

NLMorphologyConverter (Version 0.10.0.dev)

User Manual

January 2010

www.neuronland.org

Overview

“NLMorphologyConverter” is a simple command-line program for converting between the various neuron morphology data formats which are used to describe the three-dimensional physical branching structure of biological neurons. The aim is to provide coverage of all formats, old and new, in which data is available online, and/or which are supported by free and commercial software packages (e.g. software for neuron reconstruction, generation, simulation, visualization, and analysis of neuron morphology).

Currently 25 formats are supported (covering 50+ format variations). Some are associated with morphology reconstruction systems (e.g. NeuroLucida, NeuroZoom, Eutectics, NeuronJ); others are the native formats of specific software packages (e.g. simulators, editors) used by the neuron reconstruction and neuron modeling community (e.g. SWC, NeuronHOC, Genesis, PSICS); some older formats have been superseded, and others see no use nowadays (Nevin, Douglas, Oxford); others are relatively new, and designed with open data-sharing in mind, i.e. MorphML (part of the NeuroML standard – see www.neuroml.org). (A full history of the various formats would be interesting to recount, but will have to wait for another time, and more research).

There is movement within the neuroscience community towards a smaller number of formats, with the most commonly used being NeuroLucida (www.mbfbioscience.com/neurolucida), due to its dominance in the market for reconstruction tools, and SWC, a compact text format which has now been adopted by the NeuroMorpho database (www.neuromorpho.org). NeuroML should also be noted as a format whose level of adoption should grow in the future, as direct support is added to more software.

The need to convert between different neuron morphology file formats is of course not new. Various converters are available, either as part of other software packages, as standalone executables, or scripts. These converters have been fairly limited in the number of file formats supported, and the conversion is usually uni-directional. NLMorphologyConverter tries to provide a bi-directional bridge between all 3D morphology formats, helping efforts to avoid the loss of data, and facilitating compatibility between all software programs which employ one or more of the supported formats. If any new formats are adopted in the future by the neuroscience community, they will be supported too.

The NLMorphologyConverter software is currently available as a binary download for Microsoft Windows and Apple Mac OSX operating systems only.

A related software package, NLMorphologyViewer is also available from the website. It is built on top of the conversion technology developed for the NLMorphologyConverter, and provides an interactive graphical view of the neuron data, as well as access to the main functionality provided by the converter.

The testing regime to which the software is subjected is detailed on the website (www.neuronland.org). The purpose is to demonstrate the lengths being taken to ensure the correctness of the import/export algorithms, and thereby make the user more confident in the validity of the generated data files.

Main Features

- Currently 25 different morphology file formats fully or partially supported.
- These 25 formats cover 50+ format variations (different version numbers, and layout styles).
- Automatic detection of input file format.
- Faithful reproduction of output file formatting.
- Many command line options for manipulating the imported data
- Intensively tested using over 20,000 publicly available morphology data files.

All users are advised to check the generated output data before making further use of it. Although import and export algorithms are well tested and generally reliable, there may be special cases which are not imported correctly - for certain file formats, original data often contains errors which the importer may not be able to deal with correctly; other formats are not sufficiently strictly specified that correct import can be guaranteed under all circumstances.

The Future

Work towards the first non-beta version of the NLMorphologyConverter (i.e. version 1.0.0) is still ongoing. Lack of development time means progress has been unexpectedly slow since 2011. It is hoped that version 1.0.0 will see the light of day in the not too distant future. Beyond this, bug-fixes will have priority, along with improved support for existing file formats. However, the fundamental functionality now exists in a reasonably advanced form, and will hopefully not require the investment of too much more development time.

The technology created during the development of the converter will continue to be incorporated into the NLMorphologyViewer (a fairly simple neuron viewer), and the bulk of future development time will be invested in improving this visual application.

