to get some general idea of where performance is similar. And then I think you should make your choices based on issues like how well you like some of the modeling choices, what the tools let you do, how they let you do that, those kinds of qualitative decisions, because on the quantitative performance issues, it's pretty close, I would say.

Michael North: On the issue of collaboration, the learning's often a two-way street. When we've done modeling with other people, as a software engineer, I learned as much about *my* hidden assumptions as I found out about other people's assumptions. And people will say, "Well, you know, you can model it that way, but it doesn't make any sense. It doesn't match anything." I can say, "Well, it's supposed to" or "I learned that in school" or "We've done that five times." But it turns out it really doesn't make sense, and so we have to learn as well. So there's both sides.

In terms of benchmarking, speed does matter, in terms of getting things done. But how long does it take to write the code? How easy is it to understand? How easy is it to maintain and change? How easy is it to test and validate? These are just as important. And so the grid should be expanded. We should have this grid, but it should include not only speed benchmarking, but also some consideration of ease of use.

Participant: Total cost of ownership.

North: Total cost of ownership, yes, exactly. That's right. We need a J. D. Powers survey, basically. And that would be a good publication to work on, if some people would like to pool their experience.

About emergent behavior, we do see some emergence now when you run these models. Organizations form, maybe not institutions, but some kind of a structure. But at the same time it's very clear that we're not that far from the rules we put in. Is it really surprising that people who look for others like them end up with people like them? Maybe, but not entirely. What we're really talking about here is that, at its heart, emergent behavior has a creative component. People come up with a new idea, and that creates some larger structure.

Catherine Dibble: I have a comment, too, going back to the institutions. I had a private student come to me who has a Ph.D. in anthropology from UCLA and also an MBA from UCLA; she's one of the movers and shakers in the business community. And she came to me asking for some sort of private mentoring, because she wanted to think about some of these cool ideas. She's quite bright, very well read. And she asked me for a suggestion for what I saw as the $6 million question in the field, the Nobel Prize question. And my answer to her was the emergence and the evolution of institutions.

I didn't mention this in my talk, but there's another aspect to what I have in mind with this sort of network landscape and the nodes and the links. What an institution is, if you think about it, is a structure that is created or emerges and then has some *persistence* beyond the agents who might come and go as part of it. So one other thing that you can do with that very simple set of nodes and links is that you can use it as a representation for such an artifact, which could then be interpreted as an institution that persists, in case that's handy for anything.

Macal: I'd like to make one comment in regard to the original question. It relates particularly to the more practical aspects of building these models or using the toolkits. Let's go back 20 years, for example. Twenty years ago a professor or graduate student with a good idea in a noncomputer field, but perhaps quantitatively inclined, could go off and write their Fortran program of their idea, which, you know, we used to do, and then peddle it — or "make it available" — to other people in the community, and hope somebody would start paying money for it.

But the situation has changed so dramatically, probably driven by the changing nature of software and the ways of instructing computers, that that structure really doesn't hold anymore, in my opinion. Or perhaps it's one extreme of a continuum: one person, one simulation? Or do we necessarily have to have what amounts to teams of people to have any hope of developing anything with any credibility?

Hopefully, we won't have to have a hundred people developing it, but I suppose that danger exists. And certainly in a university environment, I'm sure that issue is of primary importance. I've noticed that a prevalent arrangement is a kind of duopoly, or however you want to say it, of a professor and a programmer. But I think for realistic, larger-scale applications, you need to really have what amounts to a software development team.

Picker: What does Anderson [Consulting] do? They're building these. What kind of teams does Anderson put together to build one of these?

M.V. Nagendra Prasad: What do you mean?

Picker: What sort of infrastructure does one need to do this kind of work? One possibility is my current infrastructure, which is: Randy does all the programming, Randy does all the computer work. That's my infrastructure. A step forward from that would have been me plus programmer. The hard sciences model is a team of 1,000 people. And these large organizations — where we're going to go on this I don't know. And I'm curious to know, since you're the representative from the private sector here, the guys with the dollars on the table, how do you do it?

Nagendra Prasad: We do have a research lab, actually. I belong to Anderson's technology and research lab. Usually we have some programming resources, but they're usually limited. We draw upon a pool of programmers within Anderson.

Thomas Wolsko: So you buy a service for both [building and maintenance]?

Nagendra Prasad: That's how it's looking right now. Yes.

Parker: So in that sense it may be harder in an academic setting, because then you have to have problems that are interesting to the programmers.

Macal: Well, not necessarily! [*Laughter from audience*.]

