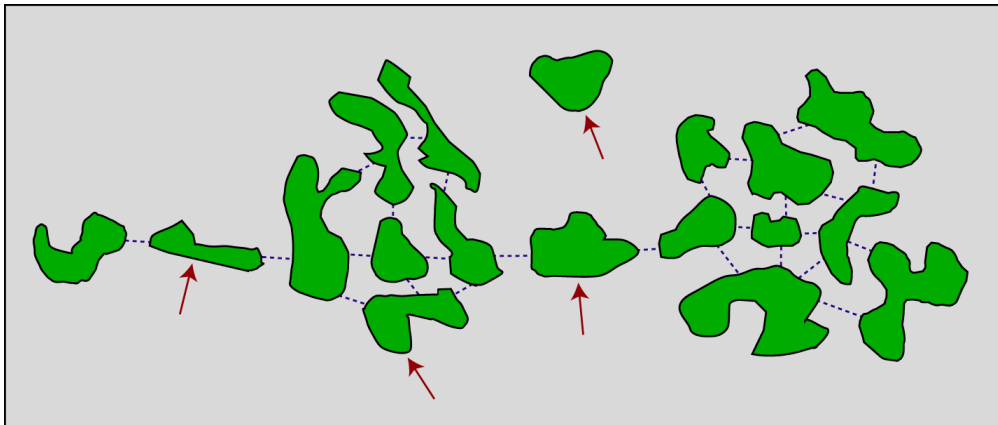




CONEFOR SENSINODE 2.2

USER'S MANUAL



**Software for quantifying the importance of habitat patches
for landscape connectivity through graphs
and habitat availability indices**

June 2007
(with minor updates after that date)

THIS USER'S MANUAL HAS BEEN DEVELOPED BY SANTIAGO SAURA AND LUCÍA PASCUAL-HORTAL
(UNIVERSITY OF LLEIDA, SPAIN)

CONEFOR SENSINODE 2.2 HAS BEEN DEVELOPED BY JOSEP TORNÉ AND SANTIAGO SAURA
(UNIVERSITY OF LLEIDA, SPAIN)

SENSINODE 1.0 WAS DEVELOPED BY DEAN L. URBAN
(DUKE UNIVERSITY, USA)

TABLE OF CONTENTS

| | |
|--|-----------|
| 1. PRODUCT INFORMATION..... | 3 |
| 1.1. AUTHORS AND ACKNOWLEDGEMENTS | 3 |
| 1.2. CONDITIONS OF USE | 3 |
| 1.3. CONTACTING THE AUTHORS | 4 |
| 1.4. SYSTEM REQUIREMENTS..... | 4 |
| 1.5. DOWNLOAD AND INSTALLATION | 4 |
| 1.6. WHAT'S NEW IN CONEFOR SENSINODE 2.2 COMPARED TO VERSION 1.0?..... | 4 |
| 2. INTRODUCTION..... | 6 |
| 2.1. WHAT IS CONEFOR SENSINODE 2.2? | 6 |
| 2.2. LANDSCAPE GRAPHS..... | 6 |
| 2.3. HABITAT AVAILABILITY INDICES | 7 |
| 3. QUICK GUIDE | 9 |
| 4. NODES | 10 |
| 4.1. HABITAT NODES..... | 10 |
| 4.2. FORMAT OF THE NODE FILE..... | 10 |
| 5. CONNECTIONS..... | 13 |
| 5.1. BINARY AND PROBABILISTIC CONNECTION MODELS | 13 |
| 5.2. CHARACTERISING CONNECTIONS BETWEEN NODES | 14 |
| 5.2.1. TYPES OF CONNECTIONS | 14 |
| 5.2.2. CHARACTERISING CONNECTIONS IN THE BINARY MODEL | 15 |
| 5.2.2.1. Connection information as distances..... | 15 |
| 5.2.2.2. Connection information as probabilities..... | 15 |
| 5.2.2.3. Connection information as links..... | 16 |
| 5.2.3. CHARACTERISING CONNECTIONS IN THE PROBABILISTIC MODEL | 16 |
| 5.2.3.1. Connection information as distances..... | 16 |
| 5.2.3.2. Connection information as probabilities..... | 17 |
| 5.3. FORMAT OF THE CONNECTION FILE | 18 |
| 6. CONNECTIVITY INDICES | 21 |
| 6.1. BINARY INDICES..... | 21 |
| 6.1.1. NL - Number of links | 21 |
| 6.1.2. NC - Number of components | 21 |
| 6.1.3. H - Harary index | 21 |
| 6.1.4. CCP - Class coincidence probability | 21 |
| 6.1.5. LCP - Landscape coincidence probability | 22 |
| 6.1.6. IIC - Integral index of connectivity | 22 |
| 6.2. PROBABILISTIC INDICES..... | 23 |
| 6.2.1. F - Flux..... | 23 |
| 6.2.2. AWF - Area-weighted flux | 23 |
| 6.2.3. PC - Probability of connectivity..... | 23 |
| 6.2.3.1. Description of the PC index | 23 |
| 6.2.3.2. Advanced settings for the PC index | 24 |
| 6.3. MAXIMUM LANDSCAPE ATTRIBUTE | 26 |
| 7. RESULTS | 28 |
| 7.1. NODE IMPORTANCES..... | 28 |
| 7.1.1. COMPUTATION OF NODE IMPORTANCES | 28 |
| 7.1.2. INTERPRETATION OF NODE IMPORTANCES | 29 |

| | |
|--|-----------|
| 7.1.3. THE NODE IMPORTANCE FILE | 30 |
| 7.2. OTHER RESULTS | 31 |
| 7.2.1. OVERALL INDEX VALUES | 31 |
| 7.2.2. COMPONENTS | 31 |
| 7.2.3. LINKS..... | 32 |
| 7.2.4. PROBABILITIES | 32 |
| 7.2.4.1. Direct probabilities..... | 32 |
| 7.2.4.2. Maximum product probabilities..... | 32 |
| 7.2.5. EXECUTION EVENTS..... | 33 |
| 8. OTHER FEATURES..... | 34 |
| 8.1. OTHER MENUS | 34 |
| 8.2. GIS EXTENSIONS FOR CALCULATING THE CONNECTION AND NODE FILES..... | 35 |
| 8.2.1. EXTENSION FOR ARCGIS 9.X..... | 35 |
| 8.2.2. EXTENSION FOR ARCVIEW 3.X..... | 39 |
| 8.2.3. OTHER SOFTWARE TOOLS AND COMPUTATION OF EFFECTIVE DISTANCES | 41 |
| 8.3. SAMPLE DATA | 41 |
| 9. LIMITATIONS AND KNOWN ERRORS..... | 44 |
| 9.1. PROCESSING CAPABILITIES AND LIMITATIONS | 44 |
| 9.2. KNOWN ERROR AND WARNING MESSAGES | 45 |
| REFERENCES..... | 55 |

This user's manual is organised in several sections that intend to provide all the necessary help and information for the use of the Conefor Sensinode 2.2 software. However, several relevant theoretical and background concepts are not fully developed in this manual, but described in the scientific papers by Pascual-Hortal and Saura (2006) and by Saura and Pascual-Hortal (2007), as well as in other papers listed in the reference section. Users are encouraged to carefully read these papers for an adequate use, understanding and interpretation of the results provided by this software.

1. PRODUCT INFORMATION

1.1. AUTHORS AND ACKNOWLEDGEMENTS

Conefor Sensinode 2.2 (hereafter CS22) has been developed by Josep Torné and Santiago Saura at the University of Lleida (Spain), and funded by the Spanish Ministry of Science and by European FEDER funds through the Conefor project (REN2003-01628, Plan Nacional de I+D+I), directed by S. Saura. CS22 also benefited from the collaboration of Lucía Pascual-Hortal and Joan Gimbert.

Sensinode 1.0 (Landgraphs package) was developed by Dean L. Urban (Duke University, USA), and was the seminal software and starting point for CS22. CS22 has been developed by modifying, reprogramming and including new indices and features in Sensinode 1.0. Dr. Dean L. Urban generously provided the source codes of Sensinode 1.0, making possible the development of CS22.

1.2. CONDITIONS OF USE

Conefor Sensinode 2.2 (CS22) is distributed free of charge for non-commercial use, with the only condition that this software and the following references are cited in any document, publication or application where CS22 is used:

- Pascual-Hortal, L. and Saura, S. 2006. Comparison and development of new graph-based landscape connectivity indices: towards the prioritization of habitat patches and corridors for conservation. *Landscape Ecology* 21 (7): 959-967.
- Saura, S. and Pascual-Hortal, L. 2007. A new habitat availability index to integrate connectivity in landscape conservation planning: comparison with existing indices and application to a case study. *Landscape and Urban Planning* 83 (2-3): 91-103.
- Saura, S. and Torné, J. 2009. Conefor Sensinode 2.2: a software package for quantifying the importance of habitat patches for landscape connectivity. *Environmental Modelling and Software* 24 (1): 135-139.

Additional papers that may be relevant for the background of this software can be found in the references section. Although the citation of the abovementioned references is preferred, in case you may need to cite directly this user's manual please follow the following format:

- Saura, S., and Pascual-Hortal, L. 2007. Conefor Sensinode 2.2 User's Manual: Software for quantifying the importance of habitat patches for maintaining landscape connectivity through graphs and habitat availability indices. University of Lleida, Spain. Available at www.conefor.org.

You may check the web (<http://www.conefor.org>) for updates of the Conefor Sensinode software package, this user's manual or other papers related to the software, the methodology, or relevant applications.

Conefor Sensinode 2.2 has been thoroughly checked by the authors in order to avoid errors and to ensure a smooth and reliable performance. However, the software is provided "as-is", without warranty of any kind. The user assumes all the responsibility for the accuracy and suitability of this program for a particular application. Use it at your own discretion and risk.

The authors appreciate hearing about the applications in which CS22 is used, as well as help in reporting bugs or suggestions for improvement (conefor@gmail.com). However,

only very limited user support may be provided and only for specific questions that cannot be solved by other means after carefully reading this user's manual and the related papers.

1.3. CONTACTING THE AUTHORS

Contact the authors for questions, comments and further information (with the limitations described above) at conefor@gmail.com and/or through the contact form available at www.conefor.org.

1.4. SYSTEM REQUIREMENTS

CS22 only requires a standard computer running a Windows operative system. CS22 has been successfully tested under Windows XP, Windows Vista, Windows 7 and Windows ME, although it may also run fine in other Windows versions. The installation of CS22 only requires about 8 MB of free space in your hard disk.

Any standard computer is usable for most of the applications of CS22. However, when dealing with very large datasets, the amount of RAM memory (GB) in your computer will limit the maximum number of nodes that can be processed in a single run with CS22. Additionally, for the most computationally intensive indices, the processing speed (GHz) of your CPU will determine how fast the node importance values can be calculated and, depending on the total number of nodes and links/connections, if the full processing can be accomplished in a reasonable time (e.g. less than 24 hours). Find more details on this in section 9.1.

1.5. DOWNLOAD AND INSTALLATION

Conefor Sensinode 2.2 can be downloaded from <http://www.conefor.org>.

From these links you can download the file "CS22install.exe". This is an auto-installation file containing the software itself (CS22.exe and software libraries), the user's manual, two relevant papers, two simple sample datasets, and an extension for ArcView 3.x and ArcGIS 9.x that allows generating the node and connection files in the format required by Conefor Sensinode. To install the software just run CS22install.exe and follow the instructions of the installation wizard. The installation only requires about 8 MB of free space in your hard disk.

1.6. WHAT'S NEW IN CONEFOR SENSINODE 2.2 COMPARED TO VERSION 1.0?

The main differences and new features of CS22 compared to the Sensinode 1.0 version are:

- CS22 includes new improved connectivity indices (integral index of connectivity, probability of connectivity) that have been shown to present relevant properties and to overcome several limitations of other existing indices, which makes them particularly suited for landscape conservation planning applications (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007). These new habitat availability indices are those most recommended for use within CS22.
- CS22 includes other commonly used indices in the context of graph-theoretical landscape connectivity analysis that were not included in the 1.0 version. Only the F and AWF indices were already included in Sensinode 1.0.

- CS22 has been developed in C++ with a visual and menu-based interface directly running in a Windows operating system (intending to be user friendly and simple to use), in contrast to the Fortran MS-DOS command prompt of Sensinode 1.0.
- CS22 includes the possibility of evaluating the connectivity improvement provided by new potential habitat sites that may be added in the landscape (through habitat creation or restoration), in addition to the quantification of the importance for maintaining connectivity of those habitat sites that already exist in the landscape (the latter was already included in Sensinode 1.0 for some indices).
- CS22 has more powerful processing capabilities than 1.0 version, being able to process much larger sets of nodes and reducing the required processing times. This is achieved through different graph algorithms and C++ programming, among others.
- CS22 computes and provides as a result the maximum product probabilities between nodes.
- CS22 allows entering the connection file as a partial file (only including those pairs of nodes that are directly connected to some degree).
- Some of the results (node importances, components) can be directly saved in DBF format, which facilitates further post-processing and directly joining the results in a GIS layer.

2. INTRODUCTION

2.1. WHAT IS CONEFOR SENSINODE 2.2?

CONEFOR Sensinode 2.2 (CS22) is a simple program for quantifying the importance of habitat patches for maintaining landscape connectivity through graph structures and habitat availability indices.

CS22 is conceived as a tool for decision-making support in landscape planning and habitat conservation.

CS22 includes several connectivity indices, but the best performing and recommended indices are the integral index of connectivity (IIC) (Pascual-Hortal and Saura 2006) and especially the probability of connectivity (PC) (Saura and Pascual-Hortal 2007).

CS22 approaches connectivity from a functional perspective. That is, not only the spatial arrangement of the habitat is considered (structural connectivity), but also the dispersal distances and the behavioural response of individuals or species to the physical structure of the landscape (functional connectivity) may be taken into account in the analysis (Tischendorf and Fahrig 2000, Theobald 2006).

The inputs for CS22 are therefore both the spatial structure and configuration of the landscape mosaic (e.g. pattern of habitat patches) and the movement ability for dispersal of the focal species (e.g. dispersal distance). Both inputs and outputs consist of numerical files (in ASCII text or DBF formats) which may be easily obtained from or incorporated into a GIS, word processor or spreadsheet program.

The software provides different outputs (see Fig. 1 and section 7), being the most outstanding one the importance of each individual node (habitat patch) for maintaining overall landscape connectivity, according to different indices. This allows ranking habitat patches (patches prioritization) by their contribution to landscape connectivity, which provides objective criteria for the selection of the most critical habitat areas for conservation planning purposes.

2.2. LANDSCAPE GRAPHS

A landscape mosaic and its intricate network of functional connections are here described and analysed within the graph theory approach. This allows optimizing the mathematical operations and overcoming the typical computational limitations that appear when dealing with large data sets and performing complex analyses regarding connectivity. Note that tackling the analysis of landscape connectivity at broad scales may easily imply considering hundreds of patches and connections for the calculations, which requires a considerable computational efficiency.

A graph is a set of nodes (or vertices) and links (or edges) such that each link connects two nodes. Nodes here represent sites of suitable habitat (patches, cells, etc.) surrounded by inhospitable habitat (non-habitat) while links symbolise the potential ability of a species to directly disperse between two nodes.

Graph structures and algorithms have been shown to be a powerful and effective way of both representing the landscape pattern as a network of functionally interconnected patches and performing complex analysis regarding landscape connectivity (Bunn et al. 2000, Ricotta et al. 2000, Urban and Keitt 2001, Jordán et al. 2003, Pascual-Hortal and Saura 2006, Pascual-Hortal and Saura 2008).

Additionally, it has been suggested that graph-theoretical indices possess the greatest benefit to effort ratio for conservation problems that require characterization of connectivity at relatively large scales. These measures provide a reasonably detailed picture of potential connectivity, but have relatively modest data requirements (Calabrese and Fagan 2004).

Nevertheless, although CS22 is developed under a complex theoretical and mathematical basis, the software is intended to be easy to use and interpret for non-expert users, and it is therefore built up within a user-friendly interface.

2.3. HABITAT AVAILABILITY INDICES

The habitat availability (reachability) concept is based in considering a patch itself as a space where connectivity occurs, integrating habitat patch area (or other patch attributes like habitat quality) and connections between different patches in a single measure (Pascual-Hortal and Saura 2006). This approach recognises that in some cases the connected habitat area existing within the patches themselves (intrapatch connectivity) may be larger than the one made available by the connections between habitat patches (interpatch connectivity). For a habitat being easily available (reachable) for an animal or population, it should be both abundant and well connected. Therefore, habitat availability for a species may be low if habitat patches are poorly connected, but also if the habitat is highly scarce even if the patches are highly connected (Pascual-Hortal and Saura 2006). This is different from the strict and traditional definition of connectivity that only takes into account the connectivity among different patches (and not within them). It has been suggested that landscape connectivity should be considered within the wider concept of habitat availability in order to be successfully integrated in landscape conservation planning applications (Pascual-Hortal and Saura 2006). In this way, connectivity is conceived (and measured) as that property of the landscape that determines the amount of reachable habitat in the landscape, no matter if such reachable habitat comes from big and/or high quality habitat patches themselves (intrapatch connectivity), from strong connections between different patches (interpatch connectivity) or, more frequently, from a combination of both.