Installation

Windows

The Win32 version comes in a standard Microsoft installer package (msi). Simply double-click on the installation package to install. Compatible with windows XP, Vista, 7, 8

Mac OSX

The Mac OSX version is currently a simple zipped binary. Installation is simply a matter of decompressing and placing the executable in a suitable location. The program will work on both PPC and Intel based Macs. (Current version targets Mac OSX 10.4 and higher.)

Linux

Linux support is currently on hold due to lack of development time.

Tutorial

For a full list of command line parameters, see the Command Line Options Reference section of this user manual.

1. *Basic Conversion*

The simplest way to convert a file containing morphology data directly from one format to another is as follows:

```
NLMorphologyConverter <input file> <output file> <output format>
```

For example,

```
NLMorphologyConverter neuron.asc neuron.swc SWC
```

In this case, a NeuroLucidaASC file is converted into an SWC file. Note that the input file format is not specified. It is automatically determined by the converter. If the input format cannot be determined, the conversion will fail. Note that the order of the parameters is important. Any command line parameters which do not start with a "--" are assumed to be, in order, the input file, output file, and output format. A list of output format names can be found in the command reference.

Several formats have several associated versions or styles, which can be selected e.g.

```
NLMorphologyConverter neuron.swc neuron.asc NeuroLucidaASC --style compact
```

and

```
NLMorphologyConverter neuron.asc neuron.xml MorphML --versionNumber 1.7.1
```

2. *Query*

Sometimes, one only wishes to query an input file, and not generate an output file. For example

```
NLMorphologyConverter neuron.asc --stats --warnings --report neuron.log
```

This will

- Output statistics for the neuron as a whole, and for the individual axonal and dendritic trees.
- Output a list of warnings (if any) indicating unusual/unrealistic/problematic data.
- Generate a detailed report file (neuron.log) containing a description of the import process, and also the statistics and warning data already displayed.

3. Modification

There are several command-line options which allow the neuron to be manipulated.

- The spatial co-ordinate of the tree can be globally manipulated, by rotating, scaling and translating.
- Various parts of the neuron structure can be omitted, using the “--omit*” options.
- The nature of axonal and dendritic can be swapped (i.e. axonal trees can be converted into dendritic trees, and vice versa).
- Spine and marker data can also be manipulated to a certain extent.

These options are all useful for fixing reconstructed data, or removing structure which is not of interest. For example, the command

```
NLMorphologyConverter neuron.asc neuron.xml MorphML --omitSomas
--translate 0 0 10.0
```

will import the NeuroLucidaASC file, omit any soma data, add 10 um to all z coordinates, then export in MorphML format.

Sometimes, it is necessary to query the tree first (using the “--stats” option), to obtain the IDs of the axonal or dendritic trees which we want to manipulate using the modify options.

```
NLMorphologyConverter neuron.asc neuron.xml MorphML
--omitTrees 1,5,9 --flipTreeType 3,4
```

In this case, the trees with ids 1, 5, and 9 are omitted before exporting the MorphML file. Trees 3 and 4 are converted from dendritic to axonal trees (or vice versa).

4. Merging

Sometimes, reconstructed neuron data is distributed across multiple files. They can be merged using the “--merge” option. For example,

```
NLMorphologyConverter --omitAxons --merge neuron1.asc --rotate 0 1 0 25
--translate 10 5 0 --omitMarkers
--merge neuron2.asc --rotate 0 1 0 75
--scale 0.9 0.9 1.0 1.0
--merge neuron3.asc
--export mergedNeuron.asc NeuroLucidaASC
```

Note that modify options can be specified for each neuron merged. Modify options specified before the first “--merge” are applied globally. Modify options specified after a specific “--merge” option, are applied to the particular file being merged. The export file name and format must be specified with the “--export” option.

5. *Splitting*

The morphology data of a neuron's individual trees can be exported using the “--split” option. For example, if the neuron data file neuron.asc only has dendritic tree data,

```
NLMorphologyConverter neuron.asc --split tree .xml MorphML
```

will export the files, tree_dend_01.xml, tree_dend_02.xml, tree_dend_03.xml.....one file for each tree.

