ATLSS Modeling: Extensions to Variable Spatial Grid Hydrologic Models

Final Report on

Requisition No. 5284-04-0050

December 1, 2005

Principal investigator:

Dr. Louis J. Gross

Department of Ecology and Evolutionary Biology

The Institute for Environmental Modeling

University of Tennessee

Knoxville, TN 37996-1610

Introduction:

The ATLSS (Across Trophic-Level System Simulation) project to develop extensions to variable spatial grid hydrologic models has involved two main components: (a) developing new Landscape classes to allow ATLSS models to utilize hydrologic models in addition to the South Florida Water Management Model (SFWMM), specifically models that have a variable spatial resolution; and (b) modification of the SESI (Spatially-explicit Species Index) models to utilize the new Landscape classes to process hydrologic model results. These tasks were undertaken in response to restoration goals that require a defensible scientific methodology to compare and contrast impacts at the various spatial and temporal scales of the alternative hydrologic plans being produced.

The sections below summarize the various components of the conversion process to extend ATLSS Landscape classes and SESI models to use variable-grid hydrology data sets. The basic tools for input conversion, formatting, and data storage have been developed and tested. The Landscape classes are now able to provide the functionality needed by the SESI models, while having the flexibility to handle different types of meshes. The new classes currently have the capability to handle points, triangles, squares, and rectangles as underlying shapes for a mesh. This functionality will allow hydrology data from SFRSM, MODBRANCH, and WASH123D models to be stored, accessed, and processed within the Landscape v3.0 classes. Conversion of the SESI White-tailed Deer Model has been completed, and trial simulations have been run and evaluated.

Landscape Classes

New C++ classes were designed, implemented and tested to handle spatial data that are based on an irregular division of space. A single set of classes handles the hydrology data from all of the different types of hydrology models that are proposed to be utilized, including the existing SFWMM hydrology data. The new classes update the functionality provided by the ATLSS Landscape v.2.0 classes

and constitute a new version of the Landscape classes, version 3.0.

Landscape v3.0 classes are designed to handle memory management and geo-spatial referencing. They accommodate input from a variety of data layers, such as hydrology, vegetation and topography and form the spatial framework for the extended SESI models.

The SESI models require such basic functionality as indexed access to cells, conversion between indexes and UTM (Universal Transverse Mercator) coordinates, and spatial access from one cell location to a specified group of neighboring cells. Implementation of these functions by the Landscape classes changes as we move away from a regular two-dimensional grid with cells of invariant size and shape. Each type of mesh poses unique challenges, but we have focused on underlying commonalities that support a generalized approach.

We have developed Landscape classes that have the flexibility to load different meshes without code changes to the SESI models, which enables us to maintain a single set of code and run a common set of algorithms for all the hydrology meshes. This flexibility has been accomplished by developing classes that generalize the shapes that make up a mesh in such a way that the functionality is preserved whether the underlying shape of mesh cells is a triangle or square.

A final task for the Landscape classes was development of input and output standards that would work across all the different meshes, while maintaining computational and storage efficiency. This goal was accomplished by implementing data storage in a NetCDF container file.

This work was performed in support of Task 1 : Basic data structures.

General structure of Landscape v3.0 classes

The Landscape v2.0 classes utilized by the original SESI models were designed to process double-indexed regular grids with row and column access. Geo-referencing was accomplished through index offsets from known corner locations. The new variable-grid meshes may have irregular cell placement, irregular cell dimensions, and varying cell sizes, as can be seen in Figures 1 and 2, which depict the SFRSM triangular mesh.

Processing variable-grids requires a completely different mechanism for implementing geo-referencing and cell connectivity that is independent of cell shape and relative position. To accomplish these goals and provide the flexibility to handle different types of meshes, the revised Landscape classes implemented a separation of data storage, cell shape information, and interconnectivity. These three components were then wrapped to provide a unit called a Landscape object.

Data storage

The storage of data in a Landscape object is handled though a one-dimensional Standard Template Library (STL) vector array. STL vectors provide an efficient means of storing, accessing, and writing data. The storage order for cell data provides an index that is used to link the shape information with the data. The one-dimensionality of arrays causes a breakdown in two dimensional access, even in the case of a regular grid, because there is no assumption of either the column-major (Fortran) order or the row-major (C) order. The single-dimension index provides spatially independent iteration over cells. A secondary requirement that we placed on the order of cells in the one-dimensional array is that the same index must access the same element across all Landscape objects derived from the same mesh, i.e. in all Landscapes derived from the same mesh, the order of cell data storage is the same. This constraint is important when cell connectivity is considered.

Cell shape information

Information about the shape of cells is encompassed in a shape class designed to hold formatting information for a single cell and provide basic functions applicable to the cell. The information stored includes the cell's location and the connection order for vertices of the shape. Examples of shape functions include computation of the area or the centroid of the cell. These functions are specific to each shape, in order to exploit inherent efficiencies in their geometric structure. Shape classes were implemented to handle the fundamental shape objects of *point*, *triangle*,

square, and *rectangle*. An extension to a more abstract polygon shape is a future possibility, in the event that a mixed-shape mesh is ever created or needed.

A minimal amount of information is stored within the shape class, to keep memory size manageable, as we envision the possibility of handling over 100,000 shape cells for each Landscape mesh generated. The shape class does not store information related to connectivity between cells, area, perimeter or other cell-specific information. This type of cell-level information has limited utility to the ATLSS models in general. For models that do require this information, functionality has been incorporated into the Landscape classes to provide it. Persistence of this information within a model can be achieved by storing it in an auxiliary Landscape class object. In this way we avoid storing unwanted data for models that do not use it, but provide a simple and efficient mechanism to make it available when necessary. The result is cleaner, more efficient, and faster-running models.

Cell connectivity

Determining and storing information about cell connectivity is the third major component of the new Landscape classes.

In a regular two-dimensional grid, the familiar sense of connectivity for a "neighborhood" of cells surrounding a source cell is implemented by incrementing the row/column index relative to that source cell. Likewise, given uniform cell dimensions, an area around a cell can also be searched by capturing a certain number of cells in each direction.

Both the ability to have relative movement between cells and the ability to capture neighborhood areas within a given radius through incrementing an index are lost when irregular meshes are adopted, because cell size is variable and positions relative to a source cell cannot be assumed to be a constant distance across the mesh.

