Active Nonlinear Tests (ANTs) of Complex Simulation Models

John H. Miller

Department of Social and Decision Sciences, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213

Simulation models are becoming increasingly common in the analysis of critical scientific, policy, and management issues. Such models provide a way to analyze complex systems characterized by both large parameter spaces and nonlinear interactions. Unfortunately, these same characteristics make understanding such models using traditional testing techniques extremely difficult. Here we show how a model's structure and robustness can be validated via a simple, automatic, nonlinear search algorithm designed to actively "break" the model's implications. Using the active nonlinear tests (ANTs) developed here, one can easily probe for key weaknesses in a simulation's structure, and thereby begin to improve and refine its design. We demonstrate ANTs by testing a well-known model of global dynamics (World3), and show how this technique can be used to uncover small, but powerful, nonlinear effects that may highlight vulnerabilities in the original model.

(*Testing Simulation Models; Nonlinear Sensitivity Analysis; Validation; World3 Model; Genetic Algorithms*)

1. Introduction

Complicated, large-scale computational models are becoming increasingly common in the analysis of critical scientific, policy, and management issues. Such models are particularly well suited for the analysis of phenomena that are characterized by high degrees of nonlinearity and enormous search spaces—conditions that confound traditional analytic methods. Unfortunately, the same conditions that make computational techniques so appealing, are also ones that make validating[1] such models inherently difficult. For example, judgments about multivariate sensitivity made on the basis of independent univariate sensitivity analysis, require *linear* relationships among the parameters to be valid. As an alternative, here we show how a computational model's structure and robustness can be tested via a simple, automatic, nonlinear search algorithm designed to actively "break" the original model's implications.

The Active Nonlinear Tests (ANTs) described below define a general class of algorithms useful for the exploration of complex simulation models. Some potential applications of ANTs to management science, include:

- *Multivariate Sensitivity Analysis*: uncovering model sensitivities to groups of parameters.
- *Model Breaking and Validation*: exploring conditions under which a model breaks down.
- *Extreme Case Scenario Discovery*: finding best or worst case scenarios that could result given reasonable parameter changes.
- *Policy Discovery*: discovering strategies for achieving some (un)desirable outcome within the context of the model.

The basic idea behind the ANTs developed here is rather simple: use a nonlinear optimization algorithm to search across a set of reasonable model perturbations with the objective of maximizing the deviation between the original model's prediction and that obtained from the perturbed model. Of course, by varying the objective function for this optimization one can stress the model in different ways—the actual choice will depend on

[1] For a discussion of model validation see Law and Kelton (1991).

0025-1909/98/4406/0820$05.00

what aspect of the model one wants to test, but presumably it will include those implications of the model that are of the greatest importance to the modeler. In a model designed to inform a particular policy or management decision, the goal might be to reverse the predictions of the model that are most in favor of a particular decision. In models designed to create theoretical predictions (see for example, Holland and Miller 1990), ANTs can find conditions under which the derived theory might fail. ANTs can also be used to enhance the exploration of ensembles of models that incorporate a variety of plausible underlying assumptions (see Bankes 1993 and Bankes and Gillogly 1994).

Perhaps the simplest application of this methodology is in testing a model's behavior vis-a-vis its parameters. If the underlying model is highly nonlinear, then information about the impact of altering individual parameters may not be useful in determining the effect of changes in groups of parameters.[2] Because ANTs search across *sets* of parameter values, they are capable of detecting important nonlinear relationships among the parameters—relationships that typically go unnoticed using standard techniques that manipulate parameters in isolation from one another. While detecting nonlinearities could be done by exhaustively searching over all possible parameter groupings, the implied combinatoric explosion makes this infeasible for even modest numbers of parameters.[3] Thus, there is a need for a more directed search mechanism that can efficiently seek out *groups* of parameters that affect the model. The use of a nonlinear search algorithm allows such a directed search to occur.

The ANTs developed here automatically probe for weaknesses in the model's behavior. While such an exercise does not give an estimate of the likelihood of such scenarios (existing techniques like Monte Carlo methods[4] can be used for this task), it does give valuable insight into the possible extremes of the model. More importantly, these searches are a means by which to uncover potential weaknesses in the model's formulation and identify key assumptions. With this information, the model can either be refined or, if it is felt to be sound, additional effort can be focused on better estimating and understanding the behavior of the key assumptions. Note that the inability to "break" a model in this way does not guarantee its quality. For example, models that are completely insensitive to their parameters can obviously not be broken in the above manner—yet, such models are also not likely to be of much value. The tradeoff between the brittleness in a model and its responsiveness to parameters will always need to be carefully considered.

The ANTs methodology can be implemented in a variety of ways. In the next section, we discuss the basic elements of ANTs. We then provide an illustration of the technique in §3 by testing a simulation model of global population and resource dynamics (the World3 model developed by Meadows et al. 1974, 1992). Underlying this model is a large set of uncertain parameters, for example, current population stocks, technological growth rates, and agricultural productivity. We will use ANTs to find new sets of parameters (constrained to lie within a narrow range of the original ones) such that some of the model's main conclusions about the future path of population growth will be most at odds with what the original parameters suggest. The use of World3 in this section was simply due to its wide availability and easy accessibility. Section 4 provides some concluding remarks.

