Transactions in GIS, 2008, 12(4): 541–561

Research Article

Choice Modeler: A Web-based Spatial Multiple Criteria Evaluation Tool

Piotr Jankowski
Department of Geography
San Diego State University

Arika-Ligmann Zielinska
Department of Geography
San Diego State University

Martin Swobodzinski
Department of Geography
San Diego State University

Abstract

This article presents a concept of a Web-based spatial multiple criteria evaluation tool for individual and group use called Choice Modeler (CM). CM was originally conceived as part of a larger Participatory Geographic Information System for Transportation project (PGIST; http://www.pgist.org) aimed at developing and evaluating Internet-hosted capabilities to support participatory decision processes. CM is designed to be either a part of a larger information system such as PGIST or a standalone tool used for evaluation of decision variants. The decision support functions provided by CM aid in reducing the cognitive complexity of the decision space characterized by multiple decision options, evaluation criteria, and criterion weights. This is achieved by incorporating in CM the sensitivity analysis functions for the identification of criteria that do not influence the decision option ranking. Users can remove such criteria from further consideration and thus lessen the cognitive burden of evaluation, which may be essential in multi-stakeholder participatory decision processes. The additional capabilities of CM include a vote aggregation function to collate individual option rankings into a group ranking, and measures of agreement/disagreement to inform the participants about a group-derived desirability of specific decision options. The design of CM was implemented using Web-service architecture. In the article we describe the design of CM and discuss its advantages and limitations.

Address for correspondence: Piotr Jankowski, Department of Geography, San Diego State University, San Diego, CA 92182-4493, USA. E-mail: piotr@geography.sdsu.edu

doi: 10.1111/j.1467-9671.2008.01111.x

1 Introduction

Multiple criteria evaluation (MCE) is a fundamental approach for screening and selecting spatially differentiated decision variants (Voogd 1983, Beinat and Nijkamp 1998). In the last fifteen years, much work has been directed toward integrating Geographic Information Systems (GIS) and multiple criteria evaluation (MCE) methods in the context of spatial decision support systems for planning, retail and services locations, land-based project selection, and environmental management (Carver 1991, Eastman et al. 1993, Pereira and Duckstein 1993, Jankowski 1995, Laaribi et al. 1996, Tkach and Simonovic 1997, Malczewski 1999, Joerin et al. 2001, Gomes and Lins 2002, Marinoni 2005). The concept of Spatial Decision Dupport Systems (SDSS), introduced in Geography 20 years ago (Cowen 1988, Densham and Rushton 1988, Densham and Goodchild 1989, Armstrong et al. 1991), was inspired by earlier work in management science on systems and tools augmenting human expert knowledge with data management, modeling, display, and reporting functions in order to support managerial decisions (Keen and Scott-Morton 1978, Sprague and Carlson 1982). The motivation for research on SDSS came largely from recognizing that there was a class of spatial decision problems characterized by the lack of structure involving an incomplete knowledge of decision objectives, evaluation criteria, decision options, and consequences/impacts of those options. Such problems often cut across various domains, organizational boundaries, individual, and group interests. In this context, MCE can be useful only after the basic structure for a decision problem, involving decision options, evaluation criteria, and the performance of decision options on the criteria, has been established. Nevertheless, over the last 20 years spatial MCE has come to be recognized as an inextricable component of SDSS (Malczewski 2006).

The decision processes aimed at resolving complex land and other resource allocation problems frequently involve multiple actors such as domain experts, managers, elected officials, and representatives of various civic groups. The recognition of the complexities accompanying spatially-explicit decision problems motivated the work on SDSS that provided spatial decision support capabilities for groups (Armstrong 1993, Faber et al. 1994). Much of that research was focused on combining the development of group work techniques using maps, computational models, and collaborative modes of work, eventually leading to the development of what is known today as geocollaboration (MacEachren 2004, MacEachren and Brewer 2004). Within that context appeared the early prototypes of SDSS tools involving MCE functions modified for group work (Malczewski 1996, Jankowski et al. 1997, Jankowski and Stasik 1997). The majority of early implementations of SDSS incorporating MCE functions were developed for the monolithic desktop computer systems using one of three strategies including loose coupling, tight coupling, and full integration (Malczewski 2006). With the development of Internet GIS (Peng and Tsou 2003), interoperability standards (OGC 2008) and Web-based technologies (Berners-Lee et al. 2001) research on SDSS in the late 1990s and early 2000s started to employ the distributed (client-server) architectures (Carver et al. 1998, Kingston et al. 2000, Zhu et al. 2001, Jankowski et al. 2001, Andrienko and Andrienko 2001, Rinner and Malczewski 2002). Rinner (2003), in his overview of a Web-based SDSS, distinguished three categories of SDSS based on: (1) sever-side applications; (2) client-side applications; and (3) mixed architectures taking advantage of both server-resident and client-resident decision support capabilities.

This classification of Web-based SDSS architectures is instrumental for situating Choice Modeler (CM) as a server-based MCE tool. The adoption of advances in Web-based technologies for MCE has been rather slow. The objective of this article is to contribute to the advancement of MCE as a stand alone methodology and a potential SDSS component by presenting a design and prototype implementation utilizing the current Web-based technologies. In the remainder of the article we provide a brief overview of recent work on Web-based SDSS with a focus on explicit MCE capabilities. This is followed by the presentation of MCE functions implemented in CM and by the discussion of its architecture. The article closes with an outlook on future directions in developing distributed MCE capabilities.

2 Web-based Spatial Decision Support Systems and Multiple Criteria Evaluation

There is a notable body of recent work on Web-based spatial decision support systems (Web-SDSS) developed in various application domains such as watershed management (Dymond et al. 2004; Choi et al. 2005a, b), public housing management (Burton et al. 2005), business allocation (Jung and Sun 2006), landslide risk assessment (Yu et al. 2006), flood forecasting (Wang and Cheng 2007), prediction of marine mammal habitats (Best et al. 2007), negotiation support (Chen et al. 2008), and collaborative planning (Stock et al. 2008). In much of the work cited here spatial decision support capabilities include application-specific models for the simulation of various what-if scenarios, and tools that enable the visualization of the simulation results. Conspicuously missing in those systems are functions and tools enabling the amplification of human judgment about the components of the decision situation (decision options, evaluation criteria, option and criterion preferences), and the support for group decision processes. In the absence of MCE functions, an exploratory visual analysis becomes the primary vehicle for supporting the evaluation of simulated decision scenarios.

Within the last six years only a small number of contributions reported on Web-SDSS developments that explicitly include MCE and group work functions. Rinner and Malczewski (2002) implemented an MCE method called ordered weighted averaging (OWA) in an Internet-based interactive mapping environment based on the CommonGIS system (Andrienko and Andrienko 1999). The OWA function was implemented as Java classes built into the client-side Java applet comprising the CommonGIS. Andrienko et al. (2003) extended their Java applet-based CommonGIS system, developed as a client-side Java applet, by adding sensitivity analysis capabilities to test the influence of shifting criterion weights on the stability of decision option ranking. They also added vote calculation functions to compute the outcome of ranked vote and the vote variance.

