RNAPoly documentation

August 27, 2018

Contents

1 Sterna **3**
1.1 Basics . . . 3
1.1.1 Setup . . . 3
1.1.2 Use . . . 4
1.1.3 Input / Output . . . 6
1.2 Settings . . . 8

2 Snac fileformat **10**
2.1 Specification . . . 10
2.2 Example . . . 10

3 Snacseq **11**
3.1 Usage . . . 11
3.2 Example . . . 12

4 Pkgen **13**
4.1 Usage . . . 13
4.2 Example . . . 13

5 Snac2ox **14**
5.1 Usage . . . 14
5.2 Example . . . 14

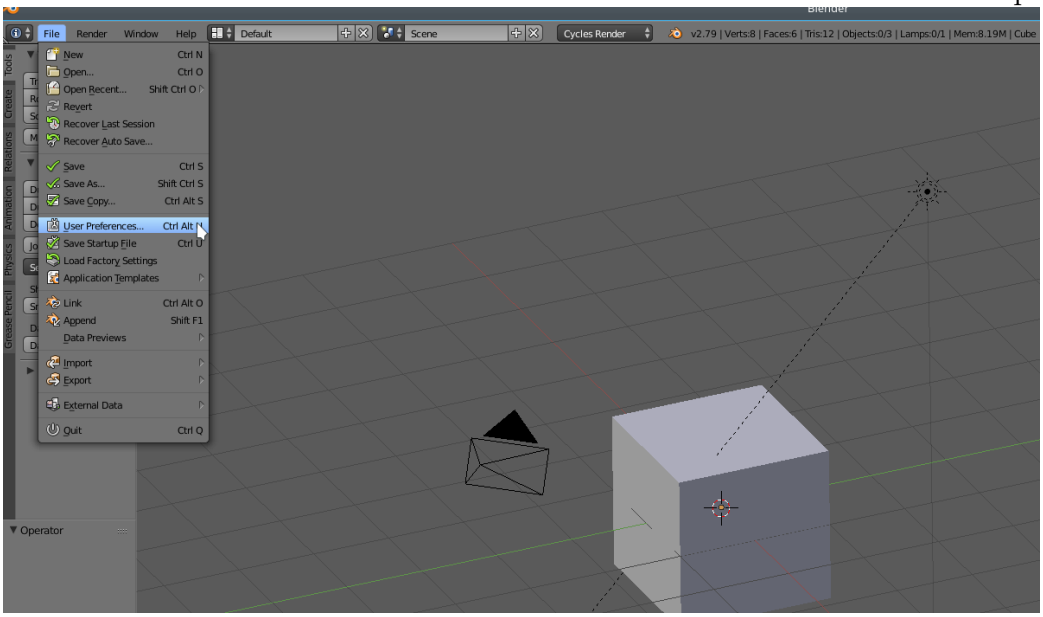
