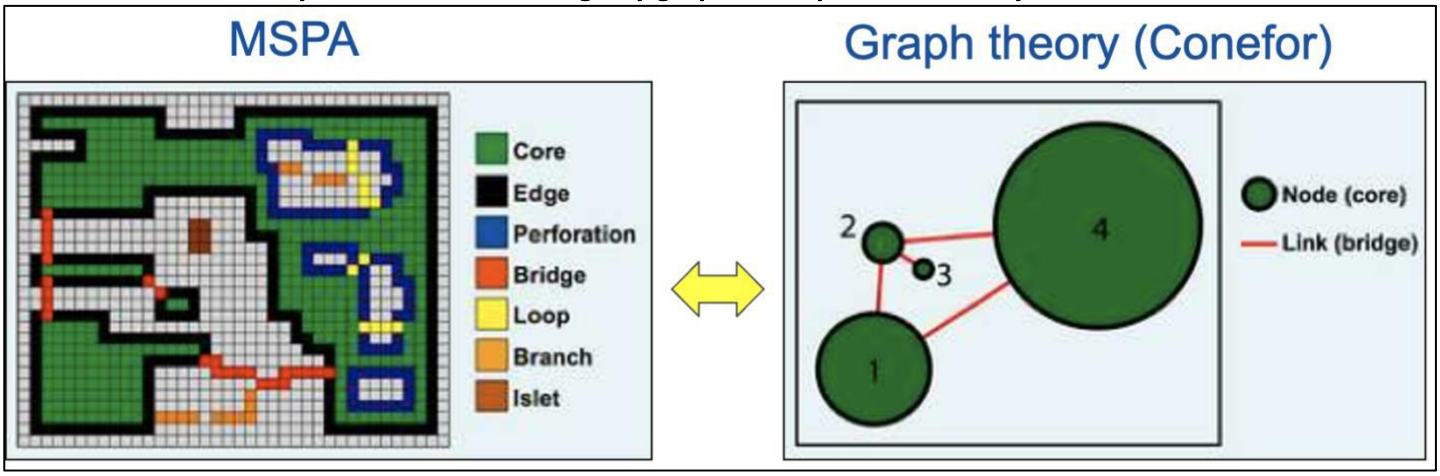
GTB-tools in container: Image Analysis → GTN(MSPA)
Task: summarize Graph Theory Network (GTN) features

All tools in this container examine **Graph Theory Network (GTN)** features. In graph theory, a network is composed of individual components and a component itself is composed of Nodes and Links. Because MSPA reliably detects Nodes and Links, it is used as a requirement for all graph theory analysis schemes: **you first must run a MSPA analysis before conducting any graph theory network analysis in GTB**.



The chart above shows the relation between MSPA and graph theory. The network illustrated above has one component, which is composed of four Nodes and four Links. The Nodes correspond to the MSPA class Core and the Links to the MSPA class Bridge. All other MSPA classes are neglected for graph theory analysis, for example the MSPA features Edge, Perforation, Loop, Branch, and Islet are removed when transitioning to a graph theory network. In addition, a graph theory Link (MSPA Bridge) is a feature without area.

GTN Components

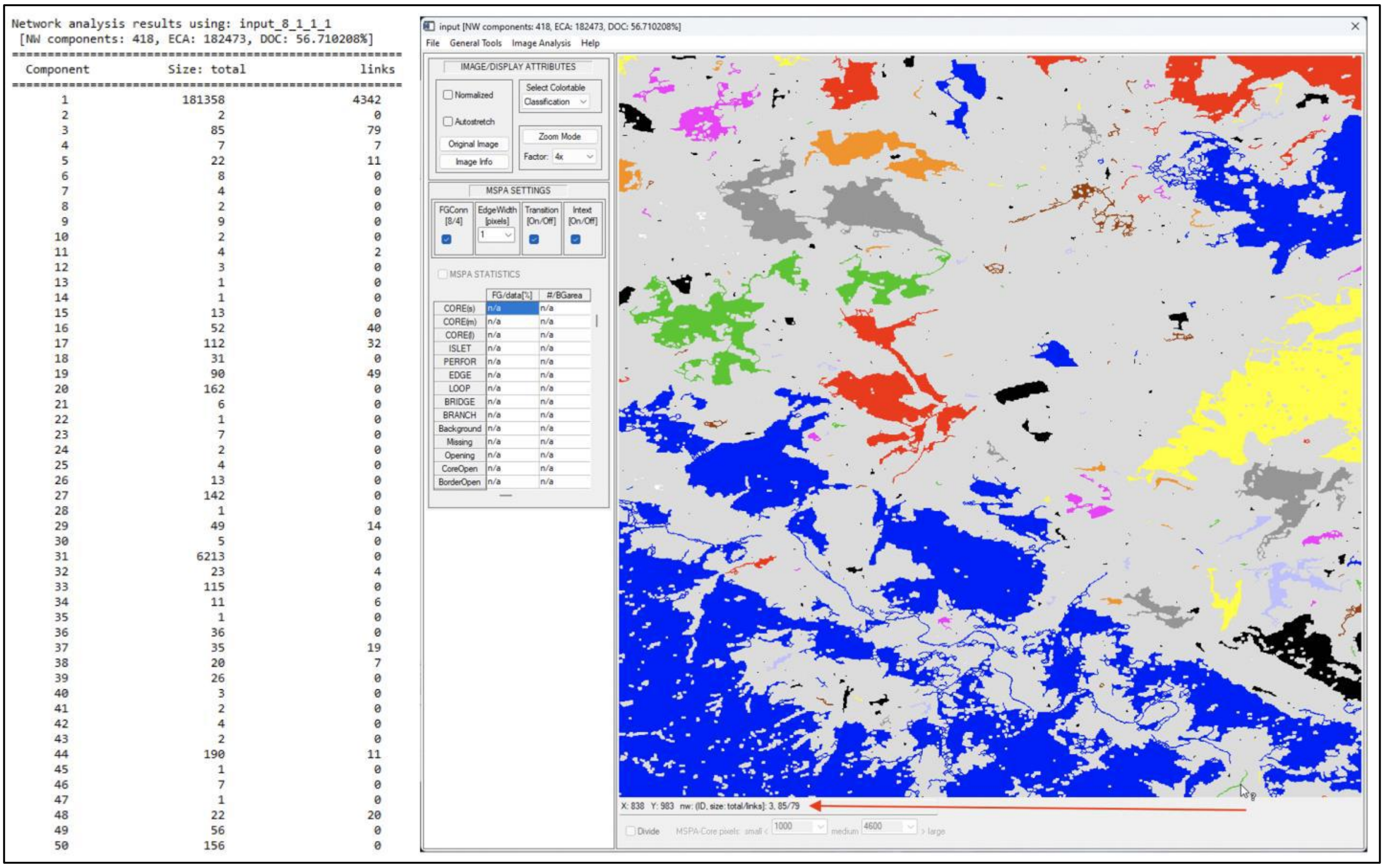
Question: how many graph theory network components are in the map? What is their size, number of Core and Links, and degree of connectivity?

The MSPA map is transferred to a graph theory network map by retaining the MSPA classes Core and Bridge only. Please note that this means that all other MSPA classes are neglected in a graph theory analysis! This graph theory network map is then analyzed to detect individual components, and their division into Nodes and Links. The equivalent connected area (ECA) and degree of connectivity (DOC) is calculated.

How: load a MSPA map and run the analysis.

Result:
Statistics: size [pixels] of each graph theory network component and its area [pixels] of Nodes and Links.
Map: color-coded map showing graph theory network components.

The chart below shows a map of color-coded graph theory network components. Components without Links, and hence consisting of a single Node only, are shown in black color. The red arrow shows the mouse cursor position at the location 838, 983 pointing at the green-colored component with the individual ID 3, and of total size of 85 pixels, out of which 79 pixels are Links, and consequently 6 pixels are Node pixels, see also the statistical summary extract in the left panel. Positioning the mouse cursor over any other location will exhibit the component ID, component size, and amount of Link pixels. The image has a total of 418 network components, with an equivalent connected area ECA = 182,473 pixels, or a degree of connectivity DOC = 56.71%.



Network analysis results using: input_8_1_1_1
[NW components: 418, ECA: 182473, DOC: 56.710208%]

Component	Size: total	links
1	181358	4342
2	2	0
3	85	79
4	7	7
5	22	11
6	8	0
7	4	0
8	2	0
9	9	0
10	2	0
11	4	2
12	3	0
13	1	0
14	1	0
15	13	0
16	52	40
17	112	32
18	31	0
19	90	49
20	162	0
21	6	0
22	1	0
23	7	0
24	2	0
25	4	0
26	13	0
27	142	0
28	1	0
29	49	14
30	5	0
31	6213	0
32	23	4
33	115	0
34	11	6
35	1	0
36	36	0
37	35	19
38	20	7
39	26	0
40	3	0
41	2	0
42	4	0
43	2	0
44	190	11
45	1	0
46	7	0
47	1	0
48	22	20
49	56	0
50	156	0

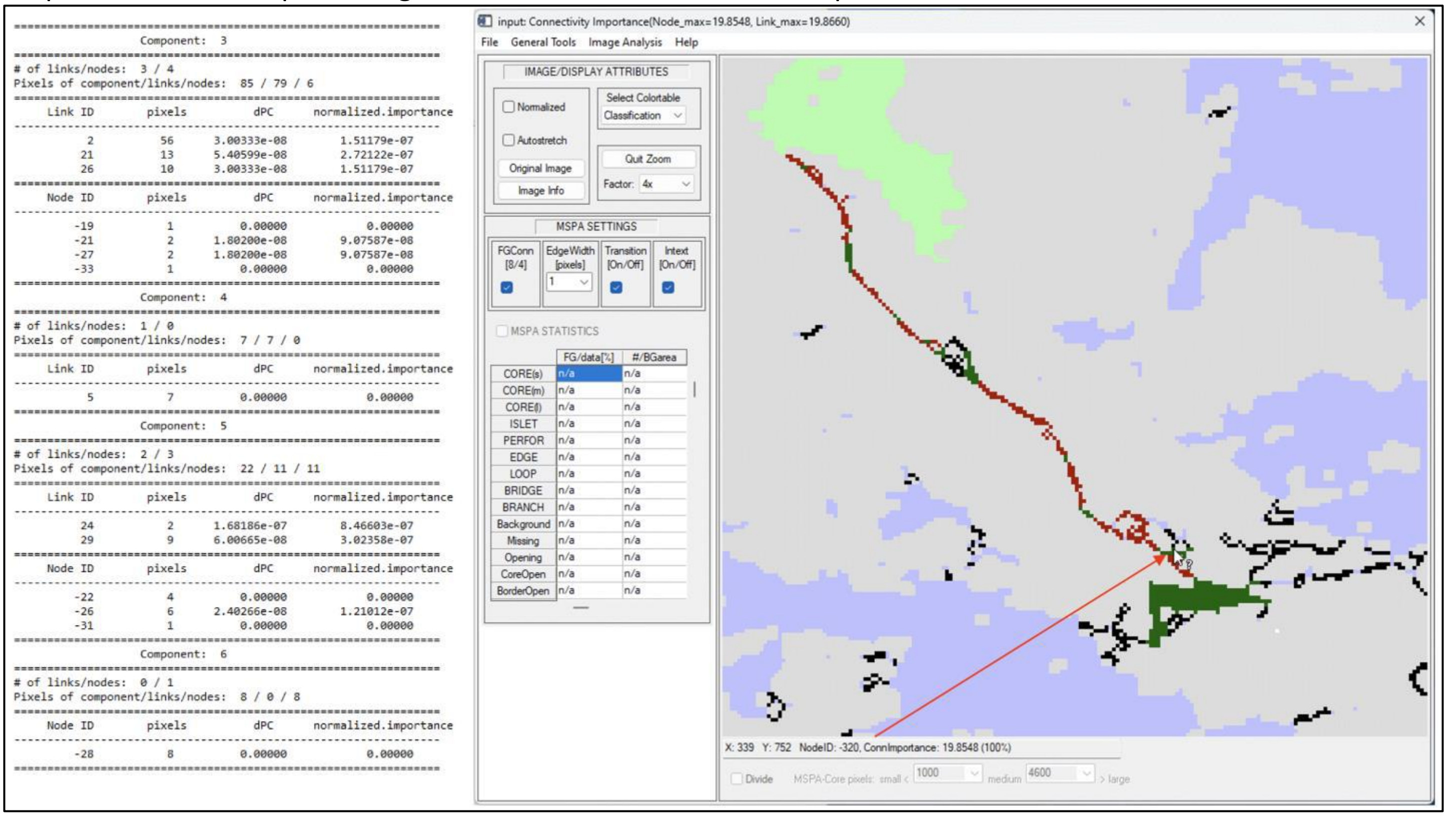
GTN Node/Link Importance

Question: how important is each Node and Link for the overall degree in network connectivity? How much will the overall degree in graph theory network connectivity decrease if a given Node or Link is removed?

The map of network components and its degree of connectivity (DOC) is used as a reference base. We then remove one single Node and calculate the new DOC. The new DOC will be lower because a Node was removed from the network. The decrease in DOC with respect to the original reference DOC is proportional to the importance of the Node that was removed. This process is repeated for removing any other Node from the original network, resulting in a list of Node importance for each Node of the network. The same scheme is applied for each Link and resulting in a list of Link importance. The importance of all Nodes is normalized such that the highest Node importance is assigned to 100%. The same normalization procedure is applied for the Links. After the normalization, each individual Link or Node will have a relative importance within [0, 100]%, with allows finding the most important Links and/or Nodes.
We then assign a decreasing intensity of green color for Nodes: dark green (highest importance: 99 $\rightarrow$ 100%), medium green (importance 95 $\rightarrow$ 99%), light green (90 $\rightarrow$ 95%) and with a decreasing intensity of red for Links: dark red (highest importance: 99 $\rightarrow$ 100%), medium red (importance 95 $\rightarrow$ 99%), light red (90 $\rightarrow$ 95%). Objects with a lower importance are color-coded with light purple color to mark Nodes (having a negative ID), while the black color denotes Links (having a positive ID to distinguish them from the Nodes).

How: use the network component map and run the analysis.

Result:
Statistics: individual Node/Link ID, size [pixels] of each network component, and normalized importance [%].
Map: color-coded map showing individual Node/Link IDs and importance.



The chart above shows a zoom-in on the connectivity importance result. The red arrow points to the location of the most important Node (100%, Node ID 320), which is color-coded in dark green. The title bar lists the values for the most important Node and Link. The left panel shows an exempt of the statistical summary output for example for component 3 mentioned before, with the IDs, area and importance of each Node and Link of each component. Please note that some Nodes/Links may have the same importance value and hence the most important Node/Link may be more than one.

MSPA ConeforInputs

Question: how can I do a more elaborate graph theory analysis?

Graph theory allows to analyze many more network aspects that are not covered in GTB. However, GTB allows setting up the two input files that are required by any graph theory software: one file describing the Nodes and one file describing the Links.

How: use the network component map and run the analysis.

Result:
Using this menu option will write out the two input files required for graph theory analysis with the Conefor software.