RECENT GENERALIZATION DEVELOPMENT AND ROAD AHEAD
(For the 5th Workshop on Progress in Automated Map Generalization, ICA)

Dan Lee
ESRI, Inc.
380 New York Street
Redlands, CA 92373
USA
Email: dlee@esri.com

Abstract

With an entirely new architecture and user environment, ArcGIS, the object-oriented generation of ESRI's GIS product, provides a spatial framework to support geographic data management and decision-making. Meanwhile, the collection of COM (component object model)-based ArcGIS components, known as ArcObjects, is the development platform for the creation of the existing ArcGIS desktop applications, such as ArcMap and ArcCatalog and is also accessible to application developers.

The integration of generalization into ArcGIS has begun with the creation of COM objects and geoprocessing tools. This paper gives a brief introduction to the Geoprocessing concepts and environment and describes our recent efforts on generalization development in these forms.

In the upcoming release, ArcGIS 9.0, geoprocessing will be one of the most powerful components for geodatabase data transformation, analysis, and manipulations. The geoprocessing environment will allow you to perform operations using many data types and provide access to hundreds of geoprocessing tools through four different methods: a dialog, a command line, a model, and a script. A major part of generalization will be operated via geoprocessing and a set of commonly needed generalization tools are being implemented within geoprocessing system. The generalization functions as COM components will be used to build advanced cartographic editing and on-demand mapping capabilities. The road ahead will be briefly discussed.

Keywords: geoprocessing, modeling, topological errors, context generalization

I. Geoprocessing in ArcGIS

Taking a relatively broad definition, geoprocessing in ArcGIS refers to the application of core GIS operations that create new spatial data from existing or derived data. The basic GIS capabilities found under this umbrella include data format conversion, spatial analysis, and data management. A typical geoprocessing operation takes an input geodatabase feature class, performs an operation on it, and returns the result of the operation as an output geodatabase feature class. For example, the Buffer operation takes a point, line, or polygon feature class and creates a buffer polygon feature class based on user-specified parameters.

To perform geoprocessing tasks, you can choose one of the following four methods: tool dialogs, command lines, model tools, and scripts.

A tool dialog can be invoked from a dockable geoprocessing toolbox window in ArcCatalog or ArcMap. The dialog gives an easy user interface for you to specify data and parameters to perform a single operation.

A command line is accessible through ArcGIS products (ArcInfo, ArcEditor, or ArcView). The command line window prompts you with the usage of the specified command, so you always know what parameter is being entered.

A model tool can be created in Modelbuilder, which provides you with a graphical environment to construct a diagram of the steps - representing a model - to complete a geoprocessing task. A model tool executes processes in chained sequence.

A script offers an efficient and effective way of managing user's geoprocessing needs, especially for tasks involving large volume of data, repetitive work, and more complex decision-making. The COM IDispatch interface makes it possible for interpretive and macro languages, such as VBScript, Jscript and Python, to access COM components. The one object in ArcGIS, GPDispatch, supports the IDispatch interface and exposes the Geoprocessor to scripting clients.

II. Developing Generalization Tools in Geoprocessing

Map generalization is a data transformation, reduction, and integration process. As part of the modern GIS and mapping systems, generalization needs a flexible and user-controlled environment. To derive smaller scale data from a master database, you may need only a single generalization operation, such as simplification or aggregation, to reduce the level of detail for a particular feature class; or perhaps a set of generalization tools can be used in conjunction with other geographic operations in a logical sequence to reach the desired result. Multiple features and feature classes can be involved and sometimes decisions need to be made, by human, based on intermediate results. The Geoprocessing in ArcGIS provides a fundamental environment for the above generalization operations.

1. The Beginning phase

The integration of generalization tools into ArcGIS has been underway in the past few months with the ultimate goals to support database generalization and cartographic generalization, as distinguished by researchers (Weibel and Jones, 1998), from geodatabases. The beginning phase includes the following tasks:

- Reevaluate the most requested generalization functions for initial implementation. With our research and experience in developing generalization commands for Workstation ArcInfo and working with users in the past, a set of generalization functions had been identified. These basic functions reflect how human thinking can be decomposed into clearly defined rules and processes, which can then be implemented and used to build towards a more comprehensive and intelligent solution. It is no doubt that the generalization functions already implemented in Workstation ArcInfo are still among the first needed; while certain level of feature conflict resolution is highly expected.

