Integrating Spatial Components into FIA Models of Forest Resources: Some Technical Aspects

Pat Terletzky[1] and Tracey Frescino[2]

Abstract.—We examined two software packages to determine their feasibility of implementing spatially explicit, forest resource models that integrate Forest Inventory and Analysis data (FIA). ARCINFO and Interactive Data Language (IDL) were examined for their input requirements, speed of processing, storage requirements, and flexibility of implementing. Implementations of two models was compared across three mapping extents in the two software packages. IDL completed the models in approximately half the time that ARCINFO did and required less memory.

The Rocky Mountain Research Station (RMRS), Interior West Forest Inventory and Analysis (IWFIA) program, Ogden, Utah, developed spatially explicit forest resource models that integrated forest inventory data with spatial data (satellite imagery and elevation data) (Frescino and Moisen 2004). Model generation was done with the Multivariate Adaptive Regression Splines program (MARS; Friedman 1991), a flexible, nonparametric regression-modeling tool that establishes relationships between a forest resource attribute and spatial information.

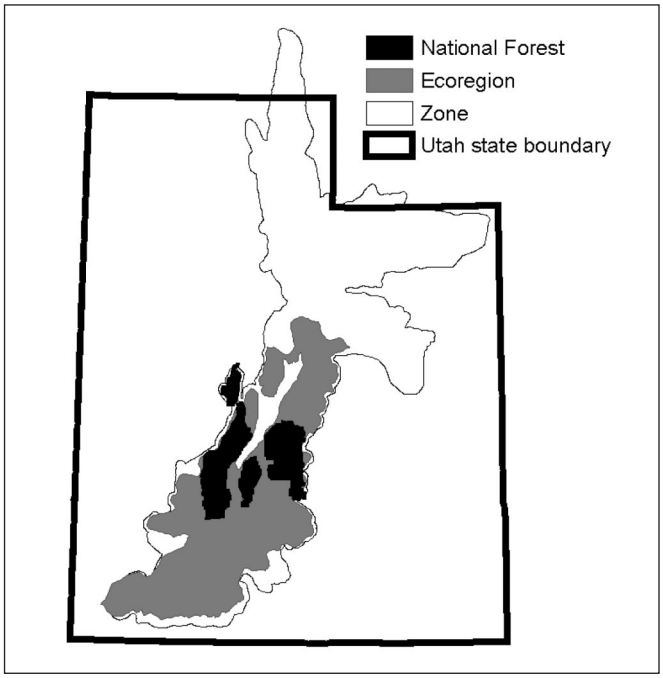
We were provided the MARS models and asked to create a user-friendly method of implementing the models to generate spatially explicit forest resource maps. Incorporating the spatial information into the forest resource models required a large amount of memory, special computing capabilities, and fast computers. We evaluated two software packages, ARCINFO (ESRI Inc., Redlands, California) and Interactive Data Language (IDL, Research Systems, Inc.; www.rsinc.com) based on four parameters: required inputs, processing speed, required memory, and ease of implementation. Although IDL is not common in the Forest Service (FS), we decided to examine it after several attempts with ARCINFO failed.

Methods

We implemented two MARS models in ARCINFO and IDL, and compared the time it took to complete the models and the storage required. The simple model had 6 input files, 13 intermediate files, and 1 output file; the complex model had 9 input files, 20 intermediate files, and 1 output file. Each model was implemented across three mapping extents (fig. 1). The largest mapping extent was a zone level, approximately 6,997,000 hectares, and the smallest extent was a forest level, approximately 627,000 hectares. The ecoregion level (Bailey 1980) was between the zone and forest levels and was 2,925,000 hectares (fig. 1).

We addressed two issues before implementing the models: (1) conversion of the spatially explicit models into the specific language for each software and (2) input and output data for-

Figure 1.—*Depiction of the three mapping scales used to examine two spatially explicit forest resource models.*



[1] Utah State University, Logan, UT.
[2] USDA FS FIA, IWFIA, Ogden, UT.

