TOWARDS ECOSYSTEM MODELING ON COMPUTING GRIDS

Dali Wang¹, Eric Carr¹, Louis J. Gross¹, and Michael W. Berry²

¹ The Institute for Environmental Modeling, 569 Dabney Hall, University of Tennessee, Knoxville, TN 37996 [wang, carr, gross]@tie.m.utk.edu

² Department of Computer Science 203 Claxton Complex, University of Tennessee, Knoxville, TN 37996 berry@cs.utk.edu

Abstract. We present a generic software architecture for ecosystem modeling on computing grids. We define and develop standards for software implementation of spatially explicit (landscape-based) ecosystem models. Our strategic goal is to build an extensible software platform that increases interoperability and productivity in ecosystem modeling, and to make it adaptable for grid computing. The Across Trophic Level System Simulation (ATLSS) is presented as a prototypical ecosystem framework, supporting a diverse range of ecological models on a heterogeneous grid using single software architecture.

Key words and phrases: ecosystem modeling, computational ecology, ecological multimodeling, grid computing, distributed database, parallel computation, software architecture, computational framework.

1. Introduction

Ecosystem modeling presents a variety of challenges. Much of classical ecological theory originates from very simple differential equations in which a single variable represents population densities; solutions of these are analyzed mathematically and then may be compared to abundance estimates from field or lab observations. Although these models have had great influence on ecological theory, their aggregated form is particularly difficult to relate to observational biology. Their application to complex natural systems with spatially and temporally varying environmental factors typically leads to models that are not analytical tractable and must be investigated numerically. Recently, increasing emphasis has been placed on integrated multi-component ecosystem models. These involve complex interactions between some or all of the trophic layers (feeding levels), resulting in models linking multiple components which may be modeled using several different mathematical approaches. These efforts in ecosystem modeling imply that monolithic software development within a traditional computing framework is hindering further sustained innovation in complex highly-integrated model simulation. Herein, we present a simple conceptual model for ecosystem modeling, and use the Across Trophic Level System Simulation (ATLSS) [1] as an example to explain the key design considerations and efforts for such modeling on a computing grid.

2. A Conceptual Model for Ecosystem Modeling

Ecosystem modeling begins with the design of a conceptual model. The conceptual model aggregates our knowledge of the target system and requires selection of components and processes judged essential for the processes under study in a given spatial-temporal context. As a simplified example of a conceptual model, consider a
linear food chain, a series of organisms in an energetic hierarchy based on feeding (trophic) relationships. Different modeling approaches might be used for organisms at different trophic positions, based upon the differing spatial and temporal scales at which the organism operates, and linking these leads to a multimodel [2]. Here, we briefly describe three different modeling approaches applied in ecological multimodeling – individual-based models (IBM), structured models (SM) and compartment models (CM).

Individual-based models are simulations which can indicate the global consequences of local interactions of members of a population [3]. These models, also called agent-based models, typically consist of an environment in which the interactions occur and some number of individuals defined in terms of their behaviors (procedural rules) and characteristic parameters. In an individual-based model, the characteristics of each individual are tracked through time and space. Structured models, on the other hand, average certain characteristics of the population and attempt to simulate changes in these averaged characteristics for the whole population. In structured models, populations are not modeled completely by following every individual, but rather according to some set of structured classes. The population’s structure is determined by a discrete set of age or size classes, and often framed as a matrix model, as is typical in human demography. The size of each structured class, along with the time step used in the model, influences the values of the model parameters and affects the rates of transition between the classes. Compartment models are typically used to describe the kinetics of several highly aggregated components in an ecosystem, expressed as a system of difference or differential equations. These may be used for entire tropic levels (producers, consumers, etc.), unstructured populations, or components such as nutrients or energy, in which case the flow of nutrients/energy across the different trophic levels of the ecosystem is followed.

3. Multimodeling: New Approaches to Integrated Ecosystem Modeling

In application, differing levels of detail are required in ecological models for organisms at different trophic levels. At a lower trophic level, more emphasis is placed on the kinetics of the ecosystem’s energy flow, nutrient flow, or pollutant transport within the food web. At a higher level, especially for endangered species with small population sizes, one may need to monitor and simulate the basic behaviors of each individual. In order to better address the complexity of ecosystem modeling, a new approach called ecological multimodeling has been proposed by Gross and DeAngelis [2] in the course of developing Across Trophic Level System Simulations (ATLSS) for South Florida (See Figure 1). ATLSS is an ecosystem modeling package designed to assess the effects on key biota of alternative water management plans for the regulation of water flow across the Everglades landscape. The immediate objective of ATLSS is to provide a quantitative modeling package to assist various stakeholders in assessing the biotic impacts of alternative future scenarios for restoration of the natural systems in South Florida. The long term goals are to aid in understanding how the biotic communities of South Florida are linked to various physical driving influences, particularly hydrology, and to provide a predictive tool for both scientific research and ecosystem management.

