Parallel Neighbourhood Modeling: Research Summary [*†]

D. Hutchinson, L. Küttner, M. Lanthier, A. Maheshwari,
D. Nussbaum, D. Roytenberg, J.-R. Sack
Carleton University, School of Computer Science

1 Motivation and Applications

In recent years the demand on Geographical Information System (GIS) technology from application areas has sharply increased. In particular, with the EOS (Earth Observation System), terabytes of spatial information will be collected daily. The increase in speed in sequential computing cannot keep up with the demand placed by many systems handling spatial data. Therefore a need for parallel computing in this area is widely recognized.

Raster processing, such as neighbourhood modeling or computing complex visibility information is, in many cases, a time consuming task. At a processing rate of 1000 cells/sec., processing a large raster of size 6000x6000 cells takes 10 hours. The problem becomes more acute when cellular automata modeling is considered. The time to process even a small number of generations, say 200, on a 6000x6000 cell raster takes over 83 days. Related problems include spatial feature extractions where "*numerous investigators report lengthy computation time*" [9]. Many users who execute such time intensive or time sensitive tasks are forced to either reduce the raster size, possibly sacrificing resolution and accuracy, or choose a simpler model which may sacrifice strength, scope and validity. Raster processing/modeling operations are employed in many application areas such as remote sensing, urban planning, and simulating biological/ medical/hydrological processes.

We are in the third year of a project funded in part by industry and the Canadian Government (NSERC) to develop a parallel GIS. The funding for the project is approximately $4,000,000 (CAN). Here, we describe our first major milestone, the design and features of a parallel system for NEighbourhood MOdeling (called NEMO) of raster data. The mandate of our system is to provide users with a fast engine for processing a variety of time-consuming tasks for raster based data. The system is designed to be platform independent and is currently implemented on the AVX series II parallel computer manufactured by Alex Informatique, a Canadian supercomputer manufacturer.

The Alex AVX Series II computer running under the Trollius operating system, is a flexible distributed memory MIMD parallel machine which can be reconfigured into a variety of standard and non-standard topologies. The AVX has 64 standard nodes and several entry (external access) nodes. A standard node consists of two processors, one Intel i860 (used for computations) with between 32 and 64MB of DRAM, and one Inmos T805 transputer (used primarily for communication) with between 8 and 16MB of DRAM.

2 System Overview

2.1 Design Issues

While parallel computers are becoming widely available, the development and implementation of (complex) algorithmic techniques to exploit the features of parallel architectures is very challenging.

Faust et. al. [6] state that "*the real issues in computing in the next decade involve innovations that will allow relatively unsophisticated users to access the power of the computer hardware, without having to become experts in programming and computer operating systems. The tools for GIS .... should become easier to use ... and at the same time be able to take advantage of the new advances in hardware and software technology.*"

[*]This R&D project is supported by the Natural Sciences and Engineering Research Council of Canada and ALMERCO Inc.

[†]The full version of this paper is available from the authors upon request.

This motivates our primary design principle of *Transparent Parallelism*. In general, parallelism can lead to many problems pertaining to timing, communication, synchronization, load balancing etc.. Transparent parallelism relieves the user of the burden of these parallel aspects of a model or application. NEMO provides transparent parallelism to allow the user to develop parallel spatial data processing applications in an environment which is free from these parallel issues. As a result, NEMO addresses the following parallel issues:

- **Architecture and Machine Independence -** ensuring that NEMO and thus user applications are easily portable and that it configures itself to the parallel environment in the best possible way.
- **Communication Bottlenecks -** avoiding interprocessor communication bottlenecks; especially, I/O communication between the parallel machine and the host. Our AVX systems have a considerable amount of internal memory and thus files can often be stored in the machines for subsequent parallel applications.
- **Data Visualization -** since NEMO operates on a distributed memory MIMD machine data visualization and display is not only an I/O issue, but also a display synchronization issue.
- **Causality Errors -** one of the major issues in parallel simulation [7]; especially noticeable when intermediate results are displayed. Due to asynchronous processing the display of intermediate results could mislead users temporarily until all data are processed.
- **Load Balancing -** minimizing the amount of processor idle time. Several data partitioning schemes have been designed, implemented and compared.
- **Data Coherence -** ensuring that data is consistent during simultaneous read and write operations.