6. *Import Hinting*

NLMorphologyConverter has no problem identifying specific file formats. However, certain formats can sometimes require a little extra help to ensure a correct interpretation of the data. Hence the “--hintxxxx” class of options. For example,

```
NLMorphologyConverter neuron.swc --stats --hintSWCStyle gulyas
```

```
NLMorphologyConverter neuron.txt --stats --hintEutecticsAxon "out"
```

7. *Export Style*

Several formats have more than 1 distinct layout style (e.g. NeuronHOC has ntscale, swc2hoc, and many others). The styles may differ in the usage of comments, or white-space, or a more fundamental difference in the ordering of sample point data, or in structure naming. Other formats have a well defined versioning system, with subtle differences introduced as the format has evolved (e.g. NeuroML).

As described before, styles can be selected using the “--style” option. Specific versions can be selected using the “--versionNumber” option (NeuroML only). In addition, structure names (for relevant file formats only) can be controlled by specifying the “--setAxonPrefix”, “--setDendPrefix”, “--setSomaPrefix” options.

Additional export style options allow some control over comments, cell naming, and structure colours, where relevant.

Command Line Options Reference

Option	Additional Parameters	Since	Description
--------	-----------------------	-------	-------------

Help Options

--version	none	0.6.0	Displays the current version number (ignored if not the unique option)
--help	none	0.6.0	Displays a list of command-line parameters which can be passed to the program, with descriptions (ignored if not the unique option)

Query Options

--stats	none	0.6.0	Displays the following statistics, for the entire neuron, and also per dendritic and axonal tree. Number of sample points Number of branch points Branching order Number of terminals Total tree branch length (microns) Tree surface area (microns ²) Tree volume (microns ³) Maximum terminal path distance (microns) Maximum terminal physical distance (microns)
--warnings	none	0.6.0	Displays the following warnings: Tree root too distant from soma Unary branch point Trinary branch point Other Non-binary branch point (≥ 4 children) Disjoint branch connection Branch has zero samples Branch formed from only one sample Sample diameter is excessively large Sample diameter is unusually small Sample diameter is zero
		0.7.0	Unusually large change in diameter Large terminal diameter Unusually long section Unusually short section Intersecting sections Encapsulated branch (a child branch lies completely within the extent of its parent).
--report <filename>	1. filename	0.7.0	Generate a report file, detailing import, modification, statistics, warnings, and export.
--verbose	none	0.7.0	Normally, only minimal information is printed to the terminal. This option provides more detailed information about the import, modification, and export processes.
--test <filename> <format> <logfile>	1. filename 2. format 3. log file	0.10.0	Run extensive test suite on specified file.

Import Options

MERGING			
--merge <filename>	1. filename	0.7.0	Specifies a neuron data file to be merged. All subsequent modify options are applied to this neuron. All modify options specified prior to this options are applied to the resultant merged neuron.
VALIDATION			
--strictValidation	none	0.10.0	Don't accept certain files which deviate from the expected file format specifications (affects only Eutectics for now. Will affect more formats in the future.)
HINTING			
--hintSWCStyle <style>	1. style name	0.9.0	It is possible to help the SWC importer if necessary, specifying the style which the input file is known to have. stdswc, cnicswc, gulyasswc (*default)
--hintSoma <substring>	1. substring to match	0.9.0	<p>NeuroLucida: Identify substring of contour names should be treated as soma data. This is in addition to "Soma", "CellBody" and "Cell Body". We include this option because in many cases soma data is not tagged with the (CellBody) property.</p> <p>MorphML: Identify cable group name of segments which should be treated as soma segments. Default: "soma".</p> <p>NeuroZoom: Identify substring of structure which should be treated as soma data. Default: "soma".</p> <p>PSICS: Identify name of points/branches which should be treated as soma segments. Default: "soma".</p>
--hintAxon <substring>	1. substring to match	0.9.0	<p>Eutectics: Identify substring of tree contours which should be treated as axon data. Default: "ax"</p> <p>MorphML: Identify cable group name of segments which should be treated as axon segments. Default: "axon".</p> <p>NeuroZoom: Identify substring of structure which should be treated as soma data. Default: "axon".</p> <p>PSICS: Identify name of points/branches which should be treated as axon segments. Default: "axon".</p>
--hintDend <substring>	1. substring to match	0.9.0	<p>Eutectics: Identify substring of tree contours which should be treated as dendrite data. Otherwise all non-axon non-soma contours will be treated as dendrites.</p> <p>MorphML: Identify cable group name of segments which should be treated as dendrite segments. Default: "dend".</p> <p>NeuroZoom: Identify substring of structure which should be treated as dendrite data. Default: "dend".</p> <p>PSICS: Identify name of points/branches which should be treated as dendrite segments. Default: "dend".</p>
--hintAllContoursAsSoma	none	0.9.0	Affects: NeuroLucidaASC, NeuroLucidaDAT, NeuroLucidaXML, NeuroZoom.

