A Browser/Server-based Prototype of Heuristic Modelling Environment for Digital Terrain Analysis

Cheng-Zhi Qin[1], Jing-Chao Jiang[1], Li-Jun Zhan[1], Yan-Jun Lu[1], A-Xing Zhu[1, 2]

[1]State Key Laboratory of Resources and Environmental Information System,
Institute of Geographic Sciences and Natural Resources Research, CAS
Beijing, China
qincz@lreis.ac.cn

[2]Department of Geography, University of Wisconsin-Madison
Madison, Wisconsin, USA

Abstract—**To narrow the "digital divide" in Digital Terrain Analysis (DTA) modelling and application, we proposed a Browser/Server (B/S)-based prototype of heuristic DTA modelling environment. In this prototype, formalized DTA knowledge was used to support heuristic and visualized modelling of user's application-specific DTA workflow. B/S structure can provide DTA users with high accessibility to DTA algorithms. Parallel DTA algorithms of computing not only local topographic attributes but also regional topographic attributes were implemented based on Message Passing Interface (MPI) to speed up the execution time of user-built DTA workflow. A case study of calculating topographic wetness index (TWI) for a low-relief catchment at fine resolution shows that employing the proposed DTA modelling environment, user can build an application-specific TWI-calculating workflow for his/her application context in a much easier way than existing DTA-assisted tools. There is less requirement of user's DTA knowledge during the modelling process. The TWI-calculating workflow can get reasonable TWI result when the execution time of the workflow was distinctly speeded up.**

INTRODUCTION

Similar to many other complex geo-spatial analyses, the practical application of Digital Terrain Analysis (DTA) is typically a non-trivial workflow-building process for users (especially for non-expert users) [1-2]. In this process, knowledge in DTA domain is necessary not only to properly organize different DTA tasks, but also to specify the algorithm with parameter-settings for each task to match the application context. However, current DTA-assisted tools often leave this to users.

Furthermore, current DTA-assisted tools are often implemented as standalone software with a serial DTA algorithm base. This makes the new-developed DTA algorithms difficult to reach the real application. And the execution of user-built DTA workflow is often too time-consuming to support real applications, especially those with large area and fine scale.

These inconveniences in current DTA-assisted tools make a so-called "digital divide" challenge [3] in DTA modelling and application. To narrow this digital divide, we proposed an attempt on web-based heuristic modelling environment for DTA.

BASIC IDEA

Heuristic and visualized DTA modelling

A graphical user interface (GUI) similar to that of ArcGIS ModelBuilder can support users' visualized modelling of DTA workflow. When DTA knowledge on such as the organization among tasks, the data flows, selection of algorithm for a specific task, etc. could be formalized and be saved in knowledge base, the formalized DTA knowledge could be used by an inference engine combining with the GUI to aid the non-expert users to model DTA workflow in a much easier way [4]. By this means when an initial target task is requested by user, the corresponding DTA workflow will be interactively expanded until all input data are ready. This heuristic and visualized modelling process of automatically expanding the application-specific DTA workflow from the last task to the first task is different to the traditional modelling process of user-assigning from the first task to the last task by most existing DTA-assisted tools. We believe that the heuristic and visualized modelling for DTA is more natural for non-expert users.

Browser/Server structure

Browser/Server (B/S) structure can provide DTA users with high accessibility to all kinds of DTA algorithms which are saved in algorithm base in server or are published as web services [5-6]. Even mobile terminal will be available for DTA modelling. Not only DTA algorithms but also open DEM dataset are increasingly being released as web services. These resources could enhance the functionality of web-based DTA modelling environment.

Parallel computing of DTA algorithms

Parallel computing is widely used to speed up the execution time for time-consuming geo-computation. More and more DTA algorithms are parallelized. They can take advantage of the high parallel computation capacity of server for faster execution of user-built DTA models.

A PROTOTYPE

Currently, we prototyped a B/S-based heuristic modelling environment for DTA (http://159.226.110.183:8080/egc/). Two views, i.e. Google map view and modelling GUI, were provided through browser for users to view geographic information and conduct modelling process, respectively. Different types of DTA knowledge can be formalized based on semantic network or case, and then saved in the knowledge base in the server. An inference engine we developed before [4,7] uses the knowledge base to support heuristic DTA modelling process which is visualized in browser.