Bernard et al. (2003) developed a prototype of a Web-based MCE tool using the concept of a Web service. Building on the recommendations and specifications of the Open Geospatial Consortium (OGC) for distributed Web-services, they discussed the design considerations for a Web-based MCE tool that aids in the selection of areas of substantial vulnerability to climate change. The authors proposed two approaches for implementing a MCE Web service within a chain of OGC-compliant Web services including Web Map Server (WMS), Web Coverage Service (WCS), Web Feature Service (WFS), and Web Catalogue Service (OGC 2008). In the first approach, the 'thick' client establishes a transparent chain between the Web services, and then retrieves and processes the necessary data directly through the standardized interfaces of the Web services. All MCE related functions (i.e. the definition of criteria, the assignment of weights to criteria, the normalization of criteria values, and the calculation of decision option scores) are handled by the 'thick' client. The second approach employs one Web service that aggregates all MCE-relevant functions. In this approach all the functionality is handled on the server-side; the 'thin' client acts merely as the user interface for displaying the results that are handed to it by the MCE Web service.

Karnatak et al. (2007) developed a client/server based MCE tool for site allocation in the context of biodiversity conversation. The tool implements the Analytic Hierarchy Process (AHP) originally developed by Saaty (1980). Similar to Bernard et al. (2003), all of the MCE functionality is located on the server. Consequently, all the calculations that are relevant to AHP take place on the server. The tool relies on ESRI's ArcSDE for remote access to an Oracle database in order to retrieve the data for multiple criteria evaluation, and on ESRI's ArcIMS for the preparation of the output of the AHP procedure in the form of a map. A 'thin' client running in any standard Web browser fetches and presents the map to the user.

In the area of support for group decision processes, Dragicevic and Balram (2004) developed a set of Web-based tools for collaborative planning. Their *Web GIS Collaborative Spatial Delphi (CSD)* framework builds upon the traditional Delphi method and combines discourse tools, digital maps, and exploratory GIS functionality within a Delphi planning process. The manipulation and visualization of spatial information is mediated through ESRI's ArcIMS and hosted on an Apache Web server, together with the tools that implement the components of the Delphi process. All functionality is concentrated on the server. A standard Web browser hosts the client application that handles the invocation of server-side operations and the display of results. A facilitator has access to additional tools that aid in the management and monitoring of the Delphi process (e.g. generation of documents, statistic tools, and participant database).

This brief overview of selected literature on Web-based SDSS begs the question about the merit of developing MCE functions and tools on the platform of distributed architectures. Should a decision about porting MCE tools from computer desktops to the Web be based on trends in software design, including the recent trend of deploying task-oriented functions as Web services, or should it be based on the analysis of trade-offs between the two different architectures (integrated vs. distributed)? In a distributed Web service architecture, Web services are encapsulated components with a clearly defined functionality. A connection to a Web service can be established through an interface that necessarily uses HyperText Transfer Protocol (HTTP) as a protocol. Hence, all exchanges of data between the client and the Web service have to be marshaled through HTTP. This might possibly lead to a performance bottleneck for applications where large quantities of data must be exchanged. The primary concerns of the developer of a Web service, however, should be: (1) that the interface is clearly defined and accessible to the client; (2) that the Web service functionality adheres to the interface definition; and (3) that the input data is in the correct format as specified by the Web service description.

An integrated solution consists of tightly coupled components that use the native communication infrastructure of the programming language to exchange messages. Access to a database takes place through direct connections (e.g. JDBC in the case of Java), and marshalling is controlled by the programming language itself and usually not exposed to the developer. Since all components reside on the same machine, performance bottlenecks resulting from data input and output are unlikely to occur. At the same time the accessibility of components within an integrated architecture is limited to the scope of the programming language, since it does not expose itself to the "outside world" as is the case with a Web service interface.

Different needs and situations call for the use of MCE as part of SDSS including individual versus group users, time-critical (such as crisis management) and time non-critical (such as planning) applications, frequently changing application domains versus stable domains, and advanced versus inexperienced users. A simple analysis of trade-offs among the integrated and distributed architectures may be conducted by evaluating a number of characteristics such as computational performance, scalability, reliability, portability of the implementation, and extensibility of MCE tools. The computational performance and reliability favor tightly integrated (monolithic) architectures. However, the scalability, portability, and extensibility favor distributed architectures. Hence, the choice of the architecture should be dictated by the application type and context.

Participatory decision situations involve dynamic decision processes, diverse user information needs, and various participation arrangements ranging from same place – same time to different place – different time. Under these circumstances, the distributed architecture for MCE tools is more advantageous than the tightly integrated architecture, due to its potential for high portability, scalability, and extensibility. In the following section we discuss the implementation of distributed architecture for CM.

3 Choice Modeler Architecture

Currently, the most common version of distributed architecture is the three-tier software architecture for system implementation (Peng and Tsou 2003). The first tier, called the *client tier*, includes a user-side Web browser and user-resident Java applets/HTML documents. Users interact with the client tier via a graphical user interface (GUI). The primary function of the client tier is to accept users' data requests and to display the results. The second tier, called the *middleware tier*, includes the Web Server and the Server Connectors (such as Servlet connectors or Active Server Pages – ASP connectors), which bridge the communication between clients and the server. The third tier is the *data storage tier*, which includes the Database Server. The three-tier software architecture can be deployed on different hardware configurations.

The specific implementation of three-tier architecture for CM is comprised of the client, the Choice Modeler server, and the database tier (Figure 1). The architecture in Figure 1 implements the concept of "server-side" processing and a thin client. The client component in the architecture is any Java-enabled Web browser, through which the user

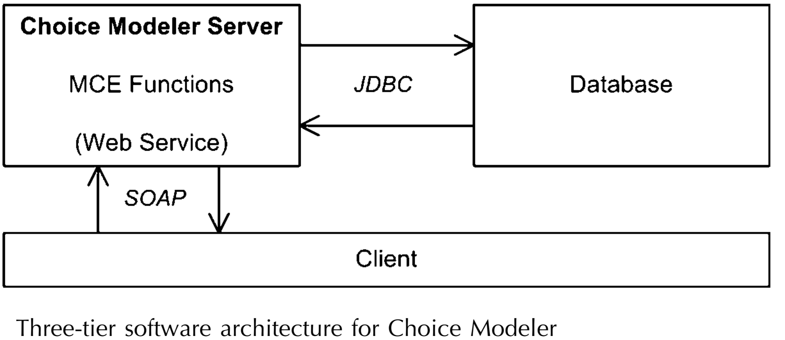

Figure 1 Three-tier software architecture for Choice Modeler

communicates with the MCE functions of the CM server and the database. The client communication with the database occurs through the CM server. The relational database stores user selections for various functional components of MCE including evaluation criteria, criteria types (benefit, cost), criteria weights, and the results of MCE functions executed by the user such as decision option ranking and sensitivity analysis. The user-specific selections and evaluation results can be stored for multiple users. A one-to-one relationship is maintained between a user and the user-specific data.

Data is organized into relations (tables) with a core user table related to other tables containing decision options criteria, criterion scores, weights, and their derivates resulting from the execution of various functions of MCE.

The CM server is responsible for retrieving data from the database and for executing MCE functions selected by the user. Two alternative implementations of MCE functions were considered, namely, implementing each MCE function as a separate Web service, or implementing all MCE functions of CM as one Web service. The former alternative may involve a 'thick' client establishing a transparent chain between the Web services while the latter involves a 'thin' client communicating with a data processing server. We chose for CM the latter alternative for its superior communication performance between the thin client and the server. The overarching concern was to reduce the number of connections between the client and the server to as few as possible under the scenario of using CM in a participatory decision making problem that might involve hundreds of users.