1 Sterna

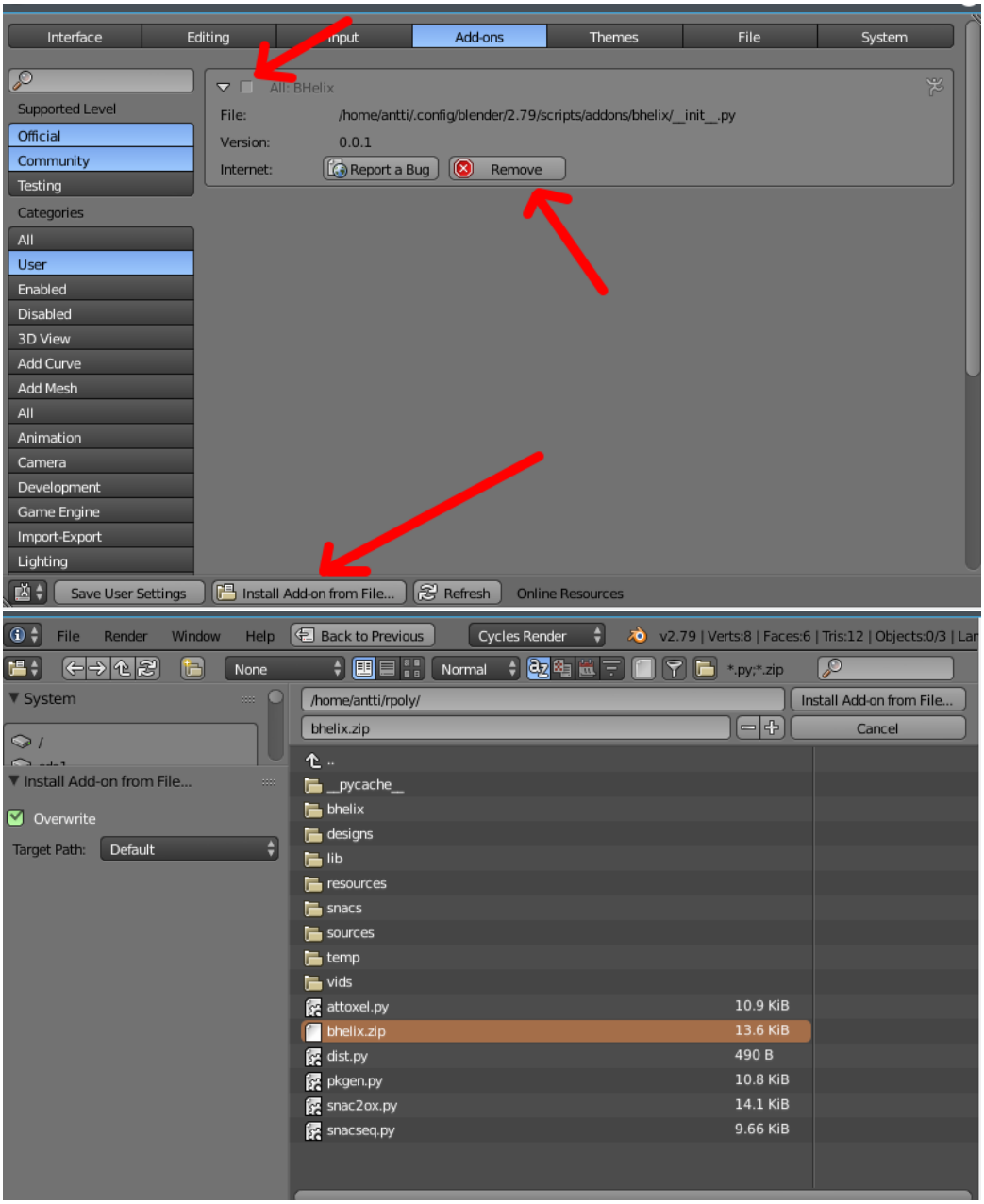
Sterna is an addon to Blender, a 3D modeling software. Sterna allows the user to design, edit and visualize RNA strands.

1.1 Basics

1.1.1 Setup

To install Sterna: `Navigate File -> Preferences -> Add-ons`. Select User category. If an older version of Sterna is already installed, deactivate it and remove it. Select Install Add-on from File. Find and select Sterna.zip.