Our solution to regaining this functionality is to generate look-up tables that allow each cell to know its neighbors. In the case of an area search within a radius surrounding a source cell, a look-up table can be generated to provide access to

target cells and the proportions of their areas that fall within the desired radius. Look-up tables can also help regain the familiar intrinsic relative positioning information that is found in a regular grid, while providing the ability to have new and expanded definitions of connectivity. In a triangle-based mesh, for example, connectivity could be defined to occur through the edges, consistently generating 3 neighbors, or through the vertices of the triangle, which could generate a varying number of neighbors depending on the size, shape, and configuration of triangles. The ability to define mesh-specific rules for connectivity will allow users to implement a range of applications within a common framework.

The generation of radius area look-up tables requires the ability to determine whether a cell is within the specified neighborhood region. At a computational level, this requires a Polygon Cutter to determine whether two polygons intersect and, if so, to define the resulting polygon of intersection. This process becomes an onerous algorithm to implement due to the pathological cases that can and do arise. For example, when two polygons overlap and create very long, narrow triangles of intersection (Figure 3), these narrow triangles can cause algorithm failure. The benefit associated with building the Polygon Cutter derives from its ability to handle all polygon shapes. The Polygon Cutter is also utilized when the grid input files are pre-processed and interpolated to the various meshes (e.g. FGAP, discussed below).

Information about cell connectivity resides in separate data objects apart from Landscape map objects. This approach was chosen so that a defined connectivity map could be applied across all mesh-identical Landscapes through the use of a consistent indexing order for mesh cells.

Input and output capabilities

A fourth component addressed in the development of the new Landscape classes is input/output capability. As in previous versions of the Landscape classes, a binary file format is required for two primary reasons. Data can be stored more compactly in a binary format, and reading from and writing to binary files is much faster. The ATLSS Landscape v3.0 classes utilize the NetCDF format to

store all the various meshes. The NetCDF interface allows easy and fast code based I/O, machine independence, and ready-made tools for ASCII conversion and exploration of data. The I/O and binary nature meet the requirements of compatibility with the original ATLSS binary format. The open standard means that the Landscape v3.0 classes will not require any special knowledge or software to access NetCDF files other than the free and publicly available tools associated with NetCDF.

NetCDF is used by the South Florida Water Management District for storage of SFRSM hydrology output. Adoption of this format by ATLSS should facilitate utilization of model results and associated information. More information about the NetCDF format can be found at their website:

`http://my.unidata.ucar.edu/content/software/netcdf.`

Other I/O formats are available and some effort has been directed towards finding a more direct method of integrating the data into ArcGIS, which would provide natural resource managers with a more flexible way of visualizing and exploring our results, as well as exploring information associated with the various sets of hydrology data. Shape files are currently being used to store each shape as an individual object. This method allows multiple data values to be stored within each shape and allows the transfer of information from the Landscape classes into Arc without interpolation or loss of resolution. However, as the number of shapes increases, the computational and display burden can become cumbersome for Arc to handle.

We are able to use the same compiled executable to load original ATLSS Landscape v2.0 binary files of hydrology data from the SFWMM, as well as files from SFRSM and other meshes. This interoperability has the major advantage of a single development track and executable that simplifies the entire process. The organization of appropriate input mesh files should not be burdensome, as it appears that mesh changes will be infrequent once production runs are in progress.

Shape Files and ARC/GIS

The capability to visualize and combine data within a GIS environment was one of the goals of the new Landscape classes. Our existing techniques to convert raster data to a GIS format could not be applied to the mesh structure of the new hydrology data sets. For this reason, we proposed and developed new software to convert the mesh data to an alternative, standard GIS format. The shapefile format was implemented as a generic means to handle all the various meshes and grids that were identified. The use of shapefiles allows a true representation of the underlying shapes used in the mesh and the association of data within the mesh cells. This format avoids the loss of information and accuracy that would result from raster interpolation. ARC/GIS has tools that can easily convert a shapefile to a raster layer and allow the user to determine the resolution necessary to maintain needed accuracy. The shapefile format is a standardized, open format that is used by many other products and can be accessed outside of ARC/GIS.20

The output method of writing mesh layers to a shapefiles decomposes each mesh cell into an individual shape. That shape is then inserted into a shapefile and the mesh cell value is inserted into the shapefile database file that relates to the shape. This process occurs for each of the mesh cells as a single step within the Landscape classes. Internally within the Landscape classes, the individual shape insertion process is accomplished through the use of the GDAL library (www.gdal.org) or the Shapelib library (shapelib.maptools.org) that has been subsumed into the GDAL library.

This work was performed in support of Task 2: Output to ARC shapefile format.

Input File Conversion

Spatial data layers used as inputs to the SESI models are being preprocessed into the new standard file format to minimize processing time while running the ATLSS SESI models. The software we developed for this purpose has been applied to SESI input files and differences between the unconverted and converted data have been evaluated. This evaluation included visualizing the unconverted and

converted data using ARC GIS.

The conversion process makes extensive use of Landscape v3.0 classes to read data in the original file format and then load them into the new spatial data classes, converting from a regular to a variable-grid format. Two examples of the conversion of input files to mesh format are described below.

A typical problem in the conversion of data for model input is the translation of geo-spatial data to a different scale of resolution and/or gridding scheme. In the case of a variable mesh, conversion may also include a change in the underlying shape and non-uniformity in the resolution of the grid. To handle the multitude of factors encountered in converting from one shape and resolution to another, a computational geometry approach was taken with regard to the data transformation. In essence, each mesh cell is treated as an individual shape and is used as a reporting region for the underlying source data set being transformed. In most cases, the source is of a much finer scale of resolution, requiring the calculation and tabulation of many intersections to generate the appropriate value for the cell. A spatial intersection is performed so that partial contributions can be computed along the boundary. Difficulties arise in this approach due to the failure of many common algorithms when anomalous shapes are occasionally generated by the intersection. For example, extremely acute triangles cause most area calculations to fail. The implementation of the intersection capability within the Landscape v3.0 classes results in a broad range of applications. In this case, it is used to facilitate the conversion of input data. An additional use would be to summarize data for various reporting regions within the model area.

This work was performed in support of Task 3 : Conversion of inputs.

FGAP **Map**

FGAP v6.6 is a common input into the SESI models to provide the underlying land cover layer (Figure 4). The compilation of FGAP information into the mesh format requires the ability to compute the intersection of 30-m x 30-m grid cells with the shape objects that make up the new mesh. Each shape in the mesh "cookie cuts" into the FGAP map and tabulates the resulting area of each vegetation type that

is intersected. This process utilizes the Polygon Cutting algorithms developed in conjunction with the Landscape v3.0 classes, described above. Figure 3 shows a typical triangular cell of SFRSM and the underlying FGAP 30-m cells that are cut out by this shape. Because all the new meshes have a resolution coarser than 30m, the resulting tabulation is typically an aggregation of several vegetation types from multiple underlying cells, with varying areas of intersection for each type.

