TEMPEST: a Fast Spatially Explicit Model of Ecological Dynamics on Parallel Machines

Bill Maniatty
Boleslaw Szymanski*
Computer Science Department
Rensselaer Polytechnic Institute
Troy, NY 12180

Tom Caraco

Department of Biological Sciences
State University of New York at Albany
Albany, NY 12222

Abstract

The spatial and temporal aspects of population dynamics are pivotal to computational biology. This paper focuses on a spatially explicit model of four species in an environment that behaves like a large probabilistic cellular automaton. The cells of the automaton represent discrete *sites* into which the environment is partitioned. Probabilistic local state transitions are executed synchronously at all sites making the simulation suitable for parallel implementation on SIMD architectures. Measuring the simulation results requires computing global parameters of the ecological environment. Such computations are challenging to implement efficiently on SIMD machines. The simulation produces a large volume of data, requiring the use of visualization for model verification and result interpretation.

In this paper, the parallel implementation of a spatially explicit two-species competition is discussed. First, the model performance is analyzed. The results indicate that the use of a massively parallel machine was necessary and efficient. There is also a discussion of implementation and use of visualization tools. Finally, the biological results are presented. Some of these results could arise only in spatially explicit ecological models.

1   INTRODUCTION

The spatial context of ecological interactions is important in understanding population dynamics, community stability, and biodiversity [Hastings 90, Kareiva 90]. However, the computational power necessary for implementing such spatially explicit models has only recently become available [Turner 89]. This paper discusses a spatially explicit model of multi-species ecological interaction implemented on a massively parallel computer.

Many natural processes, including competition, can be characterized by a global state that is defined as a vector of local states. Correspondingly, the composition of local state transitions determine the global state transition [Szymanski and Caraco 1993]. To simulate the spatial and temporal dynamics of a multi-species interaction, their environment is partitioned into a grid of sites; each site can support at most one organism. Each site's state indicates the presence or absence of species involved in a simulation. At each time step our probabilistic model synchronously computes each site's state transition as a function of its current state and the states of its neighbors. The neighboring sites which have influence on an affected site form a *stencil*. Typically, the stencil is defined as a rectangular collection of sites with the affected site located inside it (but not necessarily at the center). The size of the ecological stencil and the location of the affected site within it reflect both biotic and abiotic factors.

Reliable modeling of ecological processes is of prime importance in evolutionary biology and environmental sciences. Such processes as the spread of diseases or invasion by exotic species impact agriculture, health, economy and the integrity of natural ecosystems. The significance of these processes and the computational power required for their simulation make ecological modeling an important high-performance computing application.

Two-dimensional spatial modeling of ecological systems has seldom been attempted. Most of the models in the literature restrict spread of species by transmission to immediate neighbors. Compared to earlier work, the model considered in this paper allows far more complete analyses. The size, shape and the probabilistic dynamics of the ecological stencil can be manipulated to simulate a variety of local effects and then to observe ecological changes across a series of large-scale environments. As

shown later, the size of ecological stencil and the long-range interaction between competing species give rise to effects that are not accountable in less precise ecological models.

The paper is organized as follows. The model and its biological background is discussed in Section 2. The computational aspects of our work, including the performance results, are discussed in Section 3. Visualization tools are presented in Section 4. The ecological results of the simulations are discussed in Section 5. Finally, conclusions and future directions are given in Section 6.

2 MODEL AND ITS BIOLOGICAL BACKGROUND

In its most general form the implemented model can simulate a four-species environment composed of two competing host species, a parasite feeding on both hosts, and a pathogen that employs the parasite as a vector between host individuals. A vector-borne pathogen can infect individuals of two host species. The host populations compete for space and, once an individual of one host species occupies a site, its mortality (freeing the site) is independent of the density, local or global, of the other species.

A single parasitic species can occupy a site only if an individual of either host population is present at that site: a parasite's range cannot exceed the range of its hosts. The parasite may carry the disease from infected to susceptible hosts of either species. The parasite generally reduces a host individual's survival and reproduction to a lesser extent than the pathogen. The parasite may prefer one host species. Consequently, our simulations can address the question of how behavioral selectivity influences the dynamics of an epidemic and so governs spatial pattern in an interactive biotic community.