			All contours are treated as soma data. Convenient for forcing all data to be imported. This data can later be modified with modify options.
--hintNeuronHOCEntryProc	1. procedure name	0.9.0	Specify the name of the procedure where we should start executing the HOC code
1D FORMAT HELPERS			
--set1DBifurAngle <angle>	1. angle	0.10.0	angle is in degrees. General. Binary child branches will split at this angle. Non-binary branch points will share this angle range. default 36
--set1DBifurAngleApical <angle>	1. angle	0.10.0	angle is in degrees. overrides General setting. Defaults to General setting)
--set1DBifurAngleAxon <angle>	1. angle	0.10.0	angle is in degrees. overrides General setting. Defaults to General setting
--set1DRootAngleRange <angle> <range>	1. angle 2. range	0.10.0	angle, range are in degrees. Range around angle which the dendrites root points will be spread. Default 90, 150
-- set1DRootAngleRangeApical <angle> <range>	1. angle 2. range	0.10.0	angle, range are in degrees. Range around angle which apical dendrites root points will be spread. Default 270, 45
-- set1DRootAngleRangeAxon <angle> <range>	1. angle 2. range	0.10.0	angle, range are in degrees. Range around 180' which the axon root points will be spread. Default 180, 45)
--set1DBranchingPatten <pattern>	1. pattern	0.10.0	pattern can currently only be planar. More to come in the future.

Modify Options

TREES			
--omitAllAxons	none	0.6.0	Axon data is not passed through to the exported data file.
--omitAllDends	none	0.6.0	Dendritic tree data is not passed through to the exported data file.
--omitTrees a,b,c	List of tree IDs	0.6.0	Specific trees (axonal or dendritic) are not passed through to the exported data file. Note, the "--stats" option needs to be used to obtain the IDs of trees to be omitted.
--flipTreeType n,m,a	List of tree IDs	0.6.0	The specified axonal (dendritic) trees are passed through to export as dendritic (axonal) trees.
--allDendsAsAxons	none	0.6.0	All dendritic trees in the imported file are passed to export as axon trees.
--allAxonsAsDends	none	0.6.0	All axons in the imported file are passed to export as dendritic trees.
--reverseAllTrees	none	0.9.0	Child branch order is reversed, at all branch points of all axonal and dendritic trees.
--reverseTrees a,b,c	List of tree IDs	0.9.0	Child branch order is reversed, at all branch points of the specified axonal and dendritic trees.
SOMAS			
--omitAllSomas (prev. --omitSoma)	none	0.6.0	No soma data is not passed through to the exported data file.
--omitSomas	List of soma IDs	0.9.0	Specific soma data is not passed through to the exported data file. Note, the "--stats" option can be used to obtain the IDs of somas to be omitted.