The hierarchical spatial pattern shown in Figure 1 for the dynamic components of ATLSS arises from the association of different model types with different species or groups of species in the ecosystem. The compartment models deal with variables representing spatially localized biota, mainly the biomasses of lower trophic level organisms, such as algae. These biomass variables only interact locally. Therefore, one can represent these
variables across a landscape by means of many local uncoupled spatial unit cell models. Here the cell size is chosen small enough to represent a tract relatively homogeneous in substrate and elevation, the extent of which might be several hundreds of meters in a relatively fat landscape such as the Everglades. The age- and size-structured population and community models represent intermediate trophic levels, such as fish, macroinvertebrates, and small non-flying vertebrates. These may undergo short-distance movements in response to changes in water levels. Their spatial domains of interaction are larger, on the order of up to a square kilometer, thus potentially encompassing many smaller unit cells, coupled to allow population movements. Individual-based models are employed to represent populations of top predators or other large-bodied species, such as wading birds and panthers. Individuals of these species may move over large areas, with movements over short periods of time spanning areas of thousands of spatial unit cells. These individual-based models are rule-based approaches which can track the growth, movement, and reproduction of many thousands of individuals across the landscape. Adequate description of a single individual can easily require a number of variables detailing its current age, size, location, physiological status, and other information pertinent to its future actions. Associated with the spatial extent of the different model components are temporal scales. The vertical axis of Figure 1 shows the range in number of state variables or level of aggregation necessary to describe the system.

4. Computing Grid

The Grid has emerged as one of the most important new developments in building the infrastructure for computational science. By integrating networking, communication, computation and information, the Grid provides a virtual platform for computation and data management, just as the Internet provides a virtual platform for access to information. Using the Grid, users can access remote computers, employing networked resources for the computational challenges encountered in large-scale ecosystem modeling. For example, in ATLSS, for any single hydrological scenario, a single ecological model easily consumes hours of elapsed CPU time on a high performance SMP (symmetric multiprocessor) comprised of 14-CPUs [4].
An example computing grid is the Scalable Intracampus Research Grid (SInRG) [5]. The SInRG project, supported by the National Science Foundation, deploys a research infrastructure on the University of Tennessee, Knoxville campus that mirrors the underlying technologies and the interdisciplinary research collaborations that are characteristic of the emerging national technology grid. SInRG’s primary purpose is to provide a technological and organizational microcosm in which key research challenges underlying grid-based computing can be attacked with better communication and control than wide-area environments usually allow. Knowledge acquired in this smaller environment can help improving the national grid infrastructure. The skeleton of SInRG is illustrated in Figure 2.

The primary building block of the SInRG architecture is the Grid Service Cluster (GSC). A GSC is an ensemble of hardware and software that is constructed and administered by a single workgroup, but is also optimized to make its resources easily available for use by the grid-enabled applications of that group and others. As shown in Figure 2, first, each GSC is built around an advanced dataswitch (at least 1Gb/s on each link) in order to provide a high level of Quality of Service (QoS) within the cluster itself. Second, each GSC contains a large data storage unit attached directly to its switch in order to facilitate (for both local and remote SInRG users) the localization of remote data for efficient processing within the GSC. In Figure 2, the GSCs are typically commodity clusters, but different SInRG research projects require customized GSCs, such as an SMP or a set of adaptive computing devices. In all cases, however, both the typical and the specialized resources of a given GSC will, under appropriate conditions, be as available across the network to the other users of SInRG as they are to the GSC’s owners. Beside the advanced hardware features, the availability throughout all GSC’s of a variety of supported middleware (such as NetSolve [6] and IBP [7]) that unifies all hardware and
network infrastructure into a computational grid, distinguishes SInRG from a general-purpose networked cluster.

5. Software Architecture for Ecosystem Modeling

In this section, we illustrate the general multi-layered architecture for ecosystem modeling (illustrated in Figure 3), including an ecological modeling information processing layer, a components layer for ecological models, and an information analysis and data representation layer.

![Software Architecture for Ecosystem Modeling](在这里插入图像)

**Figure 3:** Software Architecture for Ecosystem Modeling

In Figure 4, we see an example of spatial information processing in ATLSS, demonstrating how high-resolution hydrology data can be used to model the dynamics of water management and ecosystems.

