Available online at www.sciencedirect.com
ScienceDirect
Environmental Modelling & Software 22 (2007) 269–277
Environmental Modelling & Software
www.elsevier.com/locate/envsoft

Empirical evaluation of decision support systems: Needs, definitions, potential methods, and an example pertaining to waterfowl management

Richard S. Sojda*

Northern Rocky Mountain Science Center, United States Department of the Interior – Geological Survey, 212 AJM Johnson Hall – Ecology Department, Montana State University, Bozeman, MT 59717, USA

Received 29 August 2004; received in revised form 16 March 2005; accepted 14 July 2005
Available online 23 February 2006

Abstract

Decision support systems are often not empirically evaluated, especially the underlying modelling components. This can be attributed to such systems necessarily being designed to handle complex and poorly structured problems and decision making. Nonetheless, evaluation is critical and should be focused on empirical testing whenever possible. Verification and validation, in combination, comprise such evaluation. Verification is ensuring that the system is internally complete, coherent, and logical from a modelling and programming perspective. Validation is examining whether the system is realistic and useful to the user or decision maker, and should answer the question: ''Was the system successful at addressing its intended purpose?'' A rich literature exists on verification and validation of expert systems and other artificial intelligence methods; however, no single evaluation methodology has emerged as preeminent. At least five approaches to validation are feasible. First, under some conditions, decision support system performance can be tested against a preselected gold standard. Second, real-time and historic data sets can be used for comparison with simulated output. Third, panels of experts can be judiciously used, but often are not an option in some ecological domains. Fourth, sensitivity analysis of system outputs in relation to inputs can be informative. Fifth, when validation of a complete system is impossible, examining major components can be substituted, recognizing the potential pitfalls. I provide an example of evaluation of a decision support system for trumpeter swan (*Cygnus buccinator*) management that I developed using interacting intelligent agents, expert systems, and a queuing system. Predicted swan distributions over a 13-year period were assessed against observed numbers. Population survey numbers and banding (ringing) studies may provide long term data useful in empirical evaluation of decision support.

Keywords: Decision support system; Verification; Validation; Empirical evaluation; Model; Trumpeter swan

1. Introduction

Decision support systems use a combination of models, analytical techniques, and information retrieval to help develop and evaluate appropriate alternatives (Adelman, 1992; Sprague and Carlson, 1982). Because such systems handle complex and poorly structured problems, they are difficult to empirically evaluate. However, it is still easy to argue that evaluation of all decision support systems is important. For example, decision support systems should contribute to reducing the uncertainty faced by managers when they need to make decisions regarding future options (Graham and Jones, 1988), and evaluating such a contribution quickly adds both architectural and statistical complexity. Another case in point, distributed decision making suits problems where the complexity prevents an individual decision maker from conceptualizing, or otherwise dealing with the entire problem (Boland et al., 1992; Brehmer, 1991). Designing the empirical evaluation of distributed systems is hardly straightforward because of these characteristics. This is somewhat different than the validation and calibration of individual ecological models as portrayed by Rykiel (1996).

* Fax: +1 406 994 6556.
E-mail address: sojda@usgs.gov

1364-8152/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.envsoft.2005.07.023

Ultimately, there are ecological and public policy reasons that increase the importance of ensuring that the right system has been built and been built correctly, as in the case of trumpeter swan management in North America. First, trumpeter swan numbers in some locales have not reached desired levels. Second, there is interest in fostering new migratory traditions. And third, some issues have been challenged with litigation. Such uncertainties and questions are common among many migratory bird issues around the world from both population and habitat perspectives. Any decision support or modelling efforts in this and related arenas require empirical evaluation in light of the importance they have in both environmental and socioeconomic dimensions.

In this paper, I focus on the evaluation of the knowledge-based, intelligent agent, and modelling components of decision support systems and the integration of those components. Evaluation of the overall acceptance among natural resource managers of decision support systems, or other socioeconomic measures of their success and failure, are certainly important but are not addressed.

2. Discerning differences between verification and validation

Some of the earlier definitions of verification and validation in relation to computer software and simulation modelling (Fishmann and Kiviat, 1968; Mihram, 1972; Adrion et al., 1982) have changed little over the years. Mihram (1972) is quite specific in focusing on the algorithm(s): "The determination of the rectitude of the completed model vis-à-vis its intended algorithmic structure." Verification has been defined by Adrion et al. (1982) as "demonstration of consistency, completeness, and correctness of the software." All seemed to agree that the focus of validation was the comparison of model output with observations from the real world. They also emphasized that verification should precede validation. These definitions are not absolute, but their use is becoming more definite over time. The following are from O'Keefe et al. (1987) for expert systems and were adapted from Boehm (1981) pertaining to software in general: "Validation means building the right system. Verification means building the system right." These have been frequently referenced by others in relation to decision support systems, especially artificial intelligence based systems (e.g., D'Erchia et al., 2001; Mosqueira-Rey and Moret-Bonillo, 2000; Plant and Gamble, 2003; Santos, 2001). A combined definition of verification and validation of software, provided by Wallace and Fujii (1989), was the analysis and testing "to determine that it performs its intended functions correctly, to ensure that it performs no unintended functions, and to measure its quality and reliability." The simplicity and completeness of Mihram's (1972) definition of validation in relation to simulation is attractive: "…the adequacy of the model as a mimic of the system which it is intended to represent." There has been a plethora of discussions about the semantics of evaluating more conventional models, and Johnson (2001) provides a fine summary related to natural resource management. The extensive literature review provided by Rykiel (1996) points out the less definitive nature of the semantics and concepts related to empirical evaluation of ecological models than those generally held for software and decision support systems. His description of the need to specify the ecological situation to which a model can be applied, and to specify what inferences can be made, is critical.