2. Active Nonlinear Tests

To implement ANTs, we use a nonlinear optimization algorithm to search over perturbations in the model's formulation, in order to maximize an objective function designed to "stress" the model. A wide variety of optimization algorithms are suitable for ANTs, and we discuss two possibilities below. Once the optimization

[2] For example, consider the model $y(x_1, x_2) = x_1x_2$. Then, $y(x_1 + \Delta x_1, x_2) = x_1x_2 + x_2\Delta x_1 = y(x_1, x_2) + x_2\Delta x_1$. Similarly, $y(x_1, x_2 + \Delta x_2) = y(x_1, x_2) + x_1\Delta x_2$. Yet, $y(x_1 + \Delta x_1, x_2 + \Delta x_2) = y(x_1, x_2) + x_2\Delta x_1 + x_1\Delta x_2 + \Delta x_1\Delta x_2$, which differs from the combination of the individual perturbations by the last term, $\Delta x_1\Delta x_2$. Of course, the linear approximation works well if either the perturbations are small or the degree of nonlinearity is low.

[3] For example, in a model with 100 parameters (each of which could either be high or low by a fixed amount), analyzing individual parameters requires 200 runs of the model, while 2 and 3 parameter groupings require 19,800 and 1.3 million additional runs, respectively.

[4] For reviews of methods to capture parametric uncertainty see Cox and Baybutt (1981), Iman and Helton (1988), and Morgan and Henrion (1990).

algorithm is determined, the modeler must decide on what aspects of the model's formulation and conclusions to test. There is a lot of flexibility in both of these choices, and here we simply illustrate some of the basic issues surrounding these choices. The inherent flexibility of these choices make ANTs useful for a variety of tasks.

The basic ANT algorithm has the following form:

1. Let $M_h(p)$ give the implications of the model for hypothesis h under assumptions p. The hypothesis, h, is any conclusion of interest from the model, and the assumptions, p, can include the model's underlying parameters, structural elements, etc. Let $\hat{p}$ give the original assumptions of the model.

2. Define Δp to be the set of allowable perturbations of the model's assumptions.

3. Let $\Pi(M_h(p), M_h(\hat{p}))$ be a judiciously chosen objective function designed to illuminate the model's behavior surrounding hypothesis h.

4. Use an optimization algorithm to maximize $\Pi(M_h(p), M_h(\hat{p}))$ over $p \in \Delta p$.

2.1. Two Simple Optimization Algorithms for Active Nonlinear Testing

A variety of existing optimization algorithms are suitable for use with ANTs. Good algorithms for this task should be capable of searching over nonlinear objective functions while (potentially) confronting noise, discontinuities, and enormous search spaces. Depending on the inherent nonlinearities in the model, the objective functions used in ANTs may be ripe with local optima. Moreover, discontinuities in the search space may arise either due to the nature of a simulation's parameters or to the types of allowable model perturbations under consideration, say, assumptions about the model's fundamental structures like probability distributions, structural equations, etc. Finally, it will often be necessary to economize on the number of iterations of the actual simulation model, and thus algorithms that can direct their sampling and operate with relatively few runs of the simulation model will be useful. In the example discussed in §3, we use both a simple hill-climbing algorithm and Holland's (1975) genetic algorithm (GA). Both of these algorithms are fairly robust to the difficult search conditions mentioned above. Of course, if the underlying model is continuous and well-behaved with respect to the assumptions that one desires to explore, then more classical nonlinear programming techniques (see, for example, Glasserman 1991, L'Ecuyer 1990, and Luenberger 1984) are likely to be more "efficient" than hill-climbing or a GA. The two optimization algorithms explored in the examples below are merely for illustration of the ANTs technique using a set of fairly robust and easily implemented algorithms. Obviously, alternative algorithms may be of use depending on a modeler's particular situation.

The hill-climbing algorithm is a simple, widely-used optimization procedure. This algorithm works by initially choosing a random "solution" within the search space as the status quo. At each iteration of the algorithm, a new solution is randomly chosen from a neighborhood of the status quo. If the new solution results in a higher value of the objective function, it becomes the status quo; otherwise the status quo remains unchanged. The algorithm continues this process for a fixed number of iterations.[5] At the end of these iterations, the status quo is used as the ultimate solution.

GAs have proven to be an effective search technique in complex optimization problems (see, for example, Goldberg 1989). In a GA, a population of "solutions" is initially created at random. Each solution is then tested on the problem, and receives a measure of "fitness." The GA then creates a new population of solutions by both reproducing old solutions based on their fitness (with better performers being more likely to be reproduced) and creating some new solutions through "genetic operators." GAs typically employ two genetic operators: crossover and mutation. Crossover recombines pieces of existing solutions in a way that tends to preserve those parts of solutions that likely result in good performance. Mutation randomly makes small, unique alterations in a solution and thereby prevents the system from getting prematurely trapped in local optima. Once a new population of solutions has been created, a new "generation" of the algorithm begins, and the testing, reproduction and modification stages discussed above are iterated. Although the reproduction and

[5] Simulated annealing algorithms (see, for example, Otten 1989), use a similar approach to hill-climbing. They differ from hill-climbing in that they will sometimes accept inferior new solutions with a probability that is decreasing in the size of the loss and a "temperature" parameter that makes such acceptances less likely over time.

modification stages of a GA require very little computational power to implement, they result in a sophisticated sampling scheme on the key patterns underlying effective solutions (Holland 1975).

2.2. The Search Space

The optimization algorithm is used to search over a space of allowable perturbations in the model's formulation. The choice of this search space will determine what aspects of the model's formulation are tested. One useful search space is a well-defined neighborhood of the original parameters used in the model. The extent of the neighborhood for such a space could be determined by, say, uncertainties surrounding the measurement of each parameter. For example, we could assume that all parameters are measured with a 10-percent error.[6]

While parameter perturbations are likely to be a common use of ANTs, other more general notions of model perturbations can also be used. For example, ANTs could perform searches over different formulations of the model's underlying equations. One way to implement this idea is to allow arbitrary feedback loops to develop in the model by incorporating perturbation terms into the rate equations for each state variable. Another alternative is to evolve more general functional components using techniques like genetic programming (Koza 1992). ANTs can also be use to test a model's basic operating assumptions. For example, ANTs could be allowed to search over various assumptions concerning underlying probability distributions, spatial configurations, or the timing of agent activation. The advent of object-oriented simulation packages, such as Swarm (Langton et al. 1995), should facilitate such searches.

2.3. Objective Functions for ANTs

For ANTs we need to define an objective function such that its optimization, on the search space defined above, will provide some insight into the model. By carefully choosing the objective function the modeler can employ the ANTs for a variety of tasks. Let $M_h(p)$ give the outcome of the model for hypothesis h under assumptions p, and let Δp give the set of allowable perturbations. For example, in a model of world economic and social dynamics, $M_h(p)$ could concern hypotheses based on the predicted population in 2100, the variance of the population path, the peak population, the timing of a particular event like the population exceeding 10 billion, etc. One set of useful objective functions are those that test the limits of the model's behavior, by maximizing (or minimizing) $M_h(p)$. More generally, the model's basic predictions can be challenged by maximizing objective functions of the form $(M_h(p) - M_h(\hat{p}))^2$, where $\hat{p}$ are the original model's assumptions. This objective function encourages the ANTs to find perturbed formulations that most deviate from the original prediction.

Additional criteria can also be incorporated into the objective function. For example, in the objective functions discussed above, no attempt is made to impose parsimony on the number of perturbations. One application where parsimony is useful is when ANTs are used to identify particular areas of the model in need of further investigation. One way to achieve parsimony is to restrict the search space such that only small sets of perturbations are admitted at any given time. Alternatively, we can include a penalty term in the objective function that encourages parsimony. Under this approach a cost function, $\phi(p, \hat{p})$, where $\phi(\cdot)$ is an increasing function of the number (and perhaps size) of perturbations from the original assumptions, $\hat{p}$, is incorporated into the objective function. A more sophisticated way to focus model perturbations is to use known information on the likelihood of a set of perturbations, and seek those perturbations that lead to high expected deviations by maximizing $M_h(p)P(p)$, where $P(\cdot)$ gives the local probability density of the perturbations. Whenever additional criteria are incorporated into the objective function, careful attention must be placed on properly weighting the various criteria within the objective function.

Obviously, both the flexibility and power of ANTs are contingent on a judicious choice of the underlying objective function.

3. An Example

To illustrate ANTs we apply the above techniques to the World3 model,[7] scenario 2, developed by Meadows

[6] Additional distribution assumptions are easily accommodated. For example, different parameters could have different ranges, etc.

[7] Unlike most large-scale models, World3 is readily available. Model

et al. (1974, 1992). World3 emerged from an early global systems model (Meadows et al. 1972) and was designed to simulate those systems that are of most importance to human sustainability, including population, industrial output, pollution, and agriculture. Hayes (1993) provides a basic review of the model's development and operation. World3 is used because it represents a convenient example of a large-scale simulation model. While many aspects of the model's formulation have been critiqued (see, for example, Nordhaus 1973, Forrester et al. 1974, and Nordhaus 1992), the issues of concern here—testing nonlinear simulation models—transcend such controversies. The World3 model has approximately 150 equations that link 272 model variables. Of these 272 variables, 96 need to be initialized. Moreover, there are 508 parameters required to specify various functional forms in the model.[8]

For the ANTs done here, we focus on the predicted world population level as the modeling outcome of interest. In Figure 1 (and all subsequent figures) the population path designated as "original" is the one produced by the model with no perturbations. As seen in the figure, the model predicts that world population will peak at about 9.4 billion around 2040, and then experience a relatively rapid decline ending with a population of about 3.9 billion in 2100. Our ANTs will focus on the predicted peak at 2040 (presumably, an event of great policy importance), and attempt to alter this outcome of the model by both amplifying and minimizing the peak population.

