

Implementation and Performance of Parallel Ecological Simulations *

William Maniatty, Boleslaw Szymanski^a and Tom Caraco^b

^a Department of Computer Science,
Rensselaer Polytechnic Institute,
Troy, NY 12180-3590

^bDepartment of Biological Sciences
SUNY Albany,
Albany, NY 12192

The spatial and temporal aspects of population dynamics are pivotal to computational biology. We developed a spatially explicit model of epidemics that behaves like a large probabilistic cellular automaton. The cells of the automaton are discrete *sites* into which the habitat is partitioned. Probabilistic local state transitions are executed synchronously at all sites making the simulation suitable for parallel implementation on SIMD architectures. An interpretation of the simulation results requires computing global parameters of the habitat that are challenging to implement efficiently on SIMD machines. For example, pattern detection and measurements use sophisticated image processing algorithms. In this paper, we discuss the parallel implementation of the epidemiological model and analyze its performance. The achieved results indicate that the use of a massively parallel machine was necessary and efficient.

Keyword Codes: I.6.3; C.4; D.1.3

Keywords: Simulation and Modeling, Applications; Performance Measurement of Systems; Concurrent programming

1. Introduction

Computer simulation of population densities and associated spatial patterns in ecological models can be used to predict an ecosystem's future. The spatial context of ecological interactions is fundamental to understanding population dynamics, community stability, and biodiversity [3, 4]. However, the computational power necessary to deal with the volume of data and calculations in spatially explicit models can only be provided by supercomputers [11]. In this paper we discuss the implementation of spatially explicit models of epidemics on massively parallel computers. We designed and implemented two basic but ecologically significant models of epidemics: directly transmitted diseases, and vector-borne diseases. An example of a vector-borne disease is lizard malaria [7], where mosquitoes, not themselves affected by the pathogen, transmit the disease from infective (sick), to susceptible (healthy) individuals from two lizard species.

*Supported in part by ONR Grant N00014-93-1-0076 and an IBM Corp. Grant. The content of this entry does not necessarily reflect the position or policy of the U.S. Government—no official endorsements should be inferred or implied.

Many natural systems, including epidemics, can be characterized by a global state that is defined as a vector of local states. Correspondingly, in the discussed model [5], the composition of local transitions determines each global state transition. To simulate the spatial and temporal dynamics of a multi-species ecosystem, the model partitions its habitat into a grid of sites, such that each site has space for at most one host. Each site's state indicates the presence or absence of the species involved in a simulation. A probabilistic model describes a site's state transition as a function of the states of its neighboring sites. The *stencil* is the (typically) rectangular neighborhood surrounding a site (which may not be in the stencil's center). The ecological stencil's size and the affected site's location within it are based on the biotic and abiotic characteristics of the species and the habitat.

The implemented model described in this paper is highly data parallel and displays regular communication patterns; hence, it is suitable for SIMD architectures. The implementation runs on a MasPar MP-1. The implemented software is parameterized to enable easy experimentation with different properties of the species involved in the epidemics. The simulation results include the number of sites in each state at the end of simulation, the distribution of the species in the habitat, pathogen contagion, as well as size, frequency and fractal dimensions of species' spatial clusters.

The paper begins with a description of the model in Section 2. The computational challenges are addressed in Section 3. The performance results, justifying the use of a massively parallel SIMD machine, are presented in Section 4. The ecological results are interpreted in Section 5 and conclusions and future directions are discussed in Section 6.

2. The Mathematical Model

The formal definition of the probabilistic model on which the implementation is based is presented in [5]. Here, we briefly examine those features of the model that affect the structure of the implementation.

The model partitions a two-dimensional *habitat* into *sites*. Each site can support at most one *host* organism. There are two types of hosts: the ecologically dominant species are the *good hosts*, the inferior species are the *bad hosts*. *Parasites* are organisms *infesting* hosts. *Pathogens* are organisms *infecting* hosts. Parasites and pathogens can only live in the presence of a host. Each site has a *state* indicating the presence or absence of each organism type at that site. All sites are influenced by nearby sites forming the *ecological stencil*, which is assumed to be rectangular. An affected site resides 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 describe birth, death, recovery from disease, etc. Only ecologically significant states and transitions are considered (see Figure 1).