--forceSphereSoma	1. radius	0.9.0	Remove any existing soma data, and set a single sphere soma with given properties
--forceCylinderSoma	1. radius 2. length	0.9.0	Remove any existing soma data, and set a single cylinder soma with given properties
SPINES + MARKERS			
--omitAllSpines	none	0.9.0	Spine data is not passed through to the exported data file.
--omitGlobalSpines	none	0.9.0	Global spine data is not passed through to the exported data file.
--omitDendSpines	none	0.9.0	Spine data associated with dendrites is not passed through to the exported data file.
--omitAxonSpines	none	0.9.0	Spine data associated with axons is not passed through to the exported data file.
--omitAllMarkers (prev. --omitMarkers)	none	0.9.0	Marker data is not passed through to the exported data file.
--omitGlobalMarkers	none	0.9.0	Global marker data is not passed through to the exported data file.
--omitSomaMarkers	none	0.9.0	Markers associated with soma data are not passed through to the exported data file.
--omitAxonMarkers	none	0.9.0	Markers associated with axon data are not passed through to the exported data file.
--omitDendMarkers	none	0.9.0	Markers associated with dendrite data are not passed through to the exported data file.
--markersToSpines <markername>	1. all "markername"	0.9.0	Markers of the specific marker name (or all markers, if all is specified), are converted to spines. Global markers become global spines. Dendritic markers become dendritic spines.
--spinesToMarkers <markername>	1. "marker name"	0.9.0	All spines are converted to markers, with the given marker name. Global spines become global markers. Dendritic spines become dendritic markers. Useful for retaining spine data in file formats which support markers, but not spines.
--mapGlobalSpinesToBranches	1. limit (um)	0.9.0	Map global spine data (i.e. spine data which isn't explicitly associated with a specific branch) to specific branches. <limit> is max possible distance from a branch sample (um) for spine to be considered mappable. Specify a large limit if you wish to ensure that all global spines are mapped.
--mapGlobalMarkersToBranches	1. limit (um)	0.9.0	Map global marker data (i.e. marker data which isn't explicitly associated with a specific branch) to specific branches. <limit> is max possible distance from a branch sample (um) for spine to be considered mappable. Specify a large limit if you wish to ensure that all global markers are mapped.
--forceSpinesGlobal	none	0.9.0	Spines which are associated with soma or tree structure, are disassociated, and made global (useful for NeuroZoom export)
--forceMarkersGlobal	none	0.9.0	Markers which are associated with soma or tree structure, are disassociated, and made global (useful for NeuroZoom export).
--spinesToBranches	none	0.10. 0	Spines are converted to branches. 2-sample branches, except in the case of spine contours (e.g. those imported from ImarisHDF format).

GLOBAL POSITIONING			
--scale x y z d	1. x-axis scale 2. y-axis scale 3. z-axis scale 4. diameter scale	0.6.0	All 3D co-ordinate data points (sample, and marker positions, and diameter) are scaled by the given values
--translate x y z	1. x-axis shift 2. y-axis shift 3. z-axis shift	0.6.0	All 3D co-ordinate data points (sample, and marker positions) are offset by the given 3D vector (x, y, z)
--rotate x y z A	1. x 2. y 3. z 4. A	0.7.0	All 3D co-ordinate data points (sample, and marker positions) are rotated 'A' degrees around the given direction vector (x, y, z).
--moveOriginToSoma	none	0.9.0	Translates entire neuron, so that the soma centre point - the average of all soma sample points (or tree root centre point if there is no soma) becomes position (0,0,0). Overrides --translate.