- Investigate the best techniques to be used in building generalization functions. The new Geodatabase data model without built-in topology in ArcGIS requires different techniques from those used when working with coverage data model (Lee, 2000). We still need to avoid topological errors during the generalization processes. The good news is that coming out in ArcGIS 8.3, feature topology can be built as needed and the topology engine is accessible, and that the triangulation structure in TIN engine has been enhanced to support generalization needs. We are now able to localize feature conflicts more easily and produce better results more efficiently. For example, in line simplification, if during the simplification a topological error is created, such as line crossing or coinciding, the local segment instead of the whole line as in Workstation ArcInfo, will be found and a reduced tolerance will be applied to avoid the conflict. This solution partially relies on the triangulation information.

- Create generalization functions as COM objects in ArcObjects library to be used in building high-level applications, such as generalization tool sets in Geoprocessing and other product components. According to feature types, the generalization class will contain IPointGeneralization, ILineGeneralization, IAreaGeneralization, and multiple feature types generalization interfaces. The line simplification and line smoothing for polylines and feature classes have been implemented as line generalization methods.

- Build the generalization tool set in Geoprocessing. The feature class generalization methods are used to create generalization tools under the Generalization Tools set. Each tool is made in compliance with all other Geoprocessing tools and supports the four Geoprocessing methods mentioned above.

2. The Model Tools

Generalization process is never straightforward; to model the process is always a challenge. The Modelbuilder in Geoprocessing helps us to analyze the affects of different procedures, adjust the workflow according to different themes and target maps, and make map production more effective. You can create and edit a model diagram in Modelbuilder to put the generalization steps in a desired sequence. The diagram can be saved as a model tool in a user-specified tool set and modified easily to repeat the same or similar processes for different datasets or for the same data with different parameters. The model diagram in Figure 1 illustrates an experimental generalization sequence. The input ContourL is a large-scale contour feature class with a 10-meter contour interval. The Select Tool selects features based on a query, in this case selecting the 50-meter index contours, and outputs them into a new feature class 50mCont. Then the selected contours will be simplified and smoothed. The additional output "collapsed" from Simplify Lines tool carries potentially collapsed lines (zero-length lines) as points, see more detail about it in the next section; in this case this output will not participate in further processes and therefore be deleted.

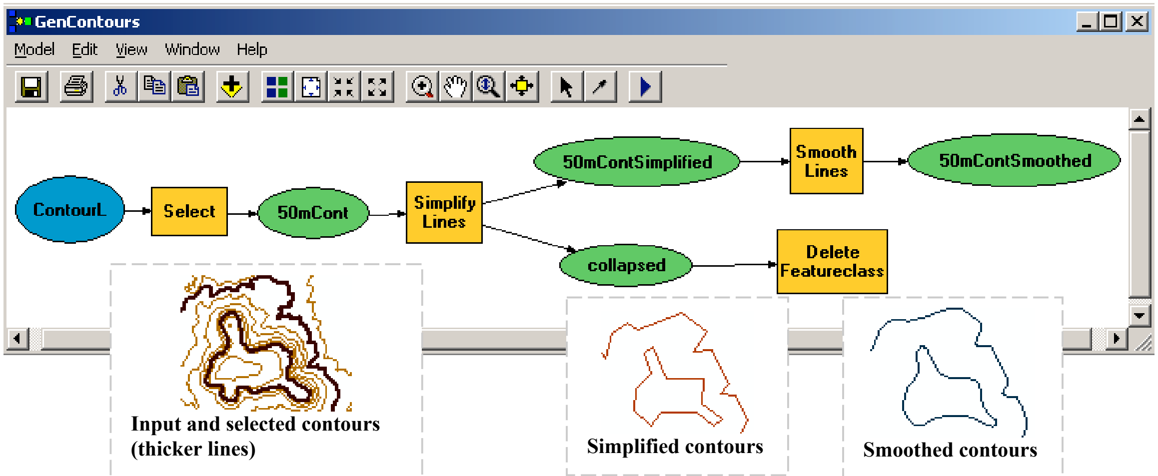

Figure 1. An experimental generalization sequence in Modelbuilder

3. General Requirements for Generalization Tools

Making a generalization tool is not just implementing an algorithm so that the input data gets reduced in the level of detail. A generalization tool must satisfy the following general requirements:

- The generalized features must be linked to their source features as either a one-to-one or a one-to-many relationship in order to facilitate attribute transfer and feature updating. A one-to-one relationship is easy to maintain because the output features can just carry the same object IDs as their corresponding input features; for example a simplified line will have the same object ID as its source line. Some generalization processes, such as aggregation, turn a group of features into a representative feature. In these cases, a relation table must be created to record the one-to-many relationships.

- Data integrity (completeness, consistency, and so on) must be ensured. In line simplification, for example, the algorithms used can cause a closed (circular) line to collapse into a zero-length line, which is invalid in geodatabase. To allow users to keep track of what and where they are, a point feature class will be generated as shown in the model diagram in Figure 1, to carry the endpoints of the collapsed or lost lines with their source line object IDs; the user can then decide to delete them, if they are indeed unimportant, retrieve the original lines, or else.

- Generalization process status and problems must be recorded and flagged to support the evaluation of the result, the analysis of the process, the post processing (interactive or automatic), and the research for enhancements in the incomplete and problem areas. Again taking line simplification as an example, when the option CHECKERRORS is specified, line-crossing and coincident lines produced by the simplification algorithms will be detected and the local segments will be re-simplified using a reduced tolerance. The iteration continues until no such topological errors occur anymore. This approach results in lines possibly simplified by the specified and reduced tolerances in different parts. To make the user aware of the situation and be able to review the cases easily, two new attributes, max_tolerance and min_tolerance (used to simplify a line), are written for each line in the output feature class. The user knows immediately what range of tolerance is used for a particular line and whether the specified tolerance is suitable for majority data.

II. Road Ahead

There is obviously a long journey ahead in developing generalization solutions and we have taken the first few steps optimistically and successfully. As the ArcGIS products continue to evolve, map generalization research and development will advance in the following areas:

- Extend the ArcObjects library to include more generalization operators and functionality that supports generalization processes, such as measurements and evaluation of the data before and after generalization, visualization of the process and results, and so on.
- Add a full set of generalization tools in Geoprocessing, including single-operation tools, multiple-features generalization tools, post-processing tools, derived model tools and script tools.
- Integrate generalization capability in the editing and map compilation environments with interactive generalization and post-editing.
- Build towards a rule-driven, intelligent generalization engine that can handle context generalization (involving multiple feature types) and make decisions on where, what, and how to generalize the target areas.
- Create smart features and enrich databases to facilitate multiple-purposes and multiple-scales data transformation and cartography.
- Meet the requirements for on-demand generalization and location-based mapping. The Internet mapping and the wireless communication requests need to be answered instantly (Koeppel, 2000) with descriptions and effectively generalized maps. On-demand map generalization may share similar nature as map compilation in terms of fitting the output scales and purposes, but the result is expected to be complete (no chance for post-editing), readable, and repeatable. This would force the generalization rules to be rather simple and decisive. In case of a conflict, a resolution has to be reached without human interactions. This is a change from the traditional thinking.

References

Koeppel, Ian, (2000). "GIS Extended to the Wireless & Internet World", ArcNews Fall 2000 issue, ESRI.

Lee, Dan, (2000). "Map Generalization in GIS: - practical solutions with Workstation ArcInfo", technical paper, under ArcGIS - Technical Papers at http://arconline.esri.com/arconline.

Weibel, Robert, and Jones, Christopher B., (1998). "Computational Perspectives on Map Generalization", GeoInformatica, Vol.2, No.4, Dec. 1998, pp. 307-314.