Environmental Modelling & Software 50 (2013) 25–36

Contents lists available at ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

A model-independent open-source geospatial tool for managing point-based environmental model simulations at multiple spatial locations

K.R. Thorp*, K.F. Bronson

USDA-ARS, U.S. Arid Land Agricultural Research Center, 21881 N Cardon Ln, Maricopa, AZ 85138, USA

ARTICLE INFO

Article history:
Received 23 January 2013
Received in revised form
7 August 2013
Accepted 5 September 2013
Available online 27 September 2013

Keywords:
Cotton
GIS
Model
Plug-in
Python
Quantum
Simulation
Spatial
Yield

ABSTRACT

A novel geospatial tool box named Geospatial Simulation (GeoSim) has been developed, which can be used to manage point-based model simulations at multiple locations using geospatial data within a geographic information system (GIS). The objectives of this paper were to describe GeoSim and demonstrate its use. GeoSim has been developed as a plug-in for Quantum GIS, and both of these software programs are open-source and freely available. An important feature of GeoSim is its model-independent nature, meaning any point-based simulation model that uses ASCII files for input and output can be managed spatially. GeoSim facilitates the transfer of geospatial data from the GIS database to the model input files and from the model output files back to the GIS database. GeoSim presently includes six software tools, each with a graphical user interface. A case study demonstrates the use of GeoSim for processing geospatial data layers at a field site, conducting spatial model simulations, and optimizing model parameters for site-specific conditions. Two cropping system models, AquaCrop and the DSSAT Cropping System Model, were implemented to simulate seed cotton yield in response to irrigation management, nitrogen management, and soil texture variability for a 14 ha study area near Lamesa, Texas. Geoprocessing tools within GeoSim were able to summarize 5592 data points within 405 polygon features in 3.8 s. Simulation tools were able to swap 33,316 and 44,550 parameters values to complete 405 spatial simulations with the AquaCrop and DSSAT models in 112 s and 398 s, respectively. These results demonstrate the utility of GeoSim for summarizing large geospatial data sets and transferring the data to the file formats of multiple models. Simulation duration was increased as compared to stand-alone model simulations without parameter swapping, which may be problematic for applications requiring large numbers of simulations. The flexible design of GeoSim is intended to support spatial modeling exercises for a variety of models and environmental applications.

Published by Elsevier Ltd.

Software availability

Name of software: Geospatial Simulation (GeoSim)
Developer: Dr. Kelly R. Thorp
Contact: kelly.thorp@ars.usda.gov
Hardware requirements: Desktop computer
Software requirements: Quantum GIS 1.7.2 or higher
Programming language: Python
Availability: Download with the Quantum GIS plug-in installer
Cost: Free

* Corresponding author.
E-mail address: kelly.thorp@ars.usda.gov (K.R. Thorp).

1364-8152/$ – see front matter Published by Elsevier Ltd.
http://dx.doi.org/10.1016/j.envsoft.2013.09.002

1. Introduction

Point-based environmental models are computer programs that simulate hydrology, gas flux, nutrient dynamics, and/or plant growth processes at one point on the landscape. A variety of these models have been developed for a wide range of applications in the past decades. For example, existing models can simulate agricultural crop productivity (Brisson et al., 2003; Jones et al., 2003; Keating et al., 2003), forest productivity (Battaglia and Sands, 1998; Xiao et al., 2004), soil nutrient transformations (Bandaranayake et al., 2003), greenhouse gas fluxes (Giltrap et al., 2010; Ito and Inatomi, 2012), and fluxes of water and energy at the land surface (Overgaard et al., 2006). Required inputs for model simulations commonly include meteorological forcing data, soil properties, plant growth factors, and/or land management practices. These are often spatially variable across the landscape.

However, for simplicity, many models are designed to simulate environmental processes at only one geographic point or within a land area considered homogeneous. Simulations at other locations require the adjustment of model input parameters to reflect conditions at those locations.

Spatial simulations with a point-based simulation model involve the application of the model at different spatial locations or within unique spatial zones across the landscape. In this way, the model can be used to understand the impacts of landscape spatial heterogeneity on processes of interest (Reed et al., 2012; Miller et al., 2007; Thorp et al., 2007, 2008b). Increased availability of geospatial data sets, including remote sensing images, land cover maps, digital soil surveys, crop yield maps, and vehicle-based plant or soil maps, have facilitated the implementation and evaluation of spatial model simulations. However, management and manipulation of geospatial data for conformance with model input and output formats can be complicated. Efforts have thus focused on development of geographic information systems (GIS) to handle geoprocessing tasks, store geospatial data, pass information to and from the simulation model, and visualize results spatially. Diverse GIS-based modeling systems have been developed for applications in agriculture (Engel et al., 1997; Hartkamp et al., 1999; McCarthy et al., 2010; Rao et al., 2000; Resop et al., 2012; Thorp et al., 2008a), hydrology (Chen et al., 2010; Feng et al., 2011; Hartkamp et al., 1999; Miller et al., 2007; Shen et al., 2005), and terrestrial and marine ecology (Reed et al., 2012; Roberts et al., 2010).

Many historic GIS-based modeling systems have been designed for a specific modeling application that addresses a unique environmental issue, such as precision agriculture decision support (Thorp et al., 2008a), regional crop yield analysis (Resop et al., 2012), anthropogenic noise impacts on ecosystems (Reed et al., 2012), or non-point source pollution impacts on watershed hydrology (Miller et al., 2007). Many systems are also model-specific, meaning they can only implement one simulation model. In extreme cases, only a single version of a particular model can be implemented, making the GIS system obsolete as model development proceeds forward (Thorp et al., 2008a). A model-independent GIS-based modeling system that functions independently of the specific modeling protocol would be more resilient to model version changes and would permit the implementation of different models within the same GIS framework. As a result, the spatial modeling tool would be more widely applicable to address a variety of issues related to land surface processes and land management. Generalization of geospatial modeling protocol has been an important research direction in recent years (Feng et al., 2011; Roberts et al., 2010).

Other limitations of historic GIS-based modeling systems are related to the choice of GIS software, which may be proprietary (Reed et al., 2012), computer platform dependent (Shen et al., 2005), now obsolete (Engel et al., 1997), or cost prohibitive for some users (Miller et al., 2007). Many of these issues are resolved within an open-source software development paradigm, which has recently been more widely accepted in GIS development and user communities (Steiniger and Bocher, 2009; Chen et al., 2010). One popular open-source GIS software is Quantum GIS (QGIS; www.qgis.org), a freely available user-friendly desktop GIS software that offers platform independence (Linux, Unix, Mac OS, and Windows), active user and development communities worldwide, and a "plug-in" system for extending software functionality. Experienced users can develop custom QGIS plug-ins, which can be freely distributed through the QGIS Official Repository. In addition to GIS software, developers of some environmental simulation models have also adopted open-source software development principles (Thorp et al., 2012). Thus, a logical course of action is to develop an open-source geospatial tool for spatially managing these environmental simulation models, which may themselves be open-source.

The main objectives of this paper were 1) to describe the features of Geospatial Simulation (GeoSim), a model-independent and open-source software plug-in for QGIS, and 2) to demonstrate its use with a case study involving two cropping system simulation models at a field site near Lamesa, Texas. GeoSim provides software tools for geoprocessing spatial data sets, for interfacing geospatial data with point-based environmental simulation models, for running model simulations based on conditions at unique spatial locations, and for calibrating model simulations to site-specific conditions. The software aims to generalize the union of geospatial data with point-based simulation models by providing a flexible interface for users to spatially implement their model of choice to address their environmental issue of choice. No specific model or environmental application is assumed. To demonstrate these features of GeoSim, a case study was designed to implement AquaCrop (Raes et al., 2009; Steduto et al., 2009) and the DSSAT Cropping System Model (Jones et al., 2003) for simulating site-specific seed cotton yield in response to variable irrigation and fertilizer management and variable soil texture. Specific objectives of the case study were 1) to demonstrate the utility of the software for geoprocessing spatial data sets and passing geospatial information to and from a model, 2) to highlight the model-independent nature of the software, and 3) to demonstrate site-specific model calibration and optimization.

2. Software overview

The purpose of GeoSim is to provide an interface to run point-based environmental simulation models using geospatial data contained in a QGIS database. The software requires a "base layer" polygon shapefile (the ESRI geospatial vector data file format), which is created externally to GeoSim. For example, the base layer shapefile could be derived from geographic coordinates collected at a field site, drawn based on feature boundaries in a remote sensing image, or created using tools available in other QGIS plug-ins, such as "ftools" (www.ftools.ca). Initial tasks for the user are to populate this base layer polygon shapefile with geospatial data that will be passed to the simulation model and to prepare the shapefile to receive key information returned by the model. Depending on the specific modeling requirements, this can sometimes be accomplished using tools already available in QGIS, but GeoSim does provide additional geoprocessing capability to extend the functionality of the existing tools and accomplish these data processing tasks more efficiently.