My specifications for verification and validation in reference to decision support systems draw almost entirely from the above authors. Verification is ensuring that the decision support system is internally complete, coherent, and logical from a modelling and programming perspective. Have the algorithms, knowledge, and other structures been correctly encoded? Are there no unintended consequences? Is there any superfluous code, e.g., production rules never used? Do input and output routines, themselves, function as intended? Thus, verification refers to adequacy of the software and computer code. Validation is less concerned with internal operation of the software and more concerned with usefulness to the user. I believe that validation can take two slightly different approaches. Foremost, validation is examining whether the decision support system achieved the project's stated purpose. For decision support systems, this is often related to helping the user(s) reach a decision(s). This could involve making better decisions, avoiding poor ones, or helping the user make them more quickly or with less data, information, and knowledge. Second, and especially in reference to individual models, validation can take the more restrictive meaning of whether a model is an adequate representation of the system it represents. This is sometimes described as black-box testing: do the inputs result in correct (and useful) outputs? Validation, typically, is comparing outputs from the model, or complete decision support system if feasible, to expectations as represented by some real-world standard. Sometimes, but not often, black-box testing is achieved on entire decision support systems; it is often difficult to do so because of their complexity. Whether stand-alone model or decision support system is being tested, I agree with Mihram (1972) and Adrion et al. (1982) that verification must occur before validation. This avoids the inadvertent situation where software provides expected outputs simply via calibration and correlation of input and outputs, rather than via scientific and logical relationships. I discuss this further in Section 3, along with more specific methods and criteria for verification and validation.

I use the term evaluation to encompass both verification and validation, but distinguish between them when used independently. I agree with Adelman (1992) that both should be part of the development process, and evaluators should specifically be part of the development team to foster iterative improvements. This is not to ignore the need for independent verification and validation of models and systems to ensure that the development team does not inadvertently err in their work.

3. Potential methods for empirical evaluation

3.1. An overview

Stuth and Smith (1993) followed the ideas of Eason (1988) and recommended iterative prototyping methods for decision

support system development. Verification and validation are part of that iterative process. Verification should be performed prior to any delivery of a working system, even if a prototype. General validation might be done at this stage as well, with detailed efforts performed later. If one agrees that software development can be a living process, then verification and validation are part and parcel to that process and need to continue as system refinements and redeployments continue (Carter et al., 1992; Stuth and Smith, 1993).

Sprague and Carlson (1982) recommend that organizations building their first decision support system recognize that it essentially is a research activity, and that evaluation should center on a general, ''value analysis''. Since then, it has become imperative that analytic and quantitative rigor be added beyond ''soft testimonials'' (Adelman, 1991, 1992; Andriole, 1989; Cohen and Howe, 1989). Sensitivity analysis can be a validation tool, especially for heuristic-based systems, and for systems where few or no test cases are available for comparison (Bahill, 1991; O'Keefe et al., 1987). Such an approach can accomplish two things. First, outputs can be examined, in relation to the domain being modelled, to see if they are approximately correct and whether they vary in expected directions with the changed inputs. Second, inputs at the extremes of continua can be shown to have appropriate outputs. Thus, sensitivity analyses can provide information about whether modelled systems have been adequately represented. Whenever validation is conducted, it is important to recognize to where inferences can be drawn, in space and time, from the validation data set. Another issue is the need to show not only how well a system performs, but also that it can avoid a catastrophic recommendation (Rushby, 1988). This is important in many natural resource venues because of the great concern for irretrievable or long term ecological changes. Other classic cases where unintended catastrophic recommendations of decision support systems must clearly be avoided include the operation of nuclear power plants and the management of endangered species.

It should be pointed out that progress during the past few decades in both theory and application development related to all aspects of decision support systems has lessened the need for some of the previously required internal verification of underlying logic and engines. Much of this is now handled within various shell environments used for developing applications. For example, logic inconsistencies are often now indicated on-the-fly during system development. This does not neglect the need to ensure that underlying causal relationships are correctly represented in models, knowledge bases, and domain ontologies. Automated methodologies for this are not usually available. Such confirmation needs to be done by careful development, intense personal checks, and objective peer review. Component testing as part of validation should also provide insights. As such, this becomes a grey area in the distinction between verification and validation. In this paper, I will continue to agree that verification should concentrate principally on ensuring that the ecological and decision making logic has been ''faithfully and accurately…translated into computer code or mathematical formalisms'' (Rykiel, 1996). The important point is that verification is not forsaken.