The decision to implement the CM server as a Web service was dictated by its potential reusability in other Web-based SDSS. In adopting the concepts of server-side processing and thin clients the architecture for CM is similar to architectural solutions for Web-based MCE proposed by others (e.g. Bernard et al. 2003, Karnatak et al. 2007). The alternative architectural solutions, which were not adopted for CM, involve a thick client and a transparent chain between the Web services. The decision not to adopt the former was made in order to promote the MCE portability, which is easier to achieve with a thin client approach. The decision not to adopt the latter was dictated primarily by choosing the efficiency and reliability of data communication over the flexibility offered by chained Web services.

Choosing a thin client running in a standard Internet browser and the reliance on Web service architecture allows for the integration of CM into a web service infrastructure. The usage of SOAP and WSDL enables service chaining and potentially allows for the usage of CM in combination with technologies such as Ajax in a Web 2.0 framework. Central to the Web 2.0 concept is the creative engagement of the user through rich user interfaces and the collaborative creation of data (O'Reilly 2005). In CM, the access to the database is limited to server-side components. In order to give the user control over data, the CM functionality would need to be extended to allow for data upload by the user.

The implementation followed the current best practices of creating a Web service, in which the service description is provided in Web Service Description Language (WSDL), the service invocation is handled according to Simple Object Access Protocol (SOAP), and the data transport protocol is HTTP. The web server and the data server may reside on the same physical machine or on separate machines. The MCE functions of the Web service were implemented in Java and the PostgreSQL object-relational database using the PostGIS plug-in to store and manage data. PostGIS adds support for storing geographic objects in the PostgreSQL database.

4 Multiple Criteria Evaluation Functions of Choice Modeler

Typical MCE functions include eliciting preferences in regard to evaluation criteria in the form of weights or trade-offs, standardizing criterion scores, combining standardized scores with criterion preferences into an overall evaluation measure for each decision option, and testing the sensitivity of evaluation results to changes in criteria weights and standardized scores. These functions typically align into a MCE workflow, which mirrors the steps of the rational model of choice composed of intelligence, design, and choice (Malczewski 1999, p. 96). In addition to these standard functions CM incorporates extensive sensitivity analysis capabilities for support of groups participating in multiple criteria evaluation processes (Figure 2).

The workflow schema presented in Figure 2 shows the major functional clusters comprising CM: global, local, and spatial sensitivity analysis, individual and group ranking, and evaluation agreement assessment. This workflow can be applied to any

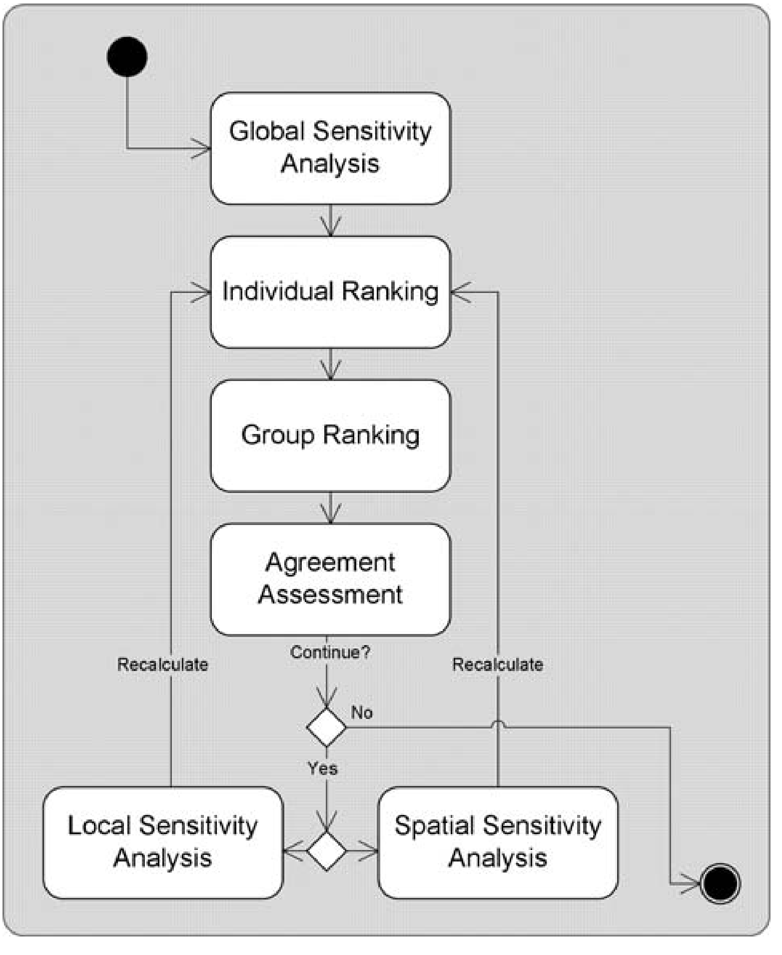

Figure 2 A workflow flowchart for Choice Modeler

spatial MCE problem once the basic structure of the problem has been established and the decision table is available. It is important to emphasize that the schema presented in Figure 2 is not theoretically limited to supporting only one specific workflow. Different alternative MCE workflows, comprised of fewer functional clusters than depicted in Figure 2, can also be supported by Choice Modeler. For example, a workflow applicable to a participatory planning situation might include only global sensitivity analysis, individual ranking, group ranking, and the ranking agreement assessment.

4.1 Functional Clusters

Figure 3 presents the workflow schema in greater detail, including functional clusters that correspond to the steps of the workflow and more specific functions that build the clusters. We describe the clusters and their functions below.

4.1.1 Global sensitivity analysis

The workflow presented in Figure 3 starts from reducing the cognitive complexity of the decision space comprised of decision options, criteria, and criteria weights. This is accomplished by identifying the evaluation criteria that contribute the most and those that contribute the least to the ranking of decision options. The computational approach to finding such criteria, called Global Sensitivity Analysis (GSA), decomposes the variance of average shift in ranks due to the variability in criteria weights, and involves the following three steps (Saisana et al. 2005):

1. Generate random weight samples of all criteria using probability distribution functions, and for each sample compute the rank order of decision options.
2. For each rank order, calculate the average shift in ranks (ASR) compared to the rank order computed with equal weights. The ASR captures the relative shift in the position of the entire set of options and quantifies it as the sum of absolute differences between the current option rank and the reference (equal weight) rank, divided by the number of all options.
3. Using results from steps (1) and (2) calculate a first order index and a total order index (S_i and S_{Ti}, respectively).

S_i quantifies a fractional contribution of a given criterion weight to the variance in option ranking taken independently from other weights. S_{Ti} captures the overall contribution of a given criterion weight including its interactivity with other criteria. As a result, it is possible to examine the complexity of the interrelationships of the evaluation criteria and exclude those criteria that play a minor role in the evaluation. The elimination of inconsequential criteria clearly helps to reduce the cognitive burden of performing the evaluation.