NEMO is capable of processing three types of time consuming raster neighbourhood models (described in Section 2.2): *Cellular Automata*, *Propagation*, and *Neighbourhood Analysis*. NEMO is not tailored to any specific application. Rather, it is designed to support applications falling under the umbrella of the above raster neighbourhood modeling families. NEMO achieves this flexibility by having five components: three application drivers (Section 2.2), namely, Cellular Automata Driver, Propagation Driver and Neighbourhood Analysis Driver components, and two client-server components (Section 2.3), namely, Display Manager and Raster Database Manager (see Figure 1).

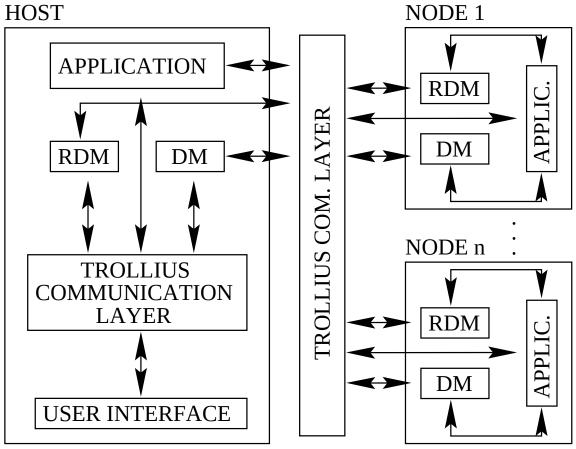

Figure 1: NEMO system overview.

2.2 Application Drivers

Raster neighbourhood modeling is performed on raster images: A neighbourhood modeling function (such as smoothing) traverses all cells of a raster and performs a local calculation based on attributes stored in the cell and its local neighbouring cells. Although the three application drivers differ in functionality, the principles under which they operate are similar. The drivers accept as input: one or more raster images, a user-defined neighbourhood function, and a local neighbourhood definition. The application drivers, using the client-server components, load the data into the parallel computer (tiling it as necessary), activate a user neighbourhood function on each cell of the raster, output the results to the database and if necessary display the resulting data. The main difference between the three drivers is the order in which the cells are processed. The Cellular Automata and Neighbourhood Analysis drivers process the cells in an arbitrary order whereas the Propagation driver processes the cells in a user-specified order (defined at run time).

The following notation is used. Let $g_{i,j}$ be a cell in an $n \times m$ grid corresponding to the location of a point in a geographic terrain (each cell stores one or more attributes, e.g., urban, rural). Cell $g_{i,j}$ may be time variant (denoted by $g_{i,j}(t)$ for time t). We define the neighbourhood $N(g_{i,j})$ of a cell $g_{i,j}$ to be a set of cells in the raster associated with $g_{i,j}$. Typically, $N(g_{i,j})$ includes all of the cells which are within a specified distance from $g_{i,j}$.

Cellular Automata Driver (CD) - Cellular automata were introduced by Codd [5] (made famous through Conway's "Game of Life" [8]) as an elegant mathematical model for a class of processes operating in discrete time and discrete space. Tobler [14] introduces the notion of *cellular geography* in which he classifies different cellular automata models covering a wide range of applications and generalizations.

GIS modeling/simulation using cellular automata has been described for forest fires [2], forest infestation [10], and earthquakes [1]. Itami [11] has studied cellular automata for residential site selection (using Tomlin's Map Analysis package). Brinch Hansen [3] describes a model program for parallel execution of cellular automata adapted to a (very simple) forest fire.

The CD is intended for general cellular automata GIS applications. It processes a generation by applying a neighbourhood function F to the whole raster at time t. It iterates this processing for the desired number of generations. It supports all of the models proposed by Tobler [14]; two of which are the *historic* and *multivariate* models. For the historic model, the function applied to each cell operates on the local neighbourhood $N(g_{i,j}(t))$ of cell $g_{i,j}$ at time t as well as the values that cell $g_{i,j}$ took during the last k generations. This model is based only on local information but also takes into account the history of the cell. In the multivariate model, the values that a cell may take depend upon several attributes present at the cell at time t.

Propagation Driver (PD) - In contrast to the CD which operates on the whole image at each generation, the PD is designed to process applications which operate on the active border principle in which only a subset of cells need be processed at a given time. For example, in a forest fire only the areas near the fire front (i.e., the active border) are of interest and must be processed. All other areas do not require processing at this moment. In the propagation model, instead of modifying the attributes of the center cell ($g_{i,j}$) in the local neighbourhood (as in the CD), the center cell ($g_{i,j}$) in the PD may affect the attributes of its neighbours. Propagation modeling can be viewed as a special case of cellular automata. However, since only a portion of the raster is active at any given time (the active border), it is more efficient, e.g., in producing cost surfaces or in calculating propagation functions such as noise propagation [13].

Neighbourhood Analysis Driver (ND) - The ND provides a framework for executing one, or a series of distinct, single pass neighbourhood functions on input rasters. Three general families of neighbourhood operations are identified in the ND design, differing primarily in the amount of intra-raster communication required:

- Point-wise operations - cell by cell combination of rasters (e.g. map overlay) requires no communication between processors.
- Local neighbourhood operations - transformation of each cell according to some function of its neighboring cells' values requires limited communication, typically between neighbouring processors (e.g. image processing operations such as noise reduction and edge enhancement).
- Global operations - computations involving an arbitrary number of cells of a raster require extensive communication of data. Examples include calculation of aggregate statistics for a raster (e.g. colour histogram).

The global operations provide the most interesting challenges to transparent parallelism. We are currently exploring possible augmentation of our model by examining the operations proposed by Tomlin's Map Algebra cartographic language [15].

2.3 Client-Server Components

NEMO contains two client-server components: Display Manager and Raster Database Manager. These components take care of I/O communication aspects between the host computer and the internal nodes. To retrieve/output data or to display a raster, each node operates, as though it was a sequential machine and not a node in a parallel environment. Each component has two subcomponents: a communication component allowing the parallel nodes to transparently interact with the host and a set of library functions executing the requests issued by the nodes.

Display Manager (DM) - The DM controls all the aspects of displaying raster images and application results. It provides a flexible environment for data display while freeing the user from having to handle any of the issues that relate to data display (e.g., scaling, clipping and geo-referencing). It provides dynamic viewing of application data by allowing continuous updating of raster images on the screen. The DM also provides a synchronization mechanism so that cellular automata applications can properly gather data from all nodes and display whole generations at a time. A typical application requires a large amount of data to be displayed from the nodes. The DM provides re-sampling (clipping and zooming) on each node in order to reduce the load on the system due to out-of-range data attempting to be displayed.

Raster Database Manager (RDM) - The RDM manages the raster database by servicing all I/O re-

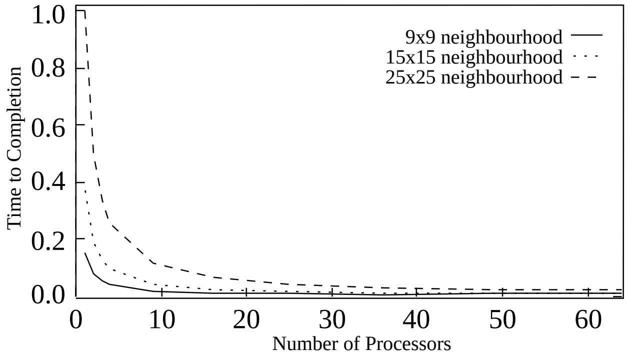

Figure 2: Timing results for a Neighbourhood Analysis application.

quests. The RDM provides a client server environment where the nodes are the clients and the RDM is the server (it controls and manipulates a central database which is stored on the host system on one or more disks). The nature of NEMO has allowed us to choose a "shared external memory" model over "distributed memory model" in order to avoid data and cache coherence problems and to avoid common deadlocks.