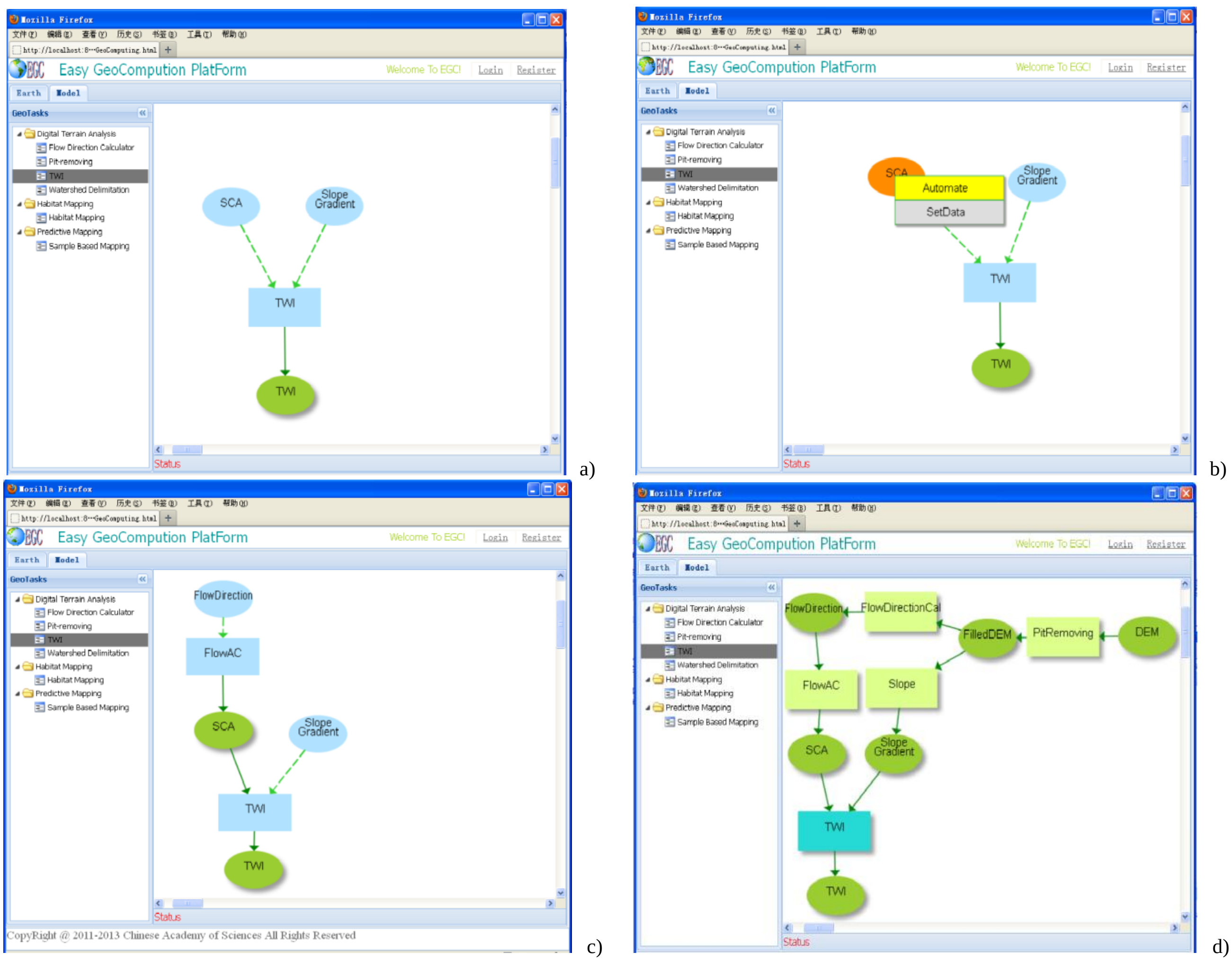

Figure 1. The heuristic and visualized modelling process of user's TWI-calculating task with the proposed DTA modelling environment. a) Drag a TWI-calculating task into canvas panel of the modelling GUI (Light blue ellipse and dotted line means that the data source is not specified yet); b) Interaction on whether the current workflow should be extended (the 'Automate' option means to automatically extend current workflow by joining a task of computing the data for current data icon); c) New task is automatically added in current DTA workflow; d) A complete TWI-calculating workflow after modeling process.

Within this modelling environment, user can further manually adjust the automatically-built DTA workflow, such as adjusting the parameter value for an algorithm, or assigning another specific algorithm for a task in this DTA workflow. The resulted application-specific DTA workflow can be stored as a file with XML format, which in the future can be loaded in this modelling environment. This file also plays the role of meta data to describe how the result is produced.