4.1.2 Individual ranking

Following the results of GSA, the user may choose to evaluate decision options using a subset of criteria identified as the influential criteria. The rank order is computed based on criteria, criterion values, and criterion weights, which are aggregated into an overall evaluation score using an Ideal Point function based on the TOPSIS technique (Hwang and Yoon 1981). The decision options are rank ordered on the basis of descending evaluation score values from the most to the least favored.

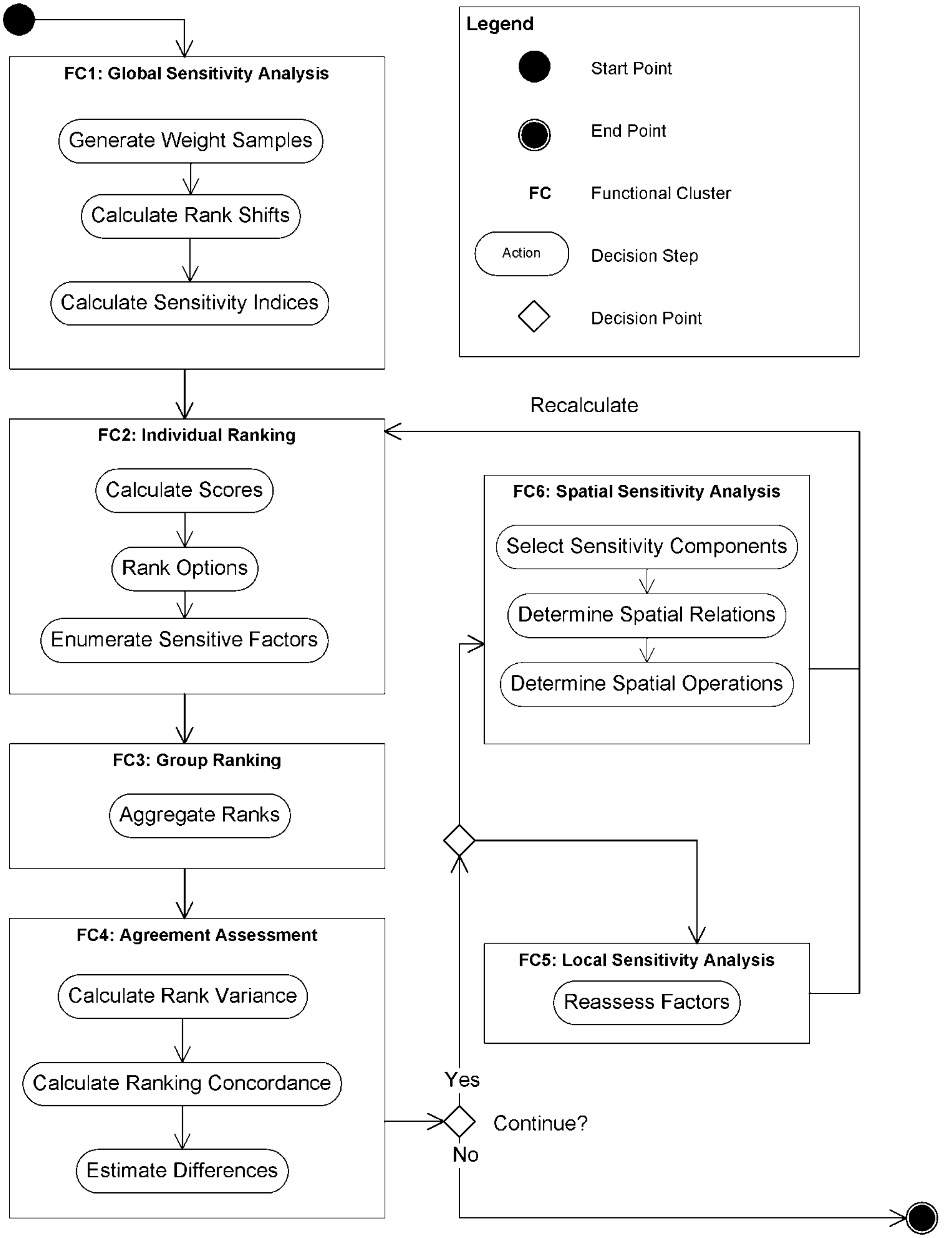

Figure 3 Detailed workflow schema for Choice Modeler

Three additional functions can be used to enumerate sensitive factors of the rank order (see FC2 in Figure 3). The Pairwise Option Path Similarity function calculates for any two options the non-weighed similarity index of all selected criteria. If, for the majority of criteria, the pairwise option difference in scores for each respective criterion is below the user-specified threshold, the two options are considered indifferent (Belton and Stewart 2002). This function may be helpful if the user wants to reduce the number of options but is uncertain as to which options to delete. If two options score similarly, then one of them may be deleted based on intangibles that are not expressed by the evaluation criteria. The other two functions include Critical Pairwise Score Difference, which pinpoints those options that score close to each other and thus are potentially prone to rank switching, and Pairwise Break Even, which calculates the necessary change in the weight of a selected evaluation criterion so that the first and second rank options can switch their ranks (Pannell 1997).

4.1.3 Group ranking

Individual rankings are synthesized into a group ranking by computing a modified Borda score. The rankings are treated as votes. The procedure for computing the Borda score takes into consideration the order of significance of what is being voted upon (like decision options). The voting position of a decision option is determined by adding the ranks for each option from every voter using the Borda social preference function (Hwang and Lin 1987). The ranked vote method prevents a contentious option, which ranks very high with some participants and very low with others, from winning. The algorithm for the modified Borda score is presented in Appendix A.

4.1.4 Agreement assessment

The role of agreement assessment functions (rank variance, ranking concordance, and peer-to-group popularity) is to inform the participants about the differences between their individual rankings and the overall group ranking. The rank variance function helps to establish the level of convergence/divergence among the group members in regard to the position of each decision option in the ranking. This is accomplished by computing a ranked vote variance for each decision option. Thus, information about the position of a given decision option in the ranking is enhanced with the level of agreement about this position (rank). The algorithm for the ranked vote variance along with an example is presented in Appendix B.

In addition to group rank order and ranking variance, CM offers a function, called Ranking Concordance, for assessing the overall ranking agreement between an individual and a group. The measure of agreement is based on determining a shift in the ranks between the user and group rankings, and is defined as follows (Saisana et al. 2005):

$$R_S = \frac{1}{M}\left(\sum_{o=1}^{M}\left|rank_{ref}(Y_o) - rank(Y_o)\right|\right) \tag{1}$$

where R_S is the average shift in ranks, M is the number of decision options under evaluation, rank Y_o is the rank of decision option o in the individual ranking, and $rank_{ref}$ Y_o is the rank of decision option o in the group ranking.

The average shift in ranks captures a relative shift in the position of the entire set of options. Rank shift is quantified as the average of the absolute differences in option ranks with respect to a reference prioritization over all M options. Here, the group prioritization is considered as the reference prioritization. Equation (1) helps in determining the average shift in ranks between the two rank orders.