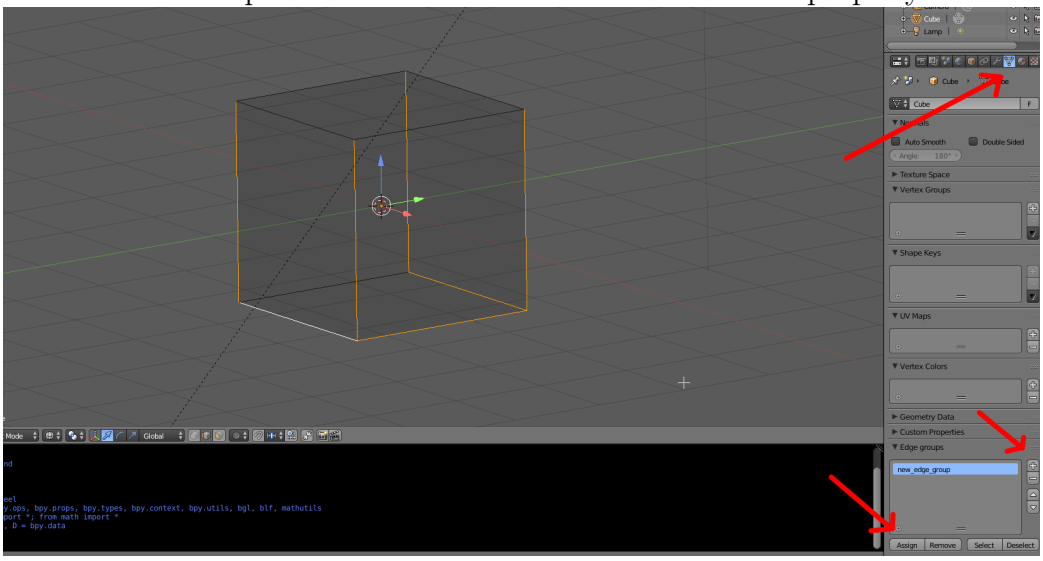


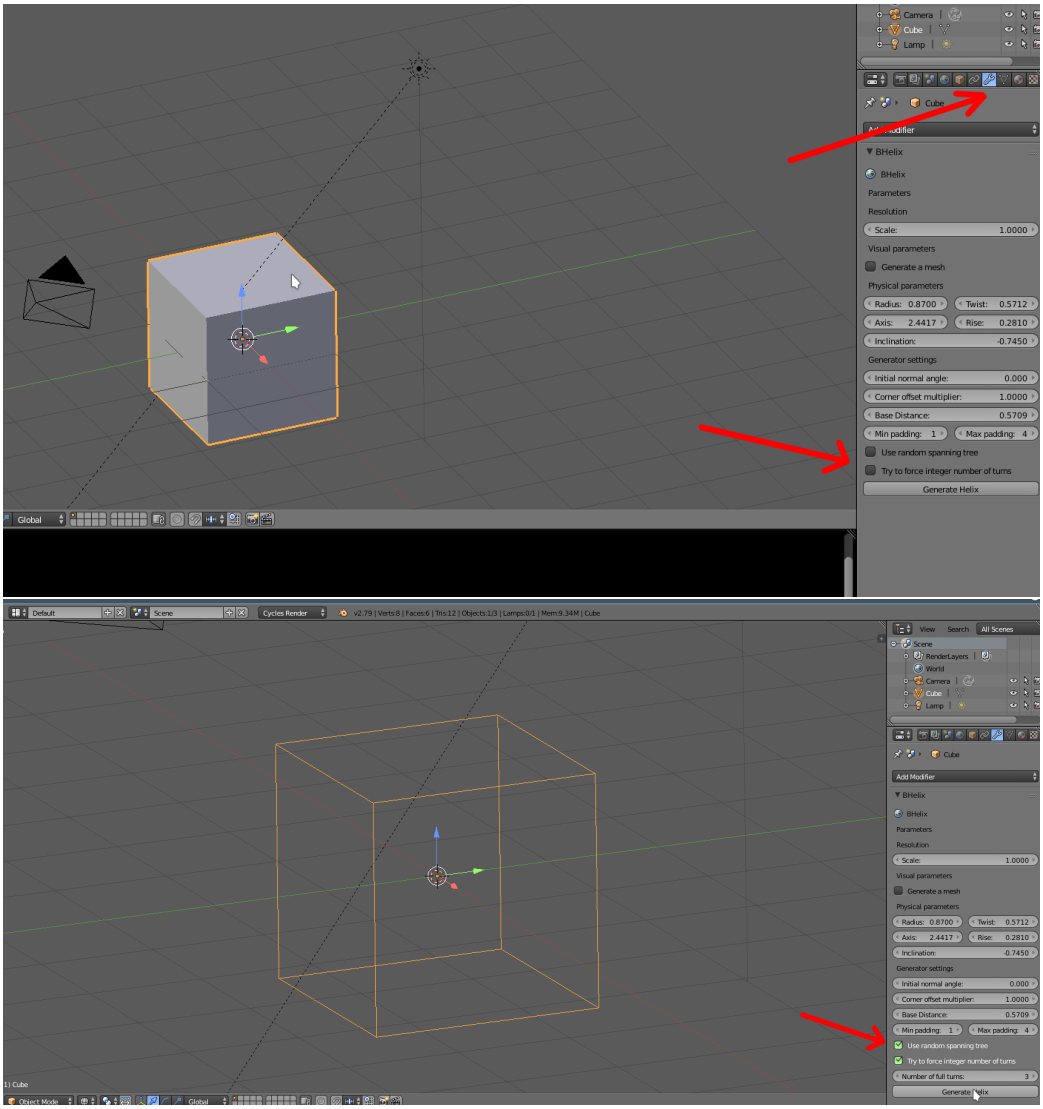
1.1.2 Use

The first step of designing an RNA strand is to choose a model. Any model will work as long as all its vertices are connected. To generate a helix: If you wish to choose the spanning tree edges, navigate to Data-tab and create a new edge group. Choose the object you wish to convert to a strand and go to edit mode by pressing TAB. Change the selection mode to edge-select by

pressing CTRL + TAB. Select the edges and press assign button under the edge group.