It is my sense that validation is often the more neglected part of evaluation of decision support systems, so I will focus there. However, I do not wish to slight verification as systems must be built based on sound cause–effect relationships and not on poorly understood relationships between input and output. If correlation analyses between model inputs and desired outputs are conducted without ensuring whether the underlying knowledge and cause–effect relationships have been correctly represented in the code, one runs the risk of unintentionally calibrating the model(s) to produce desired results. If faced with the dilemma of checking for calibration versus true validation, it can be useful to apply the system on a different data set or information base (if available), representing a different time frame or geography. Often, operating the system for extreme cases can also uncover heretofore undiscovered inconsistencies or errors.

3.2. Analogous concepts from artificial intelligence

Successful implementation of decision support and expert systems hinges on incorporating three evaluation procedures (Adelman, 1992): (1) examining the logical consistency of system algorithms (verification), (2) empirically testing the predictive accuracy of the system (validation), and (3) documenting user satisfaction.

Verification and validation of knowledge-based and other decision support systems are known to be more problematic than in general modelling for many reasons (Gupta, 1991). A few difficulties in verifying multiagent systems (O'Leary, 2001) are noteworthy, such as rule conflict, circularity, nonused or unreachable antecedents, and agent isolation. Plus, not only is it important for a system to handle common cases, it ought to be able to deal with extreme inputs. This latter ability is one characteristic often only found with human experts. However, extreme events are not only common in, but often drive, ecological systems. Wallace and Fujii (1989) provide a matrix of 41 techniques and tools that can be applied to 10 software verification and validation issues. Cohen and Howe (1989) take a slightly different approach specific to artificial intelligence methods, and they, too, discuss evaluation from the perspective of the software development life cycle. They emphasize empirical studies for such evaluation, whether focusing on verification or validation. For testing knowledge-based systems, Murrell and Plant (1997) provide a categorization of 145 automated techniques. Although the theory and application of intelligent agents and multiagent systems has blossomed for decision support system development, no widely accepted methodology pertaining to the evaluation of agent-based systems has yet emerged.

3.3. Alternative validation methods

3.3.1. Gold standard

Under some conditions, modelling researchers can test performance against a preselected gold standard. Mosqueira-Rey and Moret-Bonillo (2000) describe this for intelligent systems as having test cases with known, prior outcomes. Virvou and

Kabassi (2004) actually had such a set of cases based on expert opinion that they used for testing an intelligent graphical user interface. Often in natural resource issues, such a standard does not exist. This is particularly true with near real-time decision support that is expected to predict and guide future scenarios while those scenarios are, in fact, unfolding. Although this approach is theoretically desirable, I am not aware of an actual implementation that employs a gold standard for evaluation of an environmental decision support system. This is not surprising in a domain where problems tend to be ill-defined and the associated knowledge uncertain and incomplete. One might consider validation of a system with historic data (see Section 3.3.2) to be a type of gold standard, with two assumptions. First, the analysis of the historic data has to be of both sufficient accuracy and sufficient precision. Second, the system cannot be providing real-time recommendations or predictions. Finally, a gold standard may not feasibly exist for systems tackling NP-hard problems, as is often the case in artificial intelligence based decision support.

3.3.2. Real-time and historic data sets

In an ideal world, one could construct a decision support system and test its performance against actual scenarios as they unfold. This is not often possible because implementation of systems may need to be immediate. One alternative is to build the system using data, information, and knowledge from one set of situations and validate using an independent set, as done for crop yields (Priya and Shibasaki, 2001), for a bass bioenergetics model (Rice and Cochran, 1984), and for timber harvest (Wang and LeDoux, 2003). Prior versus post-testing is another example of this, and a decision support system for credit management was so validated by Kanungo et al. (2001). When a data-driven model is a significant part of the decision support system, sometimes the data can be randomly separated into two parts, one for model development and one for validation. Pretzsch et al. (2002) illustrate this using an extensive data set with a forest management simulator. Haberlandt et al. (2002) also took this approach for water quality assessments in river basins. A third option, when the decision support system is not data-based but rather knowledge-based, is to empirically evaluate predictions (outputs) from the system against a historic data set. This does assume that the logic underlying the system is constant over time. An example of this latter case is more fully developed in Section 4 (see tests 1 and 3A in Table 1). Consultation with statistical experts is advised, because such data analysis is often complex. For example, analyses dealing with spatial and temporal autocorrelation may be needed, multivariate analyses may be warranted, or Bayesian methodologies might be used to address evolving solutions.

3.3.3. Panel of experts

It is sometimes possible to test performance against an independent panel of experts (O'Keefe et al., 1987). The panel is considered to provide optimal (or nearly so) recommendations, predictions, or diagnoses against which the outputs from the decision support system are statistically compared.