We will apply ANTs to this simulation by probing the 96 variables that must be initialized. We consider perturbations of these variables that are within a range of 10 percent of the values used in the original model, with the actual perturbations constrained to $\{-10\%, -9\%, \ldots, 0\%, \ldots, 9\%, 10\%\}$. Thus, a "solution" to this problem will be a set of 96 values, each of which has an integer value on the range -10 to $+10$. For example, a solution may decrease the first variable by 5 percent, increase the second by 8 percent, leave the third unchanged, etc.

The two optimization algorithms previously discussed, hill-climbing and GA, are used for these ANTs. In addition, to help clarify the performance of the ANTs, we also use a random-search algorithm. In the random-search algorithm, we randomly generate a fixed number of potential solutions and then test each of them in the objective function—the best observed of these potential solutions is then used as the ultimate solution.

The hill-climbing algorithm used here begins by randomly picking a "solution" and designating it as the status quo.[9] At each iteration of the algorithm a new solution is created by taking the status quo and independently and randomly altering each perturbation in the status quo with probability $2/96$, giving an expectation of 2.0 alterations per solution. When a particular perturbation is chosen to be altered, a new perturbation value is randomly drawn from $\{-10\%, \ldots, 10\%\}$. The new solution is then tested, and replaces the status quo if it has a higher value in the objective function. Each iteration of this algorithm requires a single run of the simulation model.

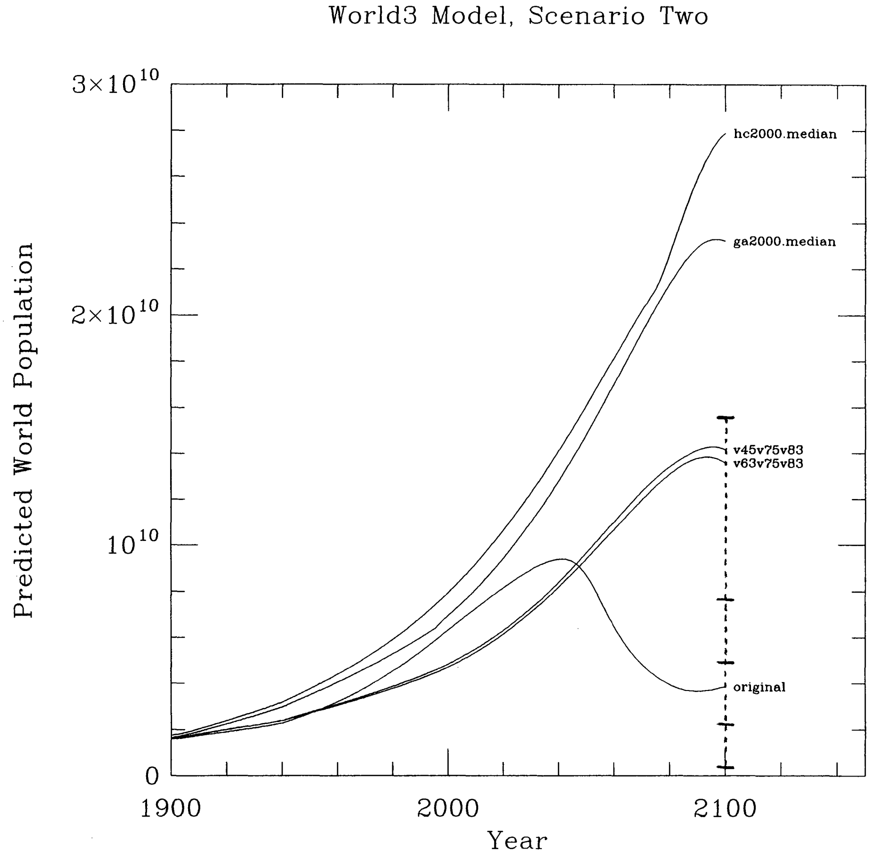
The GA used here begins with a population of 40 random solutions. During each generation of the algorithm newly formed solutions are tested on the objective function. Solutions are then reproduced by randomly selecting two old solutions (with replacement) and keeping the one with higher fitness (this procedure is known as tournament selection). This selection process is repeated forty times for each generation, resulting in a biased (by better performance) sample of forty solutions drawn from the old population. The 40 new solutions are then randomly paired. Each pair is subject to modification via genetic operators with a 0.5 probability. If modified, each solution in the pair is first mutated using the same procedure described above for altering the status quo point in hill-climbing, except that the initial probability of mutation has an expectation of 5.0 alter-

kits are available from the Institute for Policy and Social Science Research, Thompson Hall, University of New Hampshire, Durham, NH 03045.

[8] Ken Simons kindly provided the Pascal code for World3, as well as expertise on the model's structure and parameters.

[9] To create an initial set of perturbations, 96 random values are independently drawn from $\{-10\%, -9\%, \ldots, 0\%, \ldots, 9\%, 10\%\}$, with each of the 21 elements equally likely. This same procedure is also used for generating each member of the first population in the GA, and the sample points used in random-search and the Monte Carlo analysis.