The formal definition of the probabilistic model on which the implementation is based is presented in [Szymanski and Caraco 1993]. Here, the discussion focuses on those features of the model that affect the structure of the implementation.

The model partitions a two-dimensional *environment* into *sites*. Each site can support at most one *host* organism. There are two types of hosts: the ecologically dominant species is the *good host*, the inferior species is the *bad host*. *Parasites* are organisms infesting hosts. *Pathogens* are organisms infecting hosts. Parasites and pathogens can only live in the presence of a host. Each site is assigned a *state* indicating the presence or absence of each type of organism at that site. Each site is subject to the effects of activities in nearby sites which form the *ecological stencil*. The ecological stencil is assumed to be rectangular. An affected site may reside at an arbitrary position within the stencil, but its relative position is

the same for all the stencils. The local state transition is probabilistically selected for each site at every time step. The transition probability is a function of the site's current state, the states of sites in the ecological stencil, and some user defined parameters. Transitions correspond to phenomena such as birth, death, recovery from disease, etc. Only ecologically significant states and transition probabilities are considered (see Figure 1).
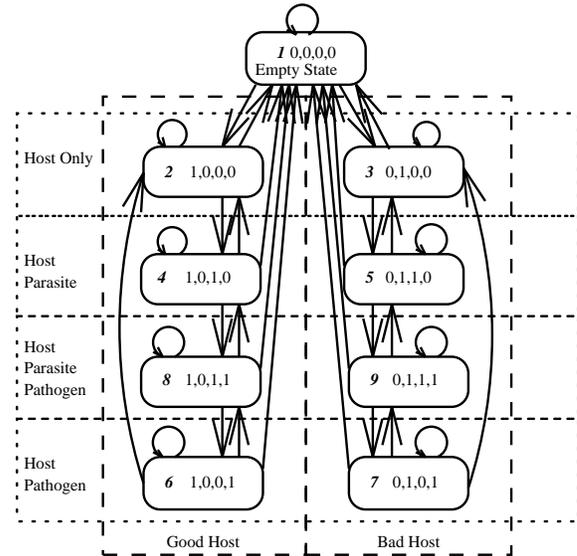
State numbers are in italicized numerals.
Organism Presence is Good, Bad, Parasite, Pathogen

Figure 1: The Model's State Transition Diagram

The implementation follows the model naturally. The simulation program iterates once for each time step, computing the stencil statistics and state transitions on each iteration. Sampling occurs at user specified intervals, at which time statistical information is gathered (sampling includes the initial and the final state of the simulation). Currently the implementation maps each site to the different processor of the MasPar computer. Only xnet communication is used to evaluate transition probabilities based on the current state of the ecological stencil.

3 COMPUTATIONAL ASPECTS AND PERFORMANCE

The simulation runs on a MasPar MP-1 SIMD architecture. The array of processing elements (PEs) of MasPar is organized in a two-dimensional grid. Each processing element has eight nearest neighbors with whom it can communicate directly using *xnet* operations. There is also a global router which allows communication between any two arbitrary PEs but imposes much higher communication cost than xnet operations.

The architecture matches the grid-partitioned environment being simulated. The sites naturally map onto the PEs. Local changes of state at each site are programmed as operations on local variables; whereas the global statistics about the temporal and spatial developments are stored in the front-end processor. The tight synchronization of SIMD architecture execution facilitates the update of the global state at each time step.

The developed algorithms can be classified into the following categories:

- *Computation of Transition Probabilities* - These probabilities are used in selecting the next state of each site. Computing them involves simultaneous evaluation of several functions over ecological stencils [Szymanski *et al.* 1993].

- *Fractal Dimension* - A measure of complexity of the spatial patterns arising in the environment: the more complex the spatial pattern, the larger the fractal dimension. For two-dimensional environments, fractal dimensions are between 1 and 2, where 1 corresponds to a line and 2 corresponds to a plane. Spatial complexity impacts many ecological processes, such as disease control (which becomes more difficult as spatial complexity increases).

- *Relative Patchiness* - A measure of ecological diversity, i.e., the average rate of change of the landscape along some direction.

In [Maniatty *et al.* 1993] the algorithms used in the implementation are discussed in more detail.