Other tools within GeoSim provide functionality for simulation control and simulation optimization. Tools for simulation control handle data flows between the GIS database and the simulation model and conduct simulations for each polygon feature within the base layer shapefile. Alternatively, simulations can be conducted for a subset of features by selecting those features in the QGIS map window or attribute table. An important characteristic of GeoSim is its ability to interface with the input and output files of any point-based simulation model. This is accomplished using "template" files to write model input files and "instruction" files to read model output files. A similar approach is used in the Parameter Estimation (PEST) model optimization software (Doherty, 2005), but the implementation is different in GeoSim. Simulation optimization tools within GeoSim extend simulation control by conducting simulations to calibrate a model, specifically by adjusting model input parameters to minimize error between model outputs and observed data for each polygon feature. A simulated annealing optimization algorithm is provided for this purpose. However, due

Table 1
Summary of software tools included in the Geospatial Simulation (GeoSim) toolbox.

Name	Function
Vector Geoprocessor	Iteratively summarizes geospatial data within base layer polygons
Raster to Vector Converter	Converts raster layers to vector layers for vector geoprocessing
Control File Creator	Develops a control file for conducting spatial model simulations
Simulation Controller	Conducts spatial model simulations and displays simulation output
Optimization File Creator	Develops an optimization file for conducting spatial model optimizations
Optimization Controller	Conducts spatial model optimizations and displays simulation and optimization output

to the model-independent nature of GeoSim, other optimization algorithms, such as PEST, could also be implemented.

Six software tools are currently provided in GeoSim: the Vector Geoprocessor, the Raster to Vector Converter, the Control File Creator, the Simulation Controller, the Optimization File Creator, and the Optimization Controller (Table 1). Each tool provides its own graphical user interface (GUI) for obtaining required inputs from the user and displaying simulation results. GeoSim can be installed by fetching the plug-in from the QGIS Official Repository. QGIS automatically handles plug-in download and installation using the tools provided under the "Plugins" menu (Fig. 1). GeoSim was developed using the Python programming language (www.python.org). Its GUIs were developed using Qt Designer (www.qt-project.org) and implemented using the PyQt libraries (www.riverbankcomputing.com). GeoSim was developed within QGIS 1.7.2, so its use with earlier versions is not suggested.

2.1. Software architecture

GeoSim conforms with the requirements for plug-in installation within the Quantum GIS environment. It has a basic object-oriented design. The top-level class, "Geospatial Simulation", is loaded when Quantum GIS starts (Fig. 2), which connects the GeoSim tools to menu items in the Quantum GIS interface (Fig. 1). Each menu item provides user access to a tool in the GeoSim toolbox (Table 1), each with an associated dialog box. When a user clicks one of these six menu items, a new class is instantiated to provide functionality to each control on the dialog box. The names for these classes end with "Dlg" on the class diagram (Fig. 2). When a user exits the dialog, the class is deconstructed. All dialog boxes are modal, so the user must complete all activities with the tool before continuing other tasks in Quantum GIS. This permits the tools to appropriately modify the base layer shapefile without interruption from the user. The classes for each dialog box each instantiate another class (not shown in Fig. 2), which instructs the program how to construct the user interface. These classes were generated automatically using Qt Designer and a Python utility (pyuic4) for converting the Qt user interface files to Python code. Additional classes were developed to read and write the information for GeoSim's control and optimization files, and the "Optimize" class was developed for managing model optimization procedures. The latter class inherits from the "Anneal" class, which is based on a simulated annealing script from Python's "SciPy" module.

2.2. Geoprocessing tools

Geoprocessing tools within GeoSim do not aim to duplicate the functionality of other QGIS plug-ins, but rather to extend them to facilitate the geoprocessing objectives required for spatial model simulations. Other QGIS tools can be used to the extent that they are helpful to prepare the base layer polygon shapefile and manage the spatial data within it. For example, ftools, mentioned previously, and the "Table Manager" plug-in are particularly useful. However, the authors found no available plug-in that was able to iteratively process geospatial data that intersected the features of a polygon shapefile. For example, to use GeoSim to manage a simulation model spatially, it may be necessary to iteratively average sensor observations that intersect each base layer polygon and store that information for later use with the model. Such tasks can be accomplished with the Vector Geoprocessor included with GeoSim (Table 1). By selecting 1) the base layer polygon shapefile, 2) another vector layer containing the

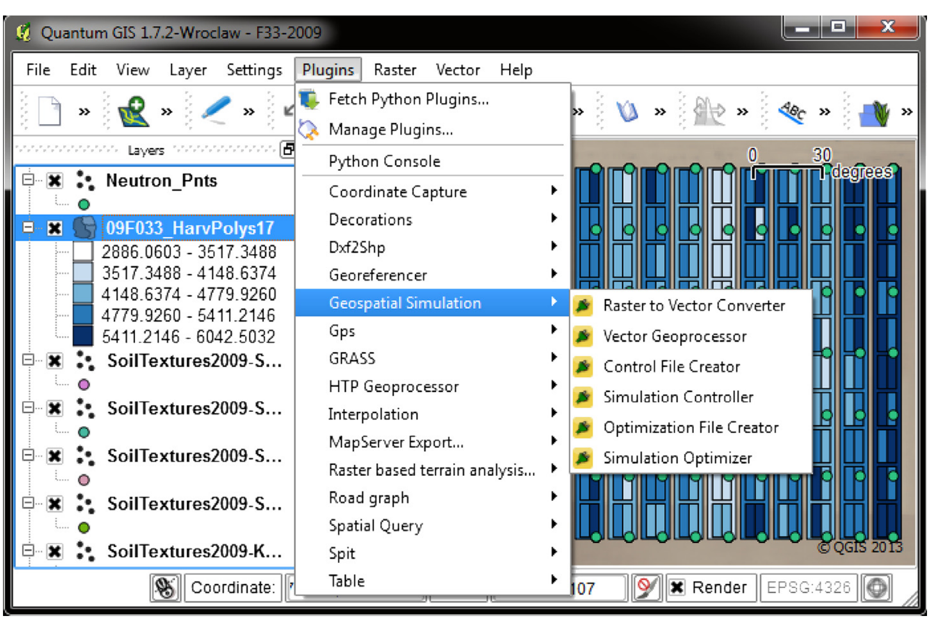

Fig. 1. The Quantum GIS user interface with the Geospatial Simulation plug-in installed, as shown in the "Plugins" menu.