The dominant vegetation type is determined, as well as the area of each vegetation type represented within the target mesh cell. In essence, a mesh-based map layer is created and stored separately for each vegetation type. These maps are stored in a NetCDF file for uniform and easy retrieval of information by the SESI models. Figure 5 shows a map of the dominant FGAP layer, clipped to the existing ATLSS and SFWMD map area boundaries. The values in black will be replaced with true values when we move to the full FGAP map as a source. The values in black will be replaced with true values when we move to the full FGAP map as a source. Ignoring the color scheme, a loss of heterogeneity can clearly be seen as we convert from the 30m FGAP to the variable SFRSM scale of resolution (Figures 4 and 5). Similar sets of FGAP input maps will be generated for each different hydrology mesh format. As an ancillary benefit, storing sets of FGAP layers in NetCDF provides easy accessibility to all layers compared to file storage formats we have used in the past.

Elevation Map

The underlying elevation map used for the meshes is also a typical input file for the SESI models. Elevation maps are provided by the various versions of hydrology models along with hydrology output data for scenario releases. One of the advantages of the NetCDF file format is that we can incorporate elevation layer information directly into the hydrology NetCDF file. This level of integration will help eliminate any possibility that users might associate incompatible elevation files with hydrology scenario data. Incorporation also insures compatibility of units across all files as part of the integration process.

ATLSS SESI Model Conversion

Code modifications have been implemented and tested that enable the SESI models to utilize Landscape v3.0 functionality to run over variable-grid meshes. The process of converting the SESI models to run on the new mesh format is a multi-step process that includes code modification, algorithmic verification of modified code and evaluation of model runs on the new scenarios.

This work was performed in support of Tasks 4 and 5 to modify the SESI model framework to handle variable spatial grids and integrate and test products from other project tasks in a working SESI model: input of data, model initialization and updating, and output of results.

Conversion Process

In order to ensure that code modifications did not result in changes in model conceptualizations and associated model projections, we established a set of baseline outputs from the current SESI models and followed a cycle of code modification and testing, in which changes were planned and implemented as a series of steps, with the models remaining functional after each step. Model output was compared to the baseline output as each step of the revision process was completed. This approach allowed any errors introduced during a step to be corrected at that time.

As code development and testing progressed to include mesh processing, existing SFWMM hydrology data sets were treated as a mesh, using Landscape v3.0 classes, rather than a grid in Landscape v2.0 format. A square shape was utilized to load the information from a grid into a Landscape v3.0 object. Model results from the original Landscape v2 SESI code, implemented exclusively for the SFWMM hydrology, could then be compared with the new results from modified SESI code utilizing SFWMM hydrology formatted by the new Landscape v3.0 classes.

Because both sets of simulations use the same input information, differing only in how data are stored and processed internally, a direct comparison could be made to verify fidelity to the original model at each step. This comparison

required the conversion of all input files and thereby provides a secondary check for the file conversion routines as well.

The next step was to run the model on test hydrology data provided for the new meshes. We performed a qualitative examination of results to evaluate changes in scale and spatial heterogeneity that were not part of the original SESI simulations. A more intensive evaluation can be done when calibration runs are available that reflect historical hydrological conditions.

We did not find it necessary to modify the class configuration of the SESI models to explicitly compartmentalize cell behavior, as existing code was sufficiently modular to separate cell functions from landscape processes. We simplified the file and directory structure, collapsing driver, timing, and index files into a single directory, and eliminated utility functions that have now been replaced by functionality in the Landscape classes. In addition to allowing the models to handle Landscape v3.0 classes and variable spatial grids, these changes have resulted in a body of code that is easier to manage and modify.

SESI White-tailed Deer Model

The conversion of the SESI Deer Model has been completed and the model has been successfully run on both a converted SFWMM hydrology scenario and test hydrology set for SFRSM. The comparison of the original versus Landscape v3.0 versions of the SFWMM hydrology showed no differences, thereby indicating preservation of the integrity of the model. The results of the SFRSM run showed qualitatively proper responses.

A sample output of the Breeding Index from the SFRSM run is shown in Figure 6. As can be seen, the extent of the map is limited to the old ATLSS/SFWMD model area. This restriction is a result of the use of input files from the SFWMM version of the SESI Deer Model to maintain compatibility during the conversion/verification process.

Other SESI Models

We have templated the process of SESI Deer Model conversion to develop protocols to apply to the conversion of other SESI models, developing a set of procedures

for migrating code from the existing configuration to one that is compatible with Landscape v3.0 classes and with mesh data. This process will also result in more standardization across models that will improve the maintainability of the code and facilitate incorporation of modifications as new data and research become available.

The SESI Short Legged Wading Bird, SESI Long Legged Wading Bird, and SESI Panther use spatial averaging functions in the Landscape classes that will be completed during the next funding cycle. We have also proposed to continue the conversion process to include additional SESI models (Cape Sable Seaside Sparrow, Snail Kite, American Alligator, Apple Snail, and Crayfish), enabling these models to use the functionality of Landscape v3.0 classes to handle variable spatial grids and to handle the refinements in spatial resolution of these hydrologic inputs. Additional changes will incorporate new information from ongoing research and monitoring of modeled species. Recent radio-telemetry locations for panthers in key segments of the model area near Shark River Slough, for which scant data have been available in the past, will be incorporated, along with the long-awaited first 24-hour GPS data available for panthers, which have been delayed by collar malfunctions, to provide a more accurate assessment of the responses of panthers to hydropatterns. These conversions will allow the relative assessments provided by ATLSS SESI models to keep pace with current research and monitoring and with changes in the hydrologic models.

Hydrology

In developing new Landscape classes to allow ATLSS models to utilize hydrologic models that have a variable spatial resolution, we maintained compatibility with the SFWMM, while focusing on three new models: SFRSM (South Florida Regional Simulation Model); MODBRANCH, a model that couples the USGS groundwater model MODFLOW with a surface water model BRANCH; and WASH123D, a 3D hydrology model developed at the University of Washington for the US Army Corps of Engineers (USACE).