Figure 1 Original World3 Model (Scenario 2) Predictions, Monte Carlo (n = 2000) Analysis, and Paths Generated by Various ANTs Maximizing 2100 Population



ations (and is then annealed over time, decaying in half every 50 generations). Each pair of solutions subject to modification also exchanges randomly chosen sequences of perturbations with one another in a crossover operation. After each pair has a chance of undergoing modification, a generation of the algorithm is concluded, the newly formed solutions are tested in the simulation, and the procedure is iterated. Each generation of the GA creates 20 new solutions, and thus requires 20 runs of the simulation model. The GA described above is quite robust to various changes in its underlying parameters and procedures, and a variety of reasonable variants appear to work equally well.

3.1. Unconstrained Maximization of Predicted 2100 Population

The first ANT of the model tries to eliminate the observed population peak by maximizing the predicted population in 2100. In essence, we are seeking a worst-case scenario of a maximum population stock at 2100, assuming only minimal deviations from our expected parameters. The objective function for this ANT is $M_{\text{pop:2100}}(p)$, where $M_{\text{pop:2100}}(\cdot)$ is the model's predicted population in 2100. Each method was run 30 separate times on the problem. For each of these 30 trials, the algorithms were randomly restarted. The resulting population paths for the median performing ANTs (among the 30 trials) are shown in Figure 1 for the hill-climbing (hc2000.median) and GA (ga2000.median) methods, requiring 2000 iterations of the simulation model. On average, the hill-climbing and GA ANTs find ending population values that are about six times higher than predicted under the original model. The best observed ANT found a set of perturbations that leads to a final population of around 29 billion, versus the model's prediction of

4 billion. The dotted vertical line at 2100 indicates the range (the two end points), mean (center tick), and one standard deviation interval (upper-middle and lower-middle ticks) of values observed in a typical Monte Carlo analysis. Note that the ending values of the ANTs lie well outside of the range observed in the Monte Carlo sample (the average value emerging from the ANTs is about 6.5 to 8 standard deviations above the Monte Carlo mean).

The descriptive statistics for the solutions emerging from the 30 separate trials (each trail beginning with a random restart) across the three search methods are given in Table 1. The numbers after each method indicate the total number of iterations of the simulation model used in the method. Thus, each of the 30 random-search-1000 trials randomly created 1000 perturbations, the best of which was used as the ultimate solution. Similarly, each hill-climbing-2000 trial iterated the status quo for 2000 steps and used the final status quo value as the ultimate solution. Lastly, each genetic-algorithm-500 used the best solution observed in the 25th generation of the GA.[10]

A number of conclusions can be drawn from Table 1. First, both the hill-climbing and GA procedures significantly outperform random search. This implies that these two algorithms can exploit some underlying structural aspects of the problem during the search process. Note that with the exception of genetic-algorithm-500, these two methods do better on average than the maximum observed during any of the 30 random-search trails. Second, for an equivalent number of iterations, hill-climbing outperforms GA *in this problem*. There is no guarantee that this performance difference will hold in general, and given that the intent of the above example is to illustrate an application of ANTs, we leave further analysis of this issue for future research. Finally, note that increasing iterations (computational resources) increases performance, but at a decreasing rate—a typical observation for such optimization problems.

Table 1 **Descriptive Statistics for Solutions Emerging from Various Methods Designed to Maximizing 2100 Population**

Method	Mean	(s.d.)	Max	Min
Random-search-500	14.9	(1.0)	19.1	13.4
Random-search-1000	15.2	(1.2)	18.0	13.3
Random-search-2000	15.7	(0.8)	18.0	14.0
		ANTS		
Hill-climbing-500	22.4	(2.3)	26.0	17.4
Hill-climbing-1000	25.1	(2.2)	27.4	20.7
Hill-climbing-2000	26.6	(2.3)	29.2	22.5
Genetic-algorithm-500	17.8	(1.4)	20.4	15.7
Genetic-algorithm-1000	20.7	(1.5)	23.4	17.7
Genetic-algorithm-2000	23.2	(1.7)	26.1	20.7

30 trials of each method, all values in billions.