CS22 includes several new indices (IIC and PC) based in the concept of habitat availability (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007). These new indices improve the characteristics of other existing indices for landscape conservation planning applications, including their abilities both for adequately reacting to relevant landscape changes (such as habitat loss) and for identifying the most critical habitat areas for the maintenance of landscape connectivity (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007, Pascual-Hortal and Saura 2007).

Note that not all the indices included in this software are habitat availability indices. CS22 also includes other conceptually different indices, which are basic or classical graph connectivity metrics, or metrics that have been used in previous landscape connectivity studies. Comparison between the results provided by the different indices may be itself interesting and provide complementary information on the connectivity of the analysed landscapes. However, it should be taken into account that some of these indices may present serious limitations for their use in the context of connectivity conservation planning (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007).

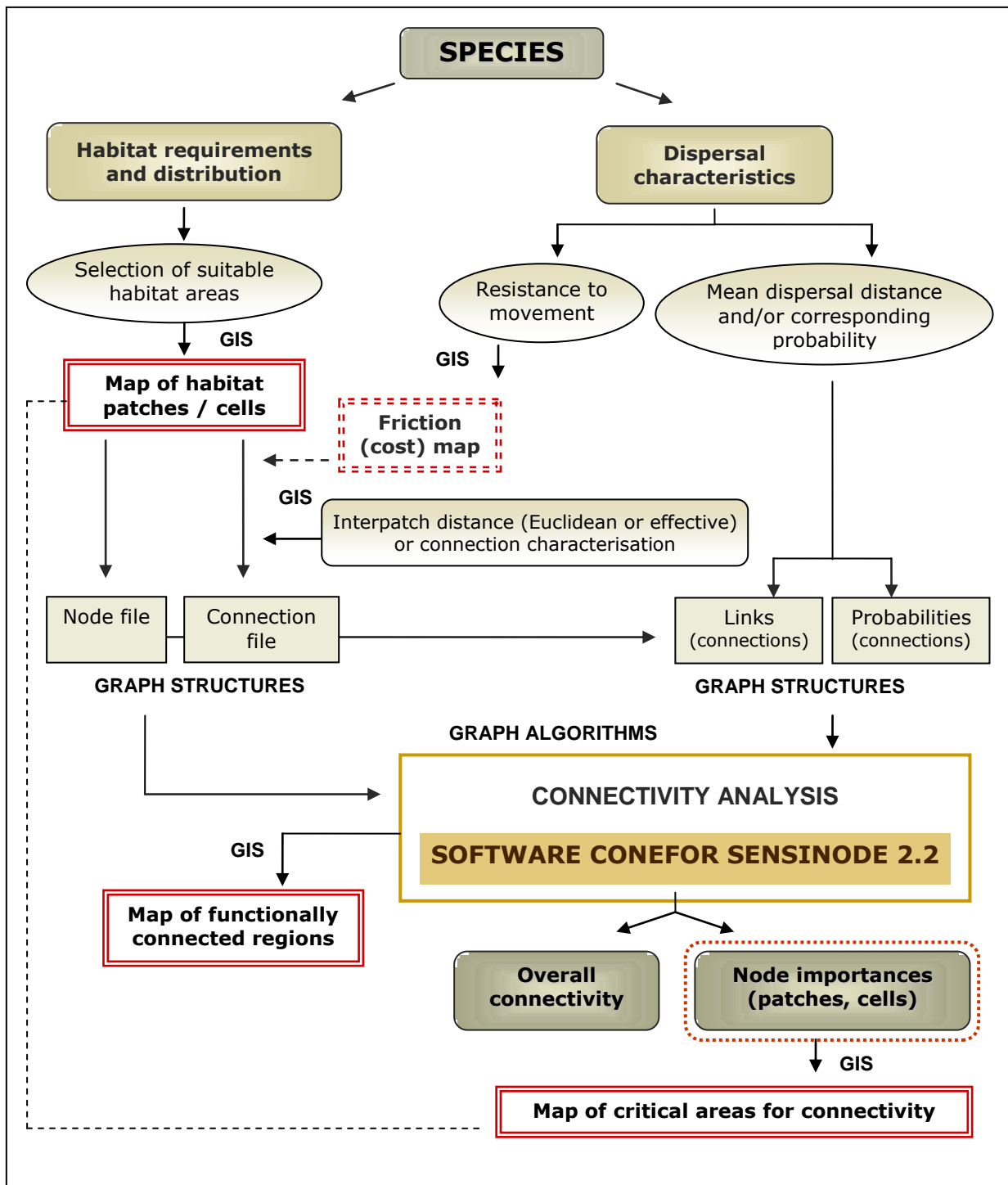


Figure 1. Schematic outline of the methodology for the analysis of landscape connectivity through the Conefor Sensinode 2.2 software.

3. QUICK GUIDE

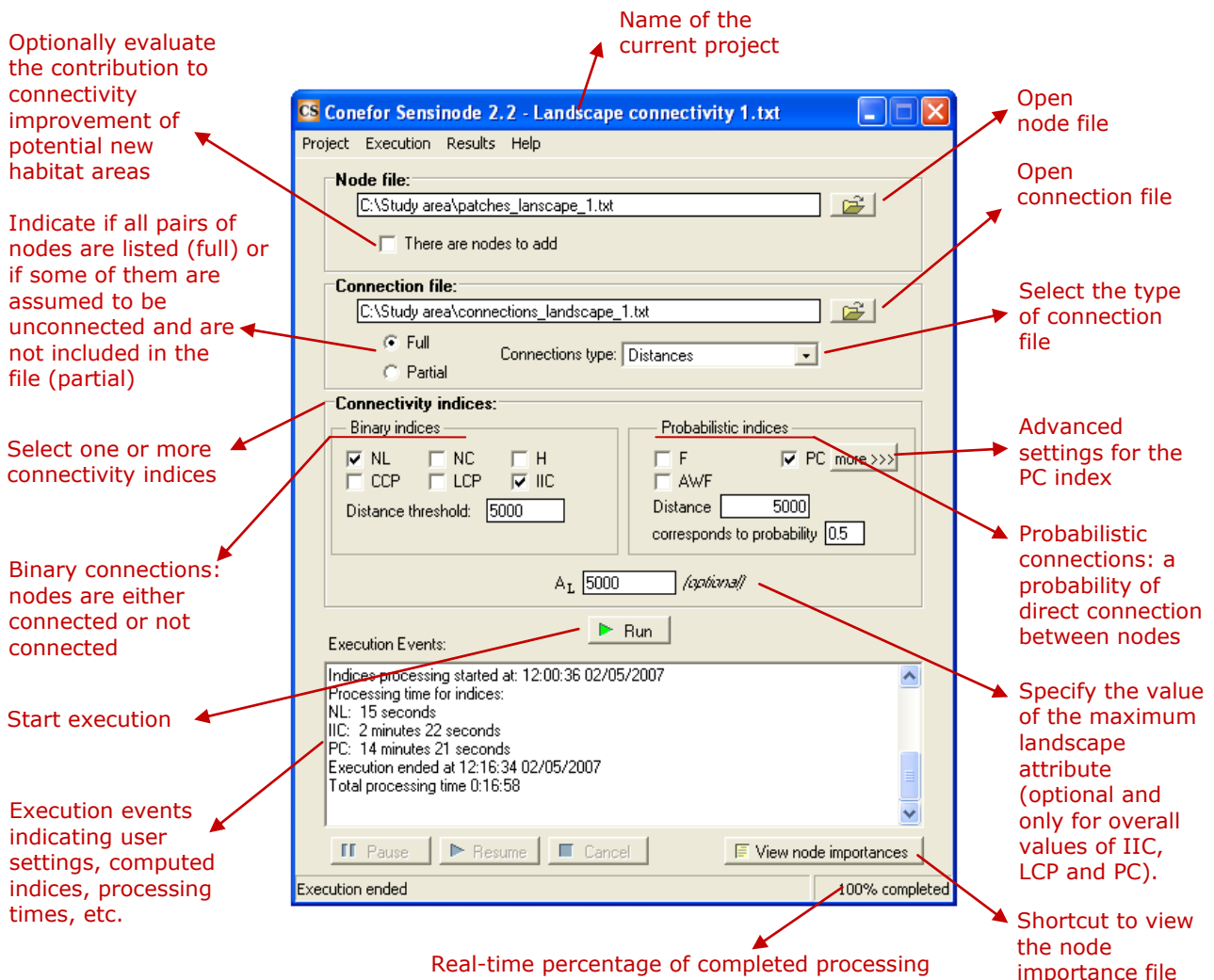


Figure 2. Main screen of the Conefor Sensinode 2.2 software.

STEP BY STEP (Typical case)

1. Browse for the *node* and *connection* files on your computer.
2. Choose between *full* or *partial* format (usually the first) for the connection file and specify the *type of connections* in that file, usually *distances* (although it may also be *probabilities* or *links*).
3. Select the desired *connectivity index* or indices for the analysis.
4. Enter the distance threshold for the binary indices and/or both the distance and corresponding probability for the probabilistic indices.
5. Click on the *Run* button to start the execution process.
6. View or save the different results and the corresponding files (node importances, overall index values, components, internode links and/or probabilities, execution events).

4. NODES

4.1. HABITAT NODES

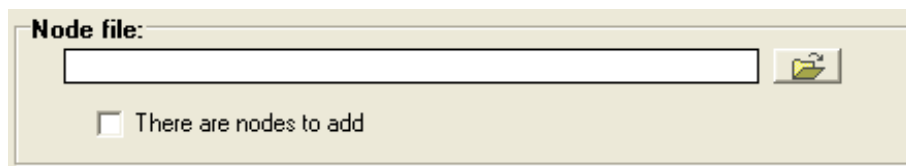
Nodes represent the differentiated spatial units (habitat areas) that are considered for the connectivity analysis. They will typically represent habitat patches or habitat cells, but they may as well correspond to any other unit discriminated in the landscape based on ecological, administrative or management criteria (e.g. ownerships, protected areas, public forests, etc.).

CS22 will determine the importance of each of the nodes for maintaining or improving landscape connectivity according to different indices, as described below (sections 6.1, 6.2 and 7.1).

The node attribute is the characteristic of the node that is considered relevant for the analysis, such as habitat area, habitat quality, quality-weighted area or some other attributes where appropriate (e.g. population density, core area, carrying capacity, habitat suitability, etc.). The node attribute corresponds to the a_i variable used for the computation of many of the connectivity indices (see section 6).

The adequate determination of the nodes and their attributes is the responsibility of the user and depends on:

- The objectives and characteristics of particular applications.
- The spatial scale of the analysis.
- The available information on the species and the landscape.



If the option "There are nodes to add" is selected in the main screen, CS22 assumes that the analysis will include, in addition to the nodes already existing in the landscape, some potential new habitat areas (nodes) that currently do not exist but that may be added in the landscape (through habitat creation or restoration). In this case, CS22 will also compute the contribution of these potential new nodes to the improvement of landscape connectivity (see section 7.1). As long as this option is not selected, CS22 will assume that only nodes currently existing in the landscape have been included in the analysis, and will just compute their importance for maintaining landscape connectivity.

4.2. FORMAT OF THE NODE FILE

The node file is a standard ASCII TEXT FILE (*.txt) with columns separated either by spaces or tabulations. The file should ONLY include NUMERICAL VALUES, and not any text characters (the columns should not have headings).

In the STANDARD CASE (no nodes to add) the node file has TWO columns:

- Column 1: node IDs as INTEGER numerical values (no other number formats or text characters should be used in this column).
- Column 2: node attributes as non-negative numerical values (either integer or floating point values are allowed in this column).

The following are simple examples of node files in the standard case:

| | |
|---|----------|
| 1 | 51.37761 |
| 2 | 32.16446 |
| 3 | 6.20032 |
| 4 | 68.77729 |
| 5 | 14.39476 |
| 6 | 4.63687 |

| | |
|----|--------|
| 1 | 0.2347 |
| 5 | 0.5274 |
| 9 | 0.9721 |
| 10 | 0.729 |
| 14 | 0.2366 |
| 17 | 0.4428 |

If the "There are NODES TO ADD" option is selected, the node file needs to have THREE columns:

- Column 1: node IDs as INTEGER numerical values (no other number formats or text characters should be used in this column).
- Column 2: node attributes as non-negative numerical values (either integer or floating point values are allowed in this column).
- Column 3: an integer value of 0 or 1 indicating for each node whether it already exists in the landscape (1) or is a candidate to be added to the landscape (0). No other values different from 0 or 1 should appear in this column.

Note that all the nodes are listed in the same file (those already existing and those that may be added to the landscape). The following are simple examples of node files when there are nodes to add:

| | | |
|------|----------|---|
| 1206 | 5.37761 | 1 |
| 2024 | 3.16446 | 0 |
| 315 | 6.20032 | 1 |
| 1476 | 68.77729 | 0 |
| 523 | 4.39476 | 1 |
| 690 | 4.63687 | 1 |
| 1722 | 11.7161 | 0 |
| 874 | 62.2495 | 1 |

| | | |
|----|-----------|---|
| 5 | 1235.37 | 1 |
| 6 | 2333.16 | 1 |
| 7 | 645.3987 | 1 |
| 8 | 72.4356 | 0 |
| 9 | 84.7231 | 1 |
| 10 | 1364.6249 | 0 |
| 11 | 1321.6299 | 1 |
| 12 | 2128.2452 | 0 |

The node IDs do not need to be consecutive nor ranked, as shown in the examples above. The nodes to add, when present, can be listed in any place in the file, and do not need to be listed together (they will just be identified by the 0 value in the third column).

In all the cases, the DECIMAL SEPARATORS need to be POINTS, not commas or other character, as in the examples above. Additionally, the numbers should NOT include any THOUSAND SEPARATOR in any of the columns. The files should always be formatted in this way. Therefore you will probably need to set the regional configuration settings on your computer accordingly.

In all the cases, the node file SHOULD END with a BLANK LINE.

The units of the node attributes are up to the user. The choice of the units does not affect the node importances for any of the indices. The only obvious requirement is that all the attributes are in the same units. The units do not neither affect the overall values of many of the indices (NL, NC, H, CCP, LCP, IIC, PC), but just the overall values of F and AWF, and the numerators of LCP, IIC and PC (LCPnum, IICnum, PCnum) (see section 6).

It is the full responsibility of the user to provide an adequately formatted node file (as described above) in order to obtain valid results through this software. Although CS22 notifies the user of some common format errors (see section 9.2), several others may not be detected or reported.

As described in section 8.2, you may use the ArcGis 9.x extension ("Conefor inputs") included with this distribution to directly generate from a GIS layer the node file in the format required by CS22.

In the "Sample data" folder (located within the folder where you installed CS22, which by default is "Conefor Sensinode 22") you can find some simple examples of node files (*.txt).

Note that the node file does not include any information on the spatial coordinates of the nodes, since the distances between nodes are computed externally to CS22 (see section 8.2). The node IDs allow you to link the results provided by CS22 to geospatial data within a GIS.

5. CONNECTIONS

5.1. BINARY AND PROBABILISTIC CONNECTION MODELS

When performing connectivity analysis, two different connection models are possible (Saura and Pascual-Hortal 2007):

- ◆ **Binary** (graphs with unweighed links)
 - ◆ **Probabilistic** (graphs with weighed links)
- The **binary connection model** considers each two nodes (e.g. habitat patches) as either connected or not, with no intermediate modulation of the connection strength or dispersal feasibility among them. It is therefore more simplified than the probabilistic model described below.

The existence of a link between a pair of nodes implies the potential ability of an organism to directly disperse between these two nodes, which are considered connected.

A link between two nodes (e.g. patches) is typically assigned in this model by comparing the distance between them and the specified threshold dispersal distance for the organism or species under study. If the internode distance is shorter than the threshold dispersal distance, a link is assigned to that pair of nodes. In this way, all the nodes within a given distance from each other (below the threshold distance) are considered econnected with the same strength.

Distances between patches (nodes) may be Euclidean (straight-line from edge to edge) or, preferably, effective (least-cost) distances by considering the different movement abilities and mortality risk of a species through different land cover types (e.g. Adriaensen et al. 2003, Theobald 2006).

- The **probabilistic connection model** characterises the connections through a probability of direct dispersal between each two nodes (e.g. habitat patches or cells), as an estimation of the strength, frequency or feasibility of that direct movement by the analysed organisms.

The probability of direct dispersal among nodes (p_{ij}) may be modelled within CS22 as a decreasing exponential function of the interpatch distance (see Eq. 1), either Euclidean or effective (minimum-cost) distance. The probabilities of direct dispersal may be also computed externally to CS22, as derived from example from actual movement patterns monitoring or mark-release-recapture methods (if those data-intensive measurements can be carried out), and provided as an input file of already calculated probabilities to CS22, as described below.

Depending on the type of information available for a particular analysis and the purposes of the application, one or other connection model may be used.

The figure below (Fig. 3) illustrates the differences between these two models.