The process of integrating the four main types of hydrology data - SFWMM,

SFRSM, MODBRANCH, and WASH123D - with Landscape v3.0 has been completed to the degree possible given availability of hydrologic output files. With existing SFWMM and SFRSM multi-year files, development has proceeded smoothly to the stage of test SESI model runs. MODBRANCH and WASH123D single-year simulations provide a greater challenge for SESI implementation, as the SESI models incorporate multi-year dependencies. The process of reading and formatting MODBRANCH has been completed. Conversion of WASH123D hydrology data was hindered by lack of availability of hydrologic output files, as described below.

SFWMM

As part of our evaluation and development process, the capability of processing SFWMM hydrology data in regular grid format has been implemented in the Landscape v3 classes by considering the grid as a type of mesh with square cells of invariant size. Providing this functionality allows the revised SESI models to continue to read and process hydrology files from the earlier SFWMM scenarios we have evaluated. Storage of SFWMM hydrology files in NetCDF format would provide some accessibility advantages if desired in the future.

SFRSM

Conversion and I/O routines for the SFRSM hydrology have been completed, due in part to the comprehensiveness of the example files provided by the District and the accessibility of the format. Conversion of input files to SFRSM mesh format has been completed as well, and test runs of the SESI DEER model on SFRSM hydrology have been made and evaluated. Handling of the mesh itself is robust to changes in the structure of the mesh, but input files will need to be regenerated for each different meshing of the study area. This flexibility is important, as the current mesh implementation of SFRSM is still under development and review.

MODBRANCH

The conversion code for MODBRANCH hydrology file access has been completed. Figure 7 shows the active mesh for MODBRANCH as read into the Landscape

classes from the GMS 2dg file format. The conversion process for hydrology required a multi-step approach due to the file format and the location of data across multiple files. The conversion centralizes this information within the NETCDF file structure for accessibility by the Landscape v3.0 classes.

The data needed to generate the hydrology and elevation files are contained in three files produced by MODBRANCH: `hbin_12hr.dat`, `(scenario_name).unix.2dg`, and `CSOP_PMs.csv`. `hbin_12hr.dat` holds the water data generated by the Fortran MODBRANCH code, implemented by the GMS software and not generally available for each run. We received output for one year of the simulation, which had a file size of 900mb. The Fortran binary water file (`.hbin`) has a heterogeneous structure containing metadata, water values, and Fortran formatting, which makes a direct parsing difficult. The simplest and most reliable way to access this information was to use a Fortran program to read the file and transfer the information directly into the NETCDF container file. The `(scenario_name).unix.2dg` contains the meshing information in a proprietary GMS file format. Parsing of this file provides the variable grid spacing and active cell information to specify insertion of the hydrology data into grid format. `CSOP_PMs.csv` contains elevation data and center-of-cell information that serves as a check for the 2dg parsing. These three files are utilized in the generation of a hydrology NETCDF file for Landscape v3.0 access.

Several decisions (e.g. period selection or averaging) need to be made with regard to the most appropriate method for combining wet, dry, and normal years as a scenario for SESI runs that are designed for multi-year periods.

A side benefit of the `CSOP_PMs.cvs` file parsing is the ability to convert any of the approximately 150 indexes located in the file into a shape file map for analysis within GIS software.

USACE Contact: Robert Evans

Email: Robert.A.Evans@saj02.usace.army.mil

Website: `http://hpm.saj.usace.army.mil/csopweb/mbr/`

WASH123D

The WASH123D model is currently under development by the US Army Corp of Engineers. The process for fully integrating this format into the ATLSS Landscape adapters has been delayed because the model has not yet reached a point in its development at which sample output files are available, as was originally anticipated (Earl Edris, Coastal and Hydraulics Laboratory, USACE, personal communication). However, subsequent discussions with USACE representatives indicate that the design of the Landscape classes will be able to handle the data once final formats and structures are determined and appropriate IO code is implemented within the Landscape classes. The following is our current understanding of where the WASH123D is heading in its development with regard to our utilization of the model as a scenario resource.

The WASH123D hydrology model utilizes a point meshing of approximately 60,000 points, with each point representing several hydrological strata within the model. The model is being implemented within GMS Hydrology Modeling Software, which is the same modeling framework used for the MODBRANCH implementation. The model output is currently being stored in a proprietary GMS file format. However, developers have indicated that the final implementation will use a more open data structure based on the XMDF geometry format, stored in a HDF5 storage file. This is a new feature developed by USACE and implemented in the new version of GMS to facilitate the communication of hydrological data. Straightforward integration of the XMDF/HDF5 format will be possible once data files are available. Internal Landscape v3.0 data structures to handle these point data types are currently in place.

The development goal of WASH123D is to produce single year simulations, due to the long computational time (days) required to run a single year. Consultation with NPS in the future will be necessary to determine an appropriate method of constructing multi-year scenarios, comparable to those developed by other models, to evaluate sequences of years with variable hydrology.

USACE Contact: Earl Edris

Email: Earl.V.Edris@erdc.usace.army.mil

Web Resources and References:

http://www.ems-i.com/GMS/GMS_Overview/gms_overview.html

http://www.wes.army.mil/ITL/XMDF/

http://www.ems-i.com/GMS/gms50new/xmdf.htm

GMS

The implementation of MODBRANCH and WASH123D occurs within the framework of the hydrology modeling application Groundwater Modeling System (GMS). GMS provides visualization, implementation and storage formats for the models. Understanding and using the GMS data storage formats is an important component of the Landscape v3.0 classes and SESI conversion process. Mesh information is stored in a proprietary GMS format, and adapters must be developed to read and extract the data. The mesh information and hydrology data will then be stored in a NetCDF file for use by the models. The ATLSS implementation limits preprocessing of hydrology data for the use by the SESI models as much as possible, but in this case preprocessing is required to combine information from several separate files.