The RDM provides a flexible data access scheme. It permits applications to access the data logically without having to handle the physical representation. The RDM provides the following functionality: **Logical Tiling -** applications can define the tile size that they wish to access; **Tile Locking -** allows applications to lock and unlock tiles; **Hardware Independence -** supports applications on different hardware; **Data Compression -** provides several compression schemes; **Data Representations -** allows applications to request the data in several representations (e.g. run-length or linear quad tree).

3 Current Status

The design and requirement specification phase of NEMO is complete. All documents and a prototype implementation have been delivered to our industrial partner who will look after commercialization. We have also developed a simulator for the AVX [12].

An image processing demonstration application has been constructed using the ND, and is running on both the AVX and a network of workstations. Figure 2 shows relative timings for various numbers of processors on the AVX executing a combination of image processing operations.

Further parallel GIS applications are being developed either for NEMO or as a stand-alone system. A parallel GIS application realistically modeling the emission of gases effecting the ozone layer in southern Ontario has successfully been completed. The speed-up over the sequential program is dramatic. In addition, forest fire modeling [4] and ice-tracking applications are being developed. We have received substantial funding to enter phase 2 of our R&D: the design and development of a prototype implementation of a vector-based GIS. (We are also investigating the design and development of a Parallel Spatial Modeling (integrated) Environment (PSME)).

References

[1] P. Bak, C. Tang, *Earthquakes as a self-organized critical phenomenon*, J. Geo phys. Res. 94, 1989, pp. 15635-15637.

[2] P. Bak, K. Chen, *A forest-fire model and some thoughts on turbulence*, Phys. Lett. A 147(5-6), 1990, 297-299.

[3] P. Brinch Hansen, *Parallel Cellular automata: A model for computational science*, Concurrency: Practice and Experience 5(5), 1993, pp. 425-448.

[4] K. Clarke, J.A. Brass and P.J. Riggan *A Cellular Automaton Model of Wildfire Propagation and Extinction*, Photogrammetric Engineering and Remote Sensing, Vol. 60, No 11, 1994, pp. 1355-1367.

[5] E.F. Codd, *Cellular Automata*, Academic Press, New York, 1968.

[6] N.L. Faust, W.H. Anderson, and J.L. Star, *Geographic Information Systems and Remote Sensing Future Computing Environment*, Photogrammetric Engineering & Remote Sensing 57(6), 1991, pp. 655-668.

[7] R. M. Fujimoto, *Parallel Discrete Event Simulation*, Communication of the ACM, Vol. 33, No. 10, 1990, pp. 30-53.

[8] M. Gardner, *The fantastic combinations of John Conway's solitaire game "Life"*, Sci. Am. 223(10), 1970, pp. 120-123.

[9] B.L. Hickmanm M.P. Bishop, and M.V. Rescigno, *Advanced Computational Methods for Spatial Information Extraction*, Computers & Geosciences 21(1), 1995, pp. 153-173.

[10] F.C. Hoppensteadt, Mathematical aspects of population biology, in L.A. Steen, ed., Mathematics Today: Twelve Informal Assays, Springer Verlag, New York, 1978, pp. 297-320.

[11] R.M. Itami, *Cellular Automatons as a framework for dynamic simulations in Geographic Information Systems*, Proc. GIS/LISI'88, Vol. 2, 1988, pp. 590-597.

[12] D. Roytenberg, J.-R. Sack, *A Simulator for the Alex AVX Series II Parallel Computer*, to appear in HPCS '96.

[13] J. Strobel, *Conceptual Modeling of Spatial Diffusion Problems - Lessons from Noise Propagation Analysis*, GISDATA Specialist Meeting in GIS and Spatial Models, Stockholm, June 14-18, 1995.

[14] W.R. Tobler, *Philosophy in Geography*, edited by S. Gale and G. Olssen Issues for Design and Implementation, Cellular Geography, 1979 .

[15] C.D. Tomlin, *Geographic Information Systems and Cartographic Modeling*, Prentice Hall, 1990.