3.2. Maximization of 2100 Population with Parsimonious Perturbations

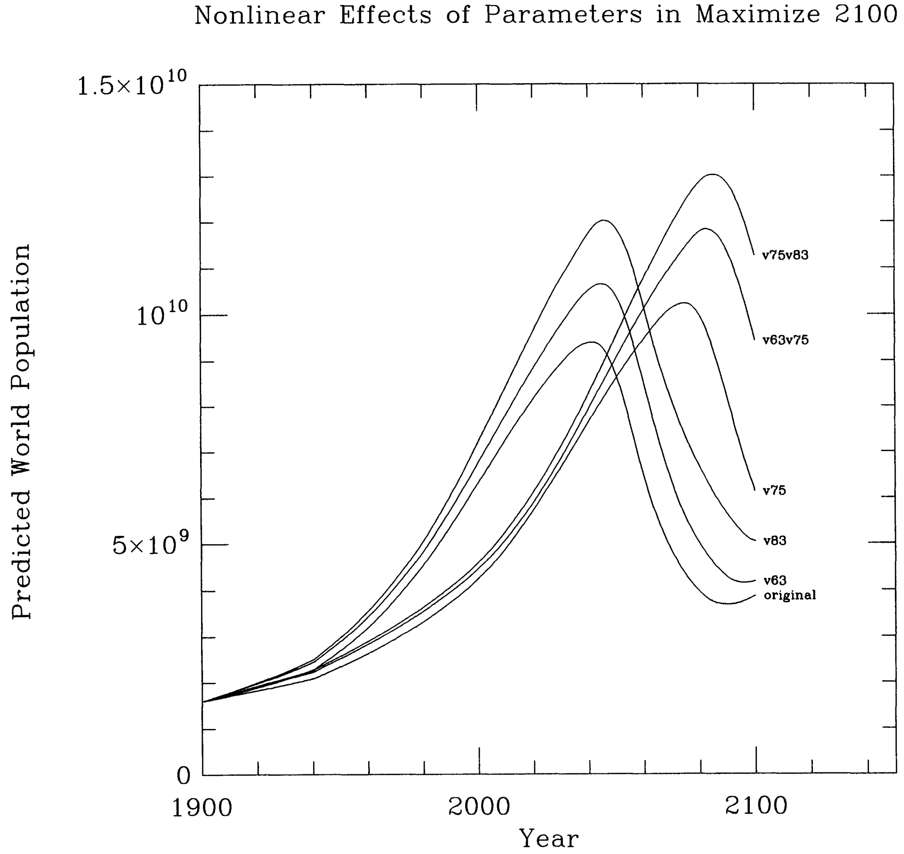
The second ANT attempts to maximize 2100 population while simultaneously minimizing the number of perturbed parameters. Such an ANT is useful for uncovering those parameters of most importance to a particular outcome, and in so doing highlighting either those parameters that should be most carefully estimated (if you believe the underlying model) or structural areas of the model that may need further investigation.

To impose parsimony, we both restrict the search space and implement a penalty on excessive perturbations. To restrict the search space, perturbations that are less than 7 percent in absolute value are not allowed—that is, each perturbation is restricted to {−10%, −9%, −8%, 0%, 8%, 9%, 10%}.[11] This modification helps the algorithm by restricting the potential search space to only extreme perturbations. We also impose a penalty for nonzero perturbations in the objective function. This is done by including a cost that increases with the square of the number of

[10] The three iteration levels in the hill-climbing and genetic algorithm methods were not generated from independent trials. For each run of these procedures, three data points (after 500, 1000, and 2000 simulation iterations) were collected. This provides some indication of the potential performance gain from additional iterations in these algorithms.

[11] Note, however that we use a random generation procedure similar to the one defined above (21 possible values), so that the likelihood of a perturbation going from a 0% value to a nonzero one is 30% (6/20).

Figure 2 Nonlinear Effects of Parametric Perturbation



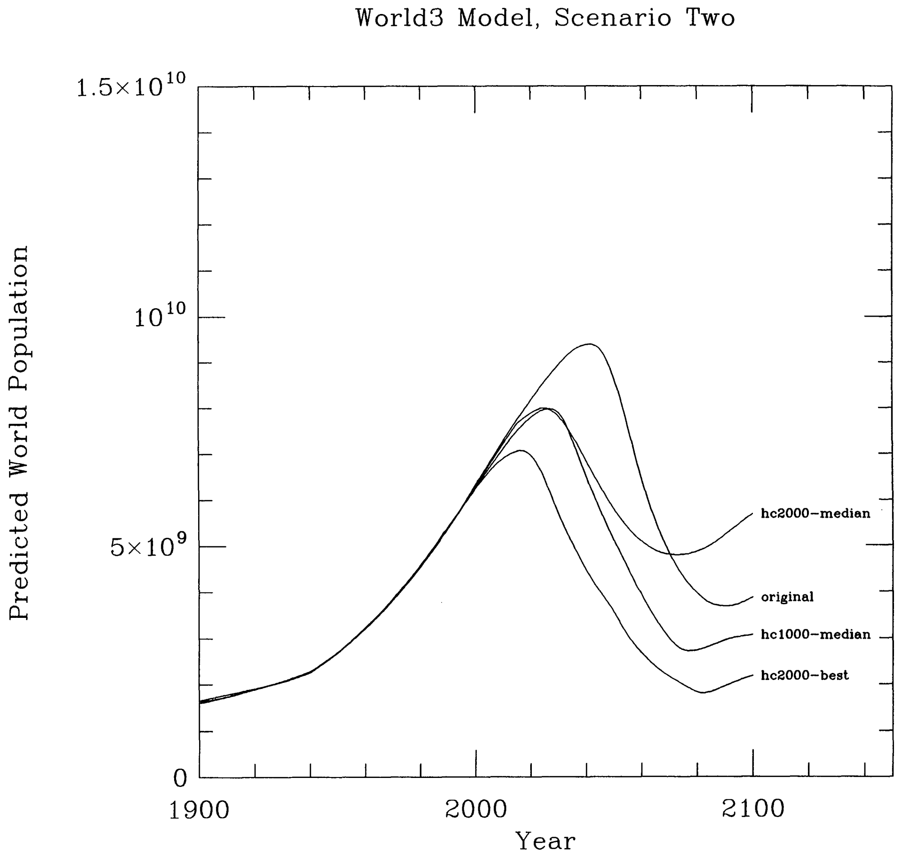
nonzero perturbations in the solution. The actual objective function was $M_{\text{pop:2100}}(p) - 10^8\phi(p, \hat{p})$, where $\phi(\cdot)$ gives the squared number of nonzero perturbations in p. By using the squared number of nonzero terms, the addition of a new perturbation is increasingly penalized. Note the use of a scaling parameter (10^8) to trade off increases in perturbations with increases in 2100 population.