Thirty six executions of the model were timed using 32 bit integer and 64 bit floating point arithmetic on the MasPar MP-1 (with 2048 PEs, each with a 1.56 MIPS and 0.0336 MFLOPS rating) and the DECstation 5000/240HX (42.9 MIPS and 6.0 MFLOPS). The simulations differ in both the number of sites in the environment (one PE is allocated to a site on the MasPar) and the area of influence of each site (i.e., the size of the ecological stencil).

The speedup on the MasPar relative to the DECstation is computed as the ratio of the corresponding wall-clock computation times. The speedup is nearly linear both with respect to the number of sites (which is also the number of MasPar PE's used) and the stencil size. The biggest speedup obtained was about 24 for a 13×13 stencil and 2048 sites (i.e., 2048 MasPar PE's).

There is a disparity between MasPar and DECstation processing speeds for floating point and integer arithmetics. On MasPar the floating point arithmetic is ≈ 46.4 slower than integer one, whereas the slow-down
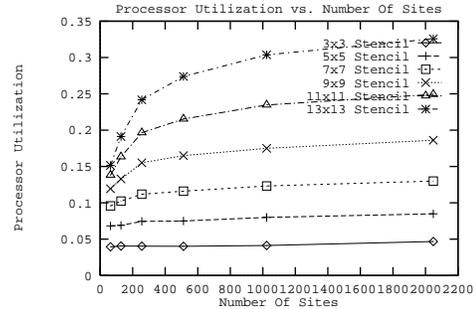

Figure 2: Processor Utilization of MasPar MP-1

is just ≈ 3.97 for the DECstation. Therefore, we introduced the measure of the integer-equivalent speed of MasPar, the speed that would be achieved if each floating point operation were replaced by 46.4 integer operations on MasPar and 3.97 operations on DECstation and the DECstation (with sequential implementation) would perform at its advertised speed. The maximum integer-equivalent speed was obtained for the 13×13 stencil with 2048 processors and it was equal $\approx 10^9$ instructions per second (about 1 GIPS).

Utilization of MasPar processors is obtained from MasPar's advertised speed and achieved integer-equivalent speed. It is plotted as a function of the number of sites and as a function of the stencil size in Figure 3. For the largest stencil and the maximal number of sites, simulation sustained high for SIMD architecture utilization of 32.6%. There are two reasons why the utilization is not constant. Firstly, conditional flow of control on SIMD architectures causes processors not taking a branch to wait for branch completion. There is a nine-way branch in computing the state transition probability inherent in the nine-state ecological model. Secondly, sublinear (logarithmic or square root) complexity of several algorithms in terms of stencil size on the MasPar results in increased efficiency of computation over larger stencils and environments.

4 VISUALIZATION TOOLS

The sheer volume of data generated by simulations makes visualization of data sets and animation of results an important issue in TEMPEST. Model validation and result interpretation require two capabilities:

1. Interactive Simulation Visualization capable of displaying the current state of the simulation.

2. Result Post-Processing capable of a simultaneous overview of a particular measurement for many executions of the simulation.

This section discusses the design and implementation of both capabilities.

4.1   Interactive Simulation Visualization

The *TEMPEST Viewer*, called *TV*, subsystem emphasizes visualization of ecological environments and ease of simulation control. The implementation of TV is based on *SUIT*, the *Simple User Interface Toolkit*, from the University of Virginia. A sample TV visualization is shown in Figure 3. Since the simulated environment can be viewed as a cellular automaton, a tomography is used to visualize the state of the ecological environment. A tomography widget was implemented in TV, because SUIT has none. The state of a site in the tomography is indicated by its color. The tomography has fixed colors, permitting fast color table look-up. The color table is defined in the creation call for the tomography, *Create-Tomography*. On average, relatively few of the sites in the environment need to be updated at a given time step so the repaint call, *PaintTomography*, has two modes: incremental repainting or a complete coloring. Hence, repaint uses *first_time* and *hit_flag* variables to determine its mode. When the user clicks the mouse with the widget located on a site, the site is said to be *hit*. The *HitTomography* routine causes the site to update its state. The updated state is selected through a radio-button display of possible values. The color code of the value is displayed in a *BlankCanvas* widget class. The user may vary the frequency of displaying the tomography from 1 to 50 time steps. This selection is made via a *bounded value* speedometer widget supplied by SUIT. The run button initiates the current step of execution. The main control of the simulation was rewritten in TV to act in a user exit fashion rather than batch looping.