Figure 2.—*Conversion of a spatially explicit MARS model (a) in ASCII text to ARCINFO AML format (b) based on the Visual Basic conversion program. The original model and the AML output contain references to grids (Veg, Elev, Lat, and Long) and intermediate products to generate the final output (BF2, BF4, BF5, and BF6).*

(a) Forest Resource Model (Mars)

BF2 = max(44 – Veg, 0)
BF4 = max(1380 – Elev, 0) x BF2
BF5 = max(Lat – 1800, 0)
BF6 = max(12336 – Long, 0) x BF4

Final_Grid = (0.992 + 0.987 x BF2 + .002 x BF4 - .032 x BF5 + .266 x BF6)

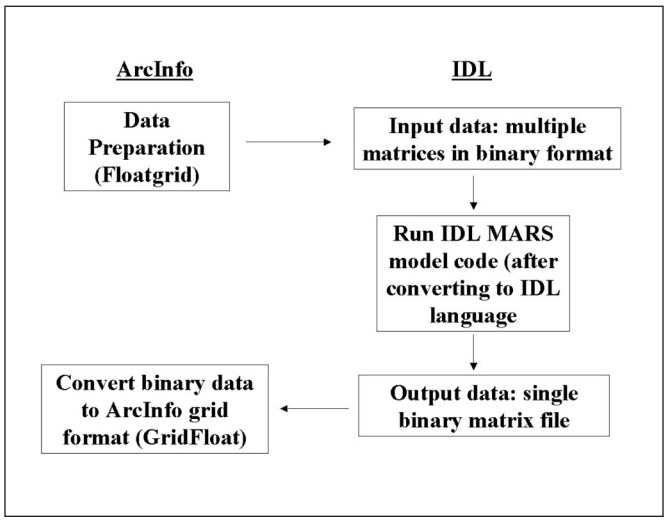
(b) Equivalent AML code

BF2 = con((44 – Veg >= 0), 44-Veg, 0)
BF4 = con((1380 – Elev >= 0), 1380 – Elev, 0) x BF2
BF5 = con((Lat – 1800 >= 0), Lat-1800, 0)
BF6 = con((12336 – Long >= 0) 12336-Long, 0) x BF4

Final_Grid = (0.992 + 0.987 x BF2 + .002 x BF4 - .032 x BF5 + .266 x BF6)

mats. The models were originally in ASCII format and required conversion to the specific language for each software package. Conversion into the ARCINFO software language (ARC Macro Language – AML) was done by a Visual Basic program, written by John Nelson, Ogden IWFIA (fig. 2). This program required user input as to file locations and file output names, and resulted in an AML code, which was then run in ARCINFO. Using this conversion program reduced the number of user input errors caused by typing the model into the computer by hand. Since there was no conversion program to create the corresponding IDL program, the MARS model was transcribed to the IDL format by hand typing the correct IDL code.

Figure 3.—*Process of implementing a spatially explicit model in IDL.*



The original input and final output data format was an ARCINFO grid. Grids are a specific type of raster data unique to ARCINFO. Raster data are simply a series of columns and rows with each cell having a value that represents a specific class (e.g., 1 = water, 2 = agriculture, 3 = urban), or a single value from continuous data (e.g., elevation, soil pH). We required ARCINFO grids as the original inputs and final outputs because it is an industry standard and many Forest Service Geographical Information Systems (GIS) facilities have the ARCINFO software. Whereas ARCINFO required no changes in input or output data, IDL required converting the input grids to binary format, and the output binary files to grids, which required ARCINFO (fig. 3).

Figure 4.—*Comparison of the time it took to run a simple mode (6 inputs, 13 intermediate grids (binary data for IDL), and one output) for three levels in the ARCINFO and the IDL software packages.*

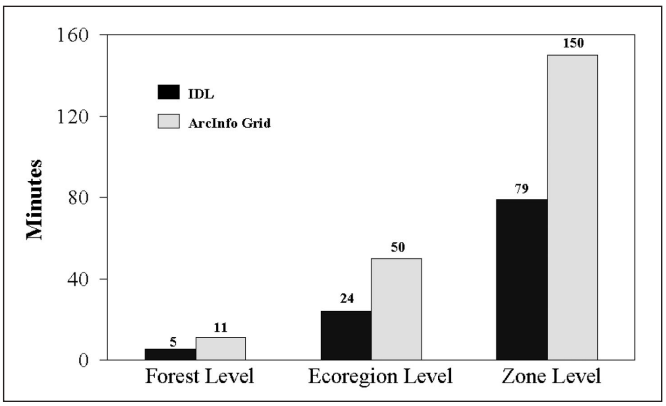


Figure 5.—*Comparison of the time it took to run a complex model (9 inputs, 20 intermediate grids (binary data for IDL) and one output) for three levels in the ARCINFO and the IDL software packages.*