The average shift in rank value (R_S) does not tell us much about the agreement between the rankings, and hence it must be related to a reference value. We use for the reference value the worse case scenario given by two symmetrically different rankings where the highest ranked option in the individual ranking is the lowest ranked option in the group ranking, etc. The reference ranking is defined formally as follows:

$$R_{S\_worst} = \frac{2}{M} * d(M - d) \tag{2}$$

where M is the number of decision options and d is defined depending on M as $d = (M + 1)/2$ when M is odd and $d = M/2$ when M is even. Finally the Ranking Concordance composite indicator is defined as follows:

$$R_C = \left(1 - \left(\frac{R_S}{R_{S\_worst}}\right)\right) * 100\% \tag{3}$$

The process of agreement assessment can be further extended by estimating the causes of differences in option rankings among the participants. This information enhances group deliberation by directing the participants to a discussion on the most conflicting issues related to the decision situation rather than forcing them to unravel the nuances of MCE mathematics in search for the sources of ranking discrepancies. In particular, the potential reasons for the differences in option rankings can be derived from the explicit comparisons of criteria lists and criteria weights between an individual participant and a group. The participant starts from comparing his or her criterion choices with the criteria lists of other participants. This comparison reveals the frequency of the corresponding criterion pairs. For each criterion its presence on the compared lists is accounted for in the following formula:

$$P_{nm} = \frac{\sum_{k=1,k\neq n}^{N} x_{km}}{N - 1} * 100 \tag{4}$$

where P_{nm} is the n-th stakeholder *peer-to-group popularity* of criterion m, n and k stand for stakeholders, N is the total number of stakeholders, and the variable x_{km} equals one if criterion m was selected by stakeholder k and zero otherwise. For a given stakeholder and an evaluation criterion, Equation (4) assesses the proportion of other participants who selected this particular criterion compared to the total number of participants in the group excluding the stakeholder himself. An example of the peer-to-group popularity measure is given in Appendix C.

4.1.5 *Local sensitivity analysis*

Following the agreement assessment the participant may decide to accept his/her current ranking of decision options or investigate the stability of the ranking with sensitivity analysis. The former ends the MCE workflow while the latter leads to two additional clusters of functions: Local Sensitivity Analysis (LSA) and Spatial Sensitivity Analysis.

In LSA, the participants may add a new criterion to a criteria list; delete a criterion from the list; or change (increase/decrease) a criterion weight. A simple rule of thumb may be followed in establishing the sensitivity of rank-order to shifting weights, which states that if the change in a criterion weight by *n*-percent results in the change in a rank-order by less than *n*-percent then one may conclude that the ranking is not sensitive to shifts in the importance of a given criterion (Longley et al. 2005).

Each of the changes involving the criterion list and criterion weights may affect the rank-order of options, and thus a new Ideal Point calculation is triggered, followed by another group aggregation and agreement assessment.

4.1.6 Spatial sensitivity analysis

In a general sense, the purpose of spatial sensitivity analysis is to evaluate: (1) the effects of uncertainty in the geographical distribution of criterion values; (2) spatial variability of criterion importance (weights); and (3) distributional effects of decision options on the stability of option ranking. CM offers one specific function for analyzing the distributional effects of decision options using distance relationships. The function computes a matrix of spatial weights based on the distances between the user-designated locations, reflecting user's spatial preferences, and the locations of decision options. Such distance-derived weights are then used to recalculate composite evaluation scores for the options. The recalculated scores may modify the rank order of options. If the user is satisfied with this spatially weighted ranking, s/he may accept it, which triggers the recalculation of group ranking and the agreement assessment.

4.1.7 Choice modeler and SDSS/PSS

CM was designed to be either a standalone tool or a component of SDSS. Consequently, CM can be also utilized in Planning Support Systems (PSS) which, according to Harris (2001) and Klosterman (2001), should include tools for the evaluation of strategic plans and the consequences of what-if scenarios. Within the realm of SDSS/PSS, CM addresses the critical issue of incorporating individual preferences into the decision process. Given the planning process, CM could enhance the analysis of the sensitivity of plans to changes in input values that are associated with the criteria used for plan assessment. The elimination of inconsequential criteria with GSA allows for shifting the focus of the 'with-without' analysis from a large set of planning criteria to those criteria that have a crucial impact on the variability of plans. The functional cluster of individual ranking (Figure 3) can be adapted as a tool that provides an index of plan performance in the form of composite scores of what-if scenarios. Given the fact that alternative plans should be as different as possible (Harris 2001), the pairwise option path similarity index can be used to eliminate plans that perform alike with respect to the planning criteria used. Finally, in the case of collaborative planning, the decision about preferred plans should be based on input from multiple stakeholders. The group ranking functional cluster is designed to accomplish this goal.

4.2 Choice Modeler and Participatory Geographic Information System for Transportation

CM was originally developed in the context of the "Participatory Geographic Information System for Transportation Decision Making" research project conducted at the University of Washington, University of Wyoming, and San Diego State University (Nyerges et al. 2006a, Participatory Geographic Information System for Transportation – PGIST;

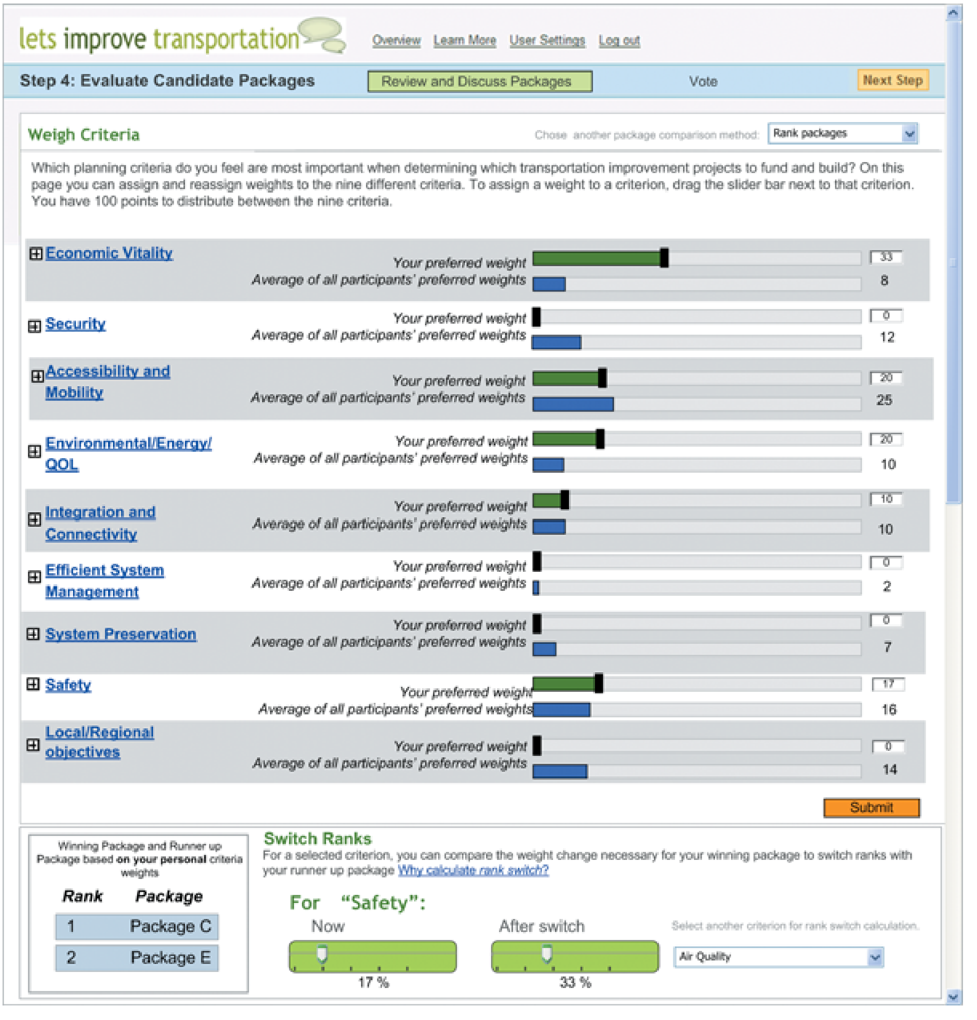

Figure 4 Criterion weighting and pairwise break even functions

http://www.pgist.org). PGIST and subsequently CM were designed as tools aimed at finding out what Internet platform designs and capabilities, including Geographic Information Systems technology, could improve public participation in analytic-deliberative transportation decision making within large groups.