The implementation follows from the model. The simulation program iterates once per time step, computing stencil statistics and state transitions on each iteration. Statistical information is gathered at user specified intervals. The user also defines the length of the simulations. The current implementation maps each site to a different processor of the

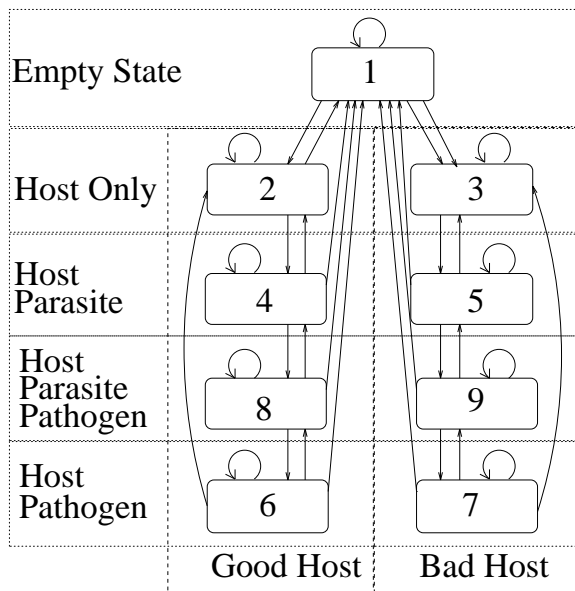


Figure 1. State Transition Diagrams

MasPar, with Xnet communication used to evaluate the current state of the stencil.

3. Computational Issues

The simulations ran on a MasPar MP-1, a SIMD mesh architecture. The MasPar's array of processing elements (PEs) is organized in a two-dimensional grid. Each PE can communicate directly with its eight nearest neighbors using *xnet* operations. There is also a global router which allows communication between any two arbitrary PEs but it imposes a 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 main algorithmic challenges were:

- *Computation of Transition Probabilities* - These probabilities are used in selecting the next state of each site. Computing them involves simultaneous reduction of an addition operator over ecological stencils, so this is very intensive but also highly parallel computation. In [9], we have presented algorithms for simultaneous reduction that take advantage of associativity and commutativity of the involved binary operator (in this case, an addition).
- *Fractal Dimension* - A measure of complexity of the spatial patterns arising in the habitat. The implementation of this measure is based on an algorithm developed by Cypher [2] for image component labeling on mesh connected SIMD machines.

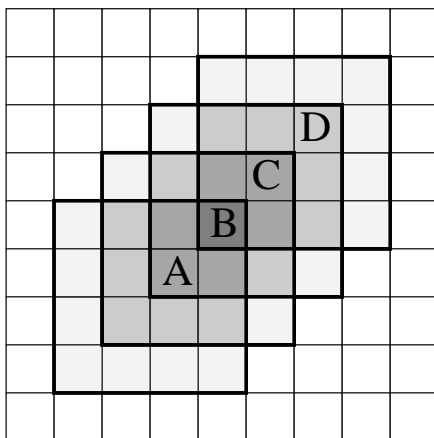


Figure 2. Overlapping Stencils for Sites A, B, C, D

This is the least parallelizable operation that involves a small fraction of the total computation.

- *Relative Patchiness* - A measure of ecological diversity, i.e., the average rate of change of the landscape along some direction. This measure is implemented using a parallel reduction algorithm.

3.1. Computation of Transition Probabilities

Each site's state transition probability can be expressed as a function of its current state, and the states of sites in the corresponding ecological stencil. Let n_1 and n_2 stand for the lengths of the sides of the ecological stencil, p_1 and p_2 be the offsets (measured from the lower left corner of the stencil) of the affected site, and $s_{i,j}$ be the current state of the site (i, j) . Then, the state transition probabilities of site (m, q) are defined as:

$$\sum_{i=m-p_1}^{m+n_1-p_1} \sum_{j=q-p_2}^{q+n_2-p_2} f(s_{m,q}, s_{i,j})$$

The above summation is evaluated for each site in the habitat and for each step of the simulation. Hence, it dominates the total execution time of the modeling. As seen in Figure 2, the ecological stencils of neighboring sites overlap. We refer to the problem of evaluating a reduction operator over overlapping regions as *simultaneous reduction*. It differs from the usually considered *parallel data reduction* in the following aspects:

- The regions where reduction is to be performed are of an arbitrary but consistent size (which may not necessarily be a power of 2). These regions correspond to the ecological stencil in the described application.
- The regions overlap.
- The result of the reduction has to be placed at a position which is arbitrary, but the same, for all the regions.