4.2   Post Processing of Results

Often visualization of a measurement across several runs is needed to show a relationship between several parameters and perhaps demonstrate temporal correlation between the runs. For TEMPEST such visualizing is useful for simulations run with the following parameters.

- The same ecological stencil size.

- The range of species death rates, plotted as x axis.

- The range of species birth rates, plotted as y axis.

- The range of simulation total running times, each total running time producing a frame of the display, as seen in Figure 4.

The displayed results are the population densities at the end of simulations (along the z axis). The displayed results are produced in many runs, so a post processing
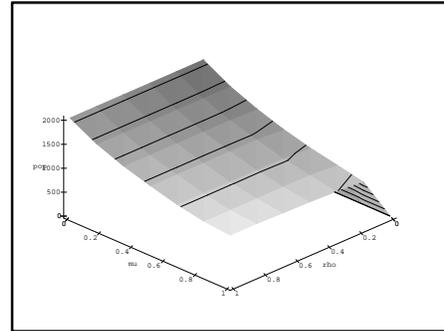

Figure 4: An Animation Frame

pass is used to format the data for input to a visualization tool. An animated surface plot is used for display. Frames are displayed in the ascending order of their total simulation time. The animation uses *Maple*, a symbolic mathematics package. TEMPEST has a Unix script to run the simulations for the given ranges in the parameter set. The post processing data formatting is done by another Unix shell script and a TEMPEST program. The animation is driven by a Maple script.

It is interesting to note that on a relatively fast sequential machine, a SUN SPARCstation 10, it takes longer to format the data for viewing, than it does to generate all the data points on the MasPar MP-1. The future generation of TEMPEST might use a more efficient visualization package than that provided by Maple.

5   BIOLOGICAL RESULTS AND INTERPRETATIONS

So far, TEMPEST has been used to simulate:

- the spread of epidemics [Maniatty *et al.* 1993],

- population dynamics of single species systems,

- populations dynamics of two competing species.

In this section, we discuss simulation results for single-species and two-species systems. The purpose of simulations was to investigate extinction trends, rates of population spread and the population densities at the equilibrium of the system. An equilibrium density is characterized by equality of mortality and birth rates. The equilibrium density is stationary if a random deviation from the equilibrium point results in a state transition that brings the system closer to the equilibrium point. Otherwise the equilibrium is non-stationary (i.e., due to random variation of the densities the system will move away from it in a finite time).
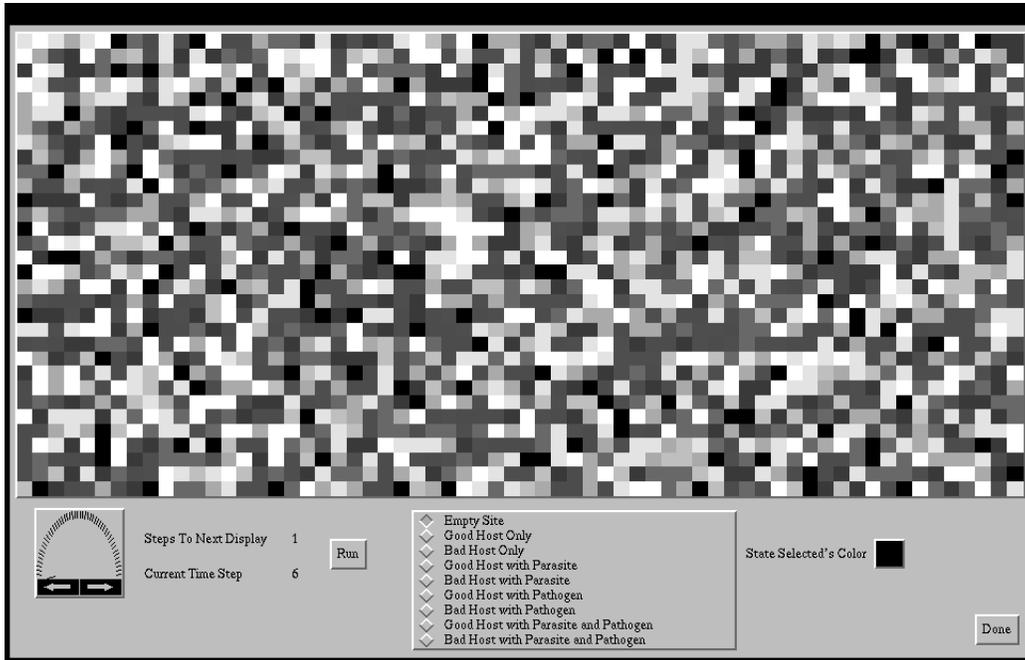
Figure 3: A TV Screen