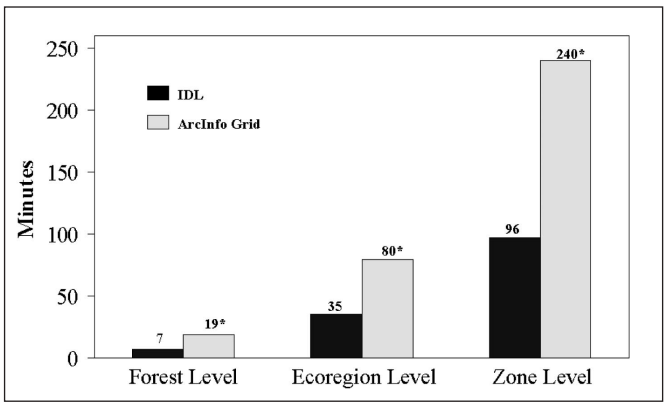
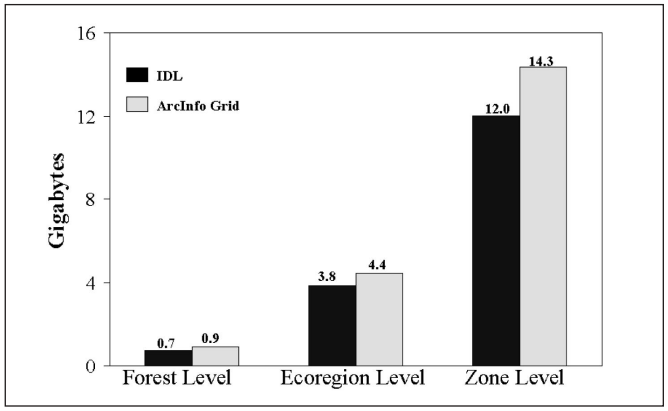


Figure 6.—*Comparison of required storage for the simple model in the ARCINFO and the IDL software packages.*



Comparison of ARCINFO and IDL Model Implementation Methods

The IDL method took about half as long as ARCINFO, although as the area of interest increased the difference lessened (figs. 4 and 5). The values for the IDL process included the time to convert the ARCINFO grids into binary files and the output binary file into an ARCINFO grid. The simple model was able to run in both ARCINFO and IDL, but the complex model resulted in erratic output in ARCINFO. Although binary files require more storage than ARCINFO grids, IDL input and output files required less total storage space than ARCINFO (fig. 6). The ARCINFO method created temporary intermediate grids that were saved to memory, whereas the IDL method processed one row of data at a time, thus requiring only the memory needed to hold one row.

Conclusion

All models were able to complete implementation in IDL and generate an output file. ARCINFO, while eventually able to generate an output file for the complex model, proved to be unreliable and unstable, often generating an output grid that showed all cell values as zero, or completely nonsensical data values. The feasibility of implementing IDL depended on more than technical results. Implementation required considering (1) the type of operating system required, (2) the computer hardware requirements, (3) the cost of obtaining and maintaining IDL, (4) the ease of use and access to people who know how to implement the software, and (5) the limited input data formats (ASCII or binary). We compared all of these aspects for ARCINFO and IDL (table 1). The advantages of ARCINFO are that many people know how to work with ARCINFO; model implementation is a simple, one-word command (&r <name of model AML>); and data inputs and output are grids. The disadvantages are that it can be unstable for complex models, and large amounts of memory are required because all input grids and intermediate grids are stored in memory. The advantages of IDL are that it is less expensive than ARCINFO; requires less memory because model implementation is done only on certain sections of the input files at a time, and the programming lan-

Table 1.—*Comparison of (1) operating systems that could run each package respectively, (2) machine requirements, (3) approximate cost, (4) potential number of experienced users, and (5) available input data formats for ARCINFO and IDL*

Comparison Factors	ARCINFO	IDL
Operating system	Windows (95, 98, 2000, NT)	Windows (95, 98, 2000, NT)
	Unix	Unix
Computer requirements		
RAM	16MB	256MB
Hard drive	25MB	1GB
Processor	486 or higher	Pentium II
Cost	$3,000 (Windows)	Windows $650/ Unix $1,250
Ease of use	Many routines written; many AML programmers	Some routines written; fewer IDL programmers
Data format	Grid, binary, ASCII	Binary, ASCII

guage is easy to learn. The disadvantages of IDL are that there are fewer people who know how to write IDL code than ARCINFO AML code, a program needs to be written to implement the model, and input and output data are restricted to binary or ASCII formats.

Literature Cited

Bailey, R.G. 1980. Description of the ecoregions of the United States. Misc. Publ. 1391. Washington, DC: Department of Agriculture.

Frescino, T.S.; Moisen, G.G. 2004. Predictive mapping of forest attributes on the Fishlake National Forest. In: Proceedings of the 4th Annual Forest Inventory and Analysis symposium; 2002 November 19–21; New Orleans, LA. Gen. Tech. Rep. NC-252. St. Paul, MN: U.S. Department of Agriculture, Forest Service, North Central Research Station.

Friedman, J.H. 1991. Multivariate adaptive regression splines. Annals of Statistics. 19: 1–141.