Hence, the preferred parallel reduction solution, *parallel prefix computation*, does not solve the simultaneous reduction problem. In [9], we presented algorithms for the simultaneous reduction problem that take advantage of associativity and commutativity of binary operators used in reduction. Additional improvement is possible if this operator also has an inverse. The efficiency of the proposed algorithms was measured on the MasPar MP-1 and reported in [6]. In the described implementation, one of the proposed algorithms is used; namely, the so called row-column algorithm that takes advantage of fast “xnet” MasPar communication. It runs in $O(\log n)$ operations and latencies, with $O(\sqrt{n})$ hops, where $n = n_1 * n_2$ is the stencil size. The code for this algorithm is parameterized by the stencil size at compile time.

Let M denote the set of states used in the model. Simultaneous reduction has to be executed $|M|$ times over the entire habitat, so complexity of this computation is $|M| \log n$ and therefore independent of the habitat size.

3.2. Fractal Dimension

Fractal dimension measures the complexity of the spatial patterns generated by the simulation. It is computed using an image component analysis technique presented in [10]. Fractal dimension is the regression of the log of the perimeter on the log of the area. Suppose there are n clusters. Let cluster c_i have area a_i , and perimeter p_i . Let $x_i = \ln a_i$, and $y_i = \ln p_i$, with means $\bar{x} = \sum_{i=1}^n x_i/n$ and $\bar{y} = \sum_{i=1}^n y_i/n$. The fractal dimension, β , can be then expressed as:

$$\beta = \frac{\sum_{i=1}^n (x_i y_i) - n \bar{x} \bar{y}}{\sum_{i=1}^n x_i^2 - n \bar{x}^2}$$

The computation consists of three phases:

- **Image Component Labeling.** This is a well known problem in image processing applications and many algorithms have been proposed as its solution. The current implementation uses an algorithm developed by Cypher [2] for mesh connected SIMD machines with time complexity $O(\sqrt{N})$, where N denotes the number of pixels in the image. In the discussed model, pixels set on corresponds to sites which have states indicating the presence of the pathogen.
- **Computing the Area and Perimeter for each Image Component.** A site is on the *perimeter* of a cluster if it is on the boundary of the habitat, i.e., if any of its neighbors has a different state. The algorithm to compute the area and perimeter of each cluster uses parallel sorting to get the records in the same cluster into contiguous strips. Then it calculates the cluster area and perimeter by executing a segmented scan over contiguous (but disjoint) strips.

The efficiency of sorting on a SIMD mesh of N processors, each storing one element, is $O(\sqrt{N})$ steps [8]. The segmented scan requires $O(\sqrt{N})$ communications and $O(\log N)$ binary operations.

- **Calculating the fractal dimension using parallel data reduction.** There is a constant number of arithmetic operations and a constant number of parallel data reduction steps involved in computing the fractal dimension. The cost of doing the calculations on a mesh of N processors each storing one pixel is $O(\log N)$.

Because the cost of the first two phases is $O(\sqrt{N})$, the total cost of the algorithm is $O(\sqrt{N})$.

3.3. Relative Patchiness

Relative patchiness measures the ecological diversity for a given state. Let M represent the set of states used in the model. Let $R_p(m)$ denote the relative patchiness for state $m \in M$. Let B be the number of edges from the given site to the adjacent sites in the model (adjacent boundary regions count as “empty” sites). The MasPar MP-1 has mesh interconnection with $B = 8$. Let N_m denote the sites that are in state m and let dir represent a direction assigned to the boundary between sites in the MasPar network topology (such as North, East, West, South). Let D_{dir} be the measure of the dissimilarity of the states on the boundary dir between sites adjacent to sites in N_m , that is $D_{dir}(m) = k$ if k sites adjacent to sites in N_m are not in state m .

For each state, m , the relative patchiness of that state, $R_p(m)$, is computed by performing a parallel computation followed by a data reduction. There is a set of edges connecting sites in N_m to adjacent sites and the spatial boundaries, denoted by $adj(N_m)$. The number of edges per site is B , so $|adj(N_m)| = B|N_m|$. If $N_m = \emptyset$, then $R_p(m) = 0$, otherwise: $R_p(m) = \frac{\sum_{dir=1}^B D_{dir}(m)}{B|N_m|}$. The complexity of this computation is $O(|M| \log N)$ because the data reduction is performed $O(|M|)$ times with $O(\log N)$ computational cost per invocation.