The simulations were run with different values of the basic parameters: (i) stencil size, (ii) death rate, and (iii) birth rate. A single species system was simulated to validate the implementation.

5.1 Extinction Versus Dynamic Equilibrium

An interesting case of density dependent dynamics arises when the two-species system is considered. One of the two species, the *good* species, is assumed to be ecologically superior to the other, the *bad* species. Suppose that the good species invades the environment at low density after the bad species has reached an equilibrium density. Our theoretical [Szymanski and Caraco 1993] and simulation results show that the bad species can remain common, in a stationary equilibrium with good species, until chance events (i.e., reasons other than the presence of the good host) induce a drastic decline in the bad species's density. Only then is the (competitively superior) good species able to drive bad species extinct. This effect requires sufficiently large ecological stencils.

When a system of two species characterized by the same ecological parameters is considered, the sum of densities of two evenly matched species behave as a density of a single species system with the equivalent parameters. Such a system has four equilibrium points:

- three non-stationary, corresponding to an empty habitat, and the two single species systems (the
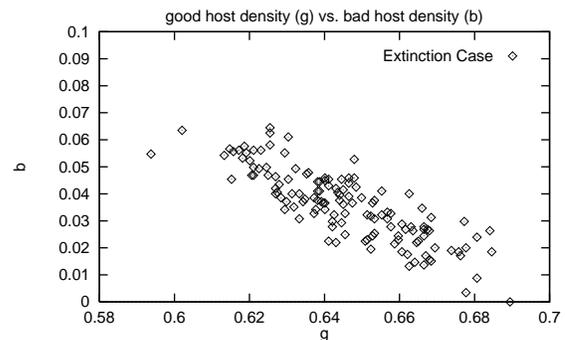

Figure 5: Extinction of Bad Species

other species are extinct), respectively,

- one stationary equilibrium with equal density of both species in the environment.

The character of each equilibrium is revealed by the values of the partial derivative in the neighborhood of each equilibrium point.

To demonstrate that multi-equilibrium systems can arise when species are not evenly matched, consider the system in which only mortality differs slightly between species, i.e., let mu_0 represent the mortality rate of the good host, and mu_1 represent the mortality rate of the bad host, then $\mu_0 = \mu + \epsilon > \mu - \epsilon = \mu_1$, for some $0 < \epsilon << 1$. Under this assumption, for some range of

values of ϵ a new stationary equilibrium point may arise [Szymanski and Caraco 1993] where both species coexist (with a somewhat higher good host density).

The simulations were used to determine the threshold value for ϵ such that larger values of ϵ will lead to extinction of bad species and smaller values will permit coexistence of both species. Search for the threshold value was done by bisection: the value of ϵ is halved when simulation results in extinction and doubled otherwise. If a simulation did not end in extinction within 2×10^4 time steps, then the stationary equilibrium was assumed.[1] The simulations covered the following cases:

1. Birth rate 0.5, death rates $0.5 + \epsilon$, $0.5 - \epsilon$ and the stencil size 9. This case was quite sensitive to small variations in ϵ. The longest run without an extinction was for 8740 time steps with $\epsilon = 0.054609$. Interestingly, a near threshold $\epsilon = 0.05004512$ extinction occurred in only 1848 time steps. The coexistence equilibrium arose for $\epsilon = 0.05004506$, with good species density 0.579102, and bad species density 0.101074.