Due to time and budget constraints only one MCE function was adopted in PGIST, namely, criteria weighting using the rating technique (Malczewski 1999), which is a component of individual ranking (see MCE workflow in Figure 3). We have included the screen mockup of criterion weighting and pairwise break even functions developed for a standard, Java-enabled Web browser in order to give the reader a sense of how CM functions could be accessed by the users (Figure 4). The function screen, just like any other function screen in PGIST, is composed of a combination of *slates* and *bars*, where the slate is the main part of the screen presenting different information structures (maps, charts, threaded conversations, etc.), and the bar is a collection of tools available for the analysis with MCE functions (Nyerges et al. 2006b).

The function screen in Figure 4 shows two MCE functions of Choice Modeler presented in the context of an online experiment called "Let's Improve Transportation", which took place between 16 October and 13 November 2007 (see http://www.letsimprovetransportation.org for additional details). The experiment, which was run on the PGIST platform installed on the server of the Department of Geography at the University of Washington, involved 135 randomly selected residents of the King, Snohomish, and Pierce counties in the western part of Washington State. The participants discussed, analyzed and recommended a package of 27 transportation improvement projects in the Puget Sound Metropolitan Area. The usability results of PGIST, including the criterion weighting function depicted in Figure 4, are still being evaluated as of the writing time of this article. The slate in Figure 4 is filled with nine criteria used by the experiment participants in evaluating various transportation improvement projects. The sliding bars allow each participant to adjust his/her weights assigned to the criteria according to the rating technique. Each bar reflecting a criterion weight assigned by a participant has a corresponding bar showing the current average weight at a given moment of time during the participatory evaluation process. The horizontal bar located at the bottom of the function screen shows another MCE function implemented in CM called Pairwise Break Even, which is one of three functions that can be used to enumerate sensitive factors of the rank order (see FC2 in Figure 3). The function returns the weight change for a selected criterion, which would be necessary for the runner-up option to switch the rank with the top-ranked option.

5 Conclusions

In this article, we outlined the rationale behind applying Web service based architecture for multiple criteria evaluation of decision options involving individuals and groups. We also presented a prototype functional cluster, called Choice Modeler, which extends multiple criteria evaluation by addressing a number of uncertainties associated with decision criteria, criteria weights, and decision options. Three functional approaches to sensitivity analysis adopted in CM, including local, global, and spatial sensitivity analysis, allow for a systematic treatment of uncertainties associated with MCE components. The CM design also includes a novel use of database to support group MCE processes through maintaining a one-to-one relationship between each individual user and the user-specific data. Such organization of the database allows both individual data storage for each user and supports group ranking and agreement assessment functions.

CM represents a first step in the direction of providing multiple criteria evaluation functions for individuals and groups in the form of distributed Web services. MCE Web services, that are based on distributed architectures, can be conceptualized as a library of plug-and-play tools that can be reconfigured and connected on an as needed basis by software engineers to support a variety of collaborative decision processes, which are idiosyncratic in nature but at the same time share certain evaluative process steps. These shared steps such as defining decision/evaluation criteria, expressing preferences in regard to the criteria, ranking of decision options, and voting on decision options can be encapsulated in Web services, which can eventually become a library of participatory spatial decision support services.

Future research efforts should pursue the agenda of developing MCE functions as open source standardized Web services, similar to work reported by Hall and Leahy (2006). Such services could then be published on the Web and create a MCE service infrastructure. Standardized services could be linked through a mechanism called service chaining, which would facilitate their reuse and support of various participatory decision making workflows. Implementing MCE functions as Web services might also spur the development of meta-knowledge, helping to select an appropriate service for a given evaluation task. An example can be a choice between utility function-based decision rules and outranking decision rules in MCE.

Acknowledgements

The authors acknowledge thoughtful comments of two anonymous reviewers who contributed to the final version of the manuscript. This research was supported in part by National Science Foundation Grant No. EIA 0325916, funded through the Information Technology Research Program, and managed in the Digital Government Program. The principal investigator on this project was Timothy L. Nyerges, and the co-principal investigators were G. Scott Rutherford and Terry Brooks at the University of Washington, Piotr Jankowski at San Diego State University, and Rhonda Young at the University of Wyoming. The authors are solely responsible for the content.

References

Andrienko N V and Andrienko G L 1999 Interactive maps for visual data exploration. *International Journal of Geographical Information Systems* 13: 355–74

Andrienko N V and Andrienko G L 2001 Intelligent support for geographic data analysis and decision making in the Web. *Journal of Geographic Information and Decision Analysis* 5: 115–28

Andrienko G L, Andrienko N V, and Jankowski P 2003 Building spatial decision support tools for individuals and groups. *Journal of Decision Systems* 12: 193–208

Armstrong M P 1993 Perspectives on the development of group decision support systems for locational problem solving. *Geographic Systems* 1: 69–81

Armstrong M P, Rushton G, Honey R, Dalziel B, Lolonis P, De S, and Densham P 1991 Decision support for regionalization: A Spatial Decision Support System for regionalizing service delivery systems. *Computers, Environment and Urban Systems* 15: 37–53

Barton J, Plume B, and Parolin B 2005 Public participation in a spatial decision support system for public housing, *Computers, Environment and Urban Systems* 29: 631–53

Beinat E and Nijkamp P 1998 Multicriteria analysis for land-use management. In Beinat E and Nijkamp P (eds) *Multiple Criteria Decision Making: An Integrated Approach*. Dordrecht, Kluwer: 253–70

Belton V and Stewart T J 2002 Performance modeling: Outranking. In Beinat E and Nijkamp P (eds) *Multiple Criteria Decision Analysis. An Integrated Approach*. Dordrecht, Kluwer: 106–11

Bernard L, Ostländer N, and Rinner C 2003 Impact assessment for the Barents Sea region: A geodata infrastructure approach. In *Proceedings of the Sixth AGILE Conference on Geographic Information Science*, Lyon, France

Berners-Lee T, Hendler J, and Lassila O 2001 The Semantic Web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American* 284(5): 34–43

Best B D, Halpin P N, Fujioka E, Read A J, Qian S S, Hazen L J, and Schick R S 2007 Geospatial Web services within a scientific workflow: Predicting marine mammal habitats in a dynamic environment. *Ecological Informatics* 2: 210–23