Website: http://www.ems-i.com/GMS/GMS_Overview/gms_overview.html

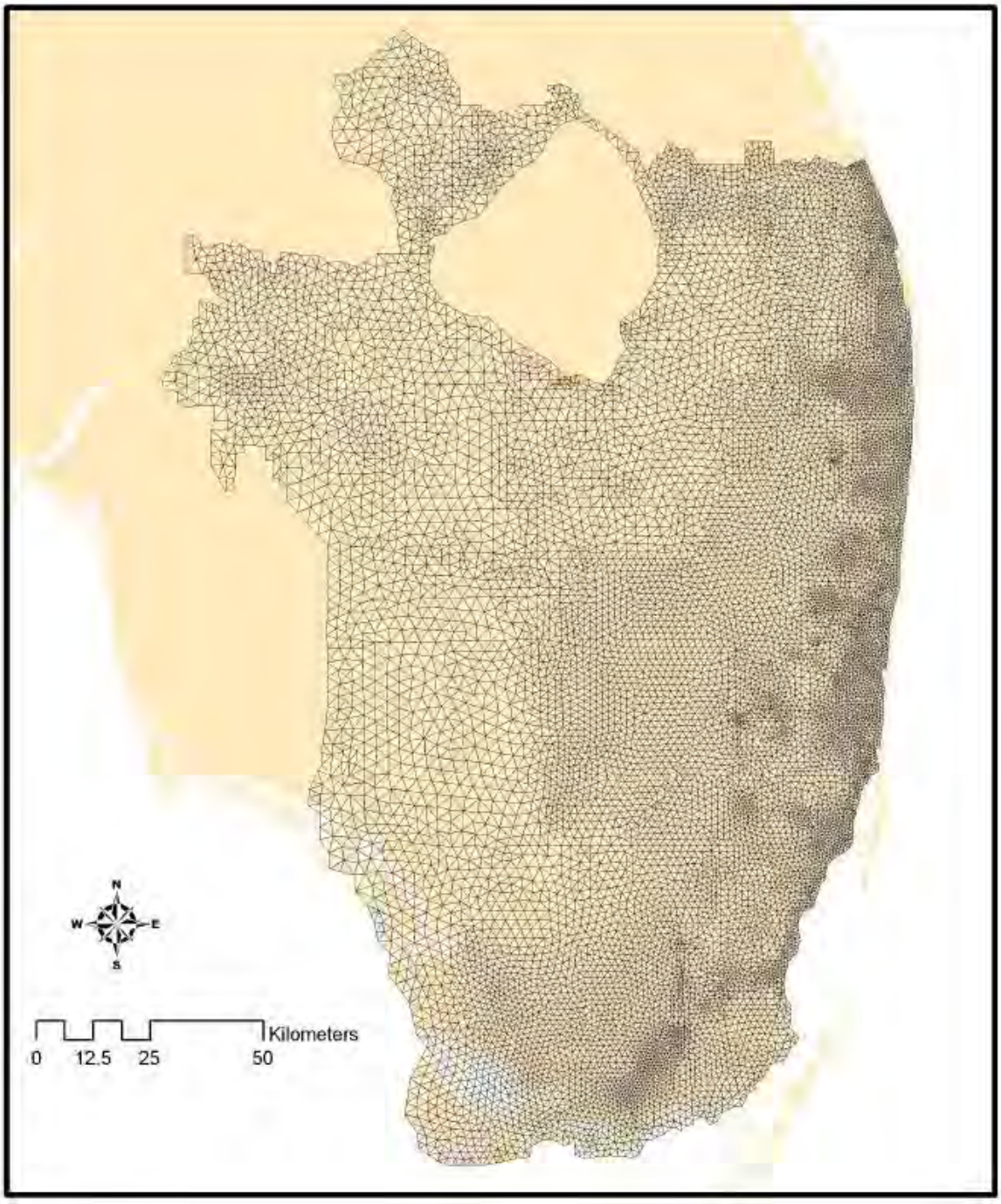

Figure 1: SFRSM triangular mesh.

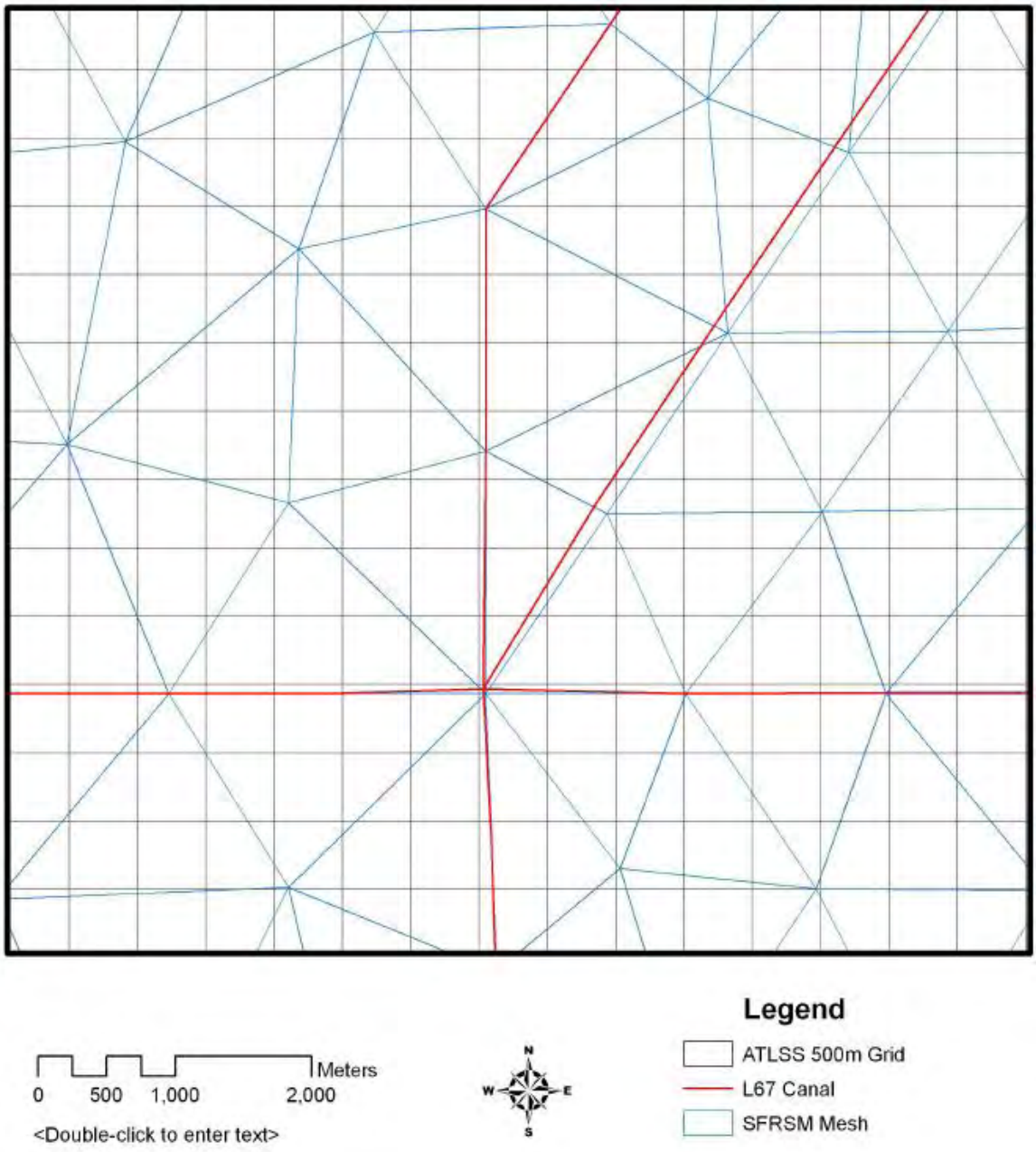

Figure 2: Comparison of triangular SFRSM mesh (blue) with ATLSS 500-m x 500-m grid cells (black) in the area of the south L67 Canal (canal overlay in red).