Table 1
Interpretation of MVPTMP analyses from 4 of 34 experimental runs of the decision support system for trumpeter swan management

Test	MVPTMP (p-value)	Interpretation from rejecting the null hypothesis
1	0.0001	Output from base queuing model similar to observed numbers
3A	0.0001	Output using all expert systems (3) and activating all (7) refuge agents similar to observed numbers
6A	–	Output using three expert systems identical to that with only the flyway expert system
7A	–	Output using alternate breeding threshold of 0.4 identical to that using the standard, 0.6

Null hypotheses were developed a priori (Sojda, 2002). No p-value is reported when output between the two groups was identical.

This is a relatively common technique in the field of artificial intelligence and recent examples include multiagent web mining (Chau et al., 2003) and graphical user interface development (Virvou and Kabassi, 2004). Two concerns must be addressed, however. First, the panel of experts needed for such an evaluation must not be connected to system development. To do so would be so confounding that no reasonable experimental design would be feasible. Second, one of the basic tenets of using decision support systems for complex issues is that such questions can be beyond the capability of single persons to conceptualize and solve (Boland et al., 1992; Brehmer, 1991).

3.3.4. Sensitivity analysis

Sensitivity analysis is often more important in model validation than in decision support system validation. This stems from the typical decision support system being highly complex, and it being difficult to isolate individual inputs, or small enough groups of inputs, to perform sensitivity analysis. Plus, some sort of gold standard or data set is still needed with which to work (see test 7A in Table 1). However, many decision support systems have underlying models integrated within their completed structure, and sensitivity analysis on the individual models can be quite useful. Whether a particular model should be sensitive or insensitive to its inputs depends on the purpose of the model. However, if it is totally insensitive, it is difficult to understand what contribution the model might be providing to the larger system, unless one is modelling a component responsible for ecological buffering. Rios-Insua et al. (2000) provide a good example of varying input values and weights and their effect on ranking of strategies for restoring radionuclide contaminated aquatic ecosystems. Their system actually provided this type of sensitivity analysis to their end users so that they could validate potential strategies in terms of the uncertainty of final recommendations. Scheller and Mladenoff (2004) varied six input parameters, such as tree mortality, to determine the effect of a model component on an output, woody debris. They then compared this result with other published estimates. They chose to use sensitivity analysis on a component ecological process because they felt that their system did not provide output similar to

traditional landscape models, i.e., an equivalent gold standard did not exist for their entire system.

3.3.5. Component testing

Sometimes it is not possible to validate a complete system, but one can test individual components. It is not uncommon, for example, to have multiple expert systems embedded in one decision support system. When one validates each component separately, however, the interactions of the components and evolutionary behavior of the full system are not known. When testing of components is the only option, it is important to acknowledge this shortcoming. Often, when separate components of a system are validated, it can be argued that this is a form of system verification, as described by Rusu (2003) (see test 6A in Table 1).

4. An example: decision support system for trumpeter swan management

4.1. Background

A multiagent system of interacting intelligent agents (Weiss, 1999) was developed as a queuing system (Dshalalow, 1995; Hillier and Lieberman, 1995) to provide decision support to waterfowl managers, allowing them to simulate the effect of management actions on swan distributions (Sojda, 2002). DECAF software [Distributed Environment Centered Agent Framework] (Graham and Decker, 2000; Graham, 2001) was used to construct the agents, allow for their interaction, and to handle user I/O. As a multiagent system, the agents wait and listen for changes in certain parameters and databases, and they may request additional information to update their beliefs (Rao and Georgeff, 1991, 1995). The system functions as a deterministic queuing model to simulate the distribution of swans geographically and temporally. The actual simulation is handled by one particular agent which has the distribution simulator code embedded within itself. The queuing system utilizes output from expert systems related to ecological aspects of the flyway management of migratory birds, especially trumpeter swans and manipulation of their habitat. These expert systems were developed separately but used as part of the overall decision support system (Sojda and Howe, 1999). They can be considered inputs to the overall multiagent system. Through the use of a configuration file, the decision support system is directed to use or ignore various components and parameters. Thus, the system can run scenarios as experiments, and this is discussed further in Section 4.4.

This decision support system was evaluated at three levels: (1) verification of individual components, as well as the overall system, (2) soft validation (i.e., individual user anecdotes) of the expert systems, and (3) validation of the whole system. It was decided not to evaluate the system against a team with expertise in flyway management of swans, primarily because it was not feasible to assemble such a panel that was independent of the people used in knowledge engineering. This was true for two related reasons. First, the total number of workers in the domain is small. Second, the cadre of such workers is closely interrelated institutionally and academically.

4.2. Verification of components and the completed systems

A key part of designing the individual expert systems was developing flowcharts of the ecological logic and using them to consult with experts for changes and refinement. Similarly, the ''planeditor'' facility in the multiagent software, DECAF, allowed me to develop graphical representations of the logic underlying each agent and consult with specialists in multiagent system design. When running the multiagent system, DECAF provided information about how each agent was functioning and about failed communications among agents. This type of scrutiny addresses the need to debug logical errors as part of verification as described for expert systems by O'Keefe et al. (1987) and for models by Mihram (1972) and Rykiel (1996).

Utilities within the expert system development shell were used for verification of logical consistency of each expert system, including a static check for problems such as incomplete rules and trees, rules that logically cannot fire, and input that never results in calls to any rules. For example, an error would be detected if more than one rule tried to set a value for a single-valued variable, or if the consequent portion of a production rule was inadvertently not provided. The utilities also dynamically checked the system with stochastic runs, and the final expert systems were each checked for internal logical consistency using 500,000 such simulated runs with no problems detected. Each expert system had large numbers of rules, 157 in the breeding habitat system, 165 in the flyway management system, and 1882 in the wetland management system. By testing them with large amounts of random inputs, the software establishes (although does absolutely guarantee) that internal, logical consistency exists.

4.3. Soft validation of the expert system components

Demonstrations of each expert system were made to waterfowl managers, biologists, and researchers. This involved meetings and telephone consultations where individuals ran actual scenarios (usually via the World Wide Web) and provided comments in response to my specific requests. In addition, the expert systems were available to anonymous users in standalone fashion on the web, both in prototype and final versions. Both types of such validation targeted the underlying ontologies, knowledge, and problem solving logic, but were not empirical. I did not tally web usage, but did occasionally note web server log files. From this, I qualitatively estimate that usage was in the magnitude of hundreds of individuals during the testing phase, mostly by users unknown to me. I did respond whenever people provided comments via email or telephone, but only several did. In all cases of soft evaluation, iterative expert system development was done, and potential end users' requirements and suggestions about the underlying ecological principles or management actions were accommodated. I discontinued further soft validation

once experts collaborating in the knowledge engineering phase, and once key personnel representing the seven refuges in the queuing system, provided no further suggestions for changes. This was an arbitrary end point.

4.4. Validation using an historic data set

4.4.1. Conceptual framework and data source for swan numbers

Based on queuing theory (Dshalalow, 1995; Hillier and Lieberman, 1995), the decision support system begins by using an observed number of swans at each of 27 geographic areas for the breeding season of one year, and then simulates the number at each of those areas for the four subsequent seasons, concluding with a simulated number for the breeding season of the subsequent year. The system simulates breeding swan numbers in one year increments. It was a comparison of the simulated number for the subsequent year versus the observed number for that same year that was the basis of my empirical testing. Such an approach follows that recommended by Rykiel (1996) for ecological modelling; although, he does not discuss decision support systems. An observed number of swans was available only for the breeding season, and not the other seasons, so analysis was limited to data for that season. Comparisons of simulated and observed data could be made for 13 years, 1988–2000. Observed numbers were those collected by the member agencies of the Pacific Flyway Council and reported by the United States Fish and Wildlife Service on an annual basis (e.g., Reed, 2000).

It was the existence of such a long term, quality data set that allowed me to do the empirical evaluation that I did. Uncovering such data sets, although key, is often most difficult. In much of the Northern Hemisphere, long term survey and banding (ringing) records of migratory birds, especially waterfowl, may be one good source of such data.

4.4.2. Data analysis

My decision support system provides output as a 27 column by 13 row matrix of numbers of swans. Columns represent each geographic area; these areas are the servers from the perspective of the queuing system terminology. Rows represent individual years. This matrix was compared statistically to a corresponding matrix of observed numbers of swans compiled from the Pacific Flyway Council's surveys.

Although all 27 areas were always used in the queuing system, swans had never been observed in seven areas during the breeding season and those areas were excluded from statistical analysis. In all such cases, the system did not simulate swans in those areas. By excluding these areas, I ensured that the consistent simulation of no swans where none were expected did not artificially inflate the evaluated accuracy and precision of the system.

Thirty-four black-box experiments were conducted to empirically validate the decision support system's ability to predict swan distributions in the flyway (Sojda, 2002). A configuration file is used automatically by the various agents to set system parameters regarding how the queuing system, itself, will be run. For example, the configuration, as built by the user, sets values that the system uses to determine whether breeding habitat is adequate for a particular refuge (server) to accept swans. Also, through the configuration file, one can choose to either ignore or use the information generated by the individual expert systems for all active refuge agents. Therefore, the multiagent system can be used in an experimental fashion testing various ecological conditions for migrating swans.

My first experiment was to test the predicted numbers of swans for 20 areas from the base system against observed numbers for a series of 13 years. The base system was the full multiagent-based queuing system run with a configuration that did not allow the expert systems to affect the use of servers by swans, essentially ignoring the three expert systems. The second experiment compared the predicted numbers of swans from running the system with its default configuration against actual observed numbers. The default configuration allowed all three expert systems to affect the availability of seven refuges to simulated migrating swans. Other experiments that I ran included varying the number of refuges in the system, varying the number of expert systems in use, and choosing alternate ways to assign the number of swans in the starting queue. In my statistical analyses, two subsequent years are treated as a pair, and output from each of the servers treated as a response. The first of the pair is simulated data; the second is either observed data or simulated data from a run of the system with a different configuration. The results from four of my experiments are provided in Table 1. For example, based on test 3A, I concluded that the complete decision support system simulated distributions of swans over time (13 years) and space that were similar to the numbers actually observed for the same 20 areas. The full complement of experiments can be found in Sojda (2002).

Multivariate matched-pairs permutation test (MVPTMP) statistical procedures (Mielke and Berry, 2001) were used for the statistical analyses. These are nonparametric methods based on Euclidean distance functions, sampled permutations, and moment approximation approaches. In the analyses, a small p-value is evidence of similarity of distributions of swans over both space and time between the two groups of data forming a pair. Statistical analyses such as provided by MVPTMP provide a mathematical representation of a multivariate comparison, but it is difficult to graphically depict comparisons of such spatial data over time because of the inherent multidimensional structure. As a simplistic graphical alternative, and communication tool, the departure of the experimental data from the observed data can be plotted as has been done for numbers of swans in Fig. 1. Additional such visualizations can be found in Sojda (2002).

5. Discussion and conclusions

The ecological domain of migratory waterfowl in which I have worked seems to be a particularly productive, yet untapped, one in which to encourage decision support system development. First, the needs are great. Relating population phenomena to habitat change is an area of deep interest to

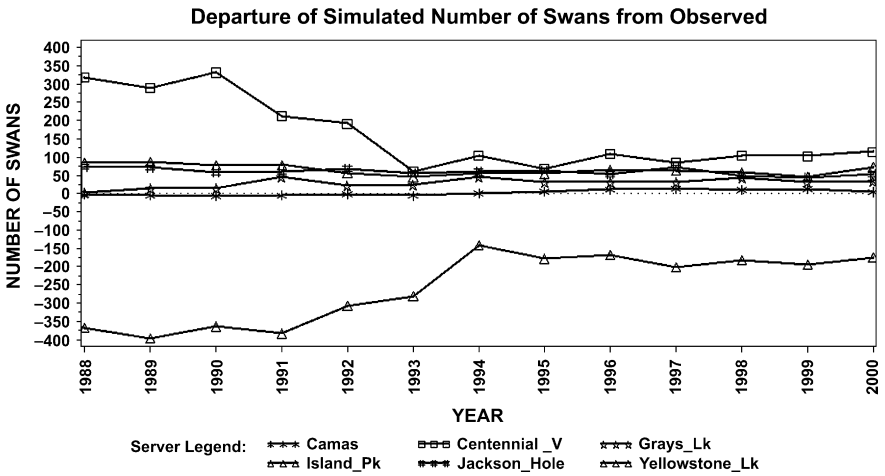

Fig. 1. An example of visualizing the departure of the simulated number from a model to the observed number is shown, in this case for trumpeter swans. Depictions close to zero would indicate strong similarity between predicted and observed numbers. Here, a server refers to the queuing system component that represents geographic areas important to trumpeter swans (Sojda, 2002).

wildlife managers, especially where decision support can provide a test bed for simulating habitat and population management activities. Mathevet et al. (2003) provide one example for ducks in the Camargue (France) using agent-based simulations. Second, many species are not within desired population levels, either being too numerous or too few. Third, because waterfowl, particularly hunted species, have been of human and government agency interest for so long, data sets are available for use in empirical evaluation. These include survey of actual numbers over time and space as well as banding (ringing) studies. As with many ecological questions, however, the theory of how to relate cause–effect relationships across spatial and temporal scales is yet to be fully developed. And, the visualization of multivariate response models is complex. Finally, missing data are always problematic. Nonetheless, development of such environmental decision support systems should be pursued, and their empirical testing encouraged.

Validation is the process of determining whether the stated purpose of the system was achieved. The purpose of my decision support system was to allow swan managers the ability to evaluate different management actions and the effect on swan distribution. I conclude that a multiagent system was an effective way to do this by simulating movement of waterfowl in a flyway and by incorporating expert systems related to management actions. The suite of management actions developed in this system were limited to breeding habitat assessment, water level management in wetlands, and implementing principles of flyway management at the local level. However, these are key waterfowl management issues. Empirical validation of the multiagent system demonstrates the effectiveness of my approach to integrating the management of seven refuges and simulating their effects on a total of 20 geographic areas over time. I have no empirical evidence to suggest whether this system could be applied to other areas, although my judgment tells me the underlying processes related to trumpeter swan ecology and management would make such application possible. To broaden it to other species would require development of different knowledge bases.

Because models are abstractions of reality, it is inherent that they will have shortcomings from not being able to accurately represent all knowledge, logical relationships, and probabilistic intricacies. This does not lessen their value, but makes empirical evaluation essential. Overall, the evidence was strong that the base system (in the decision support system for trumpeter swan management) mimicked the observed pattern of swan distributions over time, as does the system run with the default configuration. Almost all experimental runs of the decision support system showed the same pattern. From the evaluation in its entirety (i.e., verification and validation), it seems reasonable to conclude that correct underlying causal relationships are represented in the individual expert systems, in the queuing model, and in their integration within the multiagent framework. Verification was completed on both components and the full system. Soft validation was completed on components. Validation with an historic data set on the full system was also accomplished.

It seems irresponsible to deliver a decision support system that has not been adequately evaluated, including both verification and validation. Empirical evaluation in some form is critical, and can range from experiments run against a preselected gold standard to more simple testing of system components. It is imperative to understand, from an experimental and logical perspective, to what extent inferences can be made as a result of the validation. In the end, the question to answer is: Was the system successful at addressing its intended purpose? Often, searching for the right database for empirical evaluation can be as important as adequate decision support system development, itself. Otherwise, one has no scientific reference by which to judge the adequacy, performance, and fundamental credibility of the system.

Acknowledgements

I recognize A. Howe, D. Dean, P. Mielke, S. Stafford, L Fredrickson, and J. Cornely for their encouragement and for introducing many of the key concepts found in this paper.

R. Jachowski stimulated thought about objectivity and practical application of model evaluation. The programming abilities of D. Zarzhitzky were invaluable in implementing the multiagent system. C. Wright is acknowledged for assistance in the development and coding of the distribution simulator component. Funding was provided by the U.S. Department of Interior: the Geological Survey-Biological Resources Division and the Fish and Wildlife Service. This research was part of Geological Survey, Biological Resources Division Project Number 915.

References

Adelman, L., 1991. Experiments, quasi-experiments, and case studies: a review of empirical methods for evaluating decision support systems. IEEE Transactions on Systems, Man, and Cybernetics 21 (2), 293–301.

Adelman, L., 1992. Evaluating Decision Support and Expert Systems. John Wiley and Sons, New York, NY.

Andriole, S.J., 1989. Handbook of Decision Support Systems. TAB Professional and Reference Books, Blue Ridge Summit, Pennsylvania.

Adrion, W.R., Branstad, M.A., Cherniavsky, J.C., 1982. Validation, verification, and testing of computer software. ACM Computing Surveys 14 (2), 159–192.

Bahill, T.A., 1991. Verifying and Validating Personal Computer-based Expert Systems. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Boehm, B.W., 1981. Software Engineering Economics. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Boland, R.J., Mahewshwari, A.K., Te'eni, D., Schwartz, D.G., Tenkasi, R.V., 1992. Sharing perspectives in distributed decision making. In: Proceedings of the Conference on Computer-supported Cooperative Work. Association for Computing Machinery, New York, NY.

Brehmer, B., 1991. Distributed decision making: some notes on the literature. In: Rasmussen, J., Brehmer, B., Leplat, J. (Eds.), Distributed Decision Making: Cognitive Models for Cooperative Work. John Wiley and Sons, Chichester, England.

Carter, G.M., Murray, M.P., Walker, R.G., Walker, W.E., 1992. Building Organizational Decision Support Systems. Economic Press Inc., San Diego, California.

Chau, M., Zeng, D., Chen, H., Huang, M., Hendriawan, D., 2003. Design and evaluation of a multiagent collaborative Web mining system. Decision Support Systems 35, 167–183.

Cohen, P.R., Howe, A.E., 1989. Toward AI research methodology: three case studies in evaluation. IEEE Transactions on Systems, Man, and Cybernetics 19 (3), 634–646.

Dshalalow, J.H., 1995. An anthology of classical queuing models. In: Dshalalow, J.H. (Ed.), Advances in Queuing: Theory, Methods, and Open Problems. CRC Press, Boca Raton, Florida.

D'Erchia, F., Korschgen, C., Nyquist, M., Root, R., Sojda, R., Stine, P., 2001. A framework for ecological decision support systems: building the right systems and building the systems right. Information and Technology Report USGS/BRD/ITR-2001-0002. U.S. Geological Survey, Biological Resources Division, Washington, DC.

Eason, K., 1988. Information Technology and Organizational Change. Taylor and Francis Publishing, London, United Kingdom.

Fishmann, G.S., Kiviat, P.J., 1968. The statistics of discrete-event simulation. Simulation 10, 185–195.

Graham, I., Jones, P.L., 1988. Expert Systems: Knowledge, Uncertainty, and Decision. Chapman and Hall, New York, NY.

Graham, J.R., 2001. Real-time scheduling in distributed multi agent systems. PhD dissertation, University of Delaware, Newark, Delaware.

Graham, J.R., Decker, K.S., 2000. Towards a distributed, environment-centered agent framework. In: Jennings Nicholas, R., Lesperance, Y. (Eds.), Proceedings of the Sixth International Workshop on Agent, Theories, Architectures, and Languages (ATAL-99). Springer-Verlag, Berlin, Germany.

Gupta, U., 1991. Validating and Verifying Knowledge-based Systems. IEEE Computer Society Press, Washington, DC.

Haberlandt, U., Krysanova, V., Bardossy, A., 2002. Assessment of nitrogen leaching from arable land in large river basins – Part II: regionalisation using fuzzy rule based modelling. Ecological Modelling 150, 277–294.

Hillier, F.S., Lieberman, G.J., 1995. Introduction to Operations Research. McGraw-Hill, Inc., New York, NY.

Johnson, D.H., 2001. Validating and evaluating models. In: Shenk, T.M., Franklin, A.B. (Eds.), Modelling in Natural Resource Management: Development, Interpretation, and Application. Island Press, Washington, DC.

Kanungo, S., Sharma, S., Jain, P.K., 2001. Evaluation of a decision support system for credit management decisions. Decision Support Systems 30, 419–436.