Kathleen Carley: I just wanted to add one more data point about collaboration, and that was what happened in the organizations community. In that group, there's been a number of these programs that started in universities and then have spun off and have actually led to the development of small companies. Probably the most successful one to date is Vité, which came out of the VDT [Virtual Design Team] lab in Psych at Stanford. What Ray Levitt and his crowd did when VDT was still within the university, is that there was Ray plus a research scientist, and they kept working on VDT for years with a straggle of graduate students going through. But they never really made progress until they got a full-time research scientist programmer who worked on it for a year. Then they finally got a product that actually made sense, got companies interested, and then they spun off their own company. It's about 10 people. But you're talking on the order now of 20 people-years to get something that is now robust enough that other researchers can use it in their own research. That kind of time frame is about right for a lot of these big simulations that are multi-leveled — that *do* have emergent networks in them — for organizations.

Unidentified speaker: I think it's going to be hard to find people at the undergraduate or even graduate school level who know enough about two different fields to be interested enough to expend their energies to learn to do all the things you need to create a modeling language and use that efficiently, *and* develop enough expertise in another domain to be interested in those patterns applied to that particular domain. I think you might start seeing a fractionation in your group more toward what happens in hard science. For instance, if you do research in endocrinology — okay, in order to run an experiment in endocrinology you need expertise in molecular biology, cell biology, tissue culture, electron microscopy, biochemistry, and a whole series of assays which an individual researcher working at the endocrine level may not be interested in, but which are required in order for them to actually do the research that they *are* interested in.

I think that the software tools that you have are reaching a level of sophistication where it's not reasonable to expect someone who is not interested in that, in and of itself, to learn those tools enough to utilize them to their full capacity. The education of people in other domains should be directed more toward seeing the applicability of these tools to whatever they're doing and developing some sort of common grammar so they can discuss what their ideas are in terms that the modelers can understand.

Sallach: I would just like to add one other perspective on our talk about how things get done within academia or how things get done within industry. I have been in industry and am now in academia, and I would say the model is always evolving, and we don't know exactly what form it will take, but we do know that within academia there has been computational support in various forms. There have been computer labs, there have been statistical support consultants, there have been data libraries, and so forth. And I think the question is, now that we're moving on to a new generation of types of analysis and a new generation of tools, how can we focus comparable kinds of resources to provide infrastructure to allow researchers to focus on research? How can one leverage the various projects that are around, to find some sort of common infrastructure? That's what we're undertaking [here], and we're very early in the process yet, but I believe this is something that is potentially achievable.

But I think it also has curricular implications. That is, the kinds of things that we're talking about today, and in fact computational social science in its broader sense, are cross-disciplinary. It's certainly cross-disciplinary across the social sciences, and it's actually cross-disciplinary in a broader sense. And I think that to the extent that the social sciences can begin to share a common vision, integrate a common sense of what computational foundations are there, it will make it that much easier then to frame and define and generate the kinds of infrastructure resources that will be useful.

Nagendra Prasad: Back to the question you asked before, I thought a little more about what it takes, say, in a company like Anderson, to get work like this done. In our case, it did happen the way I said: I went out and asked for a programmer after we had the idea — basically I told them I'd teach them Java — and then got some help. But usually the way it happens is that [the clients] don't care if you're going to use agent-based programming. They don't care if you use system dynamics. They don’t really care what the technology is. You have to talk to them in terms of the value proposition that you give to them in their language.

So when I go give the talk to somebody inside Anderson, I rarely ever mention “agent based” or I just do it in passing. So you have to talk in terms of “I’m going to solve this problem that you have.” So they really don’t care what’s below the hood; they just bother about whether it’s a Corolla or a Camry. You have to really learn to talk in their language. Also even in terms of getting subject-matter experts, often you’ll have to work with problems that most probably you’re not an expert in. So you’ll have to create that trust relationship, both at the level of building the designs to get the knowledge and at the level of selling it once you have built the system.

Richard Burton: If I may change the subject for just a second, I want to take a moment for all of us who have been invited to this to thank the organizers in the University of Chicago and Argonne, and David [Sallach] and Tom [Wolsko] in particular, for putting this together.

And then to comment on what David was saying a moment ago, I think that we’re talking about the founding of an institution here, and those of us on the outside think that’s exciting and want to wish you the very best in this, because I think we can’t answer some of the questions that this panel has talked about without having institutions like the one that you’re forming here in Chicago.

Macal: I think the panel will conclude at this point, and I’ll turn things back over to David for one final remark.

Sallach: There’s very little left to be said, really. The first thing I want to say is just thank you all for coming, for your attention, for your interest. You’ve made it a very lively and interesting conference, and we appreciate it. Secondly, we’ve had a great team to put this together. I want to thank everybody who’s participated. Some of them you’ve seen up here, some of them you’ve seen providing transportation, working with AV equipment, providing snacks, and so forth, and some of them you haven’t seen at all, and I hope that you’ll give them a good round of applause, because a lot of people worked really hard to set it up. And third, I think both the vision that Kathleen Carley described and the problems that were pointed out by the panels, the challenges that lie ahead, indicate that this is a first step and not a last step. We are certainly interested in finding ways of moving this area forward.