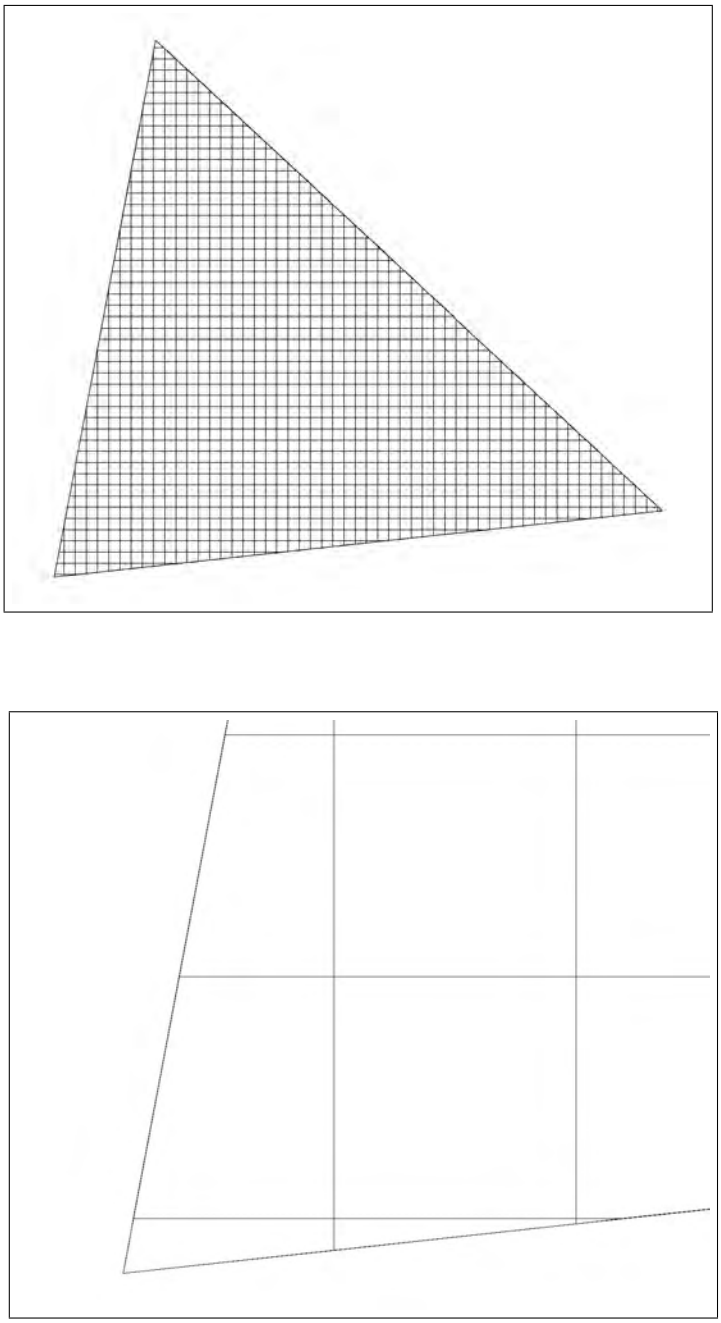

Figure 3: SFRSM triangle shape (approx. 1500 m per side) showing constituent 30-m cells (top). Detail of lower left corner, showing "cut" 30-m cells on edge (bottom).