4. Achieved Performance

In this section, two important issues connected with the choice of architecture are discussed. The first issue addresses the question of use of a massively parallel machine versus a workstation. Hence, we compare the performance of the implementation on the MasPar with the implementation on a DECstation. The second issue concerns the parallelism available in the application. The achieved high utilization of 2048 processors of the MasPar together with the sustained performance equivalent to the several dozens of DECstations allows us to conclude that the use of a massively parallel machine was fully justified for this implementation.

The implemented model was executed thirty-six times using a 32 bit integer and a 64 bit floating point arithmetic on both 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 sizes 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).

In comparing performance of the MasPar versus the DECstation, processing speed disparity of floating point and integer arithmetic on both processors must be considered. On the MasPar, the floating point arithmetic is about 46.4 times slower than integer

arithmetic; whereas slow-down is just 3.97 for the DECstation. Therefore, this paper uses the integer-equivalent speed of the MasPar; that is a speed that would be achieved if each floating point operation were replaced by 46.4 integer operations on the MasPar. The total number of operations performed by the MasPar was estimated by timing a sequential implementation on the DECstation and using its advertised speed for floating point and integer arithmetics. The results are plotted in Figure 3 as a function of the number of sites and a function of the stencil size. The maximum integer-equivalent speed was obtained for the 13×13 stencil with 2048 processors and it was slightly above 10^9 instructions per second (about 1 GIPS). There are two reasons why the integer-equivalent

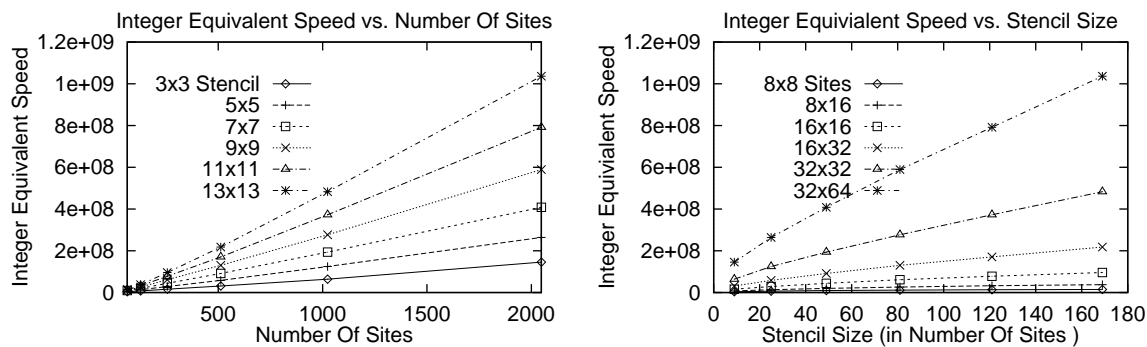


Figure 3. Integer-Equivalent Speed of the MasPar MP-1

speed curves are not strictly linear. Firstly, the conditional flow of control on SIMD architectures causes processors not taking a branch to wait for branch completion. The nine-state ecological model results in a nine-way switch in computing the state transition probability at each processor. Secondly, the sublinear complexity of the simultaneous reduction algorithm with respect to stencil area (i.e., $O(\log N)$ operation and latency costs, $O(\sqrt{N})$ communication distance) on the MasPar differs from the linear complexity of the comparable sequential algorithm for the DECstation.

The MasPar’s advertised speed and achieved integer-equivalent speed were used to derive the utilization of the MasPar’s processors. It is plotted as a function of the number of sites and as a function of the stencil size in Figure 4. For the largest stencil and the maximum number of sites, simulation sustained (high for SIMD architectures) a utilization of 32.6%.

5. Ecological Results

Each simulation started with the same ecosystem. Hosts without parasite or pathogen occupy most sites (90.2%). A thin barrier that cannot be occupied splits the environment into left and right halves. The two “islands” are ecological “stepping stones”² for the parasite and pathogen, as seen in Figure 5. A few hosts carrying the parasite, pathogen,

²This is an abstraction of the more realistic case of fox rabies [12] moving from continental Europe to Great Britain across the English Channel via a tunnel.