There’s a lot of interest in the possibility of doing this again next year. We’re not prepared to report on that today, but it will be a topic of discussion. If you have any suggestions as to how it could be improved, suggestions about priorities that you’d like to see, please send it to the e-mail address on the web site [http://www.cas.anl.gov]. We will continue to have the web site be a focal point and a way to provide additional information. And it will be there that any future plans will be announced, as well as via e-mail or other means.

So thank you again for your participation, and we look forward to working with you again in the future.

List of Attendees

Victor Aguirregabiria
University of Chicago

Gary Chun-I An
Cook County Hospital

Yukika Awazu
University of Illinois at Chicago

Robert Axtell
The Brookings Institution

Tom Baines
Argonne National Laboratory

Alessandro Teresio Barbarino
University of Chicago

Andrew Begel
University of California, Berkeley

Deborah A. Boucoyannis
University of Chicago

Gale A. Boyd
Argonne National Laboratory

Richard M. Burton
Duke University

Aaron B. Campbell
Anythink, Inc.

Allan Peter Campbell
Argonne National Laboratory

Kathleen Carley
Carnegie-Mellon University

John Hadley Christiansen
Argonne National Laboratory

Richard R. Cirillo
Argonne National Laboratory

Nicholson Collier
University of Chicago

Jean Czerlinski
University of Chicago

Marcus Daniels
Swarm Development Group

Robbie Sue Davidson
Argonne National Laboratory

Michael C. Dawson
University of Chicago

Scott D. DeWald
Naval Surface Warfare Center

Catherine Dibble
University of California, Santa Barbara

Ulrich Doraszelski
Northwestern University

Ron Fisher
Argonne National Laboratory

Charles H. Franklin
University of Wisconsin, Madison

Alice Dawson Furumoto
Rush Med. College, Cook County Hosp.

Xavier Gabaix
University of Chicago

Richard Gaylord
University of Illinois

Gene B. Gragg
University of Chicago

Bruce Patrick Hamilton
International Atomic Energy Agency

Donald Alan Hanson
Argonne National Laboratory

Patrick Heuveline
University of Chicago

Bill Horsthemke
Argonne National Laboratory

William D. Ingle
GRI

Mark J. Jusko
Argonne National Laboratory

John Kenny
University of Chicago

Vladimir Koritarov
Argonne National Laboratory

Erik R. Larsen
City University of London

Doowan Lee
University of Chicago

Irene A. Lee
Swarm Development Group

David J. LePoire
Argonne National Laboratory

Fredrik Liljeros
University of Stockholm

Victor D. Lofgreen
University of Chicago

Ian Lustick
University of Pennsylvania

Charles M. Macal
Argonne National Laboratory

Matthew Craig Marlowe
University of Chicago

Guillermo Moloche
University of Chicago

John Padgett
University of Chicago

Miles T. Parker
The Brookings Institution

Maram V. Nagendra Prasad
Andersen Consulting

Michael John North
Argonne National Laboratory

James P. Peerenboom
Argonne National Laboratory

Scott Page
University of Iowa

Randal C. Picker
University of Chicago

Edward J. Plaskacz
Argonne National Laboratory

Mark Harlan Rich
University of Wisconsin

Meredith Rolfe
University of Chicago

Xiaomin Rong
University of Chicago

Vincent P. Roske, Jr.
Joint Staff, J-8

David L. Sallach
University of Chicago

Gerald Sheblé
Iowa State University

Michael S. Shum
University of Chicago

Kathy Lee Simunich
Argonne National Laboratory

Lee M. Smith
Ernst & Young LLP

Milan Svolik
University of Chicago

Craig Swietlik
Argonne National Laboratory

Pamela J. Sydelko
Argonne National Laboratory

Adrian M. Tentner
Argonne National Laboratory

W.H. Thomas
Naval Surface Warfare Center

Jeffrey J.P. Tsai
University of Chicago

Lefteri H. Tsoukalas
Purdue University

Kaz Uekawa
University of Chicago

Chuck VanGroningen
Argonne National Laboratory

Jack C. Van Kuiken
Argonne National Laboratory

Thomas Daniel Veselka
Argonne National Laboratory

Peter D. Viechnicki
University of Chicago

Ronald G. Whitfield
Argonne National Laboratory

Thomas D. Wolsko
Argonne National Laboratory

Jing Yang
Bank of Canada