Through browser, user can submit an application-specific DTA workflow to the server for execution. Currently the DTA workflow built in this modelling environment is executed in a cluster owned by our laboratory. DTA algorithm base includes not only serial algorithms implemented before, but also parallel algorithms we implemented based on Message Passing Interface (MPI). The iterative DEM-preprocessing algorithm for removing pits and flat areas in original DEM was parallelized with a new efficient parallelization strategy [8].

After execution of an application-specific DTA workflow, the executed result will be published as WMS (Web Mapping Service) with MapServer to be shown in the Google map view. User can also download the result data through the browser.

APPLICATION

Application context

For example, a user needs calculate topographic wetness index (TWI) for a low-relief catchment (about 60 km^2) in northeastern China. In database only a DEM at 1-m resolution is available for this area.

Heuristic and visualized modelling process

When the user entry the prototyped B/S-based heuristic modelling environment for DTA through the browser, he/she can view both existing geographic information and consequent DTA results from a Google map view. In the other view (modelling GUI), he/she drags the TWI-calculating task into the canvas panel of the modelling GUI, not only a TWI task but also its links to the output data (TWI) and input data (two topographic attributes, i.e. slope gradient and SCA) are visualized (Fig. 1a). Because the necessary input data are not available, the inference engine instruct the current DTA workflow to be automatically expanded by adding additional tasks of calculating slope gradient and SCA (Fig. 1b, c). The similar process will be repeated until the finish of modelling user's TWI-calculating workflow, i.e. all input data needed in this workflow can be found in database or be calculated (Fig. 1d).

During this modelling process, for each task in workflow a specific algorithm with proper parameter-settings is chosen by the inference engine based on the knowledge base. For example, in this TWI-calculating application the MFD-md algorithm [9] instead of other flow direction algorithms was chosen for SCA-calculating task for low-relief area with 1-m resolution DEM, according to the formalized knowledge (Fig. 2).

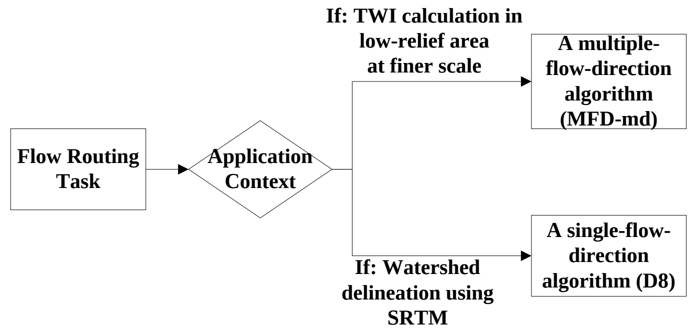

Figure 2. Selection of flow direction algorithm based on the DTA knowledge.

Execution process and result

After above modelling process, the resulted application-specific DTA workflow is ready to be submitted to the server for execution. The MPI-based parallel programs of specific DTA algorithms in this workflow were executed in cluster. The TWI result is shown in the map view, and the data is also ready for download by user (Fig. 3).

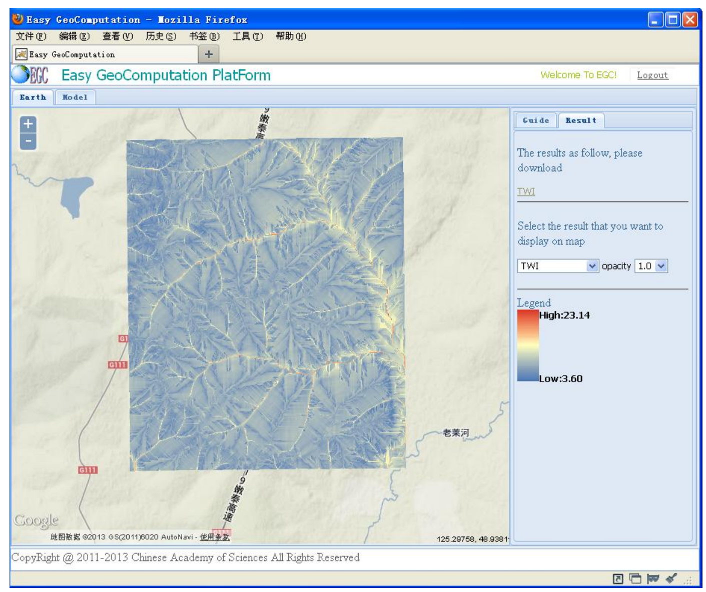

Figure 3. TWI result shown in the map view.

Discussion

By this B/S heuristic modelling environment for DTA, user can build an application-specific TWI-calculating workflow for his/her application context in a much easier way. There is less requirement of user's DTA knowledge during the modelling process. The resulted TWI-calculating workflow can get reasonable TWI result. With the aid of parallel computing, the run time of the workflow was distinctly speeded up (measured as the speedup, i.e. (time for serial algorithm) / (time for parallel algorithm with *p* processes), Fig. 4).

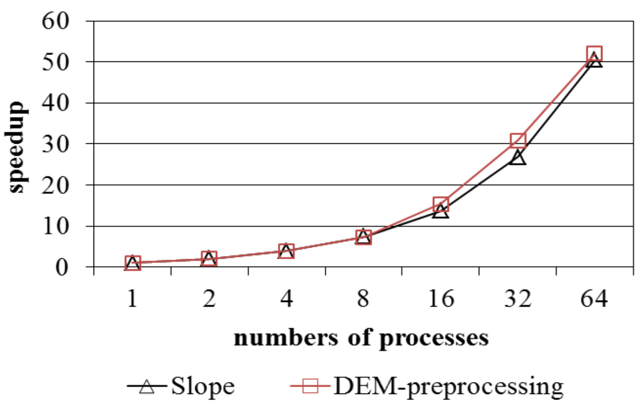

Figure 4. Speedup of the MPI-based parallel algorithms in the application-specific TWI-calculating workflow built with the proposed DTA modelling environment when were executed on an IBM SMP cluster. The cluster has 134 calculation nodes through Infiniband parallel network interconnection. Each node has 2 Intel Xeon (E5650 2.0 GHz) 6-core CPU and 24 GB DDRIII memory. The dimension of DEM is 11130 ×9320 cells. Run time (not including I/O time) for serial programs of slope gradient and DEM-preprocessing is 5.4 s and 40110 s, respectively.

ACKNOWLEDGMENTS

This study was supported by the Institute of Geographic Sciences and Natural Resources Research, CAS (No. 2011RC203), and the National High-Tech Research and Development Program of China (No. 2011AA120302).

REFERENCES

[1] Wilson, J. P., and J. C. Gallant, 2000. “Terrain Analysis: Principles and Applications”, Wiley, 479 p.

[2] Hengl, T., and H.I. Reuter (Eds), 2008. “Geomorphometry: Concepts, Software, Application”, Developments in Soil Science - Volume 33, Elsevier, 765 p.

[3] Zhu, A.-X., C.-Z. Qin, and Y.-J. Lu, 2011. “High performance computing, CyberGIS and digital divide”, In 19th International Conference on Geoinformatics, June 24-26, Shanghai, China.

[4] Qin, C.-Z., Y.-J. Lu, A.-X. Zhu, and W.-L. Qiu, 2011. “Software prototyping of a heuristic and visualized modelling environment for digital terrain analysis”, In 11th International Conference on GeoComputation, July 20-22, University College London, UK.

[5] Yue, P., L. Di, W. Yang, G. Yu, and P. Zhao, 2007. “Semantics-based automatic composition of geospatial Web service chains”, Computers & Geosciences, 33(5), 649-665.

[6] Zhao, P., T. Foerster, and P. Yue, 2012. “The geoprocessing web”, Computers & Geosciences, 47, 3-12.

[7] Lu, Y-.J., C.-Z. Qin, A.-X. Zhu, and W.-L. Qiu, 2012. “Application-matching knowledge based engine for a modelling environment for digital terrain analysis”, In 20th International Conference on Geoinformatics, June 15-17, Chinese University of Hong Kong, China.

[8] Qin, C.-Z., and L. Zhan, 2012. “Parallelizing flow-accumulation calculations on Graphics Processing Units—from iterative DEM preprocessing algorithm to recursive multiple-flow-direction algorithm”, Computers & Geosciences, 43, 7-16.

[9] Qin, C.-Z., L.-L. Bao, A.-X. Zhu, X.-M. Hu, and B. Qin, 2013. “Artificial surfaces simulating complex terrain types for evaluating grid-based flow direction algorithms”, International Journal of Geographical Information Science, 27, 1055-1072.