![Spatial Information Processing](在这里插入图像)

**Figure 4:** An Example of Spatial Information Processing in ATLSS
5.1 Ecological Modeling Information Processing Layer

The main functionalities of this layer are i) to provide a uniform structure to integrate the spatial information and information about physical data (such as hydrology) from external models; ii) to generate computational domains (meshes) for different ecological models; and iii) to determine parameter estimates for ecological models. Figure 4 shows an example of information processing in ATLSS – the procedure to generate a high resolution hydrology [8]. Hydrological data is a required component for all ATLSS models and is provided by the South Florida Water Management District (SFWMD) at a resolution of 2-miles by 2-miles. In order to simulate the dynamics of many animal populations in South Florida, higher resolution hydrological data (e.g., at 100 meters) are required. Therefore, a high-resolution topography map is created (at 30 meters) using satellite images of vegetation and information about the hydrologic ranges at which each type of vegetation can exist. From this, a high resolution hydrology map is developed by redistributing the volume of water within each 2-mile by 2-mile area.

5.2 Ecological Models Components Layer

This layer contains the different kinds of ecological models. Figure 5 shows some of the possible models in ATLSS. Based upon the ecological relationships, several models might be linked together; for example, the Lower Trophic Level Model, Fish Functional Group Model, and Wading Birds Model comprise a tightly linked model group.

![Ecological Models in ATLSS](image)

Figure 5: Ecological Models in ATLSS

5.3 Information Analysis and Data Representation Layer

The main functions of this layer are i) to analyze data using a variety of tools, for example carrying out different spatio-temporal averages; ii) to provide a specific data format for data visualization; and iii) to provide a standard data format for export of data to standard Geographic Information System (GIS) software. As an example, one of the ATLSS structured population models, the ATLSS Landscape Fish Model (ALFISH) [9] is illustrated in Figure 6. This model contains a set of habitat rules and appropriate hydrologic conditions for fish growth based upon observations and the experience of field biologists. As do all the ATLSS models, ALFISH takes as input a variety of information, such as hydrological data, Florida vegetation information (the GAP map)
and produces estimates of fish abundance that natural resource managers can use to help assess the impact of different hydrologic plans.

6. Two Key Technical Issues in Ecosystem Modeling on Grids

In this section, we discuss two major technical challenges encountered in the course of ecosystem modeling on grids and summarize our methodologies to address those issues. These are i) model aggregation and interaction; and ii) heterogeneous high performance computation on the Grid. Although the methods described are developed to address problems associated with ecosystem simulation, they have broader impacts on many research applications involving GIS-based dynamic modeling, multi-scale system simulation, and distributed heterogeneous computation as well as data intensive grid computing.

6.1 Model Aggregation and Interaction

A central issue in ecosystem modeling is the need to link dynamic models that operate across differing spatial regions and at different rates. Due to the characteristics of ecological modeling, the spatial patterns (maps) of ecosystem components are generally derived from a geographic information system (GIS). Two approaches are used to represent geographic information: raster and vector data. In a raster representation of geographic space, the space is divided into an array of cells that are usually square. All geographic variance is expressed by assigning properties to these cells, giving a very simple data model. In a vector representation, all components are captured as collections of points connected by precisely straight lines. Spatial information in a vector-based data set is specified by units with spatially homogeneous properties, (soil properties or land cover) which are defined by a polygonal object, bordered by a polygon. Such vector data lends itself naturally to bucket or compartment models developed to simulate the dynamics of spatially-aggregated system components.

In ATLSS, spatial resolution problems are addressed by the landscape library [10]. The structure of the landscape library is illustrated in Figure 7. The core of this library comprises three groups of classes (Geo-referencing, Regrid, and IODEvice classes)
which can transform and translate the spatial data to and from other forms. Geo-
referencing classes are built to extract regional information from geo-spatial data. These
classes accept any number of vertices, specified in the universal transect mercator
(UTM), which forms a polygon. In a GIS, regions are usually based on the shape and
location of the polygon, defined in terms of UTM coordinates, therefore, the same
regional information can be extracted from data sets with different resolutions,
registrations or spatial extents. Regrid classes have been developed to transform the
data between different resolutions and registrations. These Regrid classes can be
configured/initialized by providing the size of a single cell or by assigning the exact
number of rows and columns to which the data sets should be resized. Spatial data sets
can then be passed to the object, from which a new rescaled map is created and
returned. IODEvices have been designed to allow new data formats to be incorporated
into the landscape library without changing the code which depends on them. Data can
be exchanged between the models via file streams or other applications such as
databases or visualization systems.