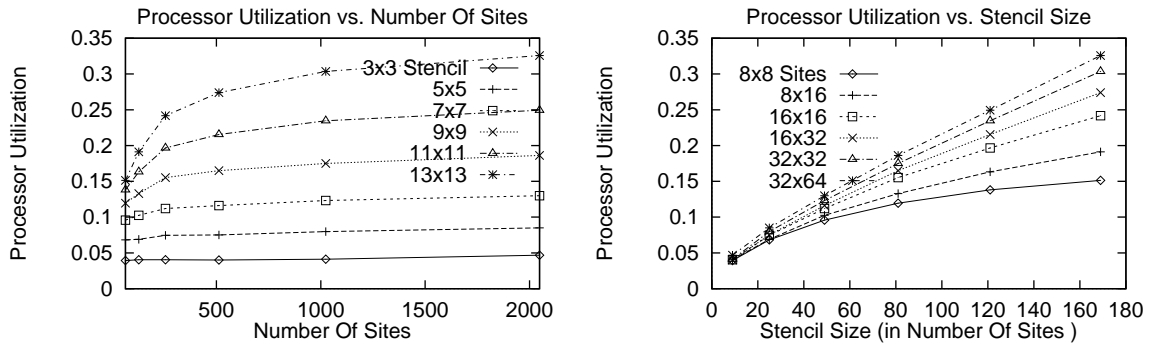


Figure 4. Processor Utilization of the MasPar MP-1

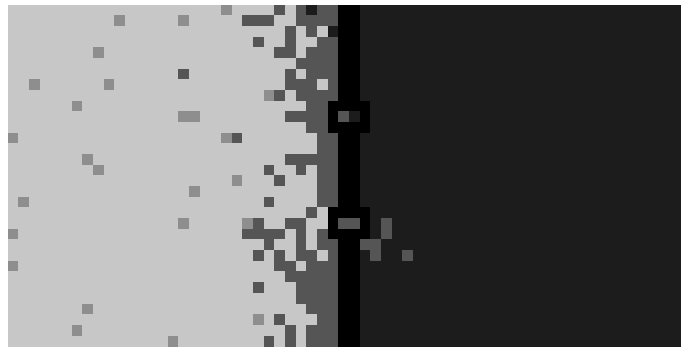


Figure 5. Tomography of epidemic about to cross the river

or both (2% of each state) are at the far left; hence, the epidemic spreads from left to right.

Two simulation results are discussed here:

1. The overall frequency of diseased hosts at each sampling points.
2. The overall biodiversity of the environment at each sampling point.

The extent of infection is an epidemic's most fundamental ecological attribute. The frequency of diseased hosts describes the epidemic's temporal evolution. Diversity increases with the number of distinct ecological states and strongly influences extinction trends. To calculate diversity, the entropy H is used as defined in Table 5. The table also lists the parameters controlling the runs. The simulations were run for 100 time steps, with sampling every 10^{th} step. The following patterns emerged:

1. $\beta = 0.001$ and $\alpha = 0.002$: Low susceptibility; neither parasite nor pathogen spread. Entries are means across μ_d values 0.001, 0.01, 0.1 and 0.5.
2. $\beta = 0.0001$ and $\alpha = 0.015$: Low susceptibility; the parasite is stopped at the

Variable	Definition
α	Probability of <i>Parasite</i> attack
β	<i>Host</i> pathogen susceptibility
H	$H = -\sum_{i=1}^S p_i \ln(p_i)$
μ_d	<i>Parasite</i> mortality rate
p_i	the fraction of sites with state i
S	The number of distinct states in the environment

Table 1
Notations of model parameters and measurements

barrier. The pathogen does not spread. Entries are means for $\mu_d = 0.001, 0.01$ and 0.1 . The parasite was extinct by the 20^{th} time step for $\mu_d = 0.5$.

3. $\beta = 0.001$ and $\alpha = 0.125$: Low susceptibility; the parasite spans the environment. Disease frequency remains low due to low susceptibility. Entries are means for $\mu_d = 0.001, 0.01$ and 0.1 . The decline in biodiversity at the sampling time 40 shows a decline in hosts without the parasite but with the pathogen.
4. $\beta = 0.5$ and $\alpha = 0.002$: High susceptibility; spread of the disease is constrained by the parasite's extinction or slow growth. For $\mu_d = 0.1$ and 0.5 , parasite extinction occurred before the simulation was through half of the time steps.
5. $\beta = 0.5$ and $\alpha = 0.015$: High susceptibility; the parasite, and thus the disease, is stopped at the barrier. For $\mu_d = 0.5$, the parasite became extinct by the 20^{th} time step. Entries are means for $\mu_d = 0.001, 0.01$ and 0.1 .
6. $\beta = 0.5$ and $\alpha = 0.5$: High susceptibility; the disease occurs throughout the environment. Entries are means for $\mu_d = 0.001, 0.01, 0.1$ and 0.5 . The decline in biodiversity after time step 20 occurs as hosts quickly contract the parasite and pathogen.