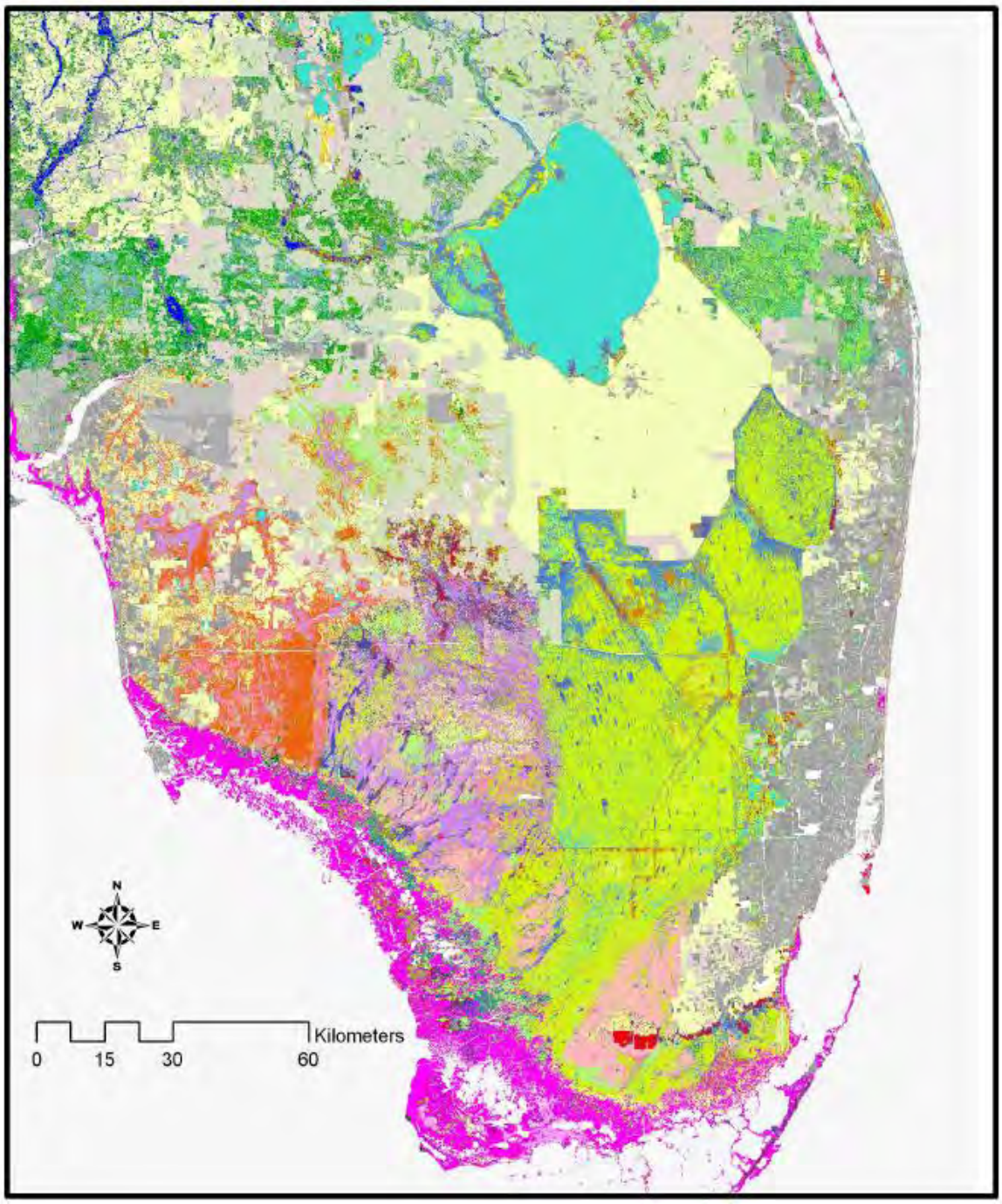

Figure 4: Florida Gap Analysis (FGAP) Vegetation Map v6.6.

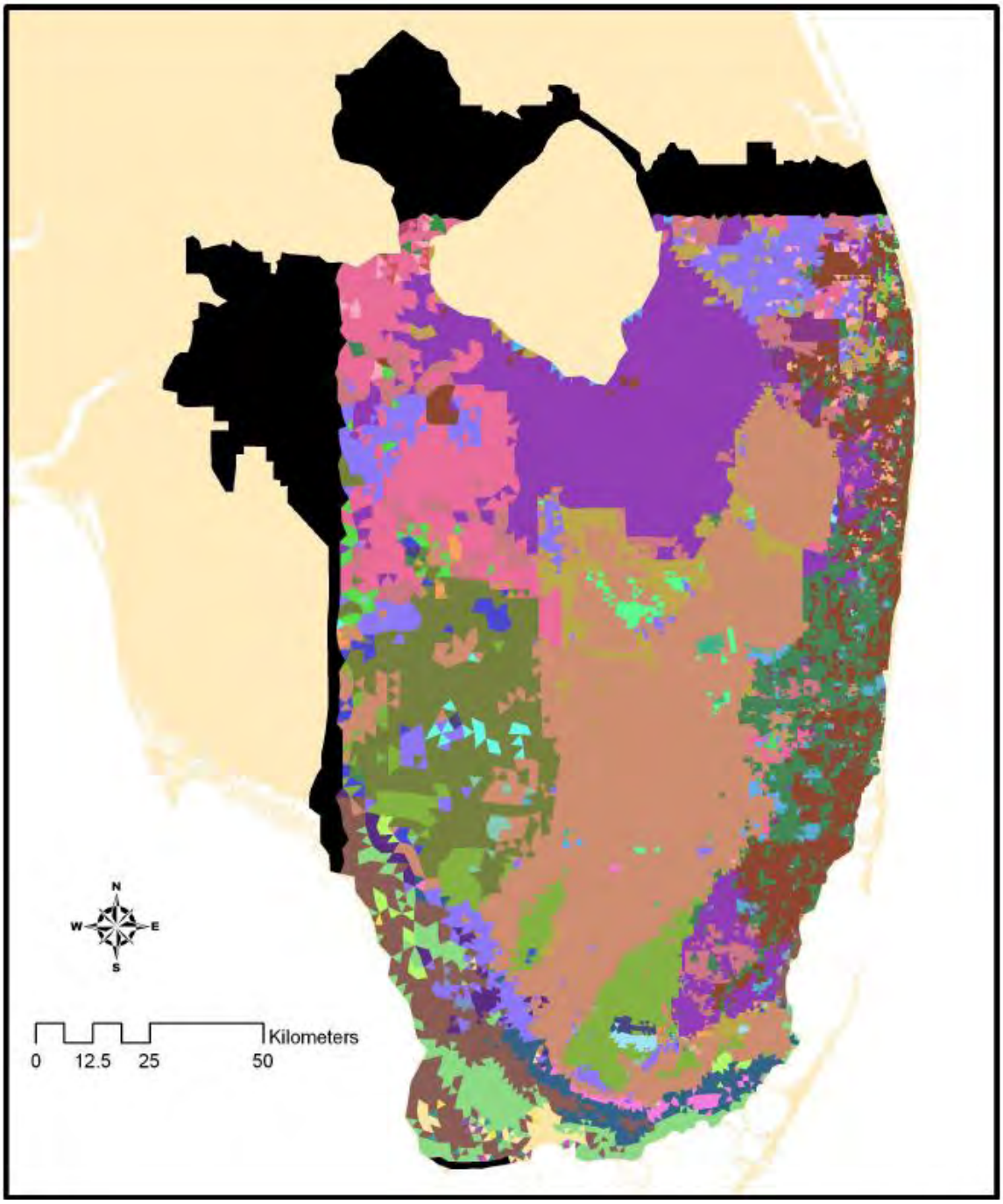

Figure 5: Interpolation of FGAP v6.6 map for SFRSM mesh.