Figure 1 shows two of the paths (each labeled by the parameters that are perturbed) discovered by the hill-climbing and GA ANTs. Both algorithms found similar sets of parameters.[12] By altering three parameters the ending population can be increased from 4 to around 14 billion. Note that encouraging parsimony lessens the magnitude of the model's breakage (this will always be the case since the optimization with parsimony is always a constrained version of the one without parsimony), yet still results in a very different outcome from that predicted by the original model.

The solutions found using the constrained procedure provide a simple demonstration of the impact of nonlinear effects. Figure 2 shows the original path along with perturbations in various model parameters (each line is labeled by the parameters involved). While each individual perturbation has an impact, the combination of the two simultaneous perturbations results in a much more extreme effect. Thus, while altering parameter 75 increases ending population by about 2.3 billion and altering parameter 83 increases it by about 1.2 billion, the combination of the two parameters causes a 7.4-billion increase. Similarly, while parameter 63 (the life expectancy normal) alone increases the ending

[12] In fact, parameters 75 (the fraction of industrial output allocated to consumption) and 83 (the reproductive lifetime of females) emerged as key perturbations in many of the tests.

Figure 3 **Original World3 Model (Scenario 2) and Paths Generated by Various ANTs Minimizing Peak Population while Conforming to Original Predictions prior to 2000**



population by only 0.3 billion, when it is combined with parameter 75 the increase is 5.6 billion.[13]

3.3. Minimizing Peak Population While Conforming Before 2000

The third ANT we explore attempts to minimize the peak population observed during any given run of the simulation, while simultaneously avoiding deviations from the original predictions during the first 100 years. One interpretation of this ANT is that we are seeking a best-case scenario while having the model still conform to known observations. If we assume that the 96 parameters are, say, policy control variables, then an alternative interpretation of this procedure is that we are using the ANT to discover useful policies (in this case, finding a sustainable population path given our past experience). To implement this ANT we minimized the peak while penalizing percentage deviations away from the predicted path before 2000. We accomplish this by maximizing the following objective function: $-M_{\text{peak}}(p) - \alpha\Sigma_t(M_{\text{pop}:t}(p)/M_{\text{pop}:t}(\hat{p}) - 1)^2$, where M_{peak} is the peak population observed, $M_{\text{pop}:t}$ is the population at time t, the summation is conducted for t = 1920, 1940, 1960, 1980, and 2000, $\hat{p}$ are the original parameters, and α was chosen so that a 1-percent average deviation was equivalent to a 1-billion drop in the peak population. Using this function, the hill-climbing algorithm was run 10 separate times for both 1000 and 2000 iterations. Figure 3 shows both median solutions and the best solution of the ten from the 2000 iteration run. As can be seen from the figure, these ANTs are able to overcome the popu-

[13] For this demonstration, parameters 63 and 75 were both increased by 10%, and parameter 83 was decreased by 10%.

lation momentum of the system generated before 2000 and substantially lessen the predicted peak population.

The above objective functions are only a small subset of those likely to yield interesting insights. The ANTs indicate that dramatic changes in the predictions of the World3 model can result from even minor changes in some parameters. As previously discussed, the occurrence of such events does not necessarily imply a faulty model—good models must be responsive to their parameters. Nonetheless, they do indicate the potential for extreme errors, as well as suggest structural areas of the model that might require further investigation and refinement.

4. Conclusions

The growing reliance on large-scale computational models is likely to continue as the cost of computation declines and as researchers and policy makers attempt to confront ever more sophisticated phenomena. The comparative advantage of such models is their ability to embrace systems characterized by large parameter spaces and rampant nonlinearities. Yet, it is these same characteristics that make testing and understanding such models inherently difficult.

ANTs allow a new class of testing to occur on computational models. Given a particular conclusion of the model, an ANT can be easily implemented by defining a corresponding objective function and an acceptable class of model perturbations. Once initiated, the ANT automatically probes the model for weaknesses. Unlike standard techniques that often look at perturbations independently from one another, ANTs are able to find important nonlinear interactions among the model's perturbations. Through the use of ANTs better models can be developed and refined in this complex environment.

ANTs can also be productively employed in those models in which one is confident of the underlying simulation's behavior. In these cases, ANTs can be used to discover worst (or best) case scenarios and therein give the user an idea about which parameters should either be altered (if possible) or most closely monitored. For example, one could use ANTs to actively try and break the operation of, say, a simulated aircraft flight system, battle field strategy, complex control system, or a computer interface or security system. In a model with policy control variables, ANTs can be used to discover the best policy given the model and a well-defined set of policy objectives. Branley et al. (1997) provide a variety of examples of such decision oriented applications using a related notion called "candle-lighting analysis."