2. Birth rate 0.3, death rates $0.7 + \epsilon$, $0.7 - \epsilon$ and the stencil size 9. This environment was less sensitive to changes in ϵ. Extinction occurred for $\epsilon = 0.006835$ in 2923 time steps. The coexistence equilibrium was reached with $\epsilon = 0.005859$. The resulting densities of good and bad species were 0.06885 and 0.45361, respectively.

3. Birth rate 0.3, death rates $0.7 + \epsilon$, $0.7 - \epsilon$ and the stencil size 25. This case required a substantially larger value of ϵ to coerce extinction, as predicted by our theoretical model. It is also interesting to note that the slowest extinction occurred in 13341 steps with $\epsilon = 0.2625$ (see Figure 5). Extinction did not occur for $\epsilon = 0.25625$ (see Figure 6).

We made few observations about the system dynamics. Systems that have coexistence equilibrium move towards it rather quickly, but an extinction could be a lengthy process. These observations reinforced the significance of the ecological stencil in our model and the need for high-performance computing to establish dynamics of the system reliably.

6 CONCLUSION AND IMPACT ON BIOLOGICAL SCIENCES

The simulation results obtained using TEMPEST have been both ecologically and computationally encouraging. The distribution and abundance of all species

---

[1]We also plotted densities of both species at the end of the simulation to verify that the bad species density is not decreasing.
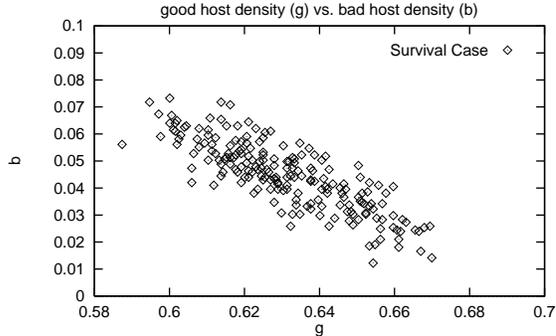

Figure 6: Survival of Bad Species

exhibit some degree of spatial variation. Spatial heterogeneity in abiotic factors or biotic processes may govern population dynamics and the resulting characteristics of ecological communities. The spatially explicit model used by TEMPEST addresses the population dynamics of (as many as) four species through simulation of the epidemiological landscape of a carrier-borne disease. Related ecological and evolutionary questions can be approached by modifying the model.

Our computational research resulted in novel parallel algorithms for problems encountered in implementing the discussed model, such as simultaneous reduction operation, computing the fractal dimension and relative patchiness. Future research directions include:

- Automated parallel code generation: Some speedups can be achieved by customizing the generated code to the required input parameters to avoid run-time condition execution that is expensive in SIMD architectures. Automated code generation also facilitates configuration and architecture independence, leading to ease of porting the software to other architectures.

- Virtualization of processors: Sites can be mapped onto logical processors so that the number of sites is not limited by the number of processors physically present in the architecture.

- Tools to compare simulation results with experimental results: This will help in determining the accuracy of a model in predicting the temporal dynamics of an ecological environment.

Future simulations will be based on more complex inter-species interactions than most analyses of spatial patterns in ecology offer.

References

[Hastings 90] Hastings, A. 1990. "Spatial heterogeneity and ecological models," in *Ecology* **71**, pp. 426-428.

[Kareiva 90] Kareiva, P. 1990. "Population dynamics in spatially complex environments: theory and data," in *Philosophical Transactions of the Royal Society*, London, B, **30**, pp. 175-190.

[Szymanski *et al.* 1993] Szymanski B.K., B. Maniatty & B. Sinharoy. 1993. "Simultaneous Parallel Reduction," in review for *Parallel Processing Letters*. Also available as technical report CS 92-31, Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY.

[Szymanski and Caraco 1993] Szymanski, B., & T. Caraco, "Spatial Analysis of Vector-Borne Disease: A Four Species Model," in press for *Evolutionary Ecology.*

[Turner 89] Turner, M.G. 1989. "Landscape ecology: the effect of pattern on process," in *Annual Review of Ecology & Systematics* **20**, pp. 171-198.

[Maniatty *et al.* 1993] Maniatty, W., B. K. Szymanski, and T. Caraco. "Implementation and Performance of Parallel Ecological Simulations," in *Proceedings IFIP WG10.3 International Conference on Applications in Parallel and Distributed Computing*, in press.