Burton J, Plume J, and Parolin B 2005 Public participation in a spatial decision support system for public housing. *Computers, Environment and Urban Systems* 29: 617–29
Carver S J 1991 Integrating multi-criteria evaluation with geographical information systems. *International Journal of Geographical Information Systems* 5: 321–39
Carver S, Evans A, Kingston R, and Turton I 1998 Geographical Information Systems on the World Wide Web: Improving public participation in environmental decision making. In *Proceedings of Conference of the European Association for the Study of Science and Technology*, Lisbon, Portugal
Chen J, He C, Jiang J, and Han G 2008 Reconciliation of inconsistent perspectives in collaborative GIS. *Cartography and Geographic Information Science* 35: 77–89
Choi J-Y, Engel B A, and Farnsworth R L 2005a Web-based GIS and spatial decision support system for watershed management. *Journal of Hydroinformatics* 7: 165–74
Choi J-Y, Engel B A, Theller L, and Harbor J 2005b Utilizing Web-based GIS and SDSS for hydrological land use change impact assessment. *Transactions of the American Society of Agricultural Engineers* 48: 815–22
Cowen D 1988 GIS versus CAD versus DBMS: What are the differences? *Photogrammetric Engineering & Remote Sensing* 54: 1551–5
Densham P J and Goodchild M F 1989 Spatial Decision Support Systems: A research agenda. In *Proceedings of GIS/LIS '89*, Phoenix, Arizona: 707–16
Densham P J and Rushton G 1988 Spatial Decision Support Systems for locational planning. In Golledge R and Timmermans H (eds) *Behavioural Modelling in Geography and Planning*. London, Croom-Helm: 56–90
Dragićević S and Balram S 2004 A Web GIS collaborative framework to structure and manage distributed planning processes. *Journal of Geographical Systems* 6: 133–53
Dymond R L, Regmi B, Lohani V K, and Dietz R 2004 Interdisciplinary Web-enabled spatial decision support system for watershed management. *Journal of Water Resources Planning and Management* 130: 290–300
Eastman J R, Kyem P A K, Toledano J, and Jin W 1993 *GIS and Decision Making*. Geneva, UNITAR
Faber B, Watts R, Hautaluoma J, Knutson J, Wallace W, and Wallace L 1994 A groupware-enabled GIS. In *Proceedings of the GIS '94 Symposium*, Vancouver, British Columbia: 551–61
Gomes E G and Lins M P E 2002 Integrating geographical information systems and multi-criteria methods: A case study. *Annals of Operations Research* 116: 243–69
Hall G B and Leahy M G 2006 Internet-based spatial decision support using open source tools. In Balram S and Dragicević S (eds) *Collaborative Geographic Information Systems*. Hershey, PA, Idea Group: 237–62
Harris B 2001 Sketch planning: Systematic methods in planning and its support. In Brail R K and Klosterman R E (eds) *Planning Support Systems Integrating Geographic Information Systems, Models, and Visualization Tools*. Redlands, CA, ESRI Press: 59–80
Hwang C-L and Lin M-J 1986 *Group Decision Making Under Multiple Attribute Decision Making: Methods and Applications*. Berlin, Springer-Verlag
Hwang C-L and Yoon K 1981 *Multiple Attribute Decision Making: Methods and Applications*. Berlin, Springer-Verlag
Jankowski P 1995 Integrating geographical information systems and multiple criteria decision making methods. *International Journal of Geographical Information Systems* 9: 251–73
Jankowski P and Stasik M 1997 Spatial understanding and decision support system: A prototype for public GIS. *Transactions in GIS* 2: 73–84
Jankowski P, Andrienko G L, and Andrienko N V 2001 Map-centered exploratory approach to multiple criteria spatial decision making. *International Journal of Geographical Information Science* 15: 101–27
Jankowski P, Nyerges T, Smith A, Moore T J, and Horvath E 1997 Spatial group choice: A spatial decision support tool for collaborative decision making. *International Journal of Geographical Information Systems* 11: 577–602
Joerin F, Theriault M, and Musy A 2001 Using GIS and outranking multi-criteria analysis for land-use suitability assessment. *International Journal of Geographical Information Science* 15: 153–74

Jung C and Sun C-H 2006 Development of a GIService on spatial data mining for location choice of convenience stores in Taipei City. In Wu H and Zhu Q (eds) *The International Society For Optical Engineering*. Bellingham, WA, The International Society For Optical Engineering doi: 10.1117/12.713149

Karnatak H C, Saran S, Bhatia K, and Roy P S 2007 Multicriteria spatial decision analysis in a Web GIS environment. *Geoinformatica* 11: 407–29

Keen P G W and Scott-Morton M S 1978 *Decision Support Systems: An Organizational Perspective*. Reading, MA, Addison-Wesley

Kingston R, Carver S, Evans A, and Turton I 2000 Web-based public participation geographical information systems: An aid to local environmental decision-making, *Computers, Environment and Urban Systems* 24: 109–25

Klosterman R E 2001 Planning support systems: A new perspective on computer-aided planning, planning support systems. In Brail R K and Klosterman R E (eds) *Planning Support Systems Integrating Geographic Information Systems, Models, and Visualization Tools*. Redlands, CA, ESRI Press: 1–23

Laaribi A, Chevallier J J, and Martel J M 1996 A spatial decision aid: A multi-criteria evaluation approach. *Computers, Environment and Urban Systems* 20: 351–66

Longley P A, Goodchild M F, Maguire D J, and Rhind D W 2005 *Geographic Information Systems and Science* (Second Edition). New York, John Wiley and Sons

MacEachren A M 2005 Moving geovisualization toward support for groupwork. In Dykes J, MacEachren A M, and Kraak M-J (eds) *Exploring Geovisualization*. Amsterdam, Elsevier: 445–62

MacEachren A M and Brewer I 2004 Developing a conceptual framework for visually-enabled geocollaboration. *International Journal of Geographical Information Science* 18: 1–34

Malczewski J 1996 A GIS-based approach to multiple criteria group decision-making. *International Journal of Geographical Information Science* 10: 955–71

Malczewski J 1999 *GIS and Multicriteria Decision Analysis*. New York, John Wiley and Sons

Malczewski J 2006 GIS-based multicriteria decision analysis: A survey of the literature. *International Journal of Geographical Information Science* 20: 703–26

Marinoni O 2005 A stochastic spatial decision support system based on PROMETHEE. *International Journal of Geographical Information Science* 19: 51–68

Nyerges T, Brooks T, Jankowski P, Rutherford G S, and Young R 2006a Web portal implementation to support public participation in transportation decision making. In *Proceedings of the Seventh Annual International Conference on Digital Government Research*, San Diego, California

Nyerges T, Ramsey K, and Wilson M 2006b Design considerations for an Internet portal to support public participation in transportation improvement decision making. In Balram S and Dragicevic S (eds) *Collaborative Geographic Information Systems*. Hershey, PA, Idea Group: 207–35

OGC 2008 OpenGIS Specifications (Standards). WWW document, http://www.opengeospatial.org/standards

O'Reilly T 2005 What is Web 2.0? Design Patterns and Business Models for the Next Generation of Software. WWW document, http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

Pannell D 1997 Sensitivity analysis of normative economic models: Theoretical framework and practical strategies. *Agricultural Economics* 16: 139–52

Pereira J M C and Duckstein L 1993 A multiple criteria decision-making approach to GIS-based land suitability evaluation. *International Journal of Geographical Information Systems* 7: 407–24

Peng Z R and Tsou M H 2003 *Internet GIS: Distributed Geographic Information Services for the Internet and Wireless Networks*. New York, John Wiley and Sons