Tools like ANTs provide a window into the structure of complex simulation models. Such a window may allow us to begin to uncover some generic properities of these types of models. For example, do complex models imply complicated, nonlinear, parameter spaces? How sensitive are these models to initial conditions? The inherent flexibility of ANTs allows us to probe these models in a variety of ways, and from these incursions begin to develop answers to these fundamental questions.

Obviously, ANTs alone are not sufficient to guarantee quality modeling. Careful thought and refinement will always be needed on the part of the modeler to insure useful models. Tools like ANTs will certainly help in this process, but cannot substitute for it. The discoveries of ANTs may force changes in the model, closing old, and perhaps opening new, vulnerabilities. The active probing of the model by ANTs provides input from a ready, and tireless, critic. Like ants seeking food at a picnic, a variety of avenues are creatively explored, and it is only with extreme care and foresight that the meal remains untouched.[14]

[14] This paper has benefited from discussions with Robert Axelrod, Paul Fischbeck, Pierre L'Ecuyer, Murray Gell-Mann, Scott Page, Laura Painton, Ken Simons, Sally Sleeper, two anonymous referees, and other colleagues at both Carnegie Mellon University and the Santa Fe Institute. Research support was provided by the National Science Foundation SBR-9411025, the Center for Integrated Study of the Human Dimensions of Global Change supported by a cooperative agreement between the NSF SBR-9521914 and Carnegie Mellon University, and the Santa Fe Institute.

References

Bankes, S., "Exploratory Modeling for Policy Analysis," *Oper. Res.*, 41 (1993), 435–449.

—— and J. Gillogly, *Proc. Third Annual Conf. on Evolutionary Programming*, Anthony V. Sebald and Lawrence J. Fogel (Eds.), World Scientific Publishing, River Edge, NJ, 353–360.

Branley, B., R. Fradin, S. O. Kimbrough, and T. Shafer, "On Heuristic Mapping of Decision Surfaces for Post-Evaluation Analysis," Working Paper, The Wharton School, University of Pennsylvania, Philadelphia, PA, 1997.

Cox, D. C. and P. Baybutt, "Methods for Uncertainty Analysis: A Comparative Survey," *Risk Analysis*, (1981), 251–258.

Forrester, J. W., G. W. Low, and M. J. Mass, "The Debate on World Dynamics: A Response to Nordhaus," *Policy Sci.*, 5 (1974), 169–190.

Glasserman, P., *Gradient Estimation Via Perturbation Analysis*, Kluwer Academic Publishing, Nowell, MA, 1991.

Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, MA, 1989.

Hayes, B., "Balanced on a Pencil Point," *American Scientist*, 81 (1993), 510–516.

Holland, J. H., *Adaption in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, 1975.

—— and J. H. Miller, "Artificial Adaptive Agents and Economic Theory," *American Econ. Rev., Papers and Proceedings*, 81 (1991), 365–370.

Iman, R. L. and J. C. Helton, "An Investigation of Uncertainty and Sensitivity Analysis Techniques for Computer Models," *Risk Analysis*, 8 (1988), 71–90.

Koza, J. R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, 1992.

Langton, C., N. Minar, and R. Burkhart, "The Swarm Simulation System: A Tool for Studying Complex Systems," Working Paper, Santa Fe Institute, Santa Fe, NM, 1995.

Law, A. M. and W. D. Kelton, *Simulation Modeling and Analysis*, 2nd Ed., McGraw-Hill, New York, 1991.

L'Eculyer, P., "A Unified View of the IPA, SF, and LR Gradient Estimation Techniques," *Management Sci.*, 36 (1990), 1364–1383.

Luenberger, D. G., *Linear and Nonlinear Programming*, 2nd Ed., Addison-Wesley, Reading, MA, 1984.

Meadows, D. L., W. W. Behrens III, D. H. Meadows, R. F. Naill, J. Randers, and E. K. O. Zahn, *Dynamics of Growth in a Finite World*, Wright-Allen Press, Cambridge, MA, 1974.

Meadows, D. H., D. L. Meadows, J. Randers, and W. W. Behrens III, *The Limits to Growth*, Signet, NY, 1992.

——, ——, and ——, *Beyond the Limits*, Chelsea Green Publishing, Post Mills, VT, 1992.

Morgan, M. G. and M. Henrion, *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*, Cambridge University Press, New York, 1990.

Nordhaus, W. D., "World Dynamics: Measurement without Data," *Economic J.* 83 (1973), 1156–1183.

——, "Lethal Model 2: The Limits to Growth Revisited," Brookings Panel on Economic Activity, draft, 1992.

Otten, R. H. J. M., *The Annealing Algorithm*, Kluwer Academic Publishers, Boston, MA, 1989.

Accepted by Pierre L'Ecuyer; received February 15, 1996. *This paper was with the author* $4\frac{1}{2}$ *months for* 2 *revisions.*