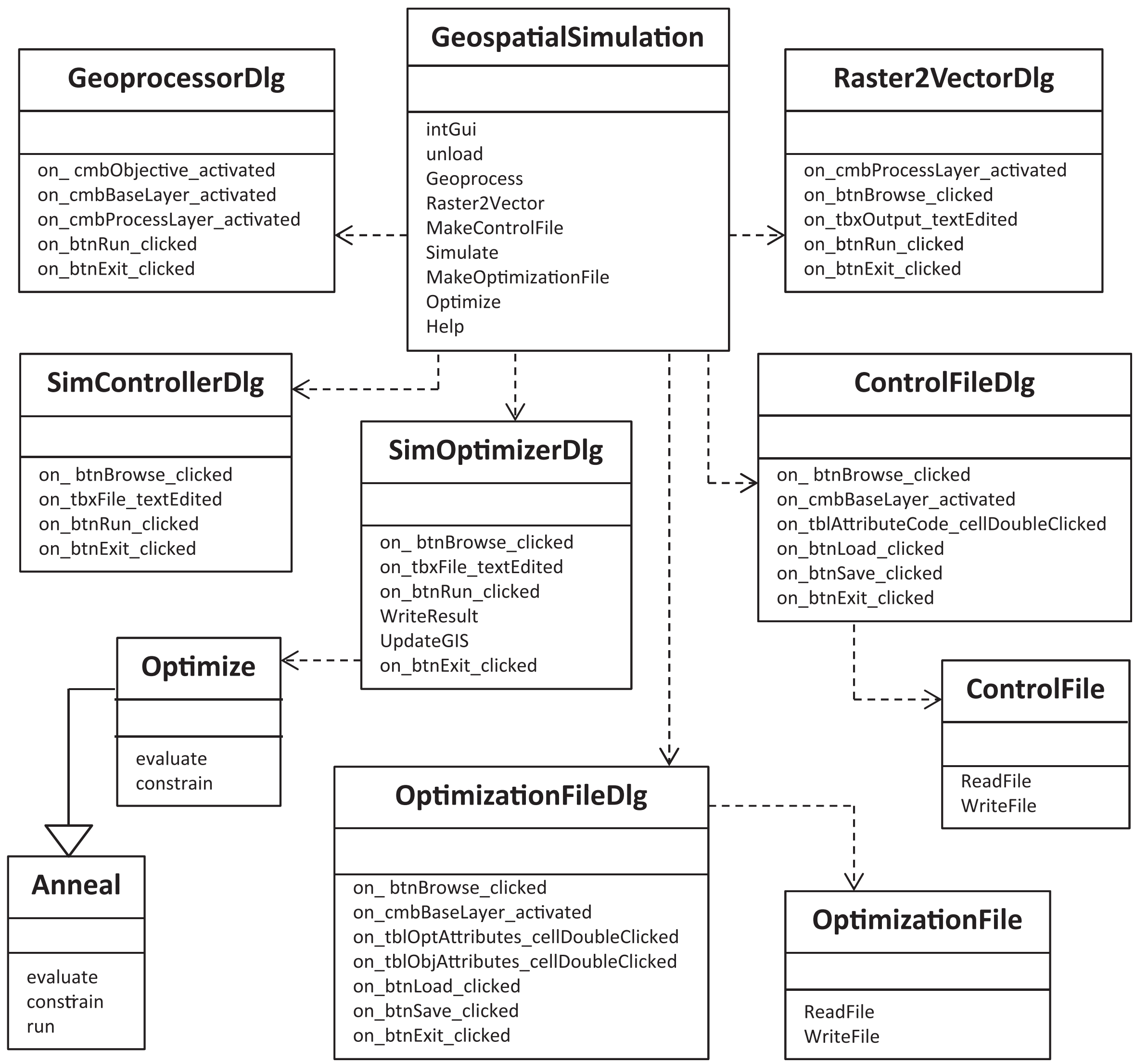

Fig. 2. A class diagram for the Geospatial Simulation (GeoSim) plug-in. The methods for each class are given, but the attributes are excluded for simplicity.

data to be processed, and 3) the attributes of interest in the layer to be processed, the tool will iteratively process the data contained within the process layer and append the results to the base layer (Fig. 3). A variety of processing objectives can be accomplished, such as finding the mean, median, maximum, or minimum of data points. A limitation of the Vector Geoprocessor is that it processes only vector data layers (points and polygons) and does not handle raster data layers. For this reason, GeoSim includes the Raster to Vector Converter (Table 1), which can be used to convert raster layers, such as interpolated soil texture maps or remote sensing images, to vector layers. The Vector Geoprocessor and other vector data processing tools within QGIS can then be used to process the data.

2.3. Simulation control

GeoSim is designed to manage spatial simulations for a point-based model that has been previously compiled as a separate executable file and that reads model input parameters and writes output data to American Standard Code for Information Interchange (ASCII) data files. It automates simulations by passing geospatial data from unique polygon features to the model input files, and it will similarly pass key model outputs back to the GIS database. It accomplishes this using "template" files to interface with the model input files and "instruction" files to interface with the model output files. A "control" file instructs the GIS how to utilize the template and instruction files to conduct simulations (Fig. 4).

A template file (*.gst file extension) must be created for each model input file that receives spatial data from the GIS. The template file is essentially a replicate of the actual input file that the model will read. Therefore, template files are highly specific to the model to be implemented, and GeoSim provides no tools for creating template files. For each geospatial data attribute that must be passed to the model, the user must assign a "unique code" or a unique combination of characters (letters, numerals, symbols, and spaces). The unique codes are then included in the template file at the location(s) where GeoSim should write the data values. A unique code should not appear anywhere else in any of the model input files other than where GIS data is to be written. Prior to running the model for a polygon feature, GeoSim will search the template files for each unique code and overwrite it with the appropriate data value for that polygon feature. The control file (discussed below) provides the relationship between each unique

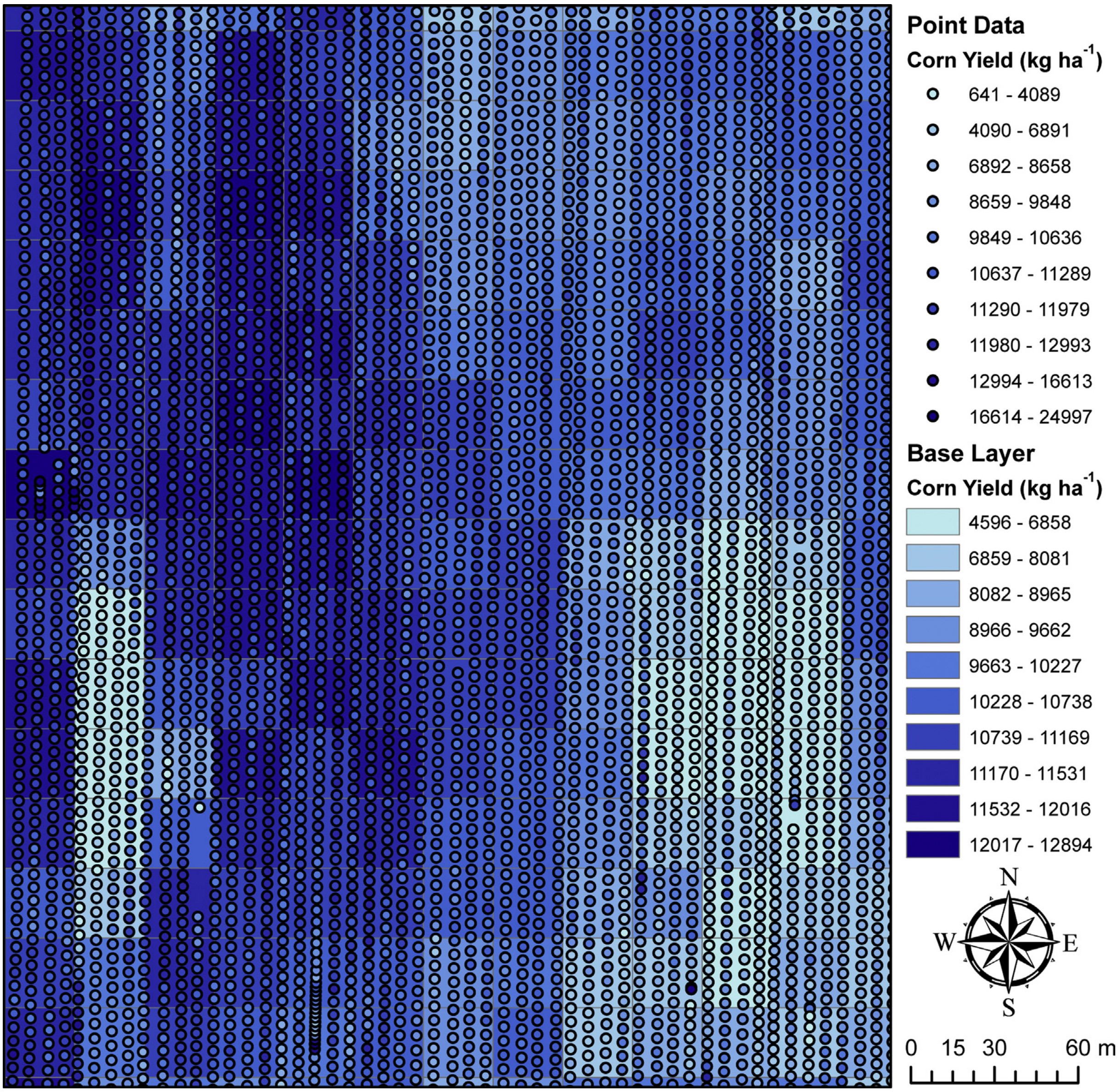

Fig. 3. The Vector Geoprocessor tool within Geospatial Simulation (GeoSim) can be used to iteratively process geospatial data (represented as points or polygons) within the polygons of the base layer shapefile. This example shows corn yield data points from a yield monitor on a grain harvester, which were iteratively averaged within and appended to the 25 m^2 zones of the base layer polygon shapefile. The data can then be used for geospatial model simulations within each of the 25 m^2 zones.

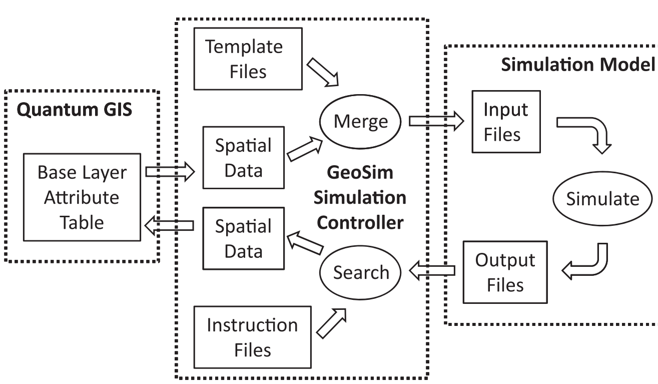

Fig. 4. The Simulation Controller tool within Geospatial Simulation (GeoSim) uses template files to interface the geospatial data from the base layer polygon shapefile with the simulation model input files. It uses instruction files to search model output files for the geospatial data to be returned to the base layer.

code and its respective attribute in the GIS database. Utilizing the information provided in the template files and the control file, GeoSim will create unique model input files as simulations are iteratively conducted for each polygon feature in the base layer shapefile.

An instruction file (*.gsi file extension) must be created for each model output file that contains spatial data to be passed back to the GIS database. The instruction file tells the GIS how to read the model output file, extract the appropriate data values, and assign the data to the appropriate attributes in the GIS database. Three commands are available to provide the instruction: “Plus”, “Find”, and “Get”. Users must develop the instruction file using these commands as necessary to search the model output file and retrieve the appropriate data values. The Plus command is used to move forward a given number of lines in the file. The Find command is used to search for a given set of unique characters in the file. These two commands, Plus and Find, are used to determine the appropriate line in the file. Then, the Get command is used to acquire the characters between two specified columns. These characters are then converted to the appropriate data type, as specified in the control file, and stored appropriately in the GIS database.

The control file (*.gsc file extension) instructs the GIS how to use the template and instruction files to conduct model simulations for polygon features in the base layer shapefile. Whereas the formats of the template and instruction files are unique to the simulation model, the format of the control file is specific to GeoSim. Therefore, GeoSim provides a tool, the Control File Creator (Table 1), for developing the control file. Seven sections of information are required: 1) the path to the model directory where all template files, instruction files, model input and output files, and the model executable file exist, 2) the name of the base layer polygon shapefile, 3) the template file names with their associated model input file names, 4) the unique codes with their corresponding attribute in the GIS database, 5) the instruction file names with their associated model output file names, 6) the data type (Real, Integer, or String) for each attribute to be updated from model output data, and 7) a command line character string to run the simulation.

After setting up the template files, instruction files, and control file, the Simulation Controller tool within GeoSim (Table 1) can be used to conduct the simulations for all polygon features in the base layer polygon shapefile (Fig. 4). Alternatively, the tool will also run simulations only for selected features in the base layer. Further information and examples for developing template, instruction, and control files are provided in the GeoSim user's manual, which is distributed with the plug-in.

2.4. Simulation optimization

It is often necessary to calibrate a simulation model uniquely for the conditions at a given location. GeoSim provides an optimization tool for this purpose. The optimization algorithm utilizes the control file (and thus the instruction files and template files as well) to conduct iterative simulations for each polygon feature with the objective to adjust model input parameters to minimize error between observed and simulated quantities. Users must prepare the observed data within the base layer polygon shapefile using the geoprocessing tools described previously. Since model optimization requires a control file to conduct simulations, it is necessary to first develop the control file and set up GeoSim for spatial simulations before optimizing the model.

The optimization file (*.gso file extension) instructs the GIS how to conduct model optimizations, and GeoSim provides the Optimization File Creator for developing this file (Table 1). Five sections of information are required: 1) the path to the control file, 2) the name of the base layer polygon shapefile, 3) the attribute names of the model parameters to be optimized with their initial values and upper and lower bounds, 4) the attribute names of the simulated and observed quantities to be compared, and 5) the values of eleven parameters that govern the performance of the built-in optimization routine. Any model parameter to be optimized must also have a unique code specified in the control and template files. Any simulated value used by the optimizer must be specified in the control and instruction files.

GeoSim includes a "simulated annealing" optimization algorithm based on the algorithm developed for Python's "SciPy" module (www.scipy.org). However, the algorithm has been largely modified for use with process-based simulation models in GeoSim. Simulated annealing is a global search algorithm that mimics the annealing process in metallurgy. Because GeoSim is model independent, users also have the option of implementing their own optimization algorithms. Custom optimizers can be called using the Simulation Controller tool, similar to its use for simulation models. In this case, the custom optimizer must handle iterative model simulations required for optimization, and GeoSim handles only the iterations over polygon features.

When using the optimization algorithm provided with GeoSim, the Simulation Optimizer (Table 1) can be implemented to conduct model optimizations for all polygon features in the base layer shapefile. Alternatively, the tool will run optimizations only for selected features in the base layer polygon shapefile. The simplest approach is to conduct optimizations independently for each polygon feature. In this case, the algorithm iterates over the polygon features (or selected features) while conducting optimizations separately for each feature. However, the algorithm can also handle grouped optimizations, where several polygon features are considered together. This is useful for optimizing a model parameter for several polygon features, while minimizing error between simulated and observed quantities across those spatial features. However, the optimization file must be properly set up for this type of optimization, as discussed in the GeoSim user's manual.

3. Case study

To demonstrate the features of GeoSim, a case study was designed to implement two point-based cropping system models for simulating site-specific seed cotton yield at a field site near Lamesa, Texas. Cropping system simulation models synthesize current knowledge of cropping system processes and utilize mass balance principles to simulate the carbon, nitrogen, and hydrologic processes and transformations that occur within a cropping system (Brisson et al., 2003; Jones et al., 2003; Steduto et al., 2009). In the agronomic sciences, cropping system models are important and useful tools for understanding the impacts of soil properties, meteorology, management practices, and cultivar selection on crop growth, development, and yield. Most are point-based models with one-dimensional representations of the soil profile and soil water balance. However, crop growth and yield responses to the environment are known to be spatially and temporally variable (Jaynes and Colvin, 1997). In the past decade, cropping system models have been used increasingly to investigate spatiotemporal variability in crop yield patterns and to develop crop management plans that optimize crop production for reduced environmental impact (Batchelor et al., 2002; McKinion et al., 2001; Thorp et al., 2006, 2007, 2008a).

3.1. Field site and experimental design

Bronson et al. (2006) described the field site and an agronomic experiment used as the basis for this case study. Briefly, the purpose of the field study was to test cotton yield responses to variable irrigation and nitrogen fertilizer management with the hypothesis that crop water and nutrient requirements would vary according to soil texture and landscape position (Fig. 5). The 14 ha field study was conducted under a 48 ha center pivot irrigation system located near the town of Lamesa in west Texas (32.77° N, 101.94° W). Soil samples were collected at 135 locations across the study area to quantify soil texture properties and initial soil nitrate. In May of 2002, 2003, and 2004, cotton was planted at the field site. Different irrigation sprinkler nozzles were installed along the length of the center pivot lateral to apply irrigation water at three different rates. Three nitrogen fertilizer treatments were tested at each irrigation level, including zero nitrogen, a blanket rate of nitrogen, and a variable-rate nitrogen application based on an interpolated map of crop nitrogen need calculated from pre-plant soil nitrate-nitrogen status. Nitrogen treatments were replicated three times within each of three replicates of the irrigation treatments. Variable crop

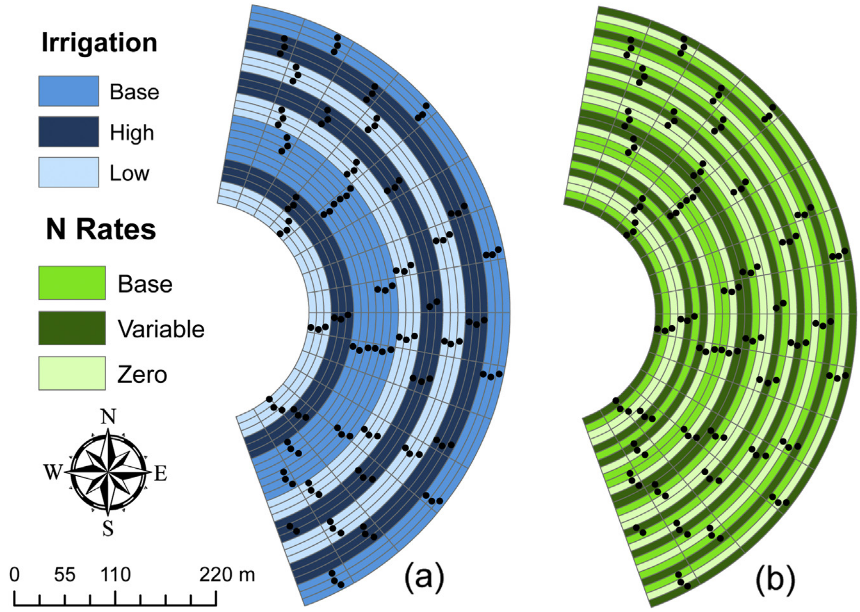

Fig. 5. Experimental design for a 14 ha agronomic study near the city of Lamesa in west Texas, where site-specific seed cotton yield was tested in response to a) variable irrigation management and b) variable nitrogen fertilizer management. Soil texture variability was characterized from soil samples at 135 sampling points (black circles) across the study area.

management practices and soil properties resulted in cotton yield variability across the study area, which was measured site-specifically each season using a cotton stripper equipped with an optical yield monitoring system. The site-specific nature of this field study complicates data analysis using point-based cropping system models, because the geospatial data layers must be summarized to satisfy model input requirements for homogeneous land areas. GeoSim was designed to perform the required geoprocessing tasks and to interface the resulting data with model file formats.

3.2. Cropping system models

Model independence was a primary design consideration to make GeoSim versatile for a variety of models and modeling applications. To demonstrate this feature, data from the field study was analyzed using two common cropping system models: the AquaCrop model (Raes et al., 2009; Steduto et al., 2009) and the DSSAT Cropping System Model (Jones et al., 2003). Both models were designed to simulate crop responses to management and environmental conditions; however, they vary substantially in detail and complexity and use different approaches to simulate some processes.

3.2.1. AquaCrop

Conceptual diagrams of the AquaCrop model (ver. 4.0) are provided by Raes et al. (2009) and Steduto et al. (2009). Briefly, AquaCrop simulates the soil water balance with the overall aim to calculate actual evapotranspiration, and crop biomass is accumulated over time as a function of water transpired. The model operates on a daily time step. Heat unit accumulation defines crop phenology. Crop development and biomass accumulation drive a canopy cover simulation, which partitions evapotranspiration into separate fluxes from crop and soil. Yield is simulated simply as a fraction of biomass accumulation. Inputs for crop and irrigation management are required by the model. Other important crop parameters include the length of the flowering period and the number of calendar days from sowing to key growth stages. Atmospheric conditions are characterized using daily inputs for maximum and minimum temperature and reference evapotranspiration (ET_0). Users must calculate ET_0 explicitly using the Penman-Monteith equation and FAO-56 guidelines (Allen et al., 1998). Soils are characterized by their water retention and conductivity characteristics, saturated hydraulic conductivity, and initial soil water status. The model simulates plant stress effects from deficit soil water conditions, which can feedback on plant growth by restricting canopy cover expansion, closing stomata to reduce transpiration, causing early plant senescence, and/or adjusting the harvest index. AquaCrop does not explicitly simulate a soil nitrogen balance.

3.2.2. DSSAT-CSM

Conceptual diagrams of the DSSAT Cropping System Model (CSM; ver. 4.5.1.005) are provided by Jones et al. (2003). Briefly, the model simulates carbon, nitrogen, and hydrologic processes and transformations that occur within a cropping system. To simulate cotton production for this case study, the CSM-CROPGRO-Cotton module was implemented. The CSM calculates cropping system processes within a homogeneous area on a daily time step, and certain subprocesses are computed hourly. Crop development proceeds through a series of growth stages based on heat unit accumulation from planting to harvest. Canopy photosynthesis is computed using leaf-level photosynthesis equations with a hedgerow model to account for row structure and canopy envelope. Assimilated carbon is partitioned to various plant components. The model simulates seed cotton yield, which is the sum of lint and seed components of the cotton boll. Simulated plant growth responds to variation in management practices, cultivar selection, soil properties, and meteorological conditions. Inputs for crop management, irrigation management, and fertilizer management are required by the model. Cultivar parameters define day length sensitivity, heat units needed to progress through growth stages, and growth potentials for specific plant parts. Soils are defined by their water retention and conductivity characteristics, bulk density, and initial conditions for water, inorganic nitrogen, and organic carbon. The FAO-56 option was implemented for evapotranspiration calculations, which requires daily inputs for minimum and maximum temperature, solar radiation, wind speed, and dew point temperature. The model simulates plant stress effects from deficit and excess soil water conditions and from deficit soil nitrogen conditions, which feedback on the daily plant growth simulation.

3.2.3. Model parameterization

Simulations with both models were conducted from two weeks before planting to the harvest date for each of the three crop growing seasons. All crop management inputs for planting, irrigation, and nitrogen fertilizer applications were specified as performed during the field investigation. Initial soil water contents were set at the drained upper limit, due to plentiful pre-season rainfall and irrigation prior to the start of simulations. Soil water retention and hydraulic parameters were specified based on a textural analysis of soil samples at the site. The Rosetta pedotransfer functions (Schaap et al., 2001) were used to calculate the required soil input parameters from textural information. Meteorological data were obtained from a West Texas Mesonet (WTM; http://www.mesonet.ttu.edu) station approximately 8 km from the field site. Additional rainfall data, although incomplete, were available at the study site to adjust WTM estimates if needed. All site-specific parameters, including soil properties, initial soil water content, irrigation rates, initial soil nitrate concentration (DSSAT-CSM only), and nitrogen fertilizer rates (DSSAT-CSM only), were managed by GeoSim.

Both models required adjustments to crop parameters to improve simulated crop growth and development as compared to field observations. For AquaCrop, the calendar days from sowing to key growth stages were adjusted to match observed values. The durations from sowing to emergence and from sowing to first flower were approximately 6 and 60 days in all growing seasons, respectively. Due to the indeterminate growth pattern of cotton, time to maturity was set based on the date of desiccant application. Maximum root depth was adjusted to 1.8 m. Maximum harvest index was adjusted to 35 for cotton in 2002 and 2003. However, in 2004, a different cotton variety was planted, requiring a maximum harvest index of 27. For the DSSAT-CSM, photothermal days from emergence to first flower (EM-FL), from first flower to first square (FL-SH), from first flower to first seed (FL-SD), and from first seed to physiological maturity (SD-PM) were adjusted from default values to 39.0, 4.0, 8.0, and 43.0, respectively. Also, the maximum fraction of daily growth partitioned to bolls (XFRT) was adjusted to 0.84.

3.3. Application of GeoSim

Due to the circular plant rows often used with center pivot irrigation, experimental treatments were arc-shaped with increasingly longer plot lengths from the pivot center to the outer circumference (Fig. 5). Vector processing tools within Quantum GIS were used to create a base layer polygon shapefile that delineated treatment plots. Plots were 8 rows wide with row spacing of 1.02 m and were radially divided into 10° arcs. This generated a total of 405 base layer polygons for the analysis. Treatment names were manually added to the base layer attribute table.

Using data from the 135 soil sampling points, the calculations of soil water retention and hydraulic properties from Rosetta, the observations of soil organic carbon, and the observations of initial soil nitrate at four soil profile depths (15, 30, 60, and 90 cm) were interpolated across the study area using universal kriging. The resulting raster layers for each soil property were polygonized using the Raster to Vector Converter (Table 1). The Vector Geoprocessor was then used to average the interpolated soil property values within each base layer polygon and append the data to the base layer attribute table. Irrigation rates were added by joining to the attribute table a data table with the appropriate rates for each irrigation treatment. Nitrogen fertilizer application rates were processed from as-applied maps generated by the fertilizer application system. Also, the yield monitor provided maps of observed spatial yield variability across the site. Both of these types of maps were point-based, so the Vector Geoprocessor was used to average the data values within each base layer polygon and append the data to the base layer attribute table. This exercise demonstrated the utility of GeoSim for summarizing multiple geospatial data layers for subsequent spatial modeling analysis.

Separate control files were created for the AquaCrop and DSSAT-CSM models using the Control File Creator (Table 1). For DSSAT-CSM, GeoSim passed the soil property data to the DSSAT-CSM soil file (TX.SOL) and management data and initial conditions were passed to separate management files (*.COX) for each growing season. For AquaCrop, GeoSim passed soil property data to the AquaCrop soil file (SANDYLOAM.SOL), the drained upper limit value to initial soil water file (TXLM.SW0), and irrigation data to separate management files (*.IRR) for each growing season. GeoSim read simulated seed cotton yield values from the appropriate output files for each model and returned these data to the base layer attribute table. Appropriate template and instruction files were created for each model to facilitate data sharing between the GIS database and the crop model files. After creation of these files, the Simulation Controller (Table 1) was used to automate AquaCrop and DSSAT-CSM simulations for each of the 405 base layer polygons for each of the three growing seasons. Error statistics between measured and simulated yield were calculated within the base layer attribute table. Model run times were also recorded.

In a final effort, both the AquaCrop and DSSAT-CSM models were spatially optimized. The Optimization File Creator (Table 1) was used to create an optimization file for each model, and the Optimization Controller was then used to adjust the drained upper limit soil parameter for each model with the objective to minimize error between measured and simulated seed cotton yield over the three growing seasons. The drained upper limit parameter was adjusted between 7% and 15%, which were the upper and lower bounds of observed drained upper limit at the site.

4. Results

To demonstrate why GeoSim was necessary for this analysis, it is helpful to momentarily consider an alternative data processing and simulation protocol, in which GeoSim was not implemented. A major issue resolved by GeoSim is scaling and collocation of geospatial information from multiple data sources to a common map, represented by the base layer shapefile (Fig. 5). This data processing step is essential for model parameterization, because 1) models are often designed to represent processes within homogeneous land areas and 2) field data is often collected at differing spatial scales and spatial locations. Demonstrating the utility of the Vector Geoprocessor within GeoSim, 5592 data points for cotton yield could be summarized within the 405 polygon features in only 3.8 s. Without this tool, a user would be required to use existing Quantum GIS tools to manually 1) find the intersection of yield data points with a polygon feature, 2) calculate the mean of the selected points, 3) record the mean yield value in an appropriate location, and 4) repeat the process for all 405 polygon features and for all other geospatial data layers. The time required for such activities would likely exceed hours or even days, and the tedium would likely lead to errors. Instead, GeoSim provides the Vector Geoprocessor to perform these tasks automatically.

If GeoSim were not used, the simulation protocol for this analysis would have to be radically different. For each of the 405 AquaCrop simulations, GeoSim swapped 80 parameters to the model input files and 3 yield values from the model output files: a total of 33,615 data values for one set of simulations. For DSSAT-CSM, 107 model input parameters and 3 yield values were swapped: a total of 44,550 data values for 405 simulations across the site. If GeoSim were not used, these data would have to be added to the model input files using another approach. Software provided by the model developers permits users to develop model input files by manually entering the data into graphical user interfaces. Alternatively, users can create model input files using a text editor. Both of these approaches become less practical as the number of parameters and the number of simulations grow large, which is typical when models are implemented repetitively over space. GeoSim expedites such simulations by providing a mechanism to swap data from the Quantum GIS to the appropriate locations in the model input files (Fig. 4). Whereas additional time is required to set up GeoSim, its parameter swapping capability can greatly simplify the model set up.

Given the impracticality of manual data handling for the simulation analyses conducted herein, it is difficult to compare the simulation results obtained through GeoSim with that from standard simulation methods. However, since GeoSim does not change any model code or adjust any model state variables and simply interacts with the model's native set of input and output files, simulation results are dependent only on GeoSim's ability to swap data accurately. This is a function of 1) the GeoSim source code and 2) the user-defined instructions for parameter swapping as given in the control, template, and instruction files. For a random sample of 10 zones out of the 405 in the above analysis, a comparison of the drained upper limit (DUL) parameter values from the Quantum GIS database to that written to the model input files showed a RRMSE of 0.28% for AquaCrop and 0.24% for DSSAT-CSM. The discrepancy was mainly due to rounding error, since the model input file formats would not receive DUL data with the same decimal precision as that stored in the GIS database. The model developers have probably designed the model to read inputs with an appropriate level of decimal precision, thus this discrepancy likely had little impact on simulation output. A comparison of the 2002 cotton yield values from the model output files with that transferred to the GIS database demonstrated no differences. Thus, the results showed that GeoSim was able to accurately swap data to and from the files of two independent cropping system models. Further testing with additional models and for other modeling applications will demonstrate if any adjustments are warranted for GeoSim to handle special cases unforeseen in the analysis presented herein.

Use of GeoSim does require additional overhead, in terms of simulation duration, to swap data between the GIS database and model input and output files. Conducting the 405 simulations across the study site using GeoSim data swapping required 112 s and 398 s with the AquaCrop and DSSAT-CSM models, respectively. Simulation tests outside the GIS environment, where 405 simulations were conducted without adjusting any parameters, demonstrated run durations of 36 s and 151 s for the AquaCrop and DSSAT-CSM models, respectively. Thus, to take advantage of GeoSim data swapping capability, simulations were slowed by a factor of 3.1 for AquaCrop and 2.6 for DSSAT-CSM. The impact was more severe for AquaCrop, because its relative simplicity makes it a faster model than DSSAT-CSM. For general use, such increases in simulation duration may not be troublesome; however, for applications requiring large numbers of simulations, productivity can be impacted. Future work will remedy this issue by identifying opportunities to improve computational efficiency in the existing GeoSim code and by exploring the potential for parallelizing simulations on multi-core desktop computers (Bryan, 2013).

Further demonstration of GeoSim can be found in the results of the simulation examples, since generating such results would be highly impractical without the tool. The AquaCrop and DSSAT-CSM models were able to simulate spatial seed cotton yield for the three growing seasons with relative root mean squared errors (RRMSE) of 16.8% and 13.3%, respectively (Fig. 6). AquaCrop underestimated yield for many zones in 2003, a year with minimal rainfall during the reproductive growth phase. The DSSAT-CSM also underestimated yield for many zones in 2003, but not as poorly as AquaCrop. Generally, DSSAT-CSM was better able to simulate yield variability in an individual growing season, likely because the model simulated yield loss resulting from both water and nitrogen deficit. GeoSim provided the flexibility to conduct spatial simulations with both of these models, permitting comparisons of spatial yield simulations that could likely not be achieved with other tools.

Maps of the relative root mean squared error between measured and simulated yield for the three growing seasons demonstrated spatial patterns in modeling error. For the AquaCrop model, there was visual evidence of spatial clustering in error patterns that were related to water management (Fig. 7a). Clusters of higher modeling error corresponded to the treatment plots with the lowest irrigation rate. Thus, the underestimated yield simulation by AquaCrop in 2003 (Fig. 6a) was likely related to error in the water balance simulation for the lowest irrigation treatment. Clusters of modeling error based on management practices were less evident for DSSAT-CSM (Fig. 7b). The northern and south central portions of the field were simulated quite well with relatively low error. This could mean that soil properties were well characterized in this area of the field. Higher error in the center portion of the field is likely due to topography. This area of the field was lower in elevation and likely

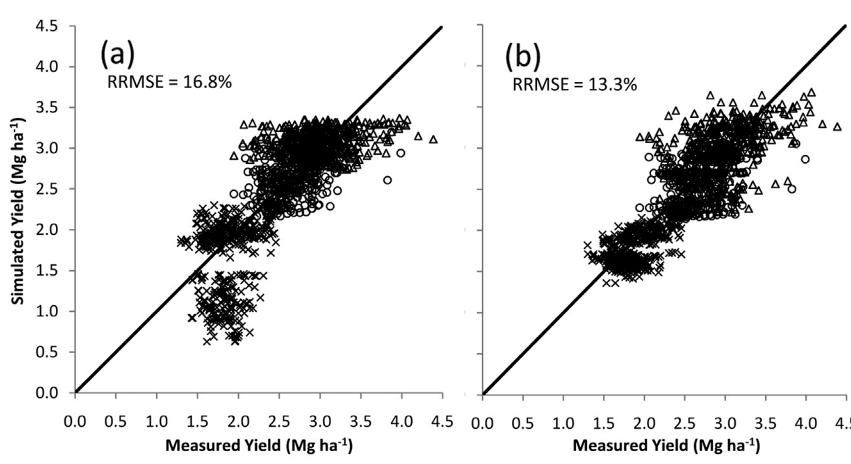

Fig. 6. Simulated versus measured seed cotton yield for three growing seasons using the a) AquaCrop model and b) the DSSAT Cropping System Model.

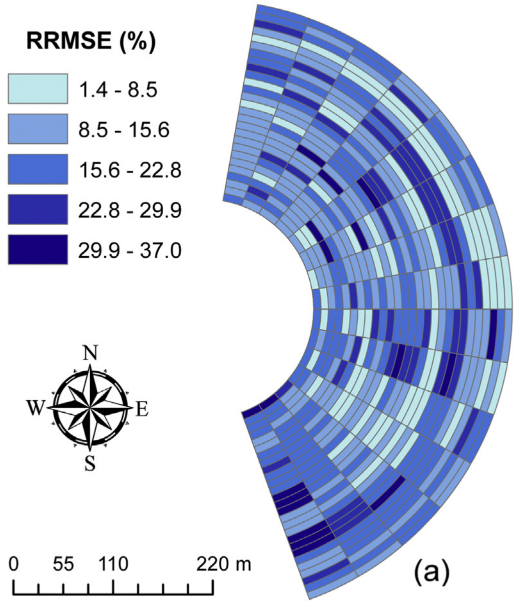

Fig. 7. Spatial distribution of relative root mean squared error (RRMSE) between simulated and measured seed cotton yield for three growing seasons using the a) AquaCrop model and b) the DSSAT Cropping System Model.

experienced run-on from other areas of the field (Bronson et al., 2006), which was not simulated by either model. Thorp et al. (2007) provided additional examples on the use of geospatial analysis to understand the underlying causes for spatial modeling error. However, their geospatial analysis was conducted separately from their simulation analysis. GeoSim provides the capability to conduct both the geospatial and the simulation analyses within the GIS environment.

By spatially optimizing the drained upper limit soil parameter for each model, the RRMSEs between measured and simulated seed cotton yield over three growing seasons were reduced to 9.7% and 8.3% for the AquaCrop and DSSAT-CSM models, respectively (Fig. 8). Adjusting the drained upper limit for AquaCrop radically improved the model's ability to simulate yield in the 2003 growing season (Fig. 8a), which was previously underestimated using the original drained upper limit estimates from kriged soil property maps (Fig. 6a). This result demonstrates the sensitivity of cropping system models to the drained upper limit parameter, especially for simulating crop yield in water limited environments. The result also demonstrates the ability of GeoSim to perform parameter adjustments to improve spatial model simulations. Although it is beyond the scope of this paper, the cross validation techniques developed by Thorp et al. (2007) could easily be implemented within GeoSim and Quantum GIS for further evaluation of these models.

5. Discussion

Given the increase in availability of geospatial data sets from a wide variety of data collection platforms and databases, there is a need for geoprocessing tools that summarize geospatial data within land units for subsequent spatial analysis with environmental simulation models. Geographic information systems are a logical tool for such analyses, because they can effectively summarize data from diverse data sources and tie these data streams together for specific land units. This is a critical first step to using the data for spatial modeling analyses. The case study presented herein

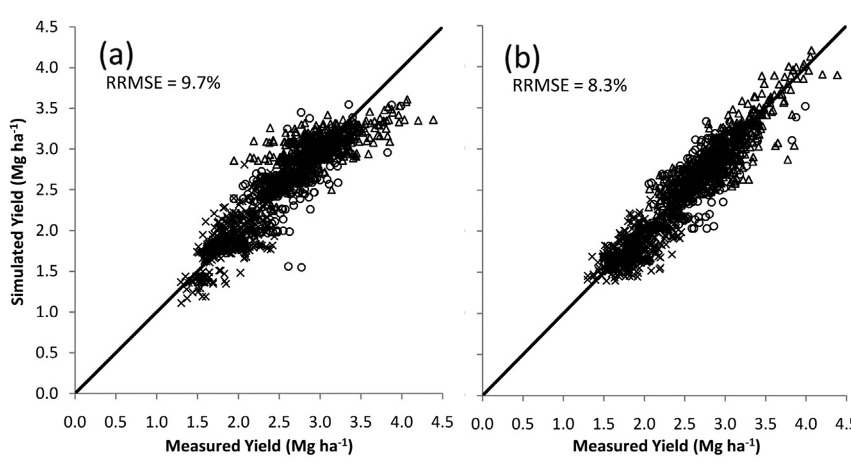

Fig. 8. Simulated versus measured spatial seed cotton yield for three growing seasons after spatially optimizing the drained upper limit soil parameter for the a) AquaCrop model and b) the DSSAT Cropping System Model.

demonstrates this capability of the open-source GeoSim software. Geospatial data from diverse sources, including soil property maps, nitrogen fertilizer application maps, and crop yield maps, were all summarized within the base layer polygons that delineated management zones at the field site. Data were summarized in a format that would permit unique realizations of the model for each spatial land unit, depending on site-specific conditions. GeoSim provides open-source tools for easily performing the necessary geospatial data processing tasks and conducting spatial model simulations using the geoprocessing results.

A main advantage of GeoSim is that it was designed to function independently from the specific modeling tool. This permits the software to be implemented with a variety of simulation models to address a variety of environmental issues. It also decouples the development of the geospatial tools from the simulation tools, such that updates to one tool do not automatically render the other incompatible. To demonstrate this feature, GeoSim was used to control two common cropping system models for the same geospatial data sets at a field site. Although the models had somewhat different data input requirements, GeoSim was able to manipulate the input data for conformance to model requirements and conduct 405 simulations across the field site with each model. An added advantage is that the geospatial data layers, processed model input data, and simulation results can be stored, visualized, and compared using the tools already available in the GIS software. Use of GeoSim requires basic knowledge of GIS software, because it was designed to be used in combination with existing Quantum GIS tools. Its flexibility permits modelers with diverse objectives to combine geospatial data processing with their modeling analyses.

To be used with GeoSim, a simulation model must satisfy three basic requirements. First, it must be point-based, meaning it is designed to simulate processes at one point on the landscape or within a land area that is considered homogeneous. Second, it must read input parameters and write output data to ASCII data files. Third, it must be callable from the command line. If these conditions are satisfied, GeoSim can likely be used to manage model simulations at multiple spatial locations.

The case study demonstrates an implementation of GeoSim using data from an agronomic field study and models designed for the agricultural sciences. However, GeoSim does generalize the software design concepts required for site-specific, point-based model simulations. For this reason, GeoSim likely has wide applicability beyond the agronomic sciences and is likely useful for any point-based environmental modeling application that makes heavy use of geospatial information. For example, gross primary productivity of forests could be simulated using remote sensing data inputs (Xiao et al., 2004) or greenhouse gas fluxes could be simulated using information from land cover maps (Ito and Inatomi, 2012). Any point-based model that communicates via ASCII file I/O and requires geospatial inputs to characterize the land surface can likely be implemented within GeoSim.

An important limitation of GeoSim is that each polygon feature in the base layer shapefile is currently considered independently from its neighbors. Although polygon boundaries were shared in the case study (Fig. 5), GeoSim does not account for spatial connectivity between the base layer polygons. Polygon boundaries could be separated in space or could potentially overlap if needed. As a result, GeoSim was not designed for simulations where model I/O is derived from a network of polygons with known connectivity. A typical example would be simulations that route water laterally across the landscape. Other tools are already available for this purpose (Arnold et al., 2012). However, if a point-based simulation model did require as input the characteristics of neighboring polygons, a new GeoSim tool could feasibly be designed to provide that functionality. It would require that the user establish polygon network connectivity, so that the simulations could be conducted in the proper order.

6. Conclusions

Geospatial Simulation (GeoSim) is a novel geospatial toolbox that facilitates spatial simulations with point-based environmental models using geospatial data in a Quantum GIS database. GeoSim advances science by generalizing the algorithms required for geospatial simulation modeling and by providing an open-source alternative for such simulations. The authors know of no other tool that accomplishes both of these tasks simultaneously, making GeoSim a unique tool with broad potential applicability. Generalization of the spatial modeling algorithms makes GeoSim flexible for use with many types of simulations models and for many environmental applications. Although this feature is demonstrated only for two models used in the agricultural sciences, GeoSim is likely applicable across many scientific disciplines. Thus, future efforts to broadly implement GeoSim with other simulation models and for other spatial modeling applications are highly encouraged. Because GeoSim generalizes the algorithms required for spatial modeling, users will likely discover new and exciting ways to implement the software for their own unique modeling applications. There may even be strategies for using GeoSim that have not yet been conceived. The authors leave this to the creativity of GeoSim users. Also, because GeoSim was developed within the open-source software paradigm, users may have ideas for improving or expanding the source code. This is a main advantage of open-source software development, and the authors are open to future collaborative efforts with anyone who finds GeoSim useful but somehow lacking. Openness and the free exchange of ideas are central to the scientific pursuit, and that is why GeoSim has been developed and distributed as open-source software. GeoSim was designed with generality, flexibility, and openness in mind, and users are encouraged to creatively explore alternative ways to implement the software for their specific geospatial modeling objectives.

References

Allen, R.G., Pereira, L.S., Raes, D., Smith, M., 1998. Crop Evapotranspiration: Guidelines for Computing Crop Water Requirements (FAO Irrigation and Drainage Paper 56). Food and Agriculture Organization of the United Nations, Rome, Italy.

Arnold, J., Moriasi, D., Gassman, P., Abbaspour, K., White, M., Srinivasan, R., Santhi, C., Harmel, R., Van Griensven, A., Van Liew, M., Kannan, N., Jha, M., 2012. SWAT: model use, calibration, and validation. Trans. ASABE 55 (4), 1491–1508.

Bandaranayake, W., Qian, Y., Parton, W., Ojima, D., Follett, R., 2003. Estimation of soil organic carbon changes in turfgrass systems using the CENTURY model. Agron. J. 95 (3), 558–563.

Batchelor, W.D., Basso, B., Paz, J.O., 2002. Examples of strategies to analyze spatial and temporal yield variability using crop models. Eur. J. Agron. 18 (2), 141–158.

Battaglia, M., Sands, P., 1998. Process-based forest productivity models and their application in forest management. For. Ecol. Manag. 102 (1), 13–32.

Brisson, N., Gary, C., Justes, E., Roche, R., Mary, B., Ripoche, D., Zimmer, D., Sierra, J., Bertuzzi, P., Burger, P., Bussière, F., Cabidoche, Y., Cellier, P., Debaeke, P., Gaudillère, J., Hénault, C., Maraux, F., Seguin, B., Sinoquet, H., 2003. An overview of the crop model STICS. Eur. J. Agron. 18 (3–4), 309–332.

Bronson, K., Booker, J., Bordovsky, J., Keeling, J., Wheeler, T., Boman, R., Parajulee, M., Segarra, E., Nichols, R., 2006. Site-specific irrigation and nitrogen management for cotton production in the Southern High Plains. Agron. J. 98 (1), 212–219.

Bryan, B., 2013. High-performance computing tools for the integrated assessment and modelling of social-ecological systems. Environ. Model. Softw. 39, 295–303.

Chen, D.Y., Shams, S., Carmona Moreno, C., Leone, A., 2010. Assessment of open source GIS software for water resources management in developing countries. J. Hydro-environ. Res. 4 (3), 253–264.

Doherty, J., 2005. PEST: Model-Independent Parameter Estimation. User Manual, fifth ed. Watermark Numerical Computing, Brisbane, Australia.

Engel, T., Hoogenboom, G., Jones, J.W., Wilkens, P.W., 1997. AEGIS/WIN: a computer program for the application of crop simulation models across geographic areas. Agron. J. 89 (6), 919–928.

Feng, M., Liu, S., Euliss, N., Young, C., Mushet, D., 2011. Prototyping an online wetland ecosystem services model using open model sharing standards. Environ. Model. Softw. 26 (4), 458–468.
Giltrap, D., Li, C., Saggar, S., 2010. DNDC: a process-based model of greenhouse gas fluxes from agricultural soils. Agric. Ecosyst. Environ. 136 (3–4), 292–300.
Hartkamp, A.D., White, J.W., Hoogenboom, G., 1999. Interfacing geographic information systems with agronomic modeling: a review. Agron. J. 91 (5), 761–772.
Ito, A., Inatomi, M., 2012. Use of a process-based model for assessing the methane budgets of global terrestrial ecosystems and evaluation of uncertainty. Biogeosciences 9 (2), 759–773.
Jaynes, D.B., Colvin, T.S., 1997. Spatiotemporal variability of corn and soybean yield. Agron. J. 89 (1), 30–37.
Jones, J.W., Hoogenboom, G., Porter, C.H., Boote, K.J., Batchelor, W.D., Hunt, L.A., Wilkens, P.W., Singh, U., Gijsman, A.J., Ritchie, J.T., 2003. The DSSAT cropping system model. Eur. J. Agron. 18 (3–4), 235–265.
Keating, B.A., Carberry, P.S., Hammer, G.L., Probert, M.E., Robertson, M.J., Holzworth, D., Huth, N.I., Hargreaves, J.N.G., Meinke, H., Hochman, Z., McLean, G., Verburg, K., Snow, V., Dimes, J.P., Silburn, M., Wang, E., Brown, S., Bristow, K.L., Asseng, S., Chapman, S., McCown, R.L., Freebairn, D.M., Smith, C.J., 2003. An overview of APSIM, a model designed for farming systems simulation. Eur. J. Agron. 18 (3–4), 267–288.
McCarthy, A.C., Hancock, N.H., Raine, S.R., 2010. VARIwise: a general-purpose adaptive control simulation framework for spatially and temporally varied irrigation at sub-field scale. Comput. Electron. Agric. 70 (1), 117–128.
McKinion, J., Jenkins, J., Akins, D., Turner, S., Willers, J., Jallas, E., Whisler, F., 2001. Analysis of a precision agriculture approach to cotton production. Comput. Electron. Agric. 32 (3), 213–228.
Miller, S., Semmens, D., Goodrich, D., Hernandez, M., Miller, R., Kepner, W., Guertin, D., 2007. The automated geospatial watershed assessment tool. Environ. Model. Softw. 22 (3), 365–377.
Overgaard, J., Rosbjerg, D., Butts, M., 2006. Land-surface modelling in hydrological perspective – a review. Biogeosciences 3 (2), 229–241.
Raes, D., Steduto, P., Hsiao, T., Fereres, E., 2009. Aquacrop – the FAO crop model to simulate yield response to water: II. main algorithms and software description. Agron. J. 101 (3), 438–447.
Rao, M.N., Waits, D.A., Neilsen, M.L., 2000. A GIS-based modeling approach for implementation of sustainable farm management practices. Environ. Model. Softw. 15 (8), 745–753.
Reed, S., Boggs, J., Mann, J., 2012. A GIS tool for modeling anthropogenic noise propagation in natural ecosystems. Environ. Model. Softw. 37, 1–5.
Resop, J., Fleisher, D., Wang, Q., Timlin, D., Reddy, V., 2012. Combining explanatory crop models with geospatial data for regional analyses of crop yield using field-scale modeling units. Comput. Electron. Agric. 89, 51–61.
Roberts, J., Best, B., Dunn, D., Treml, E., Halpin, P., 2010. Marine geospatial ecology tools: an integrated framework for ecological geoprocessing with ArcGIS, Python, R, MATLAB, and C^{++}. Environ. Model. Softw. 25 (10), 1197–1207.
Schaap, M.G., Leij, F.J., van Genuchten, M.T., 2001. ROSETTA: a computer program for estimating soil hydraulic parameters with hierarchical pedotransfer functions. J. Hydrol. 251 (3–4), 163–176.
Shen, J., Parker, A., Riverson, J., 2005. A new approach for a Windows-based watershed modeling system based on a database-supporting architecture. Environ. Model. Softw. 20 (9), 1127–1138.
Steduto, P., Hsiao, T., Raes, D., Fereres, E., 2009. Aquacrop – the FAO crop model to simulate yield response to water: I. concepts and underlying principles. Agron. J. 101 (3), 426–437.
Steiniger, S., Bocher, E., 2009. An overview on current free and open source desktop GIS developments. Int. J. Geogr. Inf. Sci. 23 (10), 1345–1370.
Thorp, K.R., White, J.W., Porter, C.H., Hoogenboom, G., Nearing, G.S., French, A.N., 2012. Methodology to evaluate the performance of simulation models for alternative compiler and operating system configurations. Comput. Electron. Agric. 81 (1), 62–71.
Thorp, K.R., Batchelor, W.D., Paz, J.O., Kaleita, A.L., DeJonge, K.C., 2007. Using cross-validation to evaluate CERES-Maize yield simulations within a decision support system for precision agriculture. Trans. ASABE 50 (4), 1467–1479.
Thorp, K.R., Batchelor, W.D., Paz, J.O., Steward, B.L., Caragea, P.C., 2006. Methodology to link production and environmental risks of precision nitrogen management strategies in corn. Agric. Syst. 89 (2–3), 272–298.
Thorp, K.R., DeJonge, K.C., Kaleita, A.L., Batchelor, W.D., Paz, J.O., 2008a. Methodology for the use of DSSAT models for precision agriculture decision support. Comput. Electron. Agric. 64 (2), 276–285.
Thorp, K.R., Jaynes, D.B., Malone, R.W., 2008b. Simulating the long-term performance of drainage water management across the midwestern United States. Trans. ASABE 51 (3), 961–976.
Xiao, X., Hollinger, D., Aber, J., Goltz, M., Davidson, E., Zhang, Q., Moore III, B., 2004. Satellite-based modeling of gross primary production in an evergreen needle leaf forest. Remote Sens. Environ. 89 (4), 519–534.