Besides the spatial resolution problems, different ecological models in a tightly linked
simulation group may use different timesteps, which introduces the challenge known as
multi-stepping. In ATLSS, the functionality of passing data between models is
implemented, but it is the responsibility of the model developers to ensure proper
processing and time-synchronization of the data.

In our research, a superstructure (class) is built for each ecological model. Generally, a
superstructure contains a model component and a unified communication interface.
Once an ecological model is constructed or adapted to fit within a superstructure, it can
seamlessly exchange data with other superstructures. Two kinds of data exchange
mechanisms are possible: message-passing and database transactions. The schematic
of ecological model coupling is demonstrated in Figure 8. In one scenario, Models 1, 2

Figure 7: Landscape Library Structure
and 3 need to be tightly coupled together, therefore the information exchange is implemented using a message-passing library (represented by blue arrows) to ensure high performance throughout (i.e., low latency). In another scenario, only Models 1 and 2 are linked together, and the intermediate results may be used later by other models. Therefore, connections between those two models (represented by red arrows) are supported by database transactions, and intermediate data are stored separately (by the database system). In this way, one can take advantage of data cache and multithreaded processing capabilities supported by most database systems, and thus maintain high performance since time-consuming IO operations have been separated from data intensive computations.

6.2 Heterogeneous High Performance Computing on Grids

Ecosystem simulation on computing grids will inevitably deploy heterogeneous computation across different computational platforms. We are developing models on an experimental metacomputing framework HARNESS (Heterogeneous Adaptable Reconfigurable Networked SystemS) [11], which exploits the services of a highly customizable and reconfigurable distributed virtual machine (DVM). A DVM is a tightly coupled computation and resource grid that provides a flexible environment to manage and coordinate parallel application execution. The system is designed to support a wide range of DVM sizes, from users building personal DVMs to enterprise and widely distributed DVMs. Collaboration and resource sharing between different entities are performed by the temporary merging and splitting of different DVMs. Virtual machine (VM) terminology, refers to a system where the computing resources on that system can be viewed as a single, large, distributed-memory computing resource. Ecosystem simulation using heterogeneous distributed computation is illustrated in Figure 9.

As shown in Figure 9, a distributed virtual machine (DVM) is established for each tightly coupled ecological model group. Within the context of a DVM, a heterogeneous
message-passing library, FT-MPI [12], provides a series of message-passing primitives, similarly to those in standard MPI, to allow high-performance information exchange between models on different computer architectures. For ecological modelers, there is no major difference in developing models using FT-MPI on a virtual machine or using standard MPI on a homogeneous networked cluster. Therefore, the concept illustrated in Figure 8 can be conveniently implemented in the context of a DVM. In addition, the data exchange (if necessary) between separated ecological model groups can be implemented via database transactions.

Another important functionality of grid computing is related to resource discovery and
fault tolerance. We use the NetSolve toolkit [6] to bring together disparate computational resources connected by computer networks. It is a remote-procedure-call (RPC) based client/agent/server system that allows one to remotely access both hardware and software components. An ecosystem simulation using the NetSolve system is illustrated in Figure 10.

For each ecological resource (database, DVM, etc.), a NetSolve server is established, and all NetSolve servers are registered into a database maintained by NetSolve agents. Therefore, when a remote user issues a job request through a dedicated website, a NetSolve agent will find the location of an appropriate server, and then the job will be routed to the appropriate ecological resource. Upon completion, the result will either be stored locally (the remote user is notified by email) or may be shipped to the remote user via Internet Backplane Protocol (IBP) [7] – a high performance internet file transfer protocol.

7. Summary and Perspectives

Grid computing offers great promise for large-scale ecosystem modeling that will potentially lead to better understanding, controls and management of the natural resources. We have focused here on a specific application of regional ecosystem simulation. With our increased understanding of the fundamentals of an ecosystem, more complex simulations will be needed, which will in turn continue to challenge grid hardware and software architects. It is the view of the authors that collaborations between ecologists and computational scientists are critical to enable advances in ecosystem modeling. From the perspective of computational ecologists, the challenges of ecosystem modeling on computing grid are: i) to provide sufficient support services to sustain the computational environment; ii) to design a flexible open software architecture to support high performance ecological multimodeling; and iii) to ensure that the ecosystem simulations performed on the Grid constitute the next generation of advances and not just proof-of-concept computations.

Acknowledgement

This research has been supported by the National Science Foundation under grant No. DEB-0219269. This research used the resources of the Scalable Intracampus Research Grid (SInRG) Project at the University of Tennessee, supported by the National Science Foundation CISE Research Infrastructure Award EIA-9972889.

Reference:


