Ecological Informatics 34 (2016) 92–101

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

A platform-independent fuzzy logic modeling framework for environmental decision support

T. Sheehan *, M. Gough

Conservation Biology Institute, Corvallis, OR, USA

ARTICLE INFO

Article history:
Received 17 November 2015
Received in revised form 28 April 2016
Accepted 1 May 2016
Available online 03 May 2016

Keywords:
Fuzzy logic
Modeling
Evaluation
Framework
Landscape evaluation
Software
Site sensitivity

ABSTRACT

Fuzzy logic modeling is a useful method for evaluating landscapes for conservation and resource planning and has been successfully used in different types of ecological and environmental studies. A variety of software packages have been produced to facilitate fuzzy logic modeling, but each is either associated with a specific computer program or does not comprise a complete modeling system. The Environmental Evaluation Modeling System (EEMS) is a platform-independent fuzzy logic modeling framework for environmental decision support. EEMS has been designed so that it can easily be adapted to work with different file types and interface with other software systems. It has been implemented to work with NetCDF and CSV file formats as a command line application, in the ArcGIS ModelBuilder environment, and as part of a web-based data exploration tool. In a performance test, EEMS was run using a dataset with four million reporting units per map layer and yielded execution times of less than 30 s. Results from an EEMS model for Utah and the Colorado Plateau show a complex pattern of site sensitivity.

1. Introduction

There are many challenges to producing quantifiable metrics for vulnerability-related concepts. The definitions of *vulnerability* and related concepts such as *risk*, *resilience*, and *adaptive capacity* vary among scientific disciplines (Füssel, 2007; McFadden, 2007; Dawson et al., 2010; Wamsley et al., 2015). Such terms have the potential to cause confusion in the field of ecosystem management (Mumby et al., 2014). Even with definitions and frameworks for these concepts (e.g. Mumby et al., 2014; Wamsley et al., 2015) each vulnerability assessment is unique (Wamsley et al., 2015) and requires appropriate thresholds and scales (Mumby et al., 2014).

Fuzzy logic (Zadeh, 1965) is one method for meeting these challenges. It is based on the concept that set membership can be partial rather than binary (inclusion or exclusion) and is well suited for fields of study in which variables are *linguistic* (Zadeh, 1973), that is to say quantified verbally and imprecisely. Rules used in fuzzy logic modeling provide a straightforward way to represent subjective or vague knowledge (Kasabov, 1996). Fuzzy logic has been successfully applied in a variety of environmental contexts including risk assessments for infrastructure placement (e. g. Bojorquez-Tapia et al., 2002; Boclin and de Mello, 2006; Aydi et al., 2013), environmental impacts for aquatic ecosystems (e. g. Cheung et al., 2005; Kaplan et al., 2014; Segui et al., 2013), and soils (McBratney and Odeh, 1997, e. g. Mays et al., 1997).

A number of software packages and frameworks implement fuzzy logic. The 'sets' package (Heaton, 2014; Meyer et al., 2015) allows users to write fuzzy logic models in R (R Foundation, 2016), a free software environment available for Windows, Macintosh (OS X), and Linux computers. Jasiewicz (2011) system provides fuzzy logic modeling for the open source GRASS GIS system (Neteler and Mitasova, 2008) available for Windows, Macintosh (OS X), and Linux. F-IND (Marchini et al., 2009) is an application for the design of fuzzy indices. For portability among operating systems, it is written in the Java programming language. ArcGIS' (ESRI, 1999-2016) raster-based fuzzy logic tools (ESRI, 1995-2012) provide fuzzy logic functionality through a graphical interface in the Windows-based Arc environment. Ecosystem Management Decision Support System (EMDS) (Reynolds et al., 2015) utilizes the NetWeaver Developer system (Miller and Saunders, 2002) and is integrated with ArcGIS. EMDS provides a rich graphical modeling environment and works with vector-based spatial data. Each of these systems is limited to running within a specific software environment (the 'sets' package, 'Jasiewicz' system, ArcGIS fuzzy logic tools, and EMDS) or does not comprise a complete fuzzy logic modeling system (F-IND).

We have developed the Environmental Evaluation Modeling System (EEMS), a fuzzy logic modeling framework designed to be easily adapted for different file types, software programs, and operating systems. It is the basis for full modeling packages for CSV and NetCDF format files on the Apple Macintosh running the OS X operating system and on personal computers running the Windows operating system. Another implementation works with feature classes in the ArcGIS ModelBuilder environment, which runs under Windows. EEMS also serves as the engine for a web-based data exploration tool on the Data

* Corresponding author at: Conservation Biology Institute, 136 SW Washington Ave, Ste 202, Corvallis, OR 97333, USA.
E-mail address: tim@consbio.org (T. Sheehan).

http://dx.doi.org/10.1016/j.ecoinf.2016.05.001
1574-9541/

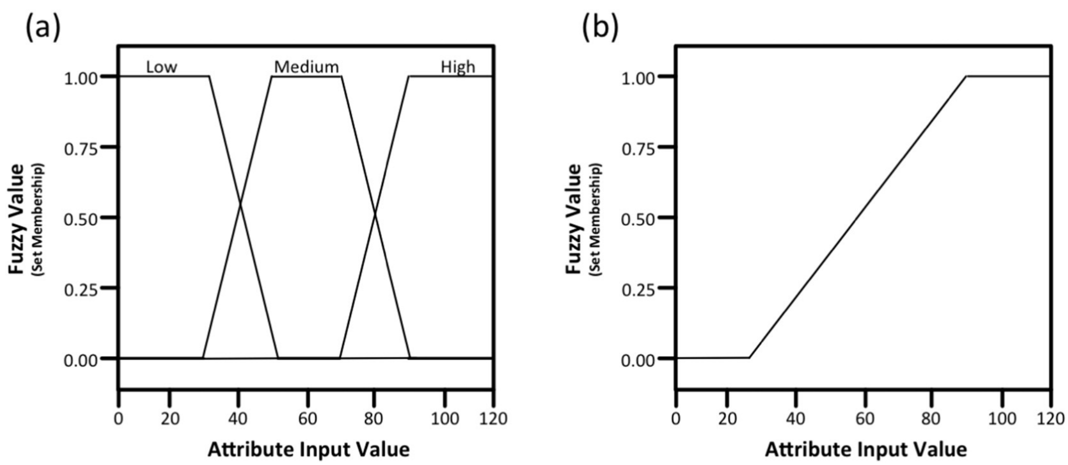

Fig. 1. Approaches for assigning fuzzy set membership from an input variable. (a) Using three conversion functions to produce overlapping fuzzy sets corresponding to membership in sets for low, medium, and high attribute input values, and (b) using a single continuous conversion function corresponding to membership in the set corresponding to high attribute input value.

Basin (Conservation Biology Institute, 2010-2016) web platform. We present a description of the EEMS fuzzy logic modeling framework, results from a performance evaluation of EEMS, and an EEMS site sensitivity model for Utah and the Colorado Plateau in the western United States.

2. EEMS fuzzy logic

Much of human reasoning is thought to be fuzzy in nature (Zadeh, 1973) and fuzzy logic has been expanded to model such reasoning using fuzzy (partial) truths, connectives (operators), and rules of inference (Zadeh, 1973; Giles, 1976). In fuzzy logic modeling (Zadeh, 1965) set membership can be partial, based on a membership function yielding a gradational continuum commonly represented by a number between 0 (indicating complete exclusion) and 1 (indicating complete inclusion) (e.g. Zadeh, 1973; Mamdani, 1977; Kasabov, 1996). A detailed description of fuzzy logic modeling methods is not necessary here since the topic is well covered in the existing literature both as a general topic (e. g. Mamdani, 1977; Kasabov, 1996) and with regard to environmental issues (e.g. Reynolds, 2001; Adriaenssens et al.,

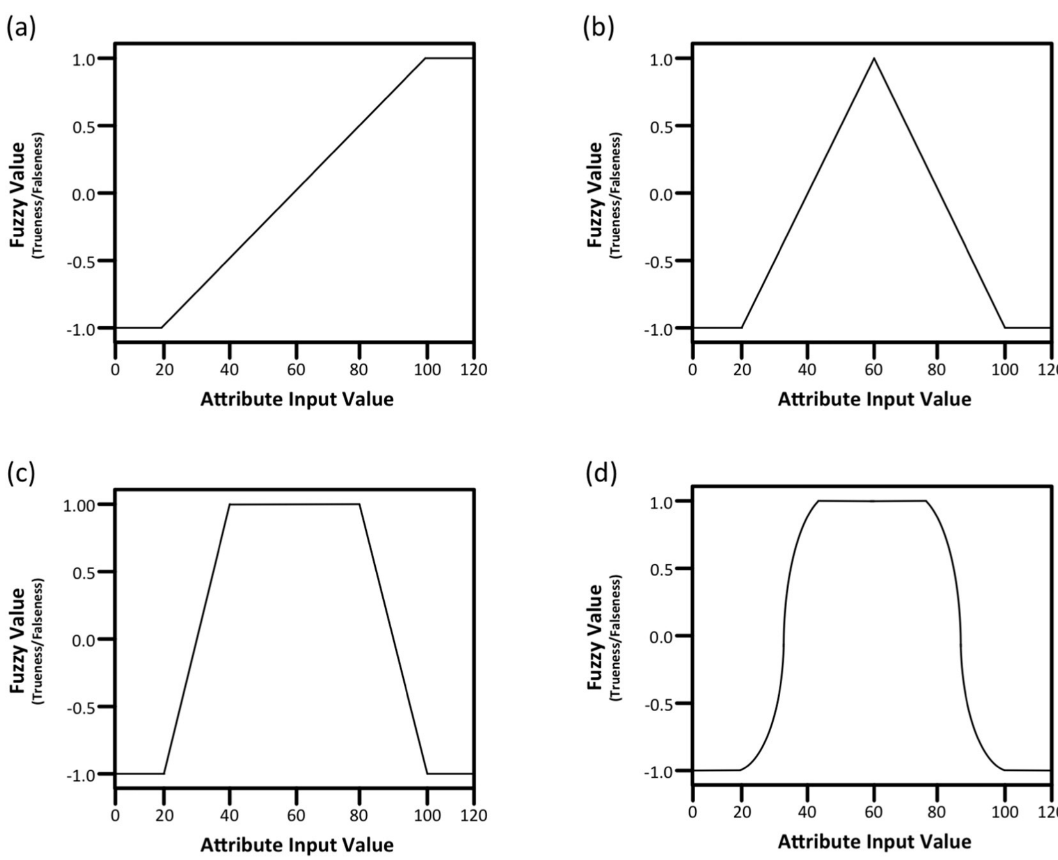

Fig. 2. Plots of conversion functions for translating attribute values into fuzzy values: (a) linear interpolation, (b) triangular, c) trapezoidal, and d) truncated curve.

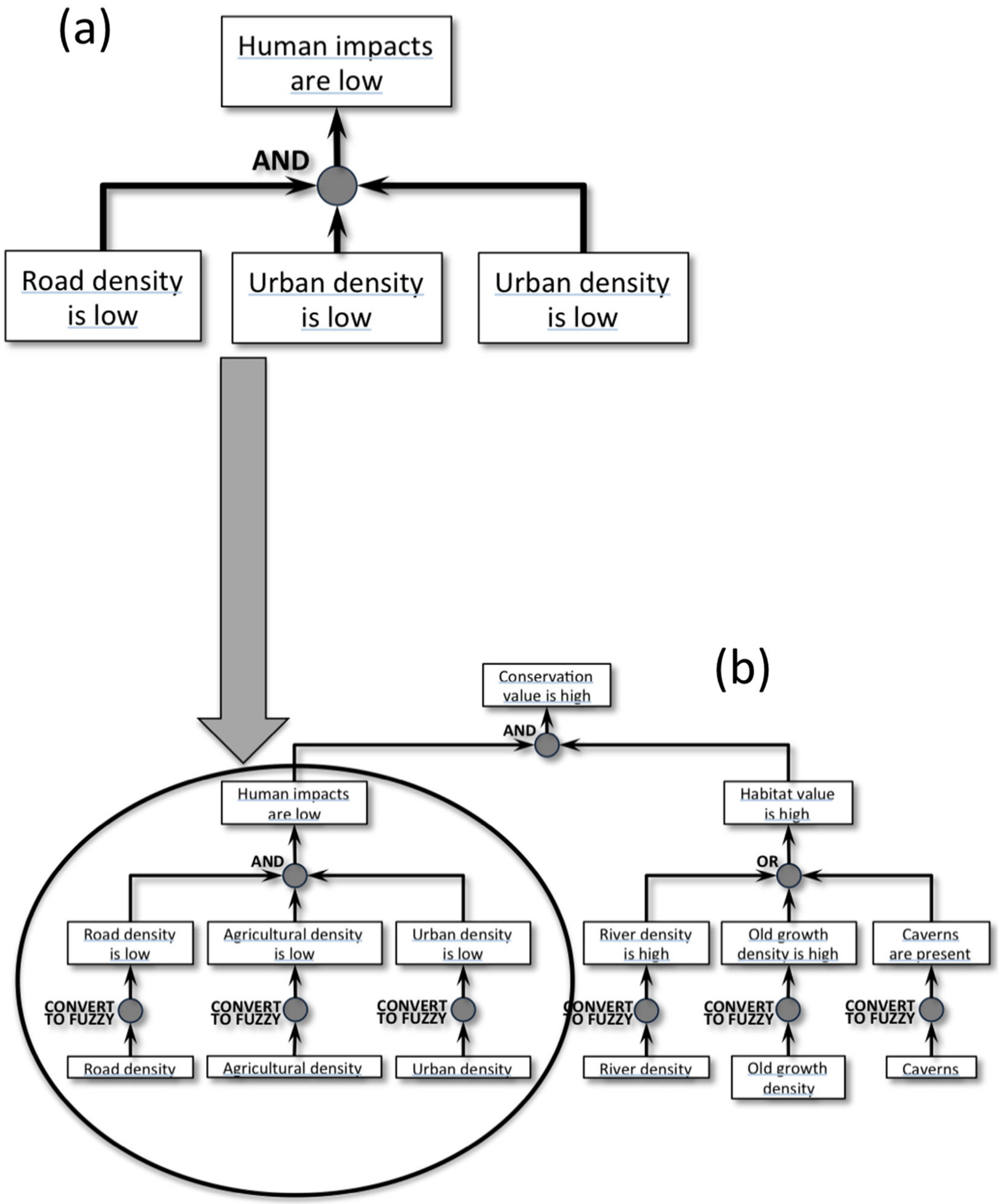

Fig. 3. Tree structured hypothetical EEMS logic model. (a) A branch with three nodes combined using the AND fuzzy logic operator; (b) full fuzzy logic tree with root node *Conservation value is high*.

2004). However the basic distinctions between two commonly used approaches bears discussion.

First, the Mamdani–Assilian method (Mamdani, 1977; Kasabov, 1996; Adriaenssens et al., 2004), multiple conversion functions are used to convert a single input variable (referred to as being crisp, as opposed to fuzzy) into multiple fuzzy variables with overlapping ranges (Fig. 1a). A logic expression is associated with every possible combination of fuzzy variables in the model,

$$\prod_{i=1}^{n} d_n \qquad (1)$$

resulting in logic expressions where n is the number of input variables and d is the number of divisions associated with input variable n. This method can also include a defuzzification (translating the fuzzy results into a crisp result) step to synthesize the results produced by the multiple logic expressions (Kasabov, 1996).

The second method (Reynolds, 2001) uses a single continuous function to convert each input variable into a single fuzzy variable (Fig. 1b). A model using this method consists of a single logic expression that can be represented by a tree with each node in the tree representing the results from a single fuzzy logic operation (e.g. Reynolds, 2001).

The Mamdani–Assilian method is commonly used when inputs are intuitively considered and grouped by degree. Examples include classifying age as *young*, *middle*, or *mature* (Kasabov, 1996) or a vulnerability index as *very low*, *low*, *high*, or *very high* (Bojorquez-Tapia et al., 2002). The method used by Reynolds (2001) is well suited for the fuzzy representation of gradual properties for example “the more a tomato is red, the more it is ripe” (Kasabov, 1996). This method has proven useful in social, political, and economic systems modeling (Kasabov, 1996) and

Table 1
EEMS scripting language commands.

Command	Use	Comments
AND	Takes the Falsest of input values	Useful when all inputs are necessary to drive the results (fuzzy operator)
COPYFIELD	Copies an existing field into a new field	
CVTTOFUZZY	Produces a fuzzy field by applying a linear interpolation function to a non-fuzzy field	Uses a single True and a single False threshold
CVTTOFUZZYCAT	Produces a fuzzy field by converting categorical integer	User defines a set of x–y pairs to specify the fuzzy value associated with input categories
CVTTOFUZZYCURVE	Produces a fuzzy field by applying linear interpolation using segments of a user-defined approximated curve	Uses a set of x–y pairs to define line segments used for linear interpolation
DIF	Take the difference of two fields	(Non fuzzy operator)
MAX	Takes the maximum of input values	(Non fuzzy operator)
MEAN	Takes the mean of input values	(Non fuzzy operator)
MIN	Takes the minimum of input values	(Non fuzzy operator)
NOT	Converts a fuzzy field into its True/False opposite	(Fuzzy operator)
OR	Take the Truest of the input fields	Useful when any of the inputs is sufficient to drive the result (fuzzy operator)
READ	Read a single field from an input file	
READMULTI	Read multiple fields from an input file	Associated method must be extended for an EEMS implementation
SELECTEDUNION	Takes the UNION (mean) of the n Truest or Falsest (user-specified) values.	Useful when n of m conditions are necessary and sufficient to drive the result (fuzzy operator)
SUM	Takes the sum of input values	(Non fuzzy operator)
UNION	Takes the UNION (mean) of fuzzy values	Useful when all inputs equally drive the result (fuzzy operator)
WTDMEAN	Takes a weighted mean of input values	(Non fuzzy operator)
WTDSUM	Takes a weighted sum of input values	Useful when two or more values contribute unequally to a result (e.g. dirt road density and paved road density) (non fuzzy operator)
WTDUNION	Takes a weighted UNION (mean) of fuzzy values	Useful when multiple inputs drive a result unequally (fuzzy operator)
XOR	Returns a value reflecting the exclusive True-ness of the Truest input	Useful when one and only one True value is required to drive the result (fuzzy operator)

landscape management decision support (e. g. Gärtner et al., 2008). The advantages of implementing a single logic expression for the Reynolds (2001) method versus the $\prod_{i=1}^{n} d_n$ logic expressions for the Mamdani–Assilian were a guiding influence for the development of EEMS.

3. EEMS fuzzy logic modeling

EEMS' method of fuzzy logic modeling is based on that used by EMDS, which uses the Reynolds (2001) method described above. Fuzzy truths and fuzzy operators are used to construct a model corresponding as a hierarchical logic tree. Fuzzy truths are represented over a continuum of −1.0 for fully false to +1.0 for fully true. 0.0 represents neither true nor false.

Input values are converted into fuzzy values via a function that maps the input value along a true/false gradient based on the input value's relationship to a logical proposition (Fig. 2). For example, we might consider the proposition *Road density is high* fully false for road density values of less than 0.25 km/km^2, fully true for values greater than 0.75 km/km^2, and falling between true and false corresponding to a linear interpolation for values between 0.25 and 0.75 km/km^2, resulting in the conversion function:

$$\begin{array}{ll} y = 1 & : x < 0.25 \\ y = 4x - 2 & : 0.25 <= x <= 0.75 \\ y = -1 & : x > 0.75 \end{array} \quad (2)$$

where x is road density and y is the fuzzy value along the true/false gradient for the proposition *Road density is high*.

Fuzzy reasoning applies one of the fuzzy logic operators (for example, AND) to fuzzy truth values, each of which corresponds to a single proposition. The result is a fuzzy value corresponding to a compound fuzzy logic proposition. As an example, road density, agricultural density, and urban density all contribute to the overall level of human impacts on a landscape. To evaluate the landscape based on the proposition *human impacts are low*, one would first evaluate each of the contributing propositions: *road density is low*, *agricultural density is low*, and *urban density is low*. This might be done using a different fuzzy conversion function for each of road density, agricultural density, and urban density. Fuzzy values for these three factors could then be combined using a fuzzy logic operator to yield a fuzzy result for the top level proposition *human impacts are low* (the AND operator in this example). The logic statement *If road density is low, and agricultural density is low, and urban density is low, then human impacts are low* can be expressed in terms of propositions and the fuzzy logic **AND** operator:

Human impacts are low = *Road density is low* **AND** *Agricultural density is low* **AND** *Urban density is low*.

A branched representation is another way to express this logic (Fig. 3a), with each node representing a proposition. A logic tree is constructed by recursively combining branches until they produce a single node, termed the root node, at the tree's apex (Fig. 3b).

EEMS was designed to work with spatial data layers whose divisions, or reporting units, are congruent. Each data layer corresponds to a node in the logic tree. Calculations are performed using corresponding reporting units between layers. Reporting unit values within layers are completely independent of one another.

4. EEMS scripting language and the EEMS parser

The EEMS scripting language is central to the EEMS framework. Either directly or through the use of a script-generating interface (as with the Arc ModelBuilder environment described below), users create a text file of commands corresponding to the nodes in an EEMS model. Commands are of the form:

$$\begin{aligned} &ResultFieldName = CommandName(\,ArgName_1 = \mathrm{ArgValue},\ ArgName_2 \\ &\quad = [ArgValue_1, ArgValue_2, \ldots]\ \ldots,\ ArgName_n = \mathrm{ArgValue}_n). \end{aligned}$$

All commands allow the optional argument **OutFileName**, which specifies that the result of the command will be written to the specified file.

In order to make scripting easier for the user, the EEMS parser checks the data type of every command argument and raises an error if an invalid value is used. When an argument is a literal string, for example with a file name, the argument value does not need to be quoted. The parser's only requirement is that each command starts on a new line, although common-sense formatting should be practiced. Additionally,

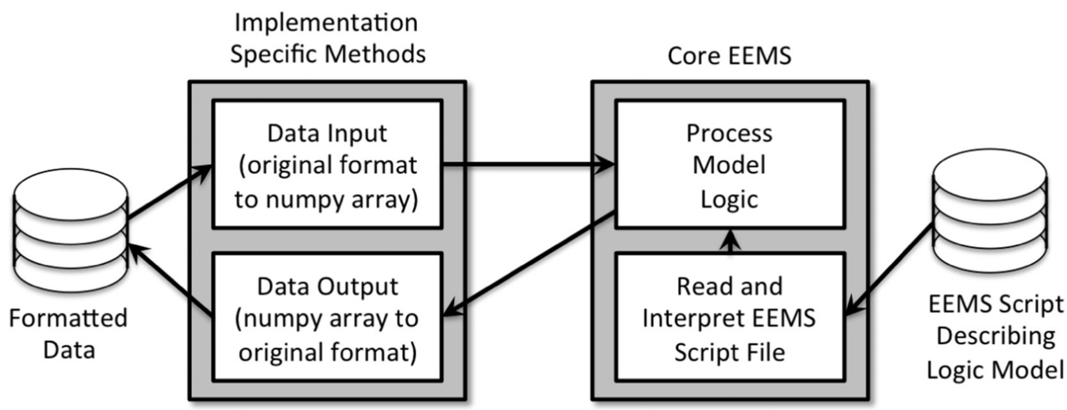

Fig 4. Schematic of four conceptual components comprising EEMS. An EEMS script is read and interpreted by one of the methods within core EEMS. Core EEMS methods are independent of data type and do not change from one implementation to another. Data are read and written by implementation specific methods targeted to a data file format such as CSV or NetCDF.

the parser insures that the script represents a valid hierarchical tree, raising an error if there is a missing dependency or a circular reference. A side effect of this checking is that EEMS creates a valid command execution order, thus the commands in an EEMS script file need not be in a particular order.

While EEMS was designed primarily as a fuzzy logic modeling framework, it includes commands to process non-fuzzy data. These commands can be used in data preparation steps or to perform other types of modeling. The EEMS commands for fuzzy data manipulation include the operators common to most formal fuzzy logic systems (OR, AND, NOT) as well as others either not included in many systems or used differently than in some other systems (e.g. UNION, SELECTEDUNION). Table 1 provides a synopsis of all EEMS commands, including descriptions of situations in which they are most useful.

5. Architecture and implementation

EEMS was written in the python programming language (Sanner, 1999). Four conceptual components make up any implementation built on top of the framework (Fig. 4): 1) script input and interpretation

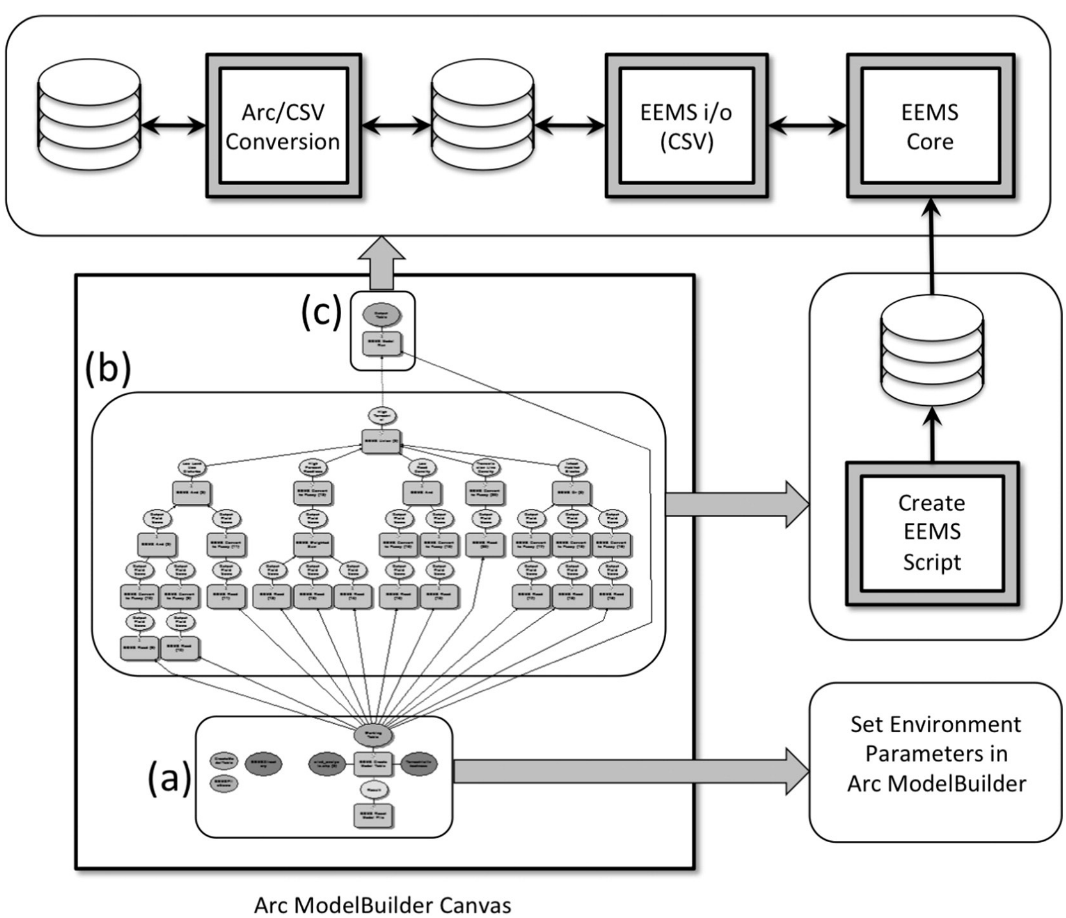

Fig. 5. Arc EEMS implementation. Using the EEMS toolbox for Arc, the user builds and executes a model on the Arc ModelBuilder canvas. (a) Environment parameters for the model and its execution, (b) the logic model that produces an EEMS script file, and (c) the model execution command.

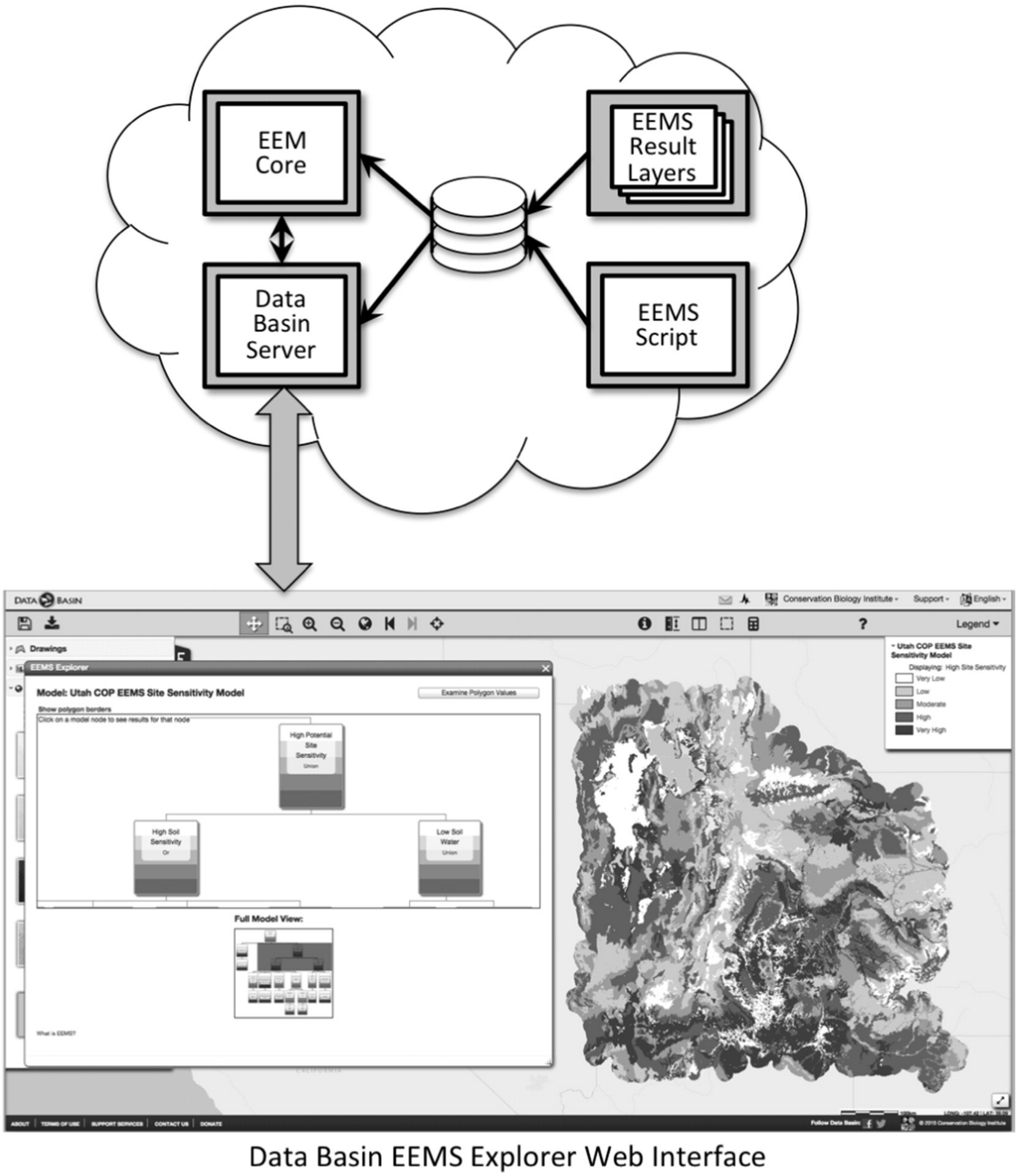

Fig. 6. Architecture of EEMS Model Explorer in Data Basin. A user accesses the cloud-based data and model through a web interface.

which reads, parses, and validates an EEMS script, then translates it into a series of executable commands; 2) logic model processing which executes model commands on the input data; 3) data input which reads data into the EEMS data structures; 4) data output which writes data from EEMS data structures. The first two of these comprise the EEMS core and do not change from one EEMS implementation to another. The second two, data input and data output, are specific to the EEMS implementation for a given data format (e.g. CSV, NetCDF). These are implemented by extending the stub methods for file reading and writing included in the EEMS base package.

EEMS has been implemented as a command line program for CSV and NetCDF file formats, as an ArcToolbox (a .zip file with a toolbox, manual, and example files is available for download at: http://consbio.webfactional.com/EEMS/EEMS2.0_ArcGIS.zip), and as an interactive data explorer in Data Basin (Conservation Biology Institute, 2010-2016, Supplemental materials 1). For the command line implementation, a user creates an EEMS language script using a text editor and then provides the script file name as an argument to the EEMS command. EEMS notifies the user of any errors, or when the run completes.

With the ArcToolbox implementation (Fig. 5), the user drags and drops tool objects onto the Arc ModelBuilder canvas, specifies options in dialog boxes, and draws connections between objects to represent data flow. When the ModelBuilder model is executed, it prepares input files, creates an EEMS script file from the ModelBuilder EEMS model representation, and applies the EEMS script to the data. For this environment, EEMS was initially implemented to read and write ESRI shapefiles and geodatabase feature classes, but processing is more efficient if ArcGIS data are translated to CSV format, processed, and translated back into the original format.

The Data Basin EEMS Explorer (Fig. 6) is built, in part, on the EEMS core. The EEMS Explorer allows a user to interactively explore the results of an EEMS model by clicking on a model node to display a

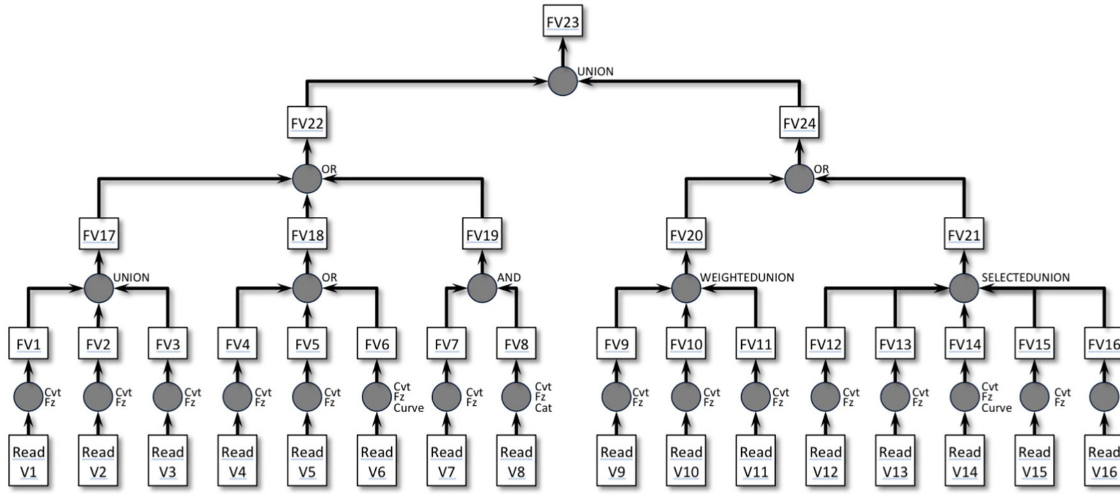

Fig. 7. EEMS logic model used for performance test. (V, input variable; FV, fuzzy variable; Cvt Fz, CONVERTTOFUZZY operator; Cvt Fz Curve, CONVERTTOFUZZYCURVE operator; Cvt Fz Cat, CONVERTTOFUZZYCAT operator.

map image of the associated layer and also by clicking on a reporting unit to display each node's value for that unit.

6. Examples

6.1. EEMS performance evaluation

An EEMS test model (Fig. 7) was prepared for performance evaluation. Each variable consisted of one data layer of four million reporting units in NetCDF format. Inputs consisted of 16 variables with one variable per file. The model computed 23 fuzzy variables for a total of 92 million fuzzy logic computations. Thirty-nine output variables (16 input variables and 23 computed variables) were written to a single NetCDF file. Model runs were executed on a Late 2011 Apple MacBook Pro with a 2.4 GHz Intel Core i7 processor, 16 GB 1333 MHz memory, 750 GB SATA hard drive, and OS X 10.7. The mean run time from five runs of the script was 29.4 s, of which 15.6 s were attributable to output.

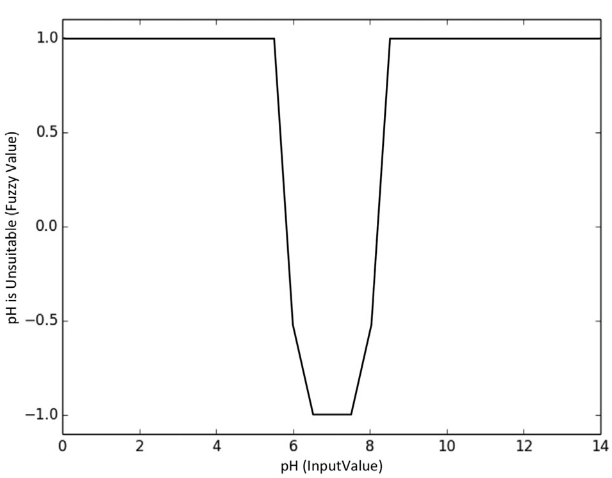

Fig. 8. Fuzzy conversion curve for *pH is unsuitable* in EEMS Utah and Colorado Plateau site sensitivity model.

6.2. Site sensitivity in Utah and the Colorado plateau

We implemented an EEMS site sensitivity model (Fig. 9) for Utah and the Colorado Plateau (USA). Model inputs were potential evapotranspiration and soil characteristics (Table 2). Resolution was 30 as. This model was designed to evaluate the study area for factors we assume could make the landscape sensitive to climate change. The two main categories of factors were general soil sensitivity to climate conditions (based on wind and water erodibility, pH, depth to bedrock, and salinity) and water retention potential (based on PET and available water capacity). We assumed that barren areas had the lowest possible sensitivity to climate conditions since many of these areas cannot be further degraded. Values used for conversions from input data values to fuzzy values are in Table 3.

Final results for the model (Fig. 10) show a complex pattern of site sensitivity across the study area. Barren areas, such as the basin west of the Great Salt Lake, the tops of some mountains, and areas within some canyons stand out with the lowest sensitivity. The most sensitive areas that occur in the southern half of the study area are characterized by low soil pH, high wind erodibility, and low available water capacity while the most sensitive areas in the northwest occur in basins characterized by high soil pH.

7. Discussion

EEMS was designed as a streamlined, fuzzy logic modeling framework. The performance evaluation results provide a strong indication that it has the potential to efficiently process large datasets or large numbers of datasets. Its flexibility as a framework is shown by implementations within the ArcGIS, the web-based Data Basin environments, and under both OS X and Windows operating systems. Implementations for both CSV and NetCDF show how it can be easily adapted to different file formats.

EEMS does not include functionality for results evaluation, sensitivity analysis, or statistics. However, the ease with which an application can be created using the EEMS framework opens the possibility for users to include fuzzy logic modeling as part of a larger modeling system. For example, a user could produce a program that loops through input values or thresholds, runs the logic model for each set of values, and then analyzes the results as desired.

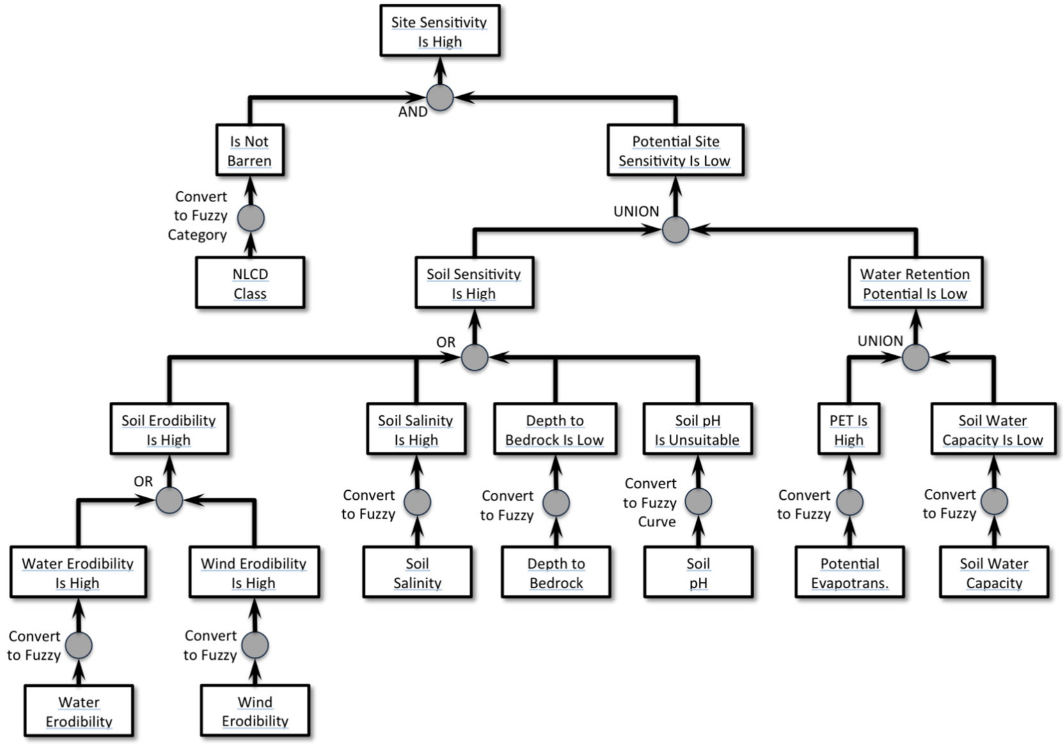

Fig. 9. Fuzzy logic model for Utah and Colorado Plateau site sensitivity.

Since EEMS has the ability to output all intermediate node results the user has the ability to do post-run analyses of all data, including spatial analyses within or between input, intermediate, and final result map layers.

8. Conclusions

EEMS was designed as a versatile fuzzy logic modeling framework that can be easily adapted to different software and hardware

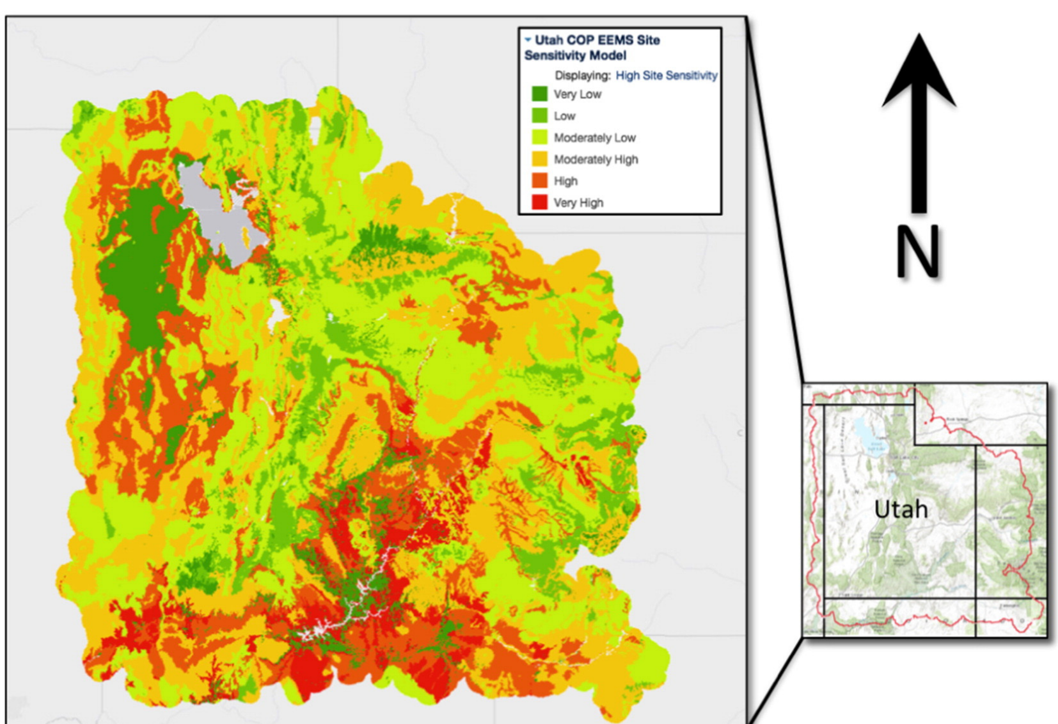

Fig. 10. Map showing the top level (root node) results for Utah/Colorado Plateau high site sensitivity model. Full model and all data layers may be explored using the Data Basin EEMS Explorer at http://goo.gl/DNWfJQ.

Table 2
Soil variables used in the Utah and Colorado Plateau EEMS site sensitivity model, their acronym, the database, calculation for each variable, and URL where the data reside.

Variable	Acronym	Database	Calculation	URL
Available water capacity	AWC	CONUS-SOIL	Weighted average for profile for map unit	http://www.soilinfo.psu.edu
K-factor	Kfact	CONUS-SOIL	Weighted average of 1st layer for map unit	http://www.soilinfo.psu.edu
pH	pH	CONUS-SOIL	Average of 1st layer for map unit	http://www.soilinfo.psu.edu
Depth to BEDROCK	RD	CONUS-SOIL	Average depth for map unit	http://www.soilinfo.psu.edu
Salinity	SAL	STATSGO	Weighted average of 1st layer	https://gdg.sc.egov.usda.gov
Wind erodibility group	WEG	STATSGO	Value for map unit	https://gdg.sc.egov.usda.gov

Table 3
Values used for converting input data values into fuzzy values for the EEMS site sensitivity model.

Input variable	Fuzzy variable	Operator	Conversion values	Units
Water erodibility	Water erodibility is high	CVTTOFUZZY	True threshold = 1 False threshold = 0	index
Wind erodibility	Wind erodibility is high	CVTTOFUZZY	True threshold = 1 False threshold = 7	index
Soil salinity	Soil salinity is high	CVTTOFUZZY	True threshold = 16 False threshold = 8	mho
Depth to bedrock	Depth to bedrock is low	CVTTOFUZZY	True threshold = 2 False threshold = 15	cm
Soil pH	Soil pH is unsuitable	CVTTOFUZZYCURVE	See Fig. 8	pH
Potential evapotranspiration	PET is high	CVTTOFUZZY	True threshold = 2 False threshold = −2	mm
Soil water capacity	Soil water capacity is low	CVTTOFUZZY	True threshold = 0 False threshold = 20	inches
2011 National Land Cover Database Class	Land is not barren	CVTTOFUZZYCAT	False for classes: 22 (high intens. dev.), 24 (low intens. dev.), 31 (barren land); True for all others	class

environments and different file types. We have demonstrated the success of this strategy. EEMS' simple modeling language allows users to easily create fuzzy logic models, and its performance efficiency provides results quickly, even with large and complex models. The full implementations of versions for CSV and NetCDF file formats give users options for running models with two commonly-used data formats. The Arc ModelBuilder implementation allows users to build models via a well-known graphical user interface and apply them to shapefiles and geodatabase feature classes. The EEMS Explorer and Data Basin provide users with an online venue to share and explore EEMS model results.

Acknowledgments

We would like to thank James Strittholt and Brendan Ward for their help with developing EEMS; Barry Baker for collaboration on the Utah and the Colorado Plateau modeling work; Barry Baker, Dominique Bachelet, Lisa Alley, and Rebecca Degagne for reviewing the drafts of this manuscript; and the reviewer for comments and guidance on improving this manuscript.

Appendix A. Supplementary data

Supplementary data to this article can be found online at http://dx.doi.org/10.1016/j.ecoinf.2016.05.001.

References

Adriaenssens, V., De Baets, B., Goethals, P.L.M., De Pauw, N., 2004. Fuzzy rule-based models for decision support in ecosystem management. Sci. Total Environ. 319, 1–12.
Aydi, A., Zairi, M., Ben Dhia, H., 2013. Minimization of environmental risk of landfill site using fuzzy logic, analytical hierarchy process, and weighted linear combination methodology in a geographic information system environment. Environ. Earth Sci. 68, 1375–1389.
Boclin, A., de Mello, R., 2006. A decision support method for environmental impact assessment using a fuzzy logic approach. Ecol. Econ. 58, 170–181.
Bojorquez-Tapia, L.A., Juarez, L., Cruz-Bello, G., 2002. Integrating fuzzy logic, optimization, and GIS for ecological impact assessments. Environ. Manag. 30, 418–433.
Cheung, W.W.L., Pitcher, T.J., Pauly, D.P., 2005. A fuzzy logic expert system to estimate intrinsic extinction vulnerabilities of marine fishes to fishing. Biol. Conserv. 124, 97–111.
Conservation Biology Institute, 2010-2016. Data Basin (website). http://www.databasin.org.
Dawson, T.P., Rounsevell, M.D.A., Kluvánková-Oravská, T., Chobotov, V., Stirling, A., 2010. Dynamic properties of complex adaptive ecosystems: implications for the sustainability of service provision. Biodivers. Conserv. 19, 2843–2853.
ESRI, 1995-2012. An overview of the overlay tools (web page). http://resources.arcgis.com/EN/HELP/MAIN/10.1/index.html#/An_overview_of_the_Overlay_tools/009z000000rm000000/.
ESRI, 1999-2016. ArcMap (Computer Software). ESRI, Inc., Redlands, CA.
Füssel, H.M., 2007. Vulnerability: a generally applicable conceptual framework for climate change research. Environ. Glob. Chang. 17, 155–167.
Gärtner, S., Reynolds, K.M., Hessburg, P.F., Hummel, S., Twery, M., 2008. Decision support for evaluating landscape departure and prioritizing forest management activities in a changing environment. For. Ecol. Manag. 256, 1666–1676.
Giles, R., 1976. Lukasiewicz logic and fuzzy set-theory. Int. J. Man Mach. Stud. 8, 313–327.
Heaton, J., 2014. Fuzzy logic in R. Forcasting Futurism 9, 35–38.
Jasiewicz, J., 2011. A new GRASS GIS fuzzy inference system for massive data analysis. Comput. Geosci. 37, 1525–1531.
Kaplan, K.A., Montero-Serra, I., Vaca-Pita, E.L., Sullivan, P.J., Suarez, E., Vinueza, L., 2014. Applying complementary species vulnerability assessments to improve conservation strategies in the Galapagos Marine Reserve. Biodivers. Conserv. 23, 1509–1528.
Kasabov, N.K., 1996. Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering. MIT Press, Cambridge, MA, and London, England (550 pp.).
Mamdani, E.H., 1977. Applications of fuzzy logic to approximate reasoning using linguistic-synthesis. IEEE Trans. Comput. 26, 1182–1191.
Marchini, A., Facchinetti, T., Mistri, M., 2009. F-IND: a framework to design fuzzy indices of environmental conditions. Ecol. Indic. 9, 485–496.
Mays, M.D., Bogardi, I., Bardossy, A., 1997. Fuzzy logic and risk-based soil interpretations. Geoderma 77, 299–315.
McBratney, A.B., Odeh, O.A., 1997. Application of fuzzy sets in soil science: fuzzy logic, fuzzy measurements and fuzzy decisions. Geoderma 77, 85–113.
McFadden, L., 2007. Vulnerability analysis in environmental management: widening and deepening its approach. Environ. Conserv. 34 (3), 195–204.
Meyer, D., Hornik, K., Buchta, C., 2015. Package 'sets'. https://cran.r-project.org/web/packages/sets/sets.pdf.
Miller, B.J., Saunders, M.S., 2002. The NetWeaver Reference Manual. Pennsylvania State University, College Park, PA, USA.
Mumby, P.J., Chollett, I., Bozec, Y.M., Wolff, N.H., 2014. Ecological resilience, robustness, and vulnerability: how do these concepts benefit ecosystem management? Curr. Opin. Environ. Sustain. 7, 22–27.
Neteler, M., Mitasova, H., 2008. Open Source GIS: a GRASS GIS approach. third ed. Springer, New York (406 pp.).
R Foundation, 2016. The R project for statistical computing (website and software download site). http://www.r-project.org.

Reynolds, K.M., 2001. Fuzzy logic knowledge bases in integrated landscape assessment: examples and possibilities. Gen. Tech. Rep. PNW-GTR-521. U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station, Portland, OR (24 pp.).
Reynolds, K., Paplanus, S., Miller, B., Murphy, P., 2015. Design features behind success of the ecosytem management decision support system and future development. Forests 6, 27–46. http://dx.doi.org/10.3390/f6010027.
Sanner, M.F., 1999. Python: a programming language for software integration and development. J. Mol. Graph. Model. 17, 57–61.
Segui, X., Pujolasus, E., Betro, S., Agueda, A., Casal, J., Ocampo-Duque, W., Rudolph, I., Barra, R., Paez, M., Baron, E., Eljarrat, E., Barcelo, D., Darbra, R.M., 2013. Fuzzy model for risk assessment of persistent organic pollutants in aquatic ecosystems. Environ. Pollut. 178, 23–32.
Wamsley, T.V., Collier, Z.A., Brodie, K., Dunkin, L.M., Raff, D., 2015. Guidance for developing coastal vulnerability metrics. J. Coast. Res. 31 (6), 1521–1530.
Zadeh, L., 1965. Fuzzy sets. Inf. Control. 8, 338–353.
Zadeh, L., 1973. Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans. Syst. Man Cybern. 3, 28–44.