Graphs of average frequency of infected hosts over time and the mean biodiversity over time for the patterns listed above are shown in Figure 6.

The epidemiological results confirm validity of the model. The model enables analysis of the full range of epidemic conditions by manipulating only a few, easily understood, probabilistic parameters. Hence, significant ecological questions can be explored in a computationally convenient manner.

6. Conclusion

The initial results have been both ecologically and computationally encouraging. The developed novel parallel algorithms for the simultaneous reduction operation performed well and enabled us to simulate ecosystems with larger ecological stencils than those considered to date. However, much work remains to be done. Future enhancements of the

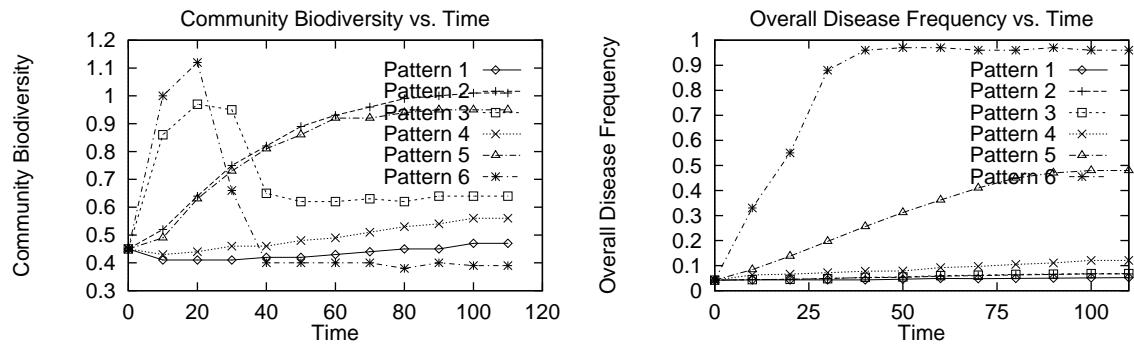


Figure 6. Ecological Results

implementation include: automated parallel code generation, virtualization of processors, interactive graphical user interface development, and tools to compare simulation results with experimental results. Future simulations will be based on more complex interspecies' interactions than most analyses of spatial patterns in ecological dynamics offer. We also want to port the current version to the more modern MasPar MP-2 and hope that on a 16K processor MP-2, large models of epidemics will sustain above 20 GIPS.

REFERENCES

1. Bartholomew, D.J. 1983. Stochastic Models for Social Processes, 3rd Edn. Wiley: New York.
2. Cypher, R. & Sanz, J. L. 1990. Algorithms for Image Component Labeling on SIMD Mesh-Connected Computers. IEEE Transactions on Computers, 39:277-281.
3. Hastings, A. 1990. Spatial heterogeneity and ecological models. Ecology 71:426-428.
4. Kareiva, P. 1990. Population dynamics in spatially complex environments: theory and data. Philosophical Transactions of the Royal Society, London, B, 30:175-190.
5. Caraco, T. & Szymanski, B. 1994. Spatial Analysis of Vector-Borne Disease: A Four Species Model. Evolutionary Ecology, in press.
6. Maniatty, B., Sinharoy, B., & Szymanski, B.K. 1993. Efficiency of Data Alignment on MasPar. SIGPLAN Notices, 28, 1:48-51.
7. Schall, J.J. 1983. Lizard malaria: parasite-host ecology. pp 84-100, in *Lizard Ecology: Studies of a Model Organism* (ed by R.B. Huey, E.R. Pianka & T.W. Schoener). Harvard University Press:Cambridge.
8. Scherson, I. D., Sen, S. 1989. Parallel Sorting in two-dimensional VLSI models of computation. IEEE Trans. on Computers, 7, 2:238-249.
9. Szymanski B.K., Maniatty, B., & Sinharoy, B. 1993. Simultaneous Parallel Reduction. Rensselaer Polytechnic Institute. Tech. Rep. CS 92-31, also submitted to Parallel Programming Letters.
10. Sugihara, G., & May, R. M., 1990. Applications of Fractals in Ecology, Trends in Ecology and Evolution, New York, Volume 5, Number 3: 79-86.
11. Turner, M.G. 1989. Landscape ecology: the effect of pattern on process. Annual Review of Ecology & Systematics 20:171-198.
12. Yachi, S., Kawasaki, K., Shigesada N., & Teramoto, E. 1989. Spatial patterns of propagating waves of fox rabies. Forma 4:3-12.