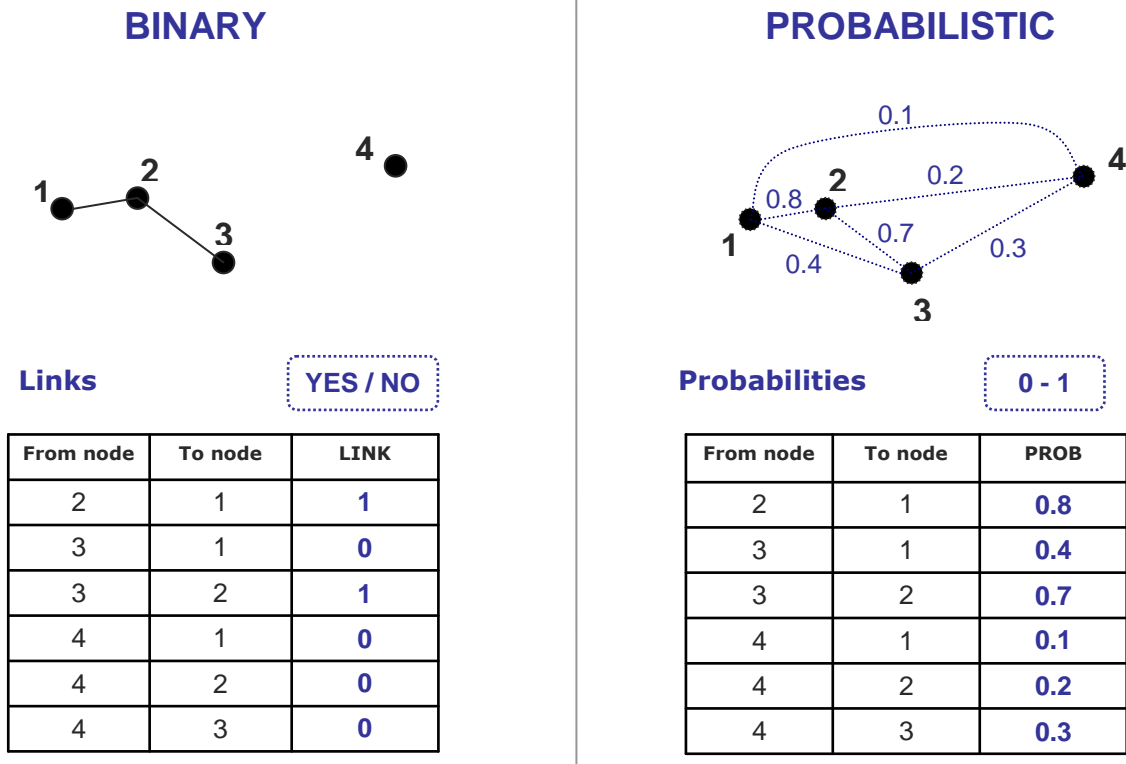


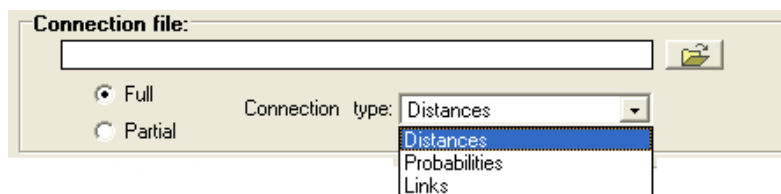
Figure 3. Differences in the determination and quantification of direct connections between nodes in the binary and probabilistic connection models.

5.2. CHARACTERISING CONNECTIONS BETWEEN NODES

The connectivity analysis implemented in CS22 considers that connections between nodes are symmetric (undirected graphs), both for the binary and the probabilistic model. This means that if node i is linked to node j , node j is linked to node i , and that the probability of direct dispersal between two nodes is the same from node i to node j (p_{ij}) than from node j to node i (p_{ji}).

5.2.1. TYPES OF CONNECTIONS

The connections between nodes may be characterised by the user through one of the following three types of connections, as selected in the main screen of CS22:



- Distances between nodes, which allows computing both the binary and probabilistic connectivity indices, as described below.

- Probabilities of direct dispersal between nodes (values ranging from 0 to 1), which allows computing both the binary and probabilistic connectivity indices, as described below.

- Links between nodes, with Boolean values indicating whether a link exists (1) or not (0) between two nodes, which only allows computing the indices based on the binary connection model.

5.2.2. CHARACTERISING CONNECTIONS IN THE BINARY MODEL

5.2.2.1. Connection information as distances

When the connection type is "Distances" and one or more of the binary indices are selected for computation, the user needs to specify a distance threshold value (within the binary indices box). In this way the links required for the binary indices can be determined. All pairs of nodes separated by a distance larger than the threshold distance will be considered directly unconnected (no link between them), while a link will be assigned to those nodes with an internode distance below or equal to the threshold. The number format in this distance threshold box should be consistent with the regional configuration settings of your computer (point or comma as decimal separator) but should not include any thousand separator.

The image shows a screenshot of a software interface. At the top, there is a dropdown menu labeled 'Connection type:' with 'Distances' selected. An arrow points down from this menu to a text input field labeled 'Distance threshold:'.

If both binary and probabilistic indices are computed for the same case study, it is the responsibility of the user to provide coherent settings in the boxes for the binary and probabilistic indices (as described below in section 5.2.3.1), so that the results for the different indices are based on comparable information.

As described in section 8.2, you may use the ArcGIS 9.x extension ("Conefor inputs") or the ArcView 3.x extension "ID Within Distance: Conefor" (both included with this distribution) to directly calculate Euclidean edge-to-edge distances among all the given polygon features (nodes, habitat patches) of a theme (shapefile). Other software tools may also be available elsewhere to compute Euclidean or effective (minimum cost) distances between nodes (see section 8.2.3).

5.2.2.2. Connection information as probabilities

When the connection type is "Probabilities" and one or more of the binary indices are selected for computation, the user needs to specify a probability threshold value (within the binary indices box). In this way the links required for the binary indices can be determined. All pairs of nodes with a probability of direct dispersal lower than the threshold probability will be considered directly unconnected (no link between them), while a link will be assigned to those nodes with an internode direct dispersal probability equal to or above the threshold. The number format in this probability threshold box should be consistent with the regional configuration settings of your computer (e.g. point or comma as decimal separator).

The image shows a screenshot of a software interface. At the top, there is a dropdown menu labeled 'Connection type:' with 'Probabilities' selected. An arrow points down from this menu to a text input field labeled 'Probability threshold:'.

If both binary and probabilistic indices are computed for the same case study, it is the responsibility of the user to provide a probability threshold value that is coherent with the probabilities that will be used for the probabilistic indices, so that the results for the different indices are based on comparable information.

5.2.2.3. Connection information as links

In this case, no other value or option is required, since all the information on the links existing in the landscape (as required for the binary indices) is already included in the input connection file.

5.2.3. CHARACTERISING CONNECTIONS IN THE PROBABILISTIC MODEL

5.2.3.1. Connection information as distances

When the connection type is "Distances" and one or more of the probabilistic indices are selected for computation, the user needs to specify the probability value corresponding to a specific distance (within the probabilistic index box).

The image shows a software interface element. At the top, there is a label 'Connection type:' followed by a dropdown menu where 'Distances' is selected. An arrow points downwards from this dropdown to a second box. This second box contains two input fields: the first is labeled 'Distance' and the second is labeled 'corresponds to probability'.

These distance-probability values may be set in different ways. For example, if the distance corresponds to the median dispersal distance of the species under analysis, you may set a probability value of 0.5 (Fig. 4). If the distance is a maximum dispersal distance, set the probability value to 0.05 or 0.01, depending on the characteristics of the available information. The number format in these distance-probability boxes should be consistent with the regional configuration settings of your computer (point or comma as decimal separator) but should not include any thousand separator.

From this information, CS22 automatically computes and internally uses the internode direct dispersal probabilities (p_{ij}) resulting from a decreasing negative exponential function of distance (e.g. Bunn et al. 2000, Keitt et al. 1997, Urban and Keitt 2001, Saura and Pascual-Hortal 2007) that matches with the probability-distance values specified by the user (Fig. 4), as follows:

$$p_{ij} = e^{-k \cdot d_{ij}} \quad (1)$$

where d_{ij} is the distance between nodes i and j (as specified in the "Distances" connection file), p_{ij} is the probability of direct dispersal between nodes i and j (i.e. the probability that an organism is able to disperse a distance equal or larger than the distance between those nodes), and k is a constant set to make the function match to the probability-distance values specified by the user.

In this way the direct dispersal probabilities required for the probabilistic indices are determined. The direct dispersal probabilities (p_{ij}) resulting from this computation can be saved and viewed as a result of the analysis (Results->Probabilities->Direct), as described in section 7.2.4.1.

Note that if both binary and probabilistic indices are computed for the same case study, it is the responsibility of the user to provide coherent settings for the binary and probabilistic indices, so that the results for the different indices are based on comparable information.

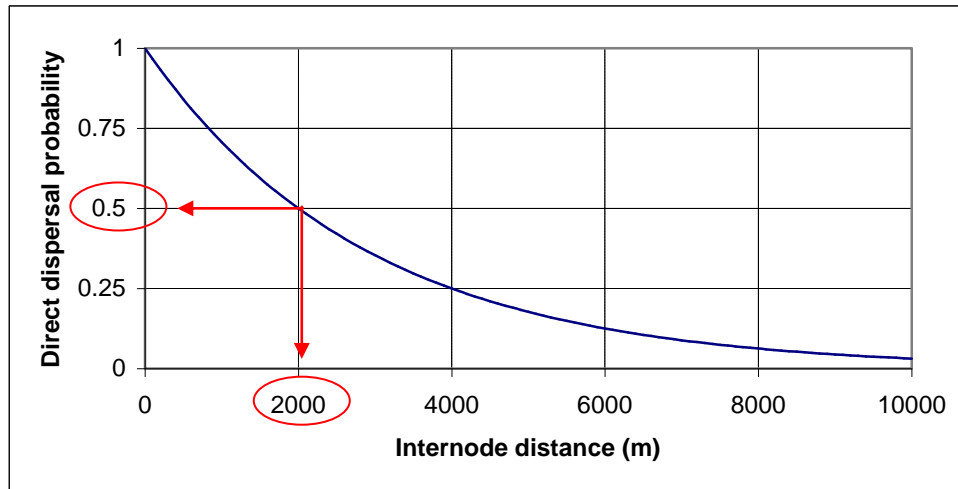


Figure 4. Probability of direct dispersal between nodes calculated as a decreasing exponential function of internode distances (Eq. 1). The user settings are in this case a distance of 2000 m corresponding to a probability of 0.5.

If the probabilities need to be calculated in a different way from the exponential function above, the user can compute them externally to CS22 and provide them directly to the software as a "Probabilities" connection file, as described below.

If needed, you may use the ArcView 3.x or ArcGIS 9.x extension (included with this distribution and described in section 8.2) to directly calculate Euclidean edge-to-edge distances among all the given polygon features (nodes, habitat patches) of a theme (shapefile). Other software tools may also be available elsewhere to compute Euclidean or effective (minimum cost) distances between nodes (see section 8.2).

5.2.3.2. Connection information as probabilities

The user can directly provide as an input a connection file with the probabilities of direct dispersal among nodes calculated in some other way (different from the negative exponential described above) or as derived from specific mark-release-recapture or radiotracking data.

In this case, no other value or option is required, since the final information on the probabilities required by this type of indices is already provided by the user.

Finally, note that in the case that the connection type is "links", only binary indices can be computed, and not any of the probabilistic ones.

5.3. FORMAT OF THE CONNECTION FILE

The connection file is a standard ASCII TEXT file (*.txt) with columns separated either by spaces or tabulations. The file should ONLY include NUMERICAL VALUES, and not any text characters (the columns should not have any headings).

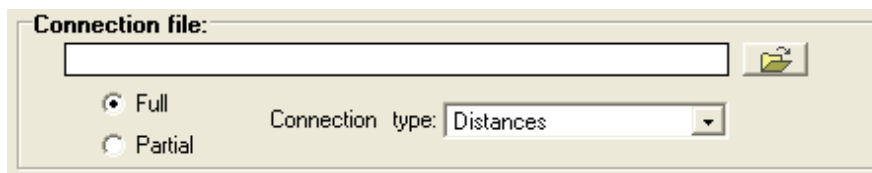
The connection file should always have the following THREE COLUMNS (no matter which type of connections is selected):

- Column 1: ID of node i (integer value).
- Column 2: ID of node j (integer value).
- Column 3: Information on the direct connection between node i and j , which may be a non-negative distance value, a probability value (ranging from 0 to 1), or a link integer value (0 or 1), depending on the type of connection file.

The nodes IDs should obviously be the same than those in the node file.

Since the internode connections are considered by CS22 as symmetric (the connection from i to j being the same than from j to i), EACH PAIR OF NODES should be LISTED ONLY ONCE in the connection file.

The user should select one of the following two formats for the connection file in the main screen of CS22:



- FULL (option by default): All pairs of nodes are listed once in the connection file. In this case, if n is the number of nodes (lines) in the node file (including the "nodes to add" if there are some), the connection file should have exactly $n(n-1)/2$ lines (note that the case $i=j$ should not be included). Otherwise the user will get an error message from CS22.

- PARTIAL: Only the pairs of nodes that are directly connected to some degree are listed in the file (along with their connection information). CS22 assumes that all the pairs of nodes that are not listed are completely unconnected (infinite distance, zero probability or no link between them, depending on the types of connections selected). This allows not including in the file the information on those nodes that are too further apart or too weakly connected to have a noticeable influence in the connectivity analysis, reducing the effort of computing or estimating too many connections and the resultant size of the connection file. If n is the number of nodes (lines) in the node file (including the "nodes to add" if there are some), the connection file should have less than $n(n-1)/2$ lines (note that the case $i=j$ should not be included).

The following are examples of full connection files (for distances, probabilities and links as the type of connections) when there are six nodes in the landscape (node file):

Distance connection file

| | | |
|---|---|------------|
| 2 | 1 | 95.0775 |
| 3 | 1 | 10702.0926 |
| 3 | 2 | 10399.1063 |
| 4 | 1 | 3815.9717 |
| 4 | 2 | 4268.6740 |
| 4 | 3 | 14702.1201 |
| 5 | 1 | 4040.1056 |
| 5 | 2 | 4207.1797 |
| 5 | 3 | 13634.3167 |
| 5 | 4 | 2036.3631 |
| 6 | 1 | 9653.8086 |
| 6 | 2 | 9918.3022 |
| 6 | 3 | 19221.6779 |
| 6 | 4 | 5596.2132 |
| 6 | 5 | 5510.1152 |

Probability connection file

| | | |
|---|---|-----------|
| 2 | 1 | 0.9869059 |
| 3 | 1 | 0.2268140 |
| 3 | 2 | 0.2365437 |
| 4 | 1 | 0.5891904 |
| 4 | 2 | 0.5533504 |
| 4 | 3 | 0.1302699 |
| 5 | 1 | 0.5711648 |
| 5 | 2 | 0.5580878 |
| 5 | 3 | 0.1510541 |
| 5 | 4 | 0.7540475 |
| 6 | 1 | 0.2622907 |
| 6 | 2 | 0.2528476 |
| 6 | 3 | 0.0696209 |
| 6 | 4 | 0.4603354 |
| 6 | 5 | 0.4658628 |

Link connection file

| | | |
|---|---|---|
| 2 | 1 | 1 |
| 3 | 1 | 0 |
| 3 | 2 | 0 |
| 4 | 1 | 1 |
| 4 | 2 | 1 |
| 4 | 3 | 0 |
| 5 | 1 | 1 |
| 5 | 2 | 1 |
| 5 | 3 | 0 |
| 5 | 4 | 1 |
| 6 | 1 | 0 |
| 6 | 2 | 0 |
| 6 | 3 | 0 |
| 6 | 4 | 0 |
| 6 | 5 | 0 |

The following are examples of PARTIAL connection files for a landscape with six nodes:

Distance connection file

| | | |
|---|---|------------|
| 2 | 1 | 95.0775 |
| 3 | 2 | 10399.1063 |
| 4 | 1 | 3815.9717 |
| 4 | 2 | 4268.6740 |
| 5 | 1 | 4040.1056 |
| 5 | 4 | 2036.3631 |
| 6 | 1 | 9653.8086 |
| 6 | 5 | 5510.1152 |

Probability connection file

| | | |
|---|---|-----------|
| 2 | 1 | 0.9869059 |
| 4 | 1 | 0.5891904 |
| 4 | 2 | 0.5533504 |
| 4 | 3 | 0.1302699 |
| 5 | 1 | 0.5711648 |
| 5 | 2 | 0.5580878 |
| 6 | 4 | 0.4603354 |
| 6 | 5 | 0.4658628 |

Link connection file

| | | |
|---|---|---|
| 3 | 2 | 0 |
| 4 | 1 | 1 |
| 4 | 2 | 1 |
| 5 | 3 | 0 |
| 5 | 4 | 1 |
| 6 | 1 | 0 |
| 6 | 2 | 0 |
| 6 | 3 | 0 |

Note that the following two pairs of FULL (left) and PARTIAL (right) connection file are completely equivalent (they will provide the same results when analysed in CS22). Pairs of nodes with no probability or link value specified in the partial file (right) will be assumed to be completely unconnected (zero probability or no link, as with the full files in the left).

Full probability connection file

| | | |
|---|---|-----------|
| 2 | 1 | 0.9869059 |
| 3 | 1 | 0 |
| 3 | 2 | 0.5533504 |
| 4 | 1 | 0 |
| 4 | 2 | 0.5711648 |
| 4 | 3 | 0.5580878 |
| 5 | 1 | 0 |
| 5 | 2 | 0 |
| 5 | 3 | 0.2483888 |
| 5 | 4 | 0.8972342 |

Partial probability connection file

| | | |
|---|---|-----------|
| 2 | 1 | 0.9869059 |
| 3 | 2 | 0.5533504 |
| 4 | 2 | 0.5711648 |
| 4 | 3 | 0.5580878 |
| 5 | 3 | 0.2483888 |
| 5 | 4 | 0.8972342 |

Full link connection file

| | | |
|---|---|---|
| 2 | 1 | 1 |
| 3 | 1 | 0 |
| 3 | 2 | 0 |
| 4 | 1 | 1 |
| 4 | 2 | 1 |
| 4 | 3 | 0 |
| 5 | 1 | 1 |
| 5 | 2 | 1 |
| 5 | 3 | 0 |
| 5 | 4 | 1 |

Partial link connection file

| | | |
|---|---|---|
| 2 | 1 | 1 |
| 4 | 1 | 1 |
| 4 | 2 | 1 |
| 5 | 1 | 1 |
| 5 | 2 | 1 |
| 5 | 4 | 1 |

In all the cases the DECIMAL SEPARATORS need to be POINTS, not commas or other character, as in the examples above. Additionally, the numbers should NOT include any THOUSAND SEPARATOR in any of the columns. The files should always be formatted in this way, no matter which are the regional configuration settings on your computer, which might need to be set accordingly. In all the cases, the connection file should end with a blank line.

When the type of connections is "distances", be sure that you use the same units for the distances in the connection file and for the distance values specified by the user for the binary (section 5.2.2.1) and probabilistic indices (section 5.2.3.1).

The IDs of nodes i and j do not need to be arranged consecutively as in the examples above. Switching i by j in some or all of the columns, or rearranging the lines in a different way from the examples above will not affect at all the results provided by CS22. However, omitted lines or errors in the file are easier to detect if it is arranged as indicated above.

Note that all the node IDs included in the connection file should be included as well in the node file. Otherwise you will get an error message when CS22 reads the connection file (see section 9.2).

In all the cases, the information on the connections should be included also for the "NODES TO ADD" if some of these are included in the analysis. They should be treated exactly as the rest of the nodes regarding the connection file. For instance, the information on the connections between an existing and non-existing node ("node to add") should also be included in the same way than for the rest of the nodes. The nodes to add, when present, can be listed in any place in the file, and do not need to be listed together at any particular position in the connection file.

It is the full responsibility of the user to provide an adequately formatted connection file (as described above) in order to obtain valid results through this software. Although CS22 notifies the user of some common format errors (see section 9.2), several others may not be detected or reported.

In the "Sample data" folder (located within the folder where you installed CS22, which by default is "Conefor Sensinode 22") you can find some simple examples of full connection files (*.txt) (see section 8.3).

As described in section 8.2, you may use the ArcView 3.x or ArcGIS 9.x extension (included with this distribution) to directly calculate and prepare the distance file in the required format for CS22. This extension computes Euclidean edge-to-edge distances among all the given polygon features (patches) of a shapefile (*.shp) dataset. Other software tools may also be available elsewhere to compute Euclidean or effective (minimum cost) distances between nodes (see section 8.2).

6. CONNECTIVITY INDICES

CS22 includes NINE CONNECTIVITY INDICES, six based on the binary connection model and three based on the probabilistic one. All of them are landscape-level indices that measure functional connectivity in different ways. For all the indices higher values indicate improved connectivity, with the exception of NC, which behaves in the opposite way. Further details on these indices can be found in Pascual-Hortal and Saura (2006) and in Saura and Pascual-Hortal (2007) and are not included in this manual. Users of CS22 are encouraged to read in detail these two papers and other related references.

Users may select one or more of these indices for analysis. CS22 allows simultaneous processing of both types of indices (binary and probabilistic) from the same set of input data. However, it is responsibility of the user to provide coherent settings in the boxes for the binary and probabilistic indices (see sections 5.2.2.1 and 5.2.3.1), so that the results for the different indices are based on compatible information.

6.1. BINARY INDICES

Here we just provide a brief description of the binary indices included in CS22. More details on their properties, background, interpretation and a comparative analysis can be found in Pascual-Hortal and Saura (2006).

6.1.1. NL - Number of links

As a landscape is more connected, it will present a larger total number of links (connections between habitat nodes in the landscape).

6.1.2. NC - Number of components

A component (or connected region) is a set of nodes in which a path exists between every pair of nodes. Thus, there is no functional relation between nodes belonging to different components. An isolated node (patch) makes up a component itself. As a landscape gets more connected, it will present fewer components.

6.1.3. H - Harary index

$$H = \frac{1}{2} \sum_{i=1}^n \sum_{j=1, i \neq j}^n \frac{1}{nl_{ij}} \quad (2)$$

where n is the total number of nodes in the landscape, and nl_{ij} is the number of links in the shortest path between patches i and j (shortest path in terms of topological distance, i.e. number of links). For patches that are not connected (belong to different components) $nl_{ij} = \infty$. Note that the case $i=j$ is not included in the sum for H. As a landscape is more connected, it will present a higher H value.

6.1.4. CCP - Class coincidence probability

CCP ranges from 0 to 1, increases with improved connectivity and is computed as:

$$CCP = \sum_{i=1}^{NC} \left(\frac{c_i}{A_C} \right)^2 \quad (3)$$

where NC is the number of components in the landscape, c_i is the total component attribute (sum of the attributes of all the nodes belonging to that component), and A_C is total habitat attribute (sum of the attributes of all habitat nodes in the landscape). Note that the node attribute values are those in the second column of the node file (section 4.2). When the node attribute is area (habitat patch area), CCP is defined as the

probability that two randomly chosen points within the habitat (class) belong to the same component; alternatively, it can be defined as the probability that two animals randomly placed within the habitat are able to find each other given the set of habitat patches and links.

6.1.5. LCP - Landscape coincidence probability

LCP ranges from 0 to 1, increases with improved connectivity and is computed as:

$$LCP = \sum_{i=1}^{NC} \left(\frac{c_i}{A_L} \right)^2 \quad (4)$$

where NC is the number of components in the landscape, c_i is the total component attribute (sum of the attributes of all the nodes belonging to that component), and A_L is the maximum landscape attribute (see section 6.3). If the node attribute is area (habitat patch area) then A_L is the total landscape area (area of the analysed region, comprising both habitat and non-habitat patches), and $LCP=1$ when all the landscape is occupied by habitat. In this case, LCP is defined in a similar way to CCP as the probability that two randomly points (or animals) located within the landscape (i.e., points can lie either in habitat or non habitat areas) belong to the same habitat component (for which both points should lie in habitat patches and also a path through links connecting both patches should exist). CCP and LCP are generalizations of the degree of coherence (Jaeger 2000) by considering components instead of individual patches.

6.1.6. IIC - Integral index of connectivity

➔ **Recommended binary index**

This index is recommended as the best binary index for the type of connectivity analysis performed by CS22 (Pascual-Hortal and Saura 2006), presenting several improved characteristics compared to other available binary indices (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007).

IIC ranges from 0 to 1 and increases with improved connectivity. It is given by:

$$IIC = \frac{\sum_{i=1}^n \sum_{j=1}^n \frac{a_i \cdot a_j}{1 + nl_{ij}}}{A_L^2} \quad (5)$$

where n is the total number of nodes in the landscape, a_i and a_j are the attributes of nodes i and j , nl_{ij} is the number of links in the shortest path (topological distance) between patches i and j , and A_L is the maximum landscape attribute (see section 6.3). For nodes that are not connected (belong to different components) the numerator in the equation for IIC equals zero ($nl_{ij}=\infty$). When $i=j$ then $nl_{ij}=0$ (no links are needed to reach a certain node from itself); this relates to the habitat availability concept that applies for IIC, in which a patch itself is considered as a space where connectivity exists. If the node attribute is area (habitat patch area) then A_L is the total landscape area (area of the analysed region, comprising both habitat and non-habitat patches) and $IIC=1$ when all the landscape is occupied by habitat.

IIC requires, like H, the calculation of shortest paths between every pair of nodes, which makes it more computationally-demanding than the other binary indices. See section 9.1 for some guidelines on processing times and the total number of nodes that can be processed in a single run.

6.2. PROBABILISTIC INDICES

Here we describe the three probabilistic indices included in CS22. More details on the properties, interpretation and a comparative analysis of these three and other connectivity indices can be found in Saura and Pascual-Hortal (2007).

6.2.1. F - Flux

$$F = \sum_{i=1}^n \sum_{j=1, i \neq j}^n p_{ij} \quad (6)$$

where n is the total number of nodes in the landscape and p_{ij} is the probability of direct dispersal between nodes i and j (as directly obtained or derived from the connection file provided by the user, see section 5.2.3). Note that the case $i=j$ is not included in the sum for the F index.

6.2.2. AWF - Area-weighted flux

$$AWF = \sum_{i=1}^n \sum_{j=1, i \neq j}^n p_{ij} \cdot a_i \cdot a_j \quad (7)$$

where n is the total number of nodes in the landscape, p_{ij} is the probability of direct dispersal between nodes i and j (as directly obtained or derived from the connection file provided by the user, see section 5.2.3), and a_i and a_j are the attributes of the nodes i and j (as obtained from the second column in the node file, see section 4.2). Note that the case $i=j$ is not included in the sum for the AWF index. F and AWF were described by Bunn et al. (2000) and Urban and Keitt (2001), and are as well equivalent to a landscape-level version of the incidence function model (IFM) measures (e.g. Hanski 1994, Moilanen and Nieminen 2002, Verheyen et al. 2004) obtained by summing the IFM values for all habitat nodes (patches) in the landscape.

6.2.3. PC - Probability of connectivity

.....> **Recommended probabilistic index**

6.2.3.1. Description of the PC index

This index is recommended as the best index for the type of connectivity analysis performed by CS22, presenting several relevant improved characteristics compared to other existing indices (Saura and Pascual-Hortal 2007). It also considers a richer connection model than IIC and it is not affected by the presence of adjacent habitat patches or cells in the analysed datasets (Saura and Pascual-Hortal 2007).

PC ranges from 0 to 1 and increases with improved connectivity. It is given by:

$$PC = \frac{\sum_{i=1}^n \sum_{j=1}^n a_i \cdot a_j \cdot p_{ij}^*}{A_L^2} \quad (8)$$

where n is the total number of habitat nodes in the landscape, a_i and a_j are the attributes of nodes i and j , A_L is the maximum landscape attribute (see section 6.3), and p_{ij}^* is the maximum product probability of all paths between patches i and j . A path is made up of a set of steps in which no node is visited more than once, where a step is a direct movement of a disperser between two nodes (without passing by any other intermediate nodes). The product probability of a path is the product of all the p_{ij} belonging to each step in that path. p_{ij}^* is defined as the maximum product probability of all possible paths between patches i and j (including single-step paths). If nodes i and j are close enough,

the maximum probability path will simply be the step (direct movement) between nodes i and j ($p_{ij}^* = p_{ij}$). If nodes i and j are more distant, the "best" (maximum probability) path will probably comprise several steps through intermediate stepping stone nodes yielding $p_{ij}^* > p_{ij}$. Figure 5 illustrates the calculation of p_{ij}^* in a simple example, where the maximum probability path between nodes A and B comprises two steps (from node A to C, and from node C to B) and presents $p_{ij}^* = 0.25$, a higher probability than a direct movement between nodes A and B ($p_{ij} = 0.2$).

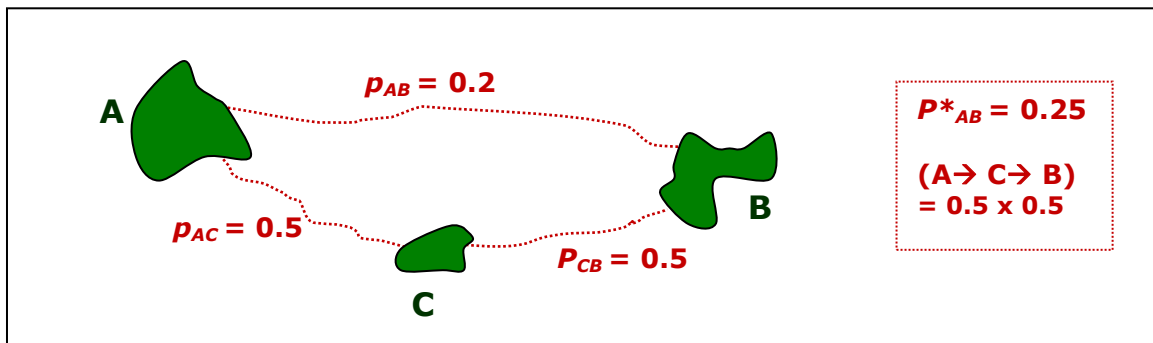


Figure 5. Example of the calculation of the maximum product probability between nodes A and B (p_{AB}^*) in a simple landscape, which results higher than the probability of direct dispersal between these two nodes (p_{AB}).

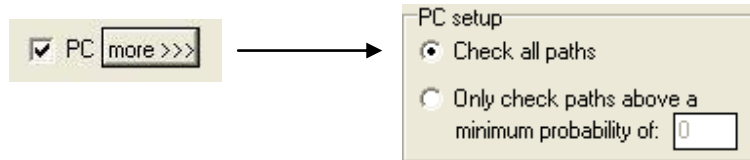
When two nodes are completely isolated from each other, either by being too distant or by the existence of a land cover impeding the movement between both nodes (e.g. a road), then $p_{ij}^* = 0$. When $i = j$ then $p_{ij}^* = 1$ (it is sure that a node can be reached from itself with the highest probability); this relates to the habitat availability concept that applies for PC, in which a habitat patch itself is considered as a space where connectivity exists. In the particular case that the direct dispersal probabilities p_{ij} are given by a negative exponential function of the internode distance (this is the function used by CS22 to compute p_{ij} when "Distances" is the type of connections, see section 5.2.3.1), the maximum probability path will be the same as the shortest path in terms of distance units.

If the node attribute is area (habitat patch area) then A_L is the total landscape area (area of the analysed region, comprising both habitat and non-habitat patches) and $PC = 1$ when all the landscape is occupied by habitat. In this case, PC can be defined as the probability that two animals randomly placed within the landscape fall into habitat areas that are reachable from each other (interconnected) given the set of habitat patches and the connections (p_{ij}) among them.

The computation of the PC index requires the application of graph-based algorithms for determining the maximum probability paths. This can be quite computationally intensive for landscapes with a large number of nodes. Therefore, PC requires more processing time and is limited to a lower maximum number of nodes than the rest of the indices described above (see section 9.1 for some guidelines in this respect).

6.2.3.2. Advanced settings for the PC index

CS22 includes some advanced settings for the PC index (PC "more" button in the interface for CS22), which are optional and should only be modified after running the software in the default mode and understanding how these settings modify the computation of PC and the potential outcome of the connectivity analysis, as described below. These advanced settings may allow reducing the total processing times for PC in some cases.



By clicking the PC “more” button you will have access to a small “PC setup” window, where you should select one of the following two options:

- **CHECK ALL PATHS (DEFAULT).** This is the default option and the one recommended for most applications. The landscape will be modelled as a complete graph with a certain probability of direct dispersal (p_{ij}) between every pair of nodes. The value of p_{ij} for all pairs of patches will be stored and processed in the graph structure even if it is very low or eventually equal to zero. In this way the information on the p_{ij} is very easy to access by the software, but the trade-off is that a large amount of information needs to be stored in the graph structure. The best paths (in terms of maximum product probability) between every two nodes will be calculated with this option, even if these two nodes have very low maximum product probabilities (p^*_{ij}) that eventually may contribute little to habitat connectivity. Therefore, the results for PC (node importance values and overall index values for this index) will be the most accurate (fully accurate) with this option. However, in some cases the processing may be slower compared to the option below, although this is not always the case and depends on the spatial arrangement of the nodes in the landscape and on the dispersal distances specified for the analysis.

- **ONLY CHECK PATHS ABOVE A MINIMUM PROBABILITY.** By choosing this option, the information on the connections will be stored and accessed internally by CS22 in a different way, probably providing faster results in very big landscapes and/or sparsely connected graphs (i.e. landscapes where most of the pairs of nodes are weakly connected or not directly connected at all, which may be the case when the analysis spans a large geographical range). In this option you should specify a minimum probability value (“min prob”). This value will be used to discard from the analysis the direct connections with a probability (p_{ij}) equal or lower than that minimum (nodes with a p_{ij} equal to or below the minimum will be treated as not directly connected at all). This “min prob” value will be also used to discard the paths with a product probability (p^*_{ij}) equal or below that specified value; in the case that the maximum product probability path between two nodes is equal or lower than that minimum, these nodes will be treated as completely unconnected. Moderate values of these minimum values may have a very minor effect on the results of the analysis, while they may considerably reduce the processing time for some landscapes. If the minimum probability value is set to 0, the results of the analysis will be identical to the previous option (check all paths), but the processing times may be still be reduced if many nodes have $p_{ij}=0$ in the initial connection file. Obviously, specifying too high values of the minimum probability will provide less accurate results (e.g. node importance values with significant differences from those resulting from the default option); this relies on the criteria and responsibility of the user. In summary, in this option the internal access to the information on the internode connections is a bit more difficult than in the former (check all paths), but a much smaller amount of information on these connections may have to be stored internally by CS22, which depending on the landscape may imply faster or slower processing. Users may try first with the default option and if the processing is too slow try then with this option (and a reasonable minimum probability value) as an alternative.

When the PC index is processed under the second option (only check paths that are over a minimum probability) the number of internode direct connections that are stored and used internally by CS22 is displayed both in the execution events (see section 7.2.5) and in the results of the overall index values (see section 7.2.1). This number will be smaller

or equal than $n(n-1)/2$, where n is the number of nodes in the landscape. In the default option (check all paths) $n(n-1)/2$ is the number of direct interpatch connections that are stored internally by CS22.

6.3. MAXIMUM LANDSCAPE ATTRIBUTE

The maximum landscape attribute (A_L) applies for the LCP, IIC and PC indices (see equations 4, 5 and 8 above). A_L is the attribute value that would correspond to a patch covering all the landscape with the best possible habitat, case in which LCP, IIC and PC would be equal to 1. If the node attribute (as obtained from the values of the second column of the node file, see section 4.2) is habitat (patch) area, A_L equals total landscape area (area of the analysed region, comprising both habitat and non-habitat patches). If the node attribute is a quality-weighted area and the quality factor ranges from 0 to 100, A_L will be equal to 100 multiplied by total landscape area, etc. Note that A_L is therefore usually greater than the sum of the habitat node attributes of a landscape (A_C , see equation 3 for the CCP index above), since A_L comprises as well all the areas in the landscape that are not currently covered by habitat.

The value of the maximum landscape attribute (A_L) can be specified by the user in a specific box in the main screen of CS22. This box is only active when at least one of these three indices (LCP, IIC, PC) is selected for analysis.



However, even when selecting LCP, IIC or PC, entering the value of A_L is optional and not really necessary for an analysis oriented to the prioritization of habitat patches for conservation. CS22 will not require you to enter a value for A_L even when analysing connectivity through LCP, IIC or PC. In fact, the value of A_L does not affect at all the importance of the nodes according to any of these indices (LCP, IIC, PC), since A_L remains constant after any node removal or addition, as noted in section 7.1. Only if the user wants to obtain the overall values of these three indices (and not only the node importance values derived from them) the value of A_L should be specified (see section 7.2.1).

If the A_L value entered by the user is smaller than the sum of the attributes of all habitat nodes in the landscape (A_C , total habitat attribute, see Eq. 3), CS22 interprets that the value of A_L is erroneous (it is not possible that the maximum landscape attribute is smaller than the current sum of node attributes), and such A_L value is discarded and not used for subsequent calculations. The result will be the same as when no A_L value is entered by the user. CS22 will notify the user of this circumstance (see section 9.2) but proceed with the analysis without using any A_L value.

When no A_L value is entered (or when the A_L value is erroneous by being smaller than A_C), CS22 will just compute the numerators of the LCP, IIC, and PC indices (shown as LCPnum, IICnum and PCnum in the results for the overall index values), as follows from the Eqs. 4, 5 and 8:

$$LCPnum = \sum_{i=1}^{NC} c_i^2 \tag{9}$$

$$IICnum = \sum_{i=1}^n \sum_{j=1}^n \frac{a_i \cdot a_j}{1 + nl_{ij}} \tag{10}$$

$$PCnum = \sum_{i=1}^n \sum_{j=1}^n a_i \cdot a_j \cdot p_{ij}^* \quad (11)$$

When a correct A_L value is entered (bigger than A_c), CS22 will compute and present in the results for the overall indices both the numerators (LCPnum, IICnum, PCnum) and the full final values of LCP, IIC and PC (Eqs. 4, 5 and 8), as long as these indices have been selected for the analysis.

7. RESULTS

All the results of the analyses performed by CS22 can be saved in standard ASCII text files (*.txt). Additionally, some of these results (node importances, components) can be saved in DBF (dBase format) files (*.dbf). Most of the results (node importances, overall index values, components, execution events) can be viewed within CS22 without need of previously saving them to any file.

The text result files have a standard format and may be opened with almost any database, spreadsheet, text processor, GIS or statistical analysis software. The format of the result text files is the same as for the node file and the connection file (see sections 4.2 and 5.3). That is, columns are separated by tabulations or spaces, the files end with a blank line, points are the decimal separators and no thousand-separator character is included in the numbers (no matter which are the regional configuration settings in your computer, the files and numbers are always saved in this way). The DBF files are also a standard format that is incorporated in many other programmes; particularly, it may allow directly joining within a GIS layer the results for the connectivity importances and the components for each node.



The different types of results are available from the drop-down "Results" menu after ending (or temporarily pausing) the execution process. Some result files may be or not available depending on the specific settings and on the indices selected for analysis. Since the individual node importance values is the most relevant and useful result provided by CS22, we describe it separately in the next section.

7.1. NODE IMPORTANCES

7.1.1. COMPUTATION OF NODE IMPORTANCES

The importance of an existing node for maintaining landscape connectivity (dI) according to a certain index (I) is calculated in CS22 as a percentage as:

$$dI(\%) = 100 \cdot \frac{I - I_{remove}}{I} \quad (12)$$

where I is the overall index value when all the initially existing nodes are present in the landscape and I_{remove} is the overall index value after the removal of that single node from the landscape (e.g. after a certain habitat patch loss).

When the option "nodes to add" is selected in CS22 (when the analysis includes some potential new habitat sites to be added in the landscape), the importance of each potential new node (dI) for improving landscape connectivity according to a certain index (I) is calculated as:

$$dI(\%) = 100 \cdot \frac{I_{add} - I}{I} \quad (13)$$

where I is the overall index value when all the initially existing nodes are present in the landscape (this includes only the nodes that actually exist in the landscape, but not any of the nodes to add) and I_{add} is the overall index value after the addition of that new node to the landscape (e.g. after a certain habitat patch creation or restoration).

Note that A_L (maximum landscape attribute, section 6.3) remains constant after the removal or addition of new nodes in the landscape. Therefore, the A_L value does not affect dI for any of the indices.

7.1.2. INTERPRETATION OF NODE IMPORTANCES

Higher dI values indicate more important nodes for all the indices but NC. The dI values should be interpreted for the different indices as follows:

- NL, LCP, H, IIC, F, AWF, PC. Only positive or zero dI values are possible. The higher dI the more important that node is for landscape connectivity, either for maintaining it (nodes already existing in the landscape) or for improving it (nodes to add).

- NC. Both negative and positive dI values can be obtained, since some types of habitat loss are interpreted by this index as a connectivity improvement (Pascual-Hortal and Saura 2006). The lower dI (more negative) the more important that node is for landscape connectivity according to this index, either for maintaining it (nodes already existing in the landscape) or for improving it (nodes to add).

- CCP. Both negative and positive dI values can be obtained, since some types of habitat loss are interpreted by this index as a connectivity improvement (Pascual-Hortal and Saura 2006). The higher dI the more important that node is for landscape connectivity according to this index, either for maintaining it (nodes already existing in the landscape) or for improving it (nodes to add).

The importance values are very useful for decision-making in landscape conservation planning, since they allow identifying the most critical nodes (e.g. habitat patches or cells) for the maintenance or improvement of landscape connectivity, in which conservation or restoration efforts should concentrate.

The results of these analyses may largely depend on the selected connectivity index. It is therefore of outstanding importance to use an adequate landscape-level connectivity index for these purposes. IIC (binary model) and PC (probabilistic model) are the connectivity indices providing the best performance and prioritization abilities in this respect (Pascual-Hortal and Saura 2006, Saura and Pascual-Hortal 2007), and are therefore recommended for these purposes.

When analysing the node importances for habitat availability indices (LCP, IIC, PC), it should be remarked that these indices integrate in a single measure the habitat area made available by the connections between nodes (interpatch connectivity) and the connected area (or other node attributes) existing within the nodes themselves (intrapatch connectivity). Therefore, the importance of a particular node (dI , Eq. 13) for a habitat availability index may be the joint result of (1) its intrinsic habitat characteristics (attributes such as area, quality, etc.) and (2) its topological position within the landscape network (e.g. a node acting as a stepping stone for dispersal between other habitat areas). Typically, the nodes with the highest dI (Eq. 13) will be those important both for interpatch (2) and intrapatch connectivity (1). However, the relative importance of each of them may largely vary for each particular node and depend as well on the inputs of the analysis (spatial arrangement of habitat nodes in the landscape, dispersal distances, etc.).

Indeed, the relative contribution to the final node importance of (1) the node attribute (e.g. patch area) and (2) the topological position of the node in the landscape network

depends on the dispersal distance and the landscape spatial pattern (Keitt et al. 1997, Saura and Pascual-Hortal 2007). When the dispersal distance is very low (eventually equal to zero) it will not be possible to disperse to any other node in the landscape, and therefore the only relevant issue for the survival of an organism is the characteristics of the habitat area in which it is currently located (node attribute). When the dispersal abilities of the analysed species are so large that the landscape is perceived as fully connected, the node attribute again gains more relevance in the determination of its importance; in this case, it would be possible to disperse directly among every pair of nodes (no key stepping-stone patches would exist), and the importance of a node would therefore be determined just by its intrinsic characteristics (attribute). At intermediate dispersal distances (relative to the landscape spatial pattern) is where the internode connectivity and topological position (2) may play a major role in determining the most important nodes (e.g key stepping stone patches), eventually contributing much more to the final node importance than the intrinsic node characteristics as quantified in the node attributes (Saura and Pascual-Hortal 2007).

Finally, note that the sum of the importances (dI) for all the existing nodes in the landscape will be in general higher than 100% (even if no "nodes to add" are included in the analysis). This sum will be larger as more critical habitat areas for connectivity exist in the landscape. For example, if quite many key-stepping stone patches were present (e.g. along a corridor) so that the loss of any of them would disconnect the remaining habitat in two disconnected halves, each of those patches would have a large impact on connectivity when lost, and eventually their dI values would, all together, sum up a percentage much larger than 100% (Saura and Pascual-Hortal 2007).

7.1.3. THE NODE IMPORTANCE FILE

The node importance file is a standard ASCII text file containing $k+2$ columns and $n+1$ lines, where k is the number of different connectivity indices selected for the analysis, and n is the number of nodes in the analysis (the extra line is for the headings of the columns). Therefore, the number of columns will range between three (when only one index has been selected for analysis) and eleven (when all the nine indices have been selected for analysis). The two first columns are always included in this file (no matter which connectivity indices have been selected), and after these two columns one more column is added for each connectivity index selected for the analysis, as follows:

- First column: Node ID (this is the same ID as in the node file and in the rest of the files in CS22).
- Second column: dA for each node, where dA is the percentage of total habitat attribute that corresponds to the attribute in that node. For example, if the node attribute is patch area, dA represents the percentage of total habitat area corresponding to that node. dA values are computed both for the existing nodes and for the nodes to add as follows:

$$dA(\%) = 100 \cdot \frac{a_i}{A_c} \quad (14)$$

where a_i is the attribute of the node for which dA is computed, and A_c is the total habitat attribute (sum of the attributes of all the habitat nodes currently existing in the landscape, not including any potential new nodes to add). dA values can be very easily computed externally to CS22, and in fact no connectivity analysis is needed for this. However, this is a basic value that is always included in the node importance file for reference and comparison with the dI for the different connectivity indices. Obviously, the sum of dA for all the existing nodes (not including the nodes to add, if there are some) equals 100% (while this does not occur in general for the connectivity indices).

- Third and subsequent columns: dI values for each node (importance of that node for maintaining or improving connectivity according to a certain index). The headings of these columns for each of the nine different connectivity indices included in CS22 are dNL , dNC , dH , $dCCP$, $dLCP$, $dIIC$, dF , $dAWF$, dPC .

The dI values for the currently existing nodes (importance for maintaining connectivity) and for the nodes to add (importance for improving connectivity, if there are some) are provided all together in the same file. The node IDs should allow the user identifying which of those nodes do already exist in the landscape and which correspond to new habitat sites that may be added in the landscape (in fact this information was already provided by the user in the third column of the node file, see section 4.2).

Note that the dI values for LCP, IIC or PC ($dLCP$, $dIIC$, dPC) are not affected by the value of the maximum landscape attribute (A_L) that may have been specified by the user (see section 6.3).

7.2. OTHER RESULTS

7.2.1. OVERALL INDEX VALUES

A single index value (I) characterises the degree of habitat connectivity for the whole landscape (mosaic of habitat patches). This provides an idea of the current status of the landscape in this respect, but is simply descriptive and not particularly relevant for specific landscape planning purposes.

This result file simply contains a few lines with the overall value for each index selected for the analysis, computed as described in section 6 and in Eqs. 2 to 8.

For the indices that include the maximum landscape attribute (A_L) in their expressions (LCP, IIC, PC, Eqs. 4, 5 and 8) this result file will always provide the value of their numerators (LCPnum, IICnum, PCnum, Eqs. 9 to 11). If a valid A_L value has been entered by the user ($A_L > A_C$, see section 6.3), the file will also contain the final index values (LCP, IIC, PC, Eqs. 2 to 8) in addition to the value of their numerators.

Note that the overall index values that are calculated in CS22 only take into account the nodes that currently exist in the landscape, and not any candidate nodes to be added in the landscapes. That is, if there are any nodes to add in the analysis, these will not be considered for the computation of the overall index values.

7.2.2. COMPONENTS

A component (also named sometimes "cluster" or "connected region") is a set of nodes for which a path exists between every pair of nodes (an isolated patch makes up a component itself). There is no functional connectivity (no path) between patches belonging to different components.

The component file has two columns and provides for each node (node ID in the first column) the ID of the component (connected region) to which that node belongs (second column). The ID of the component is just an identification number that CS22 assigns to new components as they are found in the processing, and has no other particular meaning apart from labelling the different components. A component will typically comprise several nodes (with fewer and larger components as the landscape gets more connected), although some single-node components may as well exist.

This result (component file) is automatically calculated and only available when at least one of the following binary indices has been selected for analysis: NC, CCP or LCP.

The component file always refers to the components found in the current initial landscape, without considering any node removal or addition. That is, the components refer to the landscape mosaic that comprises all the nodes currently existing in the landscape, and do not take into account any nodes to add if some of these have been included in the analysis. Therefore, the node IDs for the nodes to add (if there are some) are not reported in the component file and do not influence the identification of components for the existing set of nodes.

7.2.3. LINKS

This is a three-column file containing the ID of node i (first column), the ID of node j (second column) and a binary number indicating if a link exists (1) or not (0) between nodes i and j (third column). Note that CS22 considers symmetric connections, which implies that the connections (links) are the same from patch i to j than from patch j to i . Therefore, each pair of nodes is only listed once in this link file.

This result is automatically calculated by the program and available only when at least one binary index is selected for analysis.

The link file always refers only to the initial set of nodes existing in the landscape (without any node removal or addition), and does not take into account any nodes to add (if some of these have been included in the analysis). Therefore, the node IDs for the nodes to add (if there are some) are not included and do not influence the links between the existing nodes that are reported in this file (although they are of course considered for the node importance file, as described in section 7.1).

7.2.4. PROBABILITIES

7.2.4.1. Direct probabilities

This is a three-column file containing the ID of node i (first column), the ID of node j (second column) and the probability of direct dispersal (p_{ij}) between nodes i and j (third column). Note that CS22 considers symmetric connections, which implies that the direct dispersal probability is the same from patch i to j than from patch j to i . Therefore, each pair of nodes is only listed once in this file.

This result is automatically calculated by the program and available only if at least one probabilistic index has been selected for the analysis.

The direct probability file always refers only to the initial set of nodes currently existing in the landscape (without any node removal or addition), and does not take into account any nodes to add (if some of these have been included in the analysis). Therefore, the node IDs for the nodes to add (if there are some) are not included and do not influence the direct dispersal probabilities between the existing nodes that are reported in this file (although they are of course considered for the node importance file, as described in section 7.1).

7.2.4.2. Maximum product probabilities

This is a three-column file containing the ID of node i (first column), the ID of node j (second column) and the maximum product probability (p^*_{ij} , see section 6.2.3.1 and Eq. 8 for the PC index) between nodes i and j (third column). Note that CS22 considers symmetric connections, which implies that the maximum product probability is the same from patch i to j than from patch j to i . Therefore, each pair of nodes is only listed once in this file.

This result is automatically calculated and available only when the PC index has been selected for analysis.

The maximum product probability file always refers only to the initial set of nodes currently existing in the landscape (without any node removal or addition), and does not take into account any nodes to add (if some of these have been included in the analysis). Therefore, the node IDs for the nodes to add (if there are some) are not included and do not influence the maximum product probabilities between the existing nodes that are reported in this file (although they are of course considered for the node importance file, as described in section 7.1).

7.2.5. EXECUTION EVENTS

The execution events show the starting and ending time of the processing, the processing time for each selected index, the total number of nodes to process, the sum of node attributes (e.g. habitat area), the distance or probability threshold (binary indices) and/or the distance corresponding to a certain dispersal probability (probabilistic indices), the names and paths of the node and connection files, etc. This information is shown in the lower part of the main screen of CS22, but can also be saved in a text file through the result menu.

8. OTHER FEATURES

8.1. OTHER MENUS

Apart from the results menu, three other drop-down menus are available from the upper part of the CS22 main screen, which are the project menu, the execution menu, and the help menu, as described next.

◆ **PROJECT menu**

A project contains all the information regarding the settings (options, files, indices, values of threshold distances and probabilities, etc.) specified by the user for a particular connectivity analysis in CS22. Project files (txt format) allow the user to save and recover all the settings you entered for performing a particular analysis with CS22.

The following options are available from the project menu:

New: Opens a new blank project for a new analysis (this will produce the same result as exiting and restarting CS22), eventually discarding previous settings and results (you will be prompted to confirm this).

Open: Opens an already existing project file, where you previously saved the CS22 user settings for a particular analysis.

Save: Saves the settings of the current project.

Save as: Saves the current settings in a project file with the specified name and folder.

Exit: Exits the project and closes the application.

◆ **EXECUTION menu**

This menu controls the processing of the connectivity analysis. The following actions are available from this menu, and as icons in the lower part of the CS22 main screen:

Run: To start running the connectivity analysis for a project as currently specified in the user settings.

Pause: To temporarily pause the execution process. This may be useful either to free the CPU resources for other applications or to obtain the overall index values before waiting for the ending of the node importance processing.

Resume: To restart the execution process after a pause. The analysis will continue with the calculations at the point it was before the pause.

Cancel: To stop the execution process before the end (you will be asked for confirmation). Resuming calculations will no longer be possible, and the analysis should restart from the beginning.

The pause or cancel actions may be used to obtain the overall index values before waiting for the ending of the node importance processing. The node importance calculations get most of the total processing time, which can be quite large in some cases (see section 9.1). CS22 first calculates the overall index values (usually very fast), and later the node importances (usually slower). If you are only interested in the first result,

you can pause or cancel the CS22 processing when the node importances processing has started (this means that the overall index values have already been calculated) and view the overall index values at the results menu. It is easy to know when CS22 has already started the processing of the node importances, since from that moment CS22 shows "Processing importance of node with ID xxxx" in the status bar in the bottom left of the main screen.

The pause/resume actions will work fine in most of the situations, in the sense that the processing will continue without errors when resuming after previously pausing the calculations. However, resuming may not be possible in some cases when processing very large datasets and computationally intensive indices (e.g. PC) and if the process has been paused before the node importance processing has started.

◆ **HELP menu**

There are two options within this menu:

Conefor Sensinode User's Manual: this will automatically open this user's manual as a pdf file (CS22manual.pdf). If you have any problem in opening the manual from this menu, try to open it directly from the folder where the pdf file is located (this is the "help" folder within the parent directory where you installed CS22, typically C:\Program files\Conefor Sensinode 22\Help, or some other similar path).

Note that for specific queries or searches you may use the search tools within Acrobat Reader and the links to the different sections and subsections provided at the table of contents at the beginning of this manual.

About Conefor Sensinode 2.2: this provides a summary on the authors, conditions of use, contact e-mail, webpage and related references of CS22.

8.2. GIS EXTENSIONS FOR CALCULATING THE CONNECTION AND NODE FILES

CS22 includes custom-made extensions for ArcGis 9.x and ArcView 3.x that allow easily obtaining both the node and connection files directly in the format required by CS22 (the latter as a file of edge-to-edge Euclidean distances). Other software tools may be used to generate the connection file as an effective distance (minimum-cost) file, which may provide more detailed and realistic results for certain applications.

8.2.1. EXTENSION FOR ARCGIS 9.X

"Conefor inputs" is a custom-made GIS extension for ArcGis 9.x developed by Jeff Jenness Enterprises (www.jennessent.com) specifically for CS22. It allows easily obtaining from a theme in ArcGis (e.g. shapefile) both the node file and the connection file (as a distance file) directly in the format required by CS22. The distances in the connection file calculated by this extension are edge-to-edge Euclidean (straight-line) between all the features (e.g. patches) of a theme, with the option to analyse either all features or only those within a specified distance (see below). The text node and distance files provided by this extension are directly usable (with no change) in CS22.

8.2.1.1. Installation

Install the Conefor Inputs extension by double-clicking on the file "Conefor_inputs.exe", which will be located in the "Extensions GIS\ArcGis 9.x" folder (within the folder where you installed CS22, typically C:\Program files\Conefor Sensinode 22, or some other

similar path), and follow the instructions. The installation routine will register the Conefor_inputs.dll with all the required ArcMap components.

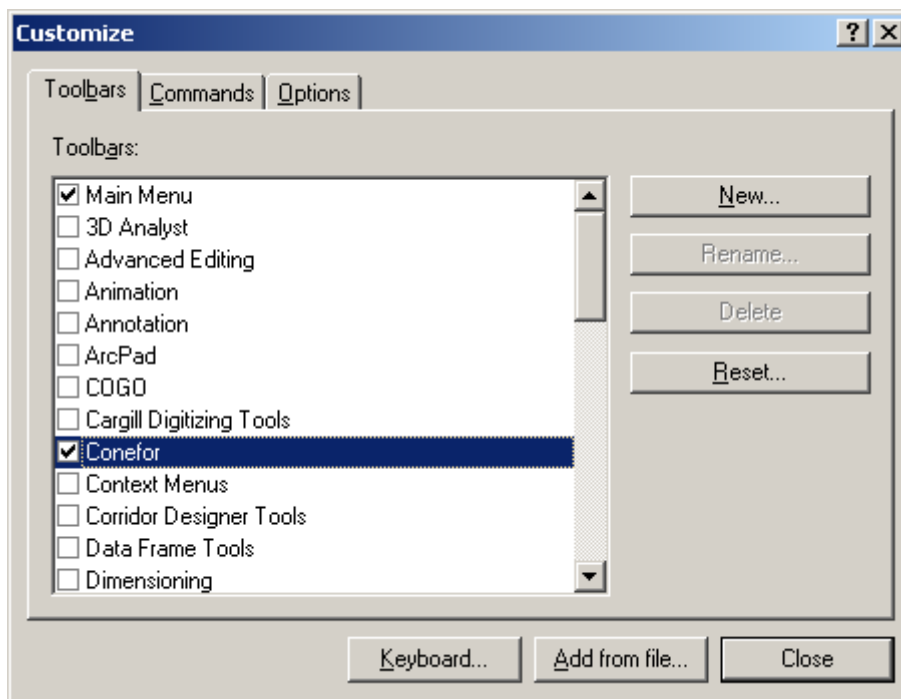
The default install location for the extension is a folder named "Conefor Inputs" inside the folder "Program Files", which will also include some additional files required by the extension and this manual. This folder should better be editable to the user (i.e. not restricted to administrators), since this extension occasionally writes to a file located in this folder. The extension should not crash if it is unable to write to this folder, but it will run better if it can.

This tool is installed as an Extension in ArcMap, although it is a type of extension that does not show up in the list of extensions available in the "Tools" menu.

After installing the extension, you should see the following new toolbar in your map (it may also be embedded in your standard ArcMap toolbars, rather than as a standalone object):

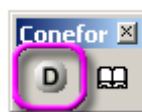


If you do not see this toolbar, then open your "Customize" tool (by either double-clicking on a blank part of the ArcMap toolbar, or by clicking the "Tools" menu, then "Customize"), click the "Toolbars" tab and check the box next to "Conefor".



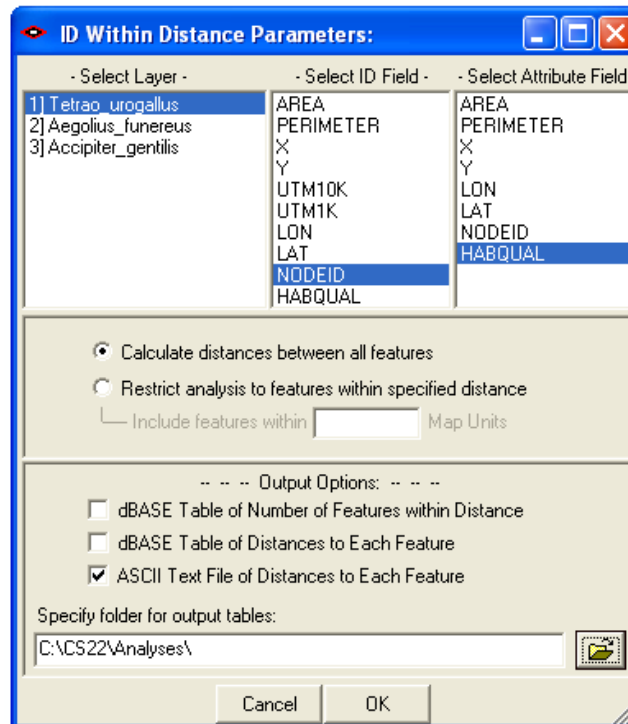
You should now see your Conefor toolbox in ArcMap.

8.2.2.2. Use of the extension (within ArcGis 9.x)



The extension tool consists of only two buttons: the one in the left ("D") opens the dialog for obtaining the node and distance files, while the one in the right just opens this help file.

After clicking the "D" button the "ID Within Distance Parameters" dialog will be shown:



In the "ID Within Distance Parameters" dialog proceed as follows:

1) Select the layer with the habitat data (where each feature will be considered as a different node in the resultant files).

The layer should be already open (added) in ArcMap/ArcGis. Note that this tool will always analyze all the features in the selected layer. It will not restrict analysis to only the selected features.

2) Select the field in the layer containing a unique ID for each node (feature in the layer).

This ID field should be already prepared in your layer before opening the dialog. In case your layer does not contain this ID field, you can easily obtain it in ArcGis 9.x by creating a new field (Add field... in the layer Attributes table), typically of Short Integer type, and by calculating the ID values in that field for each feature as 1 + the FID field in your layer. The FID field contains a unique ID for each feature in your layer, starting from 0 up to the number of features-1, and should typically be the first and leftmost field in your attribute table. FID is an 'internal' field that will not show up in the list of fields to be selected in the dialog, but the field you create from that FID as described above will appear in the dialog.

3) Select the field in the layer containing the attribute of the nodes (e.g. habitat area, habitat quality, quality-weighted area, etc.). This attribute field should be as well calculated and available in the layer before starting the dialog.

4) Select if you intend to calculate distances to either all other features in the layer (this will correspond to a full connection file, as described in section 5.3), or only those

features that are within a specified distance (this will correspond to a partial connection file, as described in section 5.3). In this latter case the distance threshold must be entered in the same units of the actual feature class projection ("Include features within xxxxx Map units").

The distance threshold option is particularly useful for reducing the processing time in the distance calculations through the GIS, when a significant proportion of the pairs of patches are too distant from each other to be directly connected. This may be the case for themes covering large geographical regions. If for example the maximum dispersal distance of the analysed species is 5 km, it makes no sense spending time in calculating distances between patches that are separated by dozens of kilometres. In this case, restrict the analysis to features within 5 km, and only the pairs of features within that distance will be included in the output. All the nodes (features) not present in the distance/connection file will be considered completely directly unconnected in CS22 (e.g. by being further apart than 5 km in the previous example), as described in section 5.3.

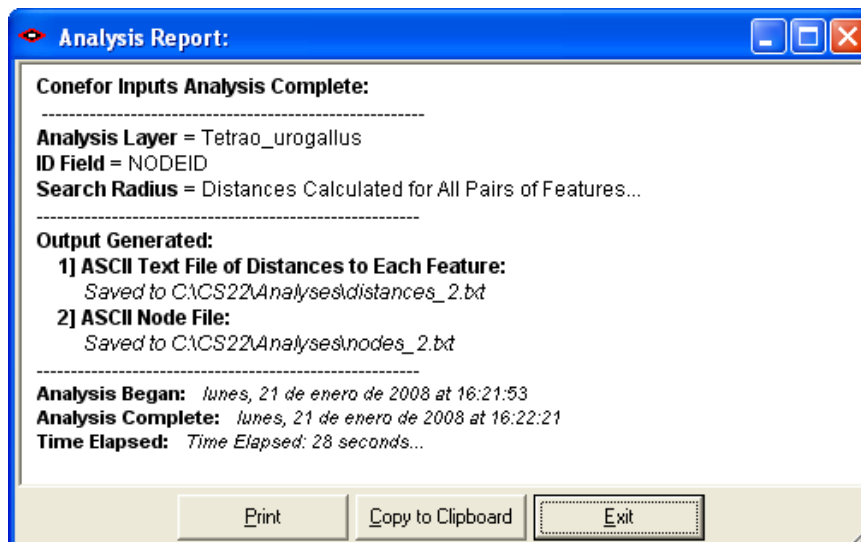
5) Select the output option "ASCII text file of distances to each feature", which is the output file that can be directly used in CS22.

The distance file will be named "Distances_X.txt", where "X" is a number that guarantees that this is a unique filename. The first such file created in this folder will be named "Distances.txt"; the second will be named "Distances_2.txt", etc. The file contains three columns of data, representing the ID of node i , the ID of node j , and the Euclidean distance between nodes i and j , as described for the connection file format in section 5.3. Note that the extension will only calculate the distance between any pair of features one time. If the tool has already calculated the distance from i to j , it will not recalculate the distance from j to i . Furthermore, the distance will only be entered into the distance file once, as required by CS22. If the file has an entry for the distance from i to j , the file will not also have an entry for the distance from j to i (as required by CS22).

The extension will also automatically generate the node file in the format required by CS22, containing two columns of data, the first representing the node ID and the second containing the attribute value for that node (see section 4.2). The file will be named "Nodes_X.txt", where "X" is a number that guarantees that this is a unique filename. The first such file created in this folder will be named "Nodes.txt"; the second will be named "Nodes_2.txt", etc.

6) Select the folder where the node and distance files will be saved. Note that you just have to select the folder (one click) and not to open the folder (the output file names are fixed as described above).

Upon completion, the tool will open all the tables and text files it produced (for you to view them), and it will also summarize the processing in a simple "Analysis report".



Note that the processing slows down with large shapefiles (that is, with increasing number of polygon features). Also, highly complex features can slow down the computation time when calculating distances between them. Polygons with lots of vertices will take longer to analyse than polygons with few vertices. Some previous processing to reduce the total number of vertices describing the polygon boundaries (especially removing redundant vertices if there are some) may benefit the speed of the distance computation through this extension.

8.2.2. EXTENSION FOR ARCVIEW 3.X

"ID Within Distance: Conefor" is a custom-made GIS extension for ArcView 3.x developed by Jeff Jenness Enterprises (www.jennessent.com). It allows measuring the edge-to-edge Euclidean (straight-line) distances between all polygon features (e.g. patches) of a theme, with the option to analyse either all features or only those within a specified distance (see below). The text distance files provided by this extension are directly usable (with no change) in CS22.


8.2.2.1. Installation

After installing CS22, the file "ID Within Distance Conefor.avx" will be located in the "Extensions GIS\ArcView 3.x" folder within the folder where you installed CS22 (typically C:\Program files\Conefor Sensinode 22, or some other similar path).

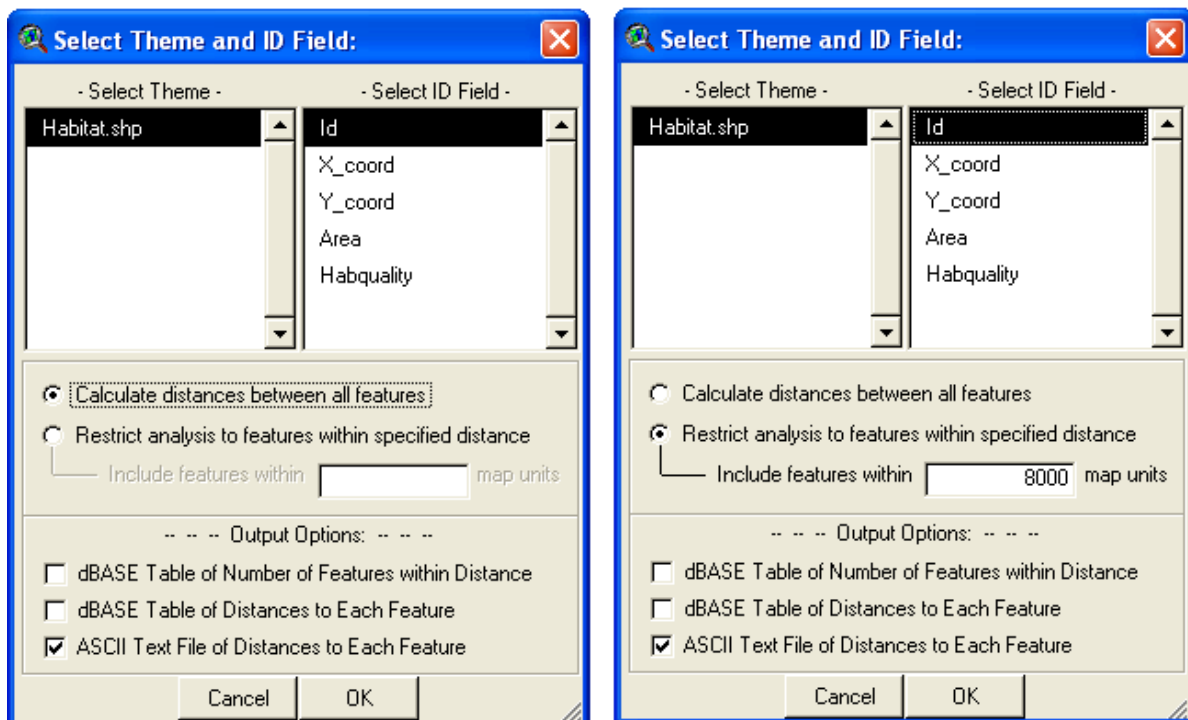
To make the extension available from ArcView 3.x, simply copy and paste the file "ID Within Distance Conefor.avx" in the folder for the ArcView 3.x extensions (Typically C:\esri\av_gis30\arcview\ext32, or some other similar path).

8.2.2.2. Use of the extension (within ArcView 3.x)

After installation, to use the extension you just need to run ArcView 3.x, open an ArcView Project and activate the extension from the "File->Extensions" dialog.

The button  will be added to the top toolbar of any open View. Click in this button and the dialog for the extension will be displayed.

Select the theme in which the analysis is to be performed and the ID field, which should be the field containing a different ID for each of the features (nodes, patches) to be analysed. These IDs should be the same than those in the second column of the nodes file (see section 4.2).



You have the option to analyse either all the features in a theme (upper figure, left) or only those features within a specified distance from each other (upper figure, right). This latter option is particularly useful for reducing the processing time in the distance calculations through the GIS, when a significant proportion of the pairs of patches are too distant from each other to be directly connected. This may be the case for themes covering large geographical regions. If for example the maximum dispersal distance of the analysed species is 5 km, it makes no sense spending time in calculating distances between patches that are separated by dozens of kilometres. In this case, restrict the analysis to features within 5 km, and only the pairs of features within that distance will be included in the output. These results can be directly provided as an input to CS22, with the "Partial" option and "Distances" type for the connection file (see sections 5.2 and 5.3), where all the nodes (features) not present in the distance/connection file will be considered completely directly unconnected (e.g. by being further apart than 5 km in the previous example).

In the same dialog you can also choose which output to produce (dBASE Table of Number of Features within Distance, dBASE Table of Distances to Each Feature, or ASCII Text File of Distances to Each Feature). The file that is directly usable in CS22 (without need for any further arrangement or format conversion) is the ASCII Text File of Distances to Each Feature. Note that in these distance files each pair of features (nodes) is listed only once, as required by CS22 (see section 5.3). If you do not change the default name, the ASCII text file of distances will be saved as "res_comp.txt".

When the analysis is completed, an output report file is opened. The output report gives you information on the specified settings for the analysis (input theme, number of records analysed, search radius and ID field), outputs generated and total time elapsed.

Note that ArcView always slows down with large shapefiles (that is, with increasing number of polygon features). Also, highly complex features can slow down the computation time when calculating distances between them. Polygons with lots of vertices will take longer to analyse than polygons with few vertices. Some previous processing to reduce the total number of vertices describing the polygon boundaries

(especially removing redundant vertices if there are some) may benefit the speed of the distance computation through this extension.

While working, there is a 'current status' dialog box indicating the estimated time remaining for the end of the computation.

8.2.3. OTHER SOFTWARE TOOLS AND COMPUTATION OF EFFECTIVE DISTANCES

Note that the extensions above are only one alternative to calculate distances among the variety of GIS and image processing software and extensions to them, and you may use any other existing application (different from ArcGis 9.x or ArcView 3.x) to get the same result.

Note also that an improved connectivity analysis will consider in some cases effective (minimum-cost) distances and not just Euclidean (straight-line) distances as provided by these extensions. Effective distances consider the variable movement abilities and mortality risk of a species through different land cover types (incorporating the effects of the matrix on the connectivity analysis), and may improve the results provided by a connectivity analysis performed through straight-line distances (e.g. Adriaensen et al. 2003, Theobald 2006). Effective distances are typically obtained through least-cost path algorithms. Several GIS or image processing applications (or extensions) may be used to calculate the effective distances between nodes.

One option to compute effective distances is to use the PathMatrix extension for ArcView 3.x that needs to be used in conjunction with the Spatial Analyst module; this extension is freely available for download from <http://cmpg.unibe.ch/software/pathmatrix/> (accessed May 2007). Another option is to use the cost distance tools within ArcGis 9.x. The processing time required to compute effective distances may be much larger than for the straight-line distances. The effective distances computed through this or other software packages can be provided as well as an input for the connectivity analysis in CS22. CS22 is general enough and adapted to analyse connectivity through either effective or straight-line distances.

8.3. SAMPLE DATA

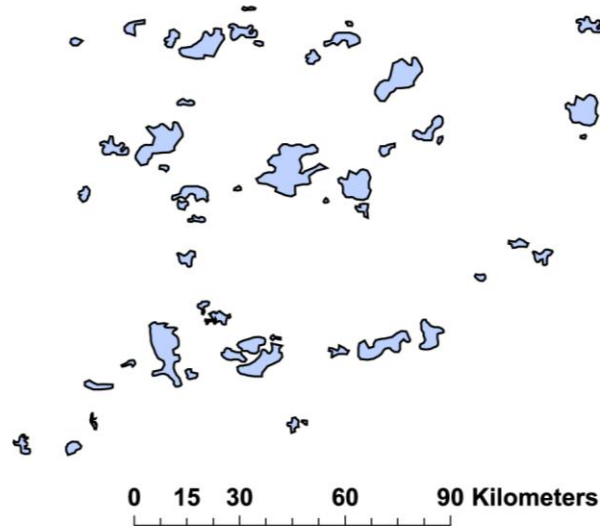
CS22 includes two simple data sets that you can use to get familiar with the software before needing to prepare your own data for your particular case studies. These two data sets are not particularly relevant and do not correspond to any real-world case study. They are just included to allow the users an easy testing of the software. These sample data are located in the "Sample data" folder, within the parent directory where you installed CS22 (typically C:\Program files\Conefor Sensinode 22\Sample data, or some other similar path).

The first sample dataset corresponds to 55 habitat patches in which the relevant attribute (a_i variable in Equations 5-8, second column in the node file) is the area of each patch. The following files are included within the folder "Sample data\Sample 1":

- Nodes1.txt: Node file, in which each node represents an existing habitat patch (no nodes to add) and the node attribute is the habitat area of each patch.
- Distances2.txt: Euclidean (straight-line) edge-to-edge distance (in meters) between every pair of patches (full connection file, see section 5.3). This distance file can be obtained as well from the ArcView 3.x or ArcGIS 9.x extension (see section 8.2) and the layer HabitatPatches.shp described next.

- HabitatPatches.shp/.shx/.sbx/.sbn/.dbf: Vector layer as a shapefile (and associated files) with the habitat polygons (patches, nodes), which can be viewed with ArcGis/ArcView or other GIS.

Note that the distance between patches (nodes) in this dataset (see figure below) ranges from a few kilometres to several dozens of kilometres, which should be taken into account when entering the distance and probability values (see sections 5.2.2.1 and 5.2.3.1).

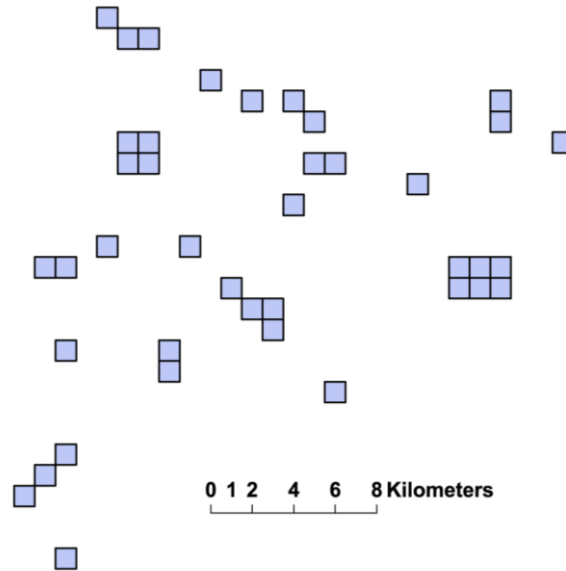


The second sample dataset corresponds to 40 equally-sized (1 x 1 km) habitat cells in which the relevant attribute (a_i variable in Equations 5-8, second column in the node file) is the habitat quality in that cell (ranging from 0 to 1). The following files are included within the folder "Sample data\Sample 2":

- Nodes2.txt: Node file, in which nodes represent existing 1 x 1 km cells (no nodes to add) and the node attribute is the habitat quality in each cell.

- Distances2.txt: Euclidean (straight-line) edge-to-edge distance (in meters) between every pair of nodes (full connection file, see section 5.3). This distance file can be obtained as well from the ArcView 3.x or ArcGIS 9.x extension (see section 8.2) and the layer HabitatCells.shp described next.

- HabitatCells.shp/.shx/.sbx/.sbn/.dbf: Vector layer as a shapefile (and associated files) with the location of the 1 x 1 km cells and their attributes (habitat quality) that can be viewed with ArcGis/ArcView or other GIS.



Note that the sample data #2 includes adjacent habitat cells (contiguous habitat cells with different habitat quality, see figure above), and not all the connectivity indices may be suited to adequately analyse this type of data (Saura and Pascual-Hortal 2007).

Note also that the distance between cells (nodes) in this dataset is in the range of a few to several kilometres, as shown in the figure above. This should be taken into account when entering the distance and probability values (see sections 5.2.2.1 and 5.2.3.1).

9. LIMITATIONS AND KNOWN ERRORS

9.1. PROCESSING CAPABILITIES AND LIMITATIONS

CS22 is a powerful software for analysing connectivity that allows processing landscapes comprising hundreds or thousands of nodes. Any standard computer is usable for most of the applications of CS22. However, the size of the datasets that can be processed in a single run is limited in practice by the following two characteristics of a computer:

- Amount of available RAM memory (which depends on the computer itself but also on the operative system and other settings). This will limit the maximum number of nodes (lines in the node file) for which the processing can start with CS22. For a personal computer with 1 GB RAM, this maximum will typically be about 20000 nodes for the binary indices and about 10000 nodes for the probabilistic indices. Increasing the amount of RAM memory in your computer may allow increasing these maximum values. However, since the required RAM is a power function of the number of nodes, doubling the available RAM will only provide a comparatively minor increase in these maximum values. Therefore, it may not be possible to process too big node files with any standard personal computer.

- CPU speed (GHz). Once the processing has started (as allowed by the RAM memory), the processing speed (GHz) of your CPU will determine how fast the node importance values can be calculated and, depending on the total number of nodes and links/connections, if the full processing can be accomplished in a reasonable time (e.g. less than 24 hours).

The processing time required to complete the analysis will not depend only on the CPU speed of your computer but mostly on the following specific characteristics of a particular connectivity analysis:

- The connectivity indices selected. There are great differences in the processing times required by the different indices, as described below.
- The landscape configuration.
- The dispersal distances of the analysed species, as quantified through the distance and probability values in the boxes for the binary and probabilistic indices.

The landscape configuration and the scale of analysis will determine the total number of nodes in the analysis, while the interaction of these two factors with the dispersal distance will determine the number of links/connections, with larger dispersal distances implying a larger number of links/connections for the same landscape. For most of the connectivity indices the processing times increase rapidly with the number of nodes and links/connections in the analysis (bigger landscapes and/or larger dispersal distances), especially for indices like IIC or PC.

Although the processing times may vary largely depending on the factors described above, the following are rough guidelines on the processing times and capabilities of CS22 for a typical landscape and a standard computer with a 3 GHz CPU:

- Most of the binary indices (NL, NC, H, CCP, LCP) are very fast to process, and usually you will be able to complete the analysis in a few minutes, or just up to a few hours even for the largest node files (as allowed by the RAM memory, about 20000 nodes for 1 GB).

- For H and IIC the processing is slower (since they require the computation of shortest paths between every two nodes), but you will probably be able to process up to 10000 nodes in less than several hours. Note that the required processing time increases as a power function of the number of nodes and links in the landscape. Therefore, analysing landscapes with about 15000 nodes may not be possible in practise since it may require several days of processing.

- F and AWF are relatively fast to process, and you may be able to complete the analysis for the largest node files (as allowed by the RAM memory, about 10000 nodes for 1 GB). For the largest number of nodes CS22 will usually require several hours of processing.

- PC is the most computationally demanding index, and you will usually have to limit the analyses to datasets with a maximum of about 2000 nodes (far below the maximum imposed by the RAM memory). Far above this value CS22 may be still able to continue with the processing, but it may need as much as several weeks to complete some analyses. Note that fixing a moderate value of the minimum probability in the option "Only check paths above a minimum probability" (advanced settings for the PC index, see section 6.2.3.2) may provide faster results in very big landscapes and/or sparsely connected graphs (i.e. landscapes where most of the pairs of nodes are weakly connected or not connected at all, which may be the case when the analysis spans a large geographical range). See section 6.2.3.2 for further details.

Increasing the CPU speed of your computer will reduce the processing times, but note that the processing times increase as a power function of the number of nodes and links for several of the indices. Therefore, processing a too large set of nodes through indices like PC may not be feasible in any standard personal computer. Therefore, in some cases the only option may be reducing the number of nodes to be processed at a time in CS22. Most of the processing time is required to calculate the node importance values. The overall index values and other results are comparatively much faster to calculate.

Note that the percentage of the execution process completed so far is shown in real time in the bottom left corner of the CS22 main screen. Based in this percentage, you may easily estimate the remaining processing time for a particular analysis.

9.2. KNOWN ERROR AND WARNING MESSAGES

This section provides a list of some of the most common known error and warning messages you may come up with while using CS22. We provide a short explanation of their possible causes and brief instructions for their solution, which in some cases implies reading in more detail some of the previous sections of this manual, as indicated next:

MESSAGE: *Node file must have 2 columns (unless there are nodes to add, which was not selected)*

Cause: Occurs when trying to use a node file with a different format from that required by the program (see section 4.2).

Solution: (1) Indicate that your node file contains nodes to add (in the node file box in the main CS22 screen) if this is the case, or (2) modify the node file so that it contains two columns, the first with the node IDs, and the second with the node attributes (see section 4.2).

MESSAGE: *Node file must have 3 columns (when there are nodes to add, as selected)*

Cause: Occurs when you have selected the "There are nodes to add" option and you are trying to use a node file with a different format from that required by the program (see section 4.2).

Solution: (1) Unselect the "There are nodes to add" option (in the node file box in the main CS22 screen) if your node file does not include nodes to add, or (2) modify the node file so that it contains three columns, the first with the node IDs, the second with the node attributes and the third with the code indicating whether it is an existing or potential node for addition (see section 4.2).

MESSAGE: *Cannot open node file or connection file xxxxxxx*

Cause: The file name xxxxxxx is not correct or the file with that name is not located in the specified folder. The file name or location may have been changed.

Solution: Browse and select the correct path and file name.

MESSAGE: *Not all the lines within the node or connection file have the same number of columns. Line xxxx in the node or connection file has yyyy columns, which is different from previous lines in the same file.*

Cause: Not all the lines in the node file have the same number of columns, or not all the lines in the connection file have the same number of columns. Some of the values for the columns in these files may be missing.

Solution: Check the format of the node file or the connection file. Note that all the lines in the node file should have two columns/values (when there are no nodes to add) or three columns/values (when there are nodes to add), and that all the lines in the connection file should have three columns/values (see sections 4.2 and 5.3).

MESSAGE: *A non-numerical node ID value has been found in line xxxxxx of the node file (first column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: Node IDs should be integer values, but non-numerical characters (e.g. text) have been found at least for one of the nodes in the node file (first column).

Solution: Check and correct the format of the node file so that it contains only integer numbers for the node IDs, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-integer node ID value has been found in line xxxxxx of the node file (first column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: Node IDs should be integer values, but a different numerical value (e.g. floating point) has been found at least for one of the nodes in the node file (first column).

Solution: Check and correct the format of the node file so that it contains only integer numbers for the node IDs, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-numerical node attribute value has been found in line xxxxxx of the node file (second column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: Node attributes should be numerical values, but non-numerical characters (e.g. text) have been found at least for one of the node attributes in the node file (second column).

Solution: Check and correct the format of the node file so that it contains only numerical values for the node attributes, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A negative node attribute value has been found in line xxxxxx of the node file (second column). Please correct. Processing will continue, but the RESULTS WILL almost surely BE INVALID because of that error in the input data. Please cancel and discard the processing and correct the format of the node file.*

Cause: Node attributes should be non-negative numbers, but negative values have been found at least for one of the node attributes in the node file (second column).

Solution: Check and correct the format of the node file so that it contains only non-negative numerical values for the node attributes, as described in section 4.2. After prompting this message CS22 will continue reading the rest of the node file and notifying about additional negative values if there are some. However, the results of the analysis will be invalid for most of the indices because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-numerical value has been found in line xxxxxx of the node file (third column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: The values in the third column of the node file (when there are nodes to add) should be just 0 or 1, indicating if each node already exists in the landscape or is a candidate to be added to the landscape (see section 4.2). However, a non-numerical value (e.g text) has been found at least in one of the lines of the node file (third column).

Solution: Check and correct the format of the node file so that it contains only 0 or 1 integer values for the third column, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-integer value has been found in line xxxxxx of the node file (third column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: The values in the third column of the node file (when there are nodes to add) should be just 0 or 1, indicating if each node already exists in the landscape or is a candidate to be added to the landscape (see section 4.2).

However, a non-integer value (e.g. floating point) has been found in at least one of the lines of the node file (third column).

Solution: Check and correct the format of the node file so that it contains only 0 or 1 integer values for the third column, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A value different from 0 or 1 has been found in line xxxxxx of the node file (third column). Please correct. No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the node file.*

Cause: The values in the third column of the node file (when there are nodes to add) should be just 0 or 1, indicating if each node already exists in the landscape or is a candidate to be added to the landscape (see section 4.2). However, a value different from 0 or 1 has been found in at least one of the lines of the node file (third column).

Solution: Check and correct the format of the node file so that it contains only 0 or 1 values for the third column, as described in section 4.2. Note that after prompting this message CS22 will not read any more lines in the node file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-numerical node ID value has been found in line xxxxxx of the connection file (first/second column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the connection file.*

Cause: Node IDs should be integer values, but non-numerical characters (e.g. text) have been found at least for one of the nodes in the connection file (first or second column, as indicated in the message itself).

Solution: Check and correct the format of the connection file so that it contains only integer numbers for the node IDs, as described in section 5.3. Note that after prompting this message CS22 will not read any more lines in the connection file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-integer node ID value has been found in line xxxxxx of the connection file (first/second column). No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the connection file.*

Cause: Node IDs should be integer values, but a different numerical value (e.g. floating point) has been found at least for one of the nodes in the connection file (first or second column, as indicated in the message itself).

Solution: Check and correct the format of the connection file so that it contains only integer numbers for the node IDs, as described in section 5.3. Note that after prompting this message CS22 will not read any more lines in the connection file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A non-numerical value has been found in line xxxxxx of the connection file (third column). Please correct. No more lines will be read from this file. RESULTS WILL BE INVALID. Please cancel and discard the processing and correct the format of the connection file.*

Cause: Numerical values should be used for characterising the direct connections (either distances, probabilities or links) between nodes. However, non-numerical characters have been found at least in one of the lines of the connection file (third column).

Solution: Check and correct the format of the connection file so that it contains only numerical values for the connection information (either distances, probabilities or links) in the third column of the connection file, as described in section 5.3. Note that after prompting this message CS22 will not read any more lines in the connection file. Although the processing will continue, the results will be invalid because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A negative distance value for the nodes with IDs xxxxxxx and yyyyyy has been found in the third column of the connection file. Please correct. Processing will continue, but the RESULTS WILL almost surely BE INVALID because of that error in the input data. Please cancel and discard the processing and correct the format of the connection file.*

Cause: When "Distances" is the connection type, non-negative distance values are expected in the third column of the connection file. However, negative distance values have been found at least in one of the lines of the connection file (third column).

Solution: Check and correct the format of the connection file so that it contains only non-negative distance values in the third column, as described in section 5.3. After prompting this message CS22 will continue reading the rest of the connection file and notifying about additional negative distance values if there are some. However, the results of the analysis will be invalid in most of the cases because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A probability value smaller than 0 or bigger than 1 for the nodes with IDs xxxxxx and yyyyyy has been found in the third column of the connection file. Please correct or check if "Distances" may be the appropriate type of connection for your connection file. Processing will continue, but the RESULTS WILL almost surely BE INVALID because of that error in the input data. Please cancel and discard the processing and correct the format of the connection file.*

Cause: When "Probabilities" is the connection type, values ranging from 0 to 1 are expected in the third column of the connection file. However, values outside that range have been found at least in one of the lines of the connection file (third column).

Solution: (1) Check if perhaps "Distances" is the correct type of connection for your connection file or, if "Probabilities" is the correct type, (2) check and correct the format of the connection file so that it contains only values ranging from 0 to 1 in the third column, as described in section 5.3. After prompting this message CS22 will continue reading the rest of the connection file and notifying about additional values out of the range if there are some. However, the results of the analysis will be invalid in most of the cases because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *A link value different from 0 or 1 for the nodes with IDs xxxxxx and yyyyyy has been found in the third column of the connection file. Please correct or check if "Distances" or "Probabilities" may be the appropriate type of connection for your connection file. Processing will continue, but the RESULTS WILL almost surely BE INVALID because of that error in the input data. Please cancel and discard the processing and correct the format of the connection file.*

Cause: When "Links" is the connection type, only 0 to 1 integer values should be included in the third column of the connection file. However, values different from 0 or 1 have been found at least in one of the lines of the connection file (third column).

Solution: (1) Check if perhaps "Distances" or "Probabilities" is the correct type of connection for your connection file or, if "Links" is the correct type, (2) check and correct the format of the connection file so that it contains only 0 or 1 values in the third column, as described in section 5.3. After prompting this message CS22 will continue reading the rest of the connection file and notifying about additional erroneous values if there are some. However, the results of the analysis will be invalid in most of the cases because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *Error in the format of the connection file. Nodes with IDs xxxxx and yyyyy are found in line zzzzz of the connection file, but at least one of them may not be listed in the node file. Please check that the node file ends with a blank line. Processing will continue without using the connection information for those nodes, but the RESULTS will probably BE INVALID because of that error in the input data. Please cancel and discard the processing and correct the format of the input files.*

Cause: The most typical cause is that the node file does not end with a blank line. In this case, the node ID in the last line of the node file is not read by CS22, causing an error when that node ID (either xxxxx or yyyyy) is found in the connection file. In other cases the node file may be correctly formatted (including that blank line at the end) but the connection file may still contain node IDs that are not previously listed in the node file, as required.

Solution: Check if the node file ends with a blank line, as required by CS22, and correct it if not formatted in that way. If the node file already includes a blank line at the end, look in line zzzzz of the connection file and check if the node IDs in that line (xxxxx and yyyyy) are listed in the node file, making the necessary corrections in either the node or the connection file. After prompting this message CS22 will continue reading the rest of the connection file and notifying about additional errors if there are some. However, the results of the analysis will be invalid in most of the cases because of this error in the input data. The user should cancel the processing and discard the results of this analysis.

MESSAGE: *Error in the format of the connection file. Some of the pairs of nodes may not be listed (as required by the full option) or too many pairs may be listed. The connection file has xxxxx lines (pairs of nodes), instead of the full yyyyy lines expected from the number of nodes in the node file.*

Cause: You have specified that the connection file is a "Full" file, but the number of lines in the connection file does not correspond to a full file for the number of nodes in the node file (see section 5.3). This may be because the connections for some of the pairs of nodes are not listed (too few lines)

or because there are more connections listed than what expected from the number of nodes (too many lines).

Solution: Check the format of the connection file (and complete missing lines if needed), or mark "Partial" instead of "Full" for the connection file if this is the case. Note that each pair of nodes should be listed only once, since connections are considered symmetric in CS22 (see section 5.3).

MESSAGE: *Error in the format of the connection file. Too many lines.*

Cause: This message appears when the "Partial" option has been selected for the connection file and the connection file contains too many lines, more lines than the maximum possible for the number of nodes in the node file (see section 5.3).

Solution: Check the format of the connection file and delete redundant or unnecessary lines (just leave the connections between those nodes that are listed in the node file). Note that each pair of nodes should be listed only once, since connections are considered symmetric in CS22 (see section 5.3).

MESSAGE: *Error in the distance threshold value (binary indices).*

Cause: Occurs when no value or an incorrect value (e.g. text characters) has been entered as the distance threshold for the analysis with the binary indices (see section 5.2.2.1). It may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start > Control Panel > Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the distance threshold in the box for the binary indices. Do not use a thousand separator when entering the number.

MESSAGE: *The value for the distance threshold cannot be negative (binary indices).*

Cause: Occurs when a negative value has been entered as the distance threshold for the analysis with the binary indices (see section 5.2.2.1).

Solution: Enter a correct (positive) value for the distance threshold in the box for the binary indices.

MESSAGE: *Error in the probability threshold value (binary indices).*

Cause: Occurs when no value or an incorrect value (e.g. text characters) has been entered as the probability threshold for analysis with binary indices (see section 5.2.2.2). It may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start -> Control Panel -> Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the probability threshold in the box for the binary indices.

MESSAGE: *The value for the probability threshold must range between 0 and 1 (binary indices).*

Cause: Occurs when a numerical value out of the range for a probability (from 0 to 1) has been entered as the probability threshold for the analysis with the binary indices (see section 5.2.2.2).

Solution: Enter a value ranging from 0 to 1 for the probability threshold in the box for the binary indices.

MESSAGE: *Error in the distance value (probabilistic indices).*

Cause: Occurs when no value or an incorrect value (e.g. text characters) has been entered as the distance (corresponding to a certain probability) for the analysis with the probabilistic indices (see section 5.2.3.1). It may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start -> Control Panel -> Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the distance in the box for the probabilistic indices. Do not use a thousand separator when entering the number.

MESSAGE: *The distance value cannot be negative (probabilistic indices).*

Cause: Occurs when a negative value has been entered as the distance (corresponding to a certain probability) for the analysis with the probabilistic indices (see section 5.2.3.1).

Solution: Enter a correct (positive) value for the distance in the box for the probabilistic indices.

MESSAGE: *Error in the probability value (probabilistic indices).*

Cause: Occurs when no value or an incorrect value (e.g. text characters) has been entered as the probability for the analysis with the probabilistic indices (see section 5.2.3.1). It may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start -> Control Panel -> Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the probability in the box for the probabilistic indices.

MESSAGE: *The probability value must range between 0 and 1 (probabilistic indices).*

Cause: Occurs when a numerical value out of the range for a probability has been entered as the probability for the analysis with the probabilistic indices (see section 5.2.3.1).

Solution: Enter a value for the probability ranging from 0 to 1 in the box for the probabilistic indices.

MESSAGE: *Error in the minimum probability value (PC index).*

Cause: Occurs when no value or an incorrect value (e.g. text characters) has been entered as the minimum probability in the advanced "PC setup" menu for the PC index (see section 6.2.3.2). It may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start -> Control Panel -> Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the minimum probability in the "PC setup" box (after clicking the "more" button next to the PC index).

MESSAGE: *The minimum probability value (PC index) must range between 0 and 1.*

Cause: Occurs when a numerical value out of the range for a probability (from 0 to 1) has been entered the minimum probability in the advanced "PC setup" menu for the PC index (see section 6.2.3.2).

Solution: Enter a value for the minimum probability ranging from 0 to 1 in the "PC setup" box (after clicking the "more" button next to the PC index).

MESSAGE: *Error in the value of the maximum landscape attribute (A_L).*

Cause: An incorrect value (e.g. text characters) has been entered as the maximum landscape attribute (A_L , see section 6.3). This error may also occur when using a comma as a decimal separator if a point is specified as the decimal separator in the regional settings of your computer (Start -> Control Panel -> Regional Settings), or viceversa.

Solution: Enter a correct numerical value for the maximum landscape attribute when appropriate. Do not use a thousand separator when entering the number. Note that it is optional to enter the value for A_L , and that if no value is entered no error will be reported and the processing will start with normality (see section 6.3). Note also that the value of the maximum landscape attribute can only be entered if LCP, IIC or PC have been previously selected for the analysis (see section 6.3).

MESSAGE: *Maximum landscape attribute (A_L) is lower than the sum of node attributes. Only the numerators of the overall LCP, IIC and PC values will be computed.*

Cause: The user has entered a value of the maximum landscape attribute (A_L , see section 6.3) that is lower than the sum of node attributes (including only the nodes currently existing in the landscape, not any nodes to add if there are some in the analysis). This makes no sense since A_L is the attribute value that would correspond to a patch covering all the landscape with the best possible habitat, which is always bigger (or at least equal) than the sum of the node attributes in the present landscape.

Solution: This is not an error but just a warning message. Since CS22 does not need the value of A_L for the node importances nor for other results (only for the overall values of LCP, IIC and PC), this incoherent value of A_L entered by the user will be discarded and the processing will continue with normality. The only difference is that only the numerators of these three indices (LCPnum, IICnum, PCnum) will be computed as the overall index values (see sections 6.3 and 7.2.1). It is up to the user to correct the value of A_L in a new processing if considered important.

MESSAGE: *Not enough memory. Please restart the software and try with a lower number of nodes.*

Cause: There are too many nodes. Such a big set of nodes cannot be successfully processed in your computer.

Solution: You may try to run the same analysis in another computer with a larger amount of RAM memory. Larger free RAM memory will allow processing landscapes with a larger number of nodes, but note that the amount of RAM needed grows as a power function of the number of nodes. Therefore, processing a too large set of nodes may not be feasible in any standard computer. In addition, even when the RAM memory is large enough (and the processing is then possible in this respect) the processing may require too much time to be successfully completed for some indices like PC (see further details on this in section 9.1). Therefore, if this is the case the only option may be reducing the number of nodes to be processed at a time in CS22. Note also that not all the indices included in CS22 have the same RAM requirements; depending on the index or indices selected for analysis the maximum number of nodes that can be processed with your computer may vary (see section 9.1). After getting this error, the current session of

CS22 will be unable to continue processing this or other analyses. You should close and restart the software.

MESSAGE: *An error occurred while attempting to initialise the Borland Database Engine (error \$2108).*

Cause: The Database Engine used by CS22 to save the results in DBF format has not been properly installed in the computer, or the connection to the Database Engine has failed.

Solution: (1) Check if the executable "CS22.exe" has been moved from its original location (usually C:\Program files\Conefor Sensinode 22\, or some other similar path), and if this is the case place it again there, in the same folder as the software libraries (*.dll). This will usually solve the problem. If not, (2) avoid saving dbf files when running two different CS22 sessions at the same time, (3) reinstall the software, or if none of the previous worked (4) simply save the results in txt format instead of dbf format.

REFERENCES

- Adriaensen, F., Chardon, J.P., De Blust, G., Swinnen, E., Villalba, S., Gulinck, H., and Matthysen, E. 2003. The application of 'least-cost' modelling as a functional landscape model. *Landscape and Urban Planning* 64 (4), 233–247.
- Bunn, A.G., Urban, D.L., and Keitt, T.H., 2000. Landscape connectivity: a conservation application of graph theory. *Journal of Environmental Management* 59 (4), 265-278.
- Calabrese, J.M., and Fagan, W.F. 2004. A comparison-shopper's guide to connectivity metrics. *Frontiers in Ecology and the Environment* 2 (10), 529-536.
- Hanski, I. 1994. A practical model of metapopulation dynamics. *Journal of Animal Ecology* 63, 151–162.
- Jaeger, J. A. G. 2000. Landscape division, splitting index, and effective mesh size: new measures of landscape fragmentation. *Landscape Ecology* 15 (2), 115-130.
- Jordán, F., Baldi, A., Orci, K. M., Racz, I. and Varga, Z. 2003. Characterizing the importance of habitat patches and corridors in maintaining the landscape connectivity of a *Pholidoptera transsylvanica* (Orthoptera) metapopulation. *Landscape Ecology* 18 (1), 83-92.
- Keitt, T.H., Urban, D.L. and Milne, B.T. 1997. Detecting critical scales in fragmented landscapes. *Conservation Ecology* [online] 1 (1): 4. URL: <http://www.consecol.org/vol1/iss1/art4>.
- Moilanen, A., and Nieminen, M., 2002. Simple connectivity measures in spatial ecology. *Ecology* 83 (4), 1131–1145.
- Pascual-Hortal, L. and Saura, S. 2006. Comparison and development of new graph-based landscape connectivity indices: towards the prioritization of habitat patches and corridors for conservation. *Landscape Ecology* 21 (7): 959-967.
- Pascual-Hortal, L. and Saura, S. 2008. Integrating landscape connectivity in broad-scale forest planning through a new graph-based habitat availability methodology: application to capercaillie (*Tetrao urogallus*) in Catalonia (NE Spain). *European Journal of Forest Research* 127: 23-31.
- Pascual-Hortal, L. and Saura, S. 2007. Impact of spatial scale on the identification of critical habitat patches for the maintenance of landscape connectivity. *Landscape and Urban Planning* 83 (2-3): 176-186.
- Ricotta, C., Stanisci, A., Avena, G.C. and Blasi, C. 2000. Quantifying the network connectivity of landscape mosaics: a graph-theoretical approach. *Community Ecology* 1 (1), 89-94.
- Saura, S., and Pascual-Hortal, L. 2007. A new habitat availability index to integrate connectivity in landscape conservation planning: comparison with existing indices and application to a case study. *Landscape and Urban Planning* 83 (2-3): 91-103.
- Saura, S. and Torné, J. 2009. Conefor Sensinode 2.2: a software package for quantifying the importance of habitat patches for landscape connectivity. *Environmental Modelling and Software* 24 (1): 135-139.

- Theobald, D.M. 2006. Exploring the functional connectivity of landscapes using landscape networks. In: Crooks, K.R., Sanjayan, M. (Eds.), *Connectivity Conservation*. Cambridge University Press, New York, pp. 416–443.
- Tischendorf, L., and Fahring, L. 2000. On the usage and measurement of landscape connectivity. *Oikos* 90 (1), 7–19.
- Urban, D. and Keitt, T. 2001. Landscape connectivity: a graph-theoretic perspective. *Ecology* 82 (5), 1205-1218.
- Verheyen, K., Vellend, M., Van Calster, H., Peterken, G., and Hermy, M. 2004. Metapopulation dynamics in changing landscapes: a new spatially realistic model for forest plants. *Ecology* 85 (12), 3302–3312.