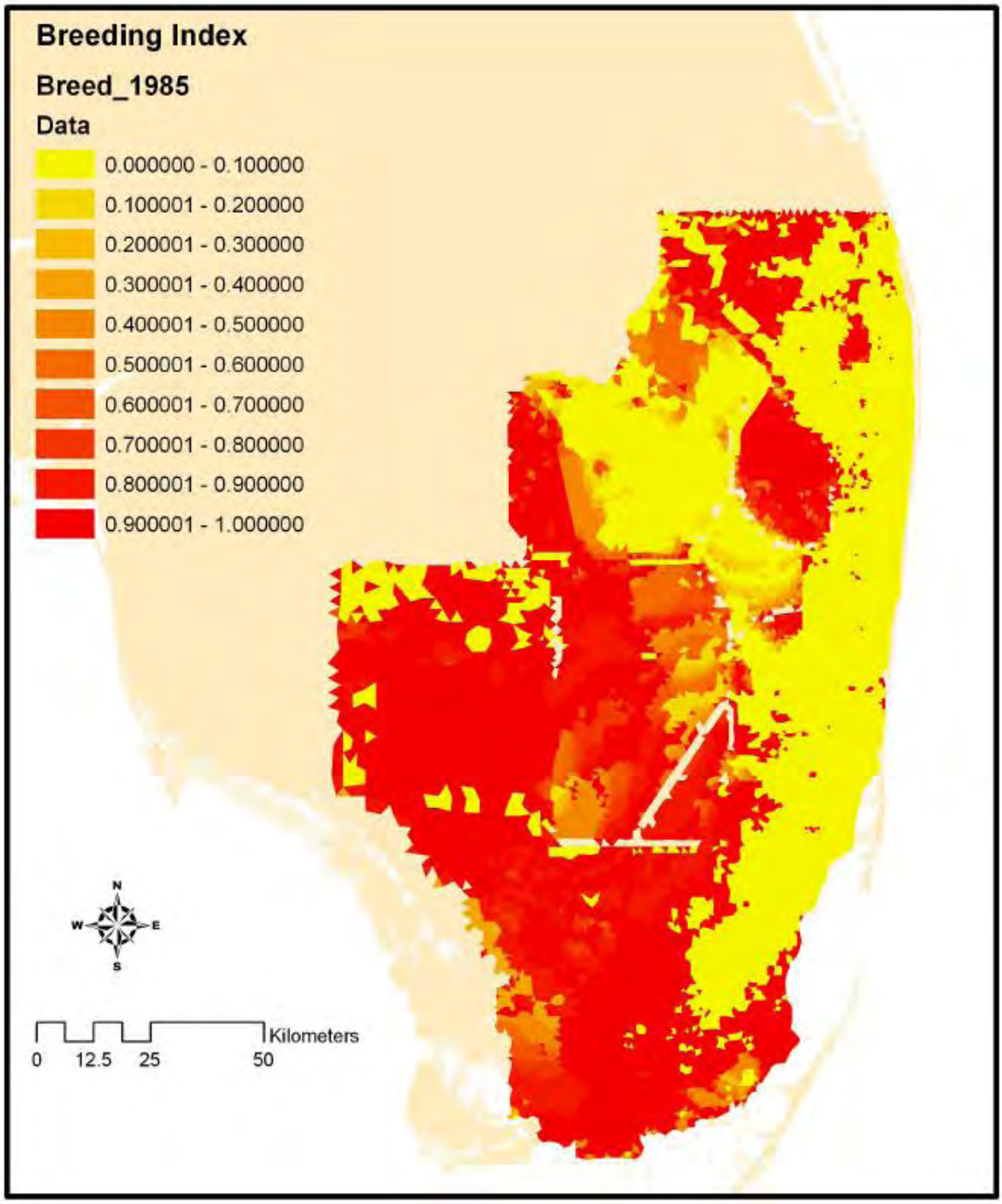

Figure 6: SESI DEER Breeding Index map, SFRSM 1985.

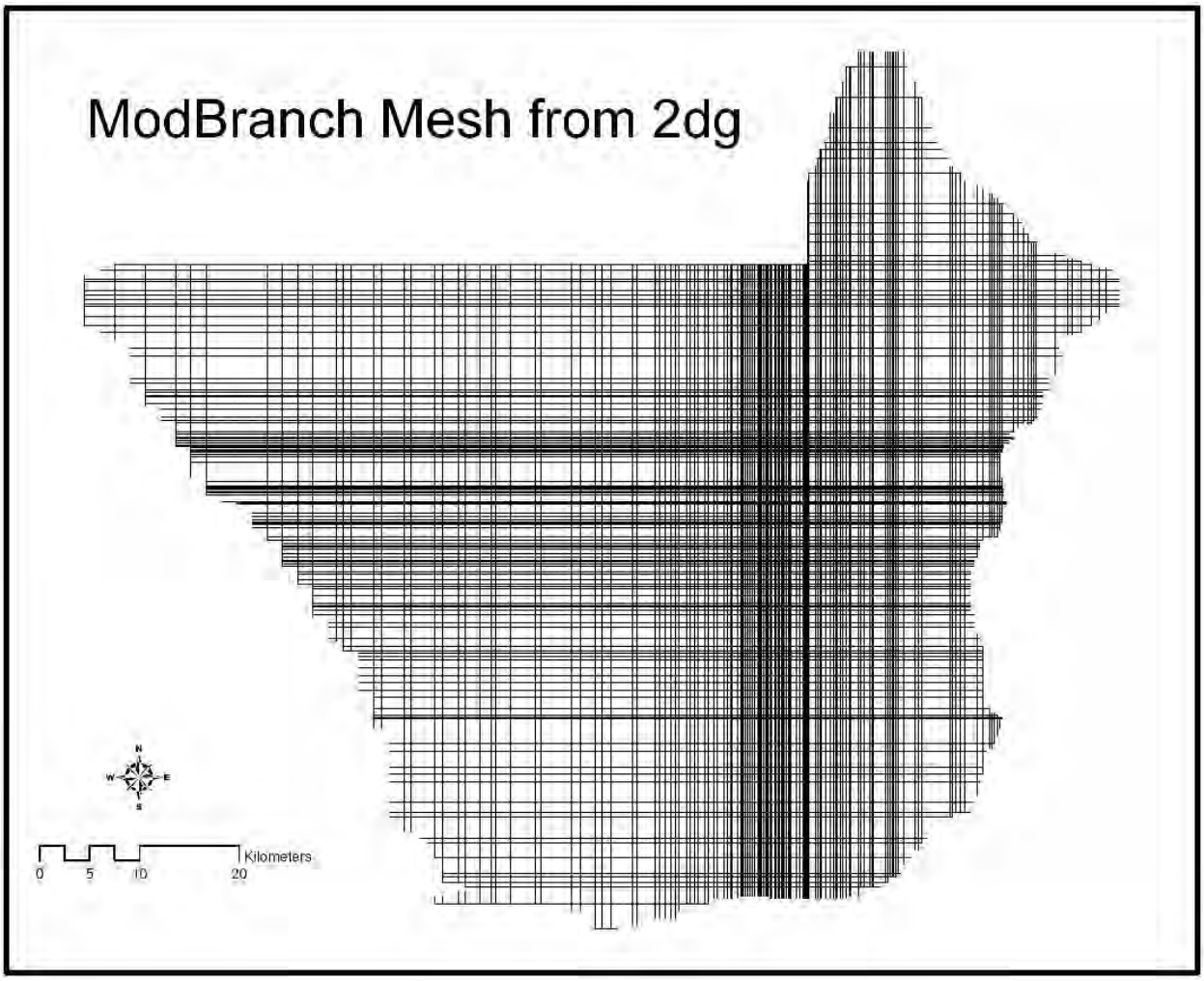

Figure 7: Active mesh for MODBRANCH, GMS 2dg file format.