Mathevet, R., Bousquet, F., Le Page, C., Antona, M., 2003. Agent-based simulations of interactions between duck population, farming decisions and leasing of hunting rights in the Camargue (Southern France). Ecological Modelling 165, 107–126.

Mielke, P.W., Berry, K.J., 2001. Permutation Methods: A Distance Function Approach. Springer-Verlag, New York, NY.

Mihram, G.A., 1972. Some practical aspects of the verification and validation of simulation models. Operational Research Quarterly 23 (1), 17–29.

Mosqueira-Rey, E., Moret-Bonillo, V., 2000. Validation of intelligent systems: a critical study and a tool. Expert Systems with Applications 18, 1–16.

Murrell, S., Plant, R.T., 1997. A survey of tools for the validation and verification of knowledge-based systems: 1985–1995. Decision Support Systems 21, 307–323.

O'Keefe, R.M., Balci, O., Smith, E.P., 1987. Validating expert system performance. IEEE Expert 2 (4), 81–90.

O'Leary, D.O., 2001. Verification of multiple agent knowledge-based systems. International Journal of Intelligent Systems 16, 361–376.

Plant, R., Gamble, R., 2003. Methodologies for the development of knowledge-based systems, 1982–2002. The Knowledge Engineering Review 81 (1), 47–81.

Pretzsch, H., Biber, P., Dursky, J., 2002. The single tree-based stand simulator SILVA: construction, application and evaluation. Forest Ecology and Management 162, 3–21.

Priya, S., Shibasaki, R., 2001. National spatial crop yield simulation using GIS-based crop production model. Ecological Modelling 135, 113–129.

Rao, A.S., Georgeff, M.P., 1991. Modeling rational agents within a BDI-architecture. In: Allen, J., Fikes, R., Sandewall, E. (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference. Morgan Kaufmann Publishers, San Mateo, California, pp. 1–18.

Rao, A.S., Georgeff, M.P., 1995. BDI agents: from theory to practice. In: Proceedings of the First International Conference on Multiagent Systems. AAAI Press, Menlo Park, California, pp. 312–319.

Reed, T., 2000. 2000 Fall trumpeter swan survey. Unpublished Report, U.S. Fish and Wildlife Service, Lakeview, Montana.

Rice, J.A., Cochran, P.A., 1984. Independent evaluation of a bioenergetics model for largemouth bass. Ecology 65 (3), 732–739.

Rios-Insua, D., Gallego, E., Mateos, A., Rios-Insua, S., 2000. MOIRA: a decision support system for decision making on aquatic ecosystems contaminated by radioactive fallout. Annals of Operations Research 95, 341–364.

Rushby, J., 1988. Validation and testing of knowledge-based systems: how bad can it get? In: Gupta, U. (Ed.), Validating and Verifying Knowledge-based Systems IEEE Computer Society Press, Los Alamitos, California.

Rusu, V., 2003. Combining formal verification and conformance testing for validating reactive systems. Software Testing, Verification and Reliability 13, 157–180.

Rykiel Jr., E.J., 1996. Testing ecological models: the meaning of validation. Ecological Modelling 90, 229–244.

Santos Jr., E., 2001. Verification and validation of Bayesian knowledge-bases. Data and Knowledge Engineering 37, 307–329.

Scheller, R.M., Mladenoff, D.J., 2004. A forest growth and biomass module for a landscape simulation model, LANDIS: design validation, and application. Ecological Modelling 180, 211–229.

Sojda, R.S., 2002. Artificial intelligence based decision support for trumpeter swan management. PhD dissertation, Colorado State University, Fort Collins, Colorado.

Sojda, R.S., Howe, A.E., 1999. Applying cooperative distributed problem solving methods to trumpeter swan management. In: Cortes, U., Sanchez-Marre, M. (Eds.), Environmental Decision Support Systems and Artificial Intelligence. American Association for Artificial Intelligence Technical Report WS-99-07. AAAI Press, Menlo Park, California.

Sprague Jr., R.H., Carlson, E.D., 1982. Building Effective Decision Support Systems. Prentice-Hall, Englewood Cliffs New Jersey.

Stuth, J.W., Smith, M.S., 1993. Decision support for grazing lands: an overview. In: Stuth, J.W., Lyons, B.G. (Eds.), Decision Support Systems for the Management of Grazing Lands. Man and the Biosphere Series Volume 11: Papers From the International Conference on Decision Support Systems for Resource Management. The Parthenon Publishing Group, Pearl River, New York.

Virvou, M., Kabassi, K., 2004. Evaluating an intelligent graphical user interface by comparison with human experts. Knowledge-based Systems 17, 31–37.

Wallace, D.R., Fujii, R.U., 1989. Software verification and validation: an overview. IEEE Software 6 (3), 10–17.

Wang, J., LeDoux, C.B., 2003. Estimating and validating ground-based timber harvesting production through computer simulation. Forest Science 49 (1), 64–76.

Weiss, G., 1999. Prologue: multiagent systems and distributed artificial intelligence. In: Weiss, G. (Ed.), Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press, Cambridge, Massachusetts.