Rinner C 2003 Web-based spatial decision support: Status and research directions. *Journal of Geographic Information and Decision Analysis* 7: 14–31

Rinner C and Malczewski J 2002 Web-enabled spatial decision analysis using ordered weighted averaging. *Journal of Geographical Systems* 4: 385–403

Saaty T 1980 *The Analytic Hierarchy Process*. New York, McGraw Hill

Saisana M, Saltelli A, and Tarantola S 2005 Uncertainty and sensitivity analysis techniques as tools for the quality assessment of composite indicators. *Journal of Royal Statistical Society* 168: 307–23

Sprague R H and Carlson E D 1982 *Building Effective Decision Support Systems*. Englewood Cliffs, NJ, Prentice-Hall
Stock C, Bishop I D, O'Connor A N, Chen T, Pettit C J, and Aurambout J-P 2008 SIEVE: Collaborative decision-making in an immersive online environment. *Cartography and Geographic Information Science* 35: 133–44
Tkach R J and Simonovic S P 1997 A new approach to multi-criteria decision making in water resources. *Journal of Geographic Information and Decision Analysis* 1: 25–43
Wang L and Cheng Q 2007 Design and implementation of a Web-based spatial decision support system for flood forecasting and flood risk mapping. In *Proceedings of the Geoscience and Remote Sensing Symposium*, Barcelona, Spain
Yu F-C, Chen C-Y, Lin S-C, Lin Y-C, Wu S-Y, and Cheung K-W 2006 A web-based decision support system for slopeland hazard warning. *Environmental Monitoring and Assessment* 127: 419–28
Zhu X, McCosker J, Dale A P, and Bischof R J 2001 Web-based decision support for regional vegetation management. *Computers, Environment and Urban Systems* 25: 605–27

Appendix A

The algorithm for computing ***Ranked Vote*** Score (a modified Borda score)

Step 1:

Determine the number of unique items voted for (N)
Find the voter with the highest count of items voted for (C)
Determine the worst item position in a vote as follows:

If C + 1 < N:

W = C + 1

Else:

W = N

Step 2:

∀ item voted for:

$$T_i = \sum_{j=1}^{K} W - P_j^i$$

where T_i denotes the sum of the position values for the *i*-th item voted for; K denotes the number of voters; P_j^i denotes the position the *i*-th vote item has in the *j*-th voter's ordered vote; starting from 0 (the highest position in the vote is zero). If the *j*-th voter did not vote for the *i*-th item then P_j^i equals the number of items in the *j*-th voter's vote.

Step 3:

∀ item voted for:

$$S_i = \frac{T_i}{K * W} * 100$$

where S_i denotes the score (modified Borda) for the ith vote item.

The voted items are then ordered based on the S_i score.

Every item that a voter did not vote on is given a position of one beyond the last position for this voter. This means that the items not voted on are treated equally and they occupy the same last position in the ranked vote.

Appendix B

The algorithm for computing the ***Variance*** of Ranked Vote Score.

Step 1:

Same as in "Ranked Score" algorithm.

Step 2:

$\forall$ item voted for:

$$T_i^V = \sum_{j=1}^{K} \sum_{m=j+1}^{K} \left| P_j^i - P_m^i \right|$$

where T_i^V denotes the sum of the absolute differences for the *i*-th item for every pair of votes; K denotes the number of voters; P_j^i denotes the position the *i*-th vote item has in the *j*-th voter's ordered vote starting from 0 (the highest position in the vote is zero). If the *j*-th voter did not vote on the i-th item P_j^i equals the number of items in the *j*-th voter's vote, and $j, m \in K$.

Step 3:

$\forall$ item voted for:

$$V_i = \frac{T_i^V}{(W-1) * \left(> \frac{K}{2}\right) * \left(< \frac{K}{2}\right)} * 100$$

where V_i denotes the variance for the *i*-th vote item; $(> K/2)$ represents the number of vote pairs (P_j^i, P_m^i) for which the absolute difference in the voting position $|P_j^i - P_m^i|$ is greater than the ratio $K/2$ where K denotes the number of voters; $(> K/2)$represents the number of vote pairs (P_j^i, P_m^i) for which the absolute difference in the voting position $|P_j^i - P_m^i|$ is smaller than the ratio $K/2$.

Numeric Example of Computing the *Ranked Voted Score* and Its *Variance*

Assume there are three individuals (voters) participating in an evaluation (V1, V2, V3) and five decision options being evaluated (vote items: A, B, C, D, E). The matrix below represents the result of a group vote, such that each voter has voted ones an ordered list of vote items:

Vote position P_j^i	Voters		
	V1	V2	V3
0	A	B	E
1	B	C	D
2	C	A	A
3	D		
C + 1 = 5			
N = 5			
K = 3			

Borda Score for A-item

$T_A = (5 - 0) + (5 - 2) + (5 - 2) = 11$

$$S_A = \frac{11}{3 * 5} * 100 = 73$$

Variance for A-item

$$T_A^V = |0 - 2| + |0 - 2| + |3 - 3| = 4$$

$$\frac{K}{2} = 1.5$$

$$V_A = \frac{4}{(5 - 1) * 2 * 1} * 100 = 50$$

Appendix C

An example of computing the **peer-to-group popularity** measure.

Table C1 demonstrates the peer-to-group popularity heuristics for one of seven participants that selected five out of seven criteria. As an example, consider the popularity of criterion C_1 which was chosen by four participants apart from S_7. Therefore, the popularity of this criterion amounts to 4/6 (67%) of other users.

The information about the criteria choices can be augmented with information about the importance assigned to the favored criteria. Equation (5) provides a simple measure to accomplish this task:

$$W_{nm} = \frac{\sum_{k=1,k\neq n}^{N} i_{km}}{\sum_{k=1,k\neq n}^{N} x_{km}} * 100 \tag{5}$$

where *n, k, m,* and x_{km} has the same meaning as in Equation (4), i_{km} is a weight assigned to criterion *m* by participant *k*, and W_{nm} represents participant's *n* peer-to-group weight

Table C1 An example of a decision table for computing peer-to-group heuristics for stakeholder S_7: S_n – stakeholder *n*, C_m – criterion *m*, P_{7m} – peer-to-group popularity of criterion *m* for S_7, W_{7m} – peer-to-group weight of criterion *m* for S_7. Note that the criteria selected by a particular stakeholder are in bold frames, whereas values in S_nC_m cells represent criteria weights.

	C_1	C_2	C_3	C_4	C_5	C_6	C_7
S_1	40	10		5		45	
S_2		15	35			50	
S_3		40		40		20	
S_4	20	15	50			15	
S_5	50			25		18	7
S_6	15	40	5	5		20	15
S_7	14		26		20	25	15
Example for S_7:							
P_{7m}	67	83	50	67	0	100	33
W_{7m}	31	24	30	19	NA	28	11

for criterion *m*. The above equation averages the weight assigned to a selected criterion by other users that chose it as their favored criterion. Similarly to the peer-to-group popularity measure, the proposed peer-to-group weight approach excludes the participant for whom the measure is calculated.

Following the example in Table C1 we can calculate a peer-to-group weight for the first criterion and the participant S_7, which is simply the average of weights derived from decision makers S_1, S_4, S_5, and S_6, and equals 125/4 (31%).