Export Options

FORMAT/STYLE/CUSTOMIZATION			
[--export] <filename> <format>	1. filename 2. format	0.6.0	Specify output filename and morphology format. This option is only essential when using the "--merge" option, otherwise it can be omitted (specify just filename and format directly). Format can be one of the following ArborVitae Douglas3D Eutectics Genesis Glasgow GulyasTree NeuronAmiraMesh (0A0: renamed to AmiraMesh) MaxSim MorphML (0A0: renamed to NeuroML) NeuroLucidaClassic NeuroLucidaV3 (080: renamed to NeuroLucidaASC) NeuroZoom NeuronGeneric (090: renamed to NeuronHOC) NeuronNTSCable (090: incorporated into NeuronHOC) NeuronSWC2HOC (090: incorporated into NeuronHOC) Nevin NINDS3D Oxford SWC
		0.8.0	NeuroLucidaASC (renamed from NeuroLucidaV3) NeuroLucidaDAT (first and currently only binary format)
		0.9.0	NeuroLucidaXML NeuronHOC (replaces NeuronGeneric, etc) PSICS

		0.10.0	Amaral AmiraMesh (replaced LNeuronAmiraMesh) Korogod NeuroML (replaces MorphML)	
--versionNumber <string>		0.9.0	Specify the version number of the exported format. Currently only affects NeuroML export. * is default 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.7.1, 1.7.2, 1.7.3, 1.8.0, 1.8.1, 1.8.2*, 2.0	
--style <string>		0.9.0	Specify a style for the exported format. (styles are equivalent representations of the data) * default export style for each format.	
			AmiraMesh	Neuron*, skeletongraph, spatialgraph (added version: 0.10.0)
			NeuroLucidaASC	modern*, verbose, classictags, nocomments, compact, nmpml
			NeuroLucidaDAT	badmarkersize*, corrected
			NeuroML	neuroml*, morphml
			NeuronHOC	slsr*, swc2hoc, cnic, ntscable, geometry_to_neuron, Oz_neuron, topol, morphmlxsl
			NeuroZoom	longpath*, shortpath
	SWC	stdswc*, cnicswc		
--colourAxon, --colorAxon --colourDend, --colorDend --colourSoma, --colorSoma --colourSpine, --colorSpine --colourMarker, --colorMarker	1. Colour name RGB(R:G:B)	0.9.0	Colour setting, for formats which support such data - NeuroLucidaASC, NeuroLucidaDAT, NeuroLucidaXML, NeuroZoom. Note American (color) or British (colour) spelling permitted. Applied to all elements of specified type, i.e. currently there is no fine control. Valid colour names: Red, Green, Blue, Yellow, Cyan, Magenta, White, DarkRed, DarkGreen, DarkBlue, DarkYellow, DarkCyan, DarkMagenta, Orange, MediumGray, BrightGray, SkyBlue, MoneyGreen, Cream For RGB specification, R, G, and B are integers in the range 0-255.	
--setPrefixAxon <string>	1. prefix	0.9.0	Use this to override default axon structure identifier. Relevant for Eutectics (default "ax"), Genesis (default "a"), MorphML (default "axon_group"), NeuroZoom (default "axon").	
--setPrefixDend <string>	1. prefix	0.9.0	Use this to override default dendrite structure identifier. Relevant for Eutectics (default "d"), Genesis (default "d"), MorphML (default "dendrite_group"), NeuroZoom (default "dendrite").	
--setPrefixSoma <string>	1. prefix	0.9.0	Use this to override default soma structure identifier. Relevant for Eutectics (default "cell"), Genesis (default "soma"), MorphML (default "soma_group"), NeuroZoom (default "soma"), NeuroLucidaXXX (default: "CellBody")	

--identifyNL	none	0.9.0	Where a file format supports comments, a comment will be generated which identifies NLMorphologyConverter as the creator.
--cellName <string>	1. cell name	0.9.0	Assigns a name to the cell - only relevant where a file format supports comments, or a specific mechanism for naming the cell. (Default: use import file name).
--addComment <string>	1. comment string	0.9.0	For suitable formats, add an arbitrary comment
SPLITTING			
--split <prefix> <extension>	1. prefix 2. extension	0.7.0	The imported data is exported one file per tree, with filenames of the form prefix_dend_XX.extension , or prefix_axon_XX.extension.

References

todo...

Links

Neuronland website - www.neuronland.org

NeuroMorpho morphology data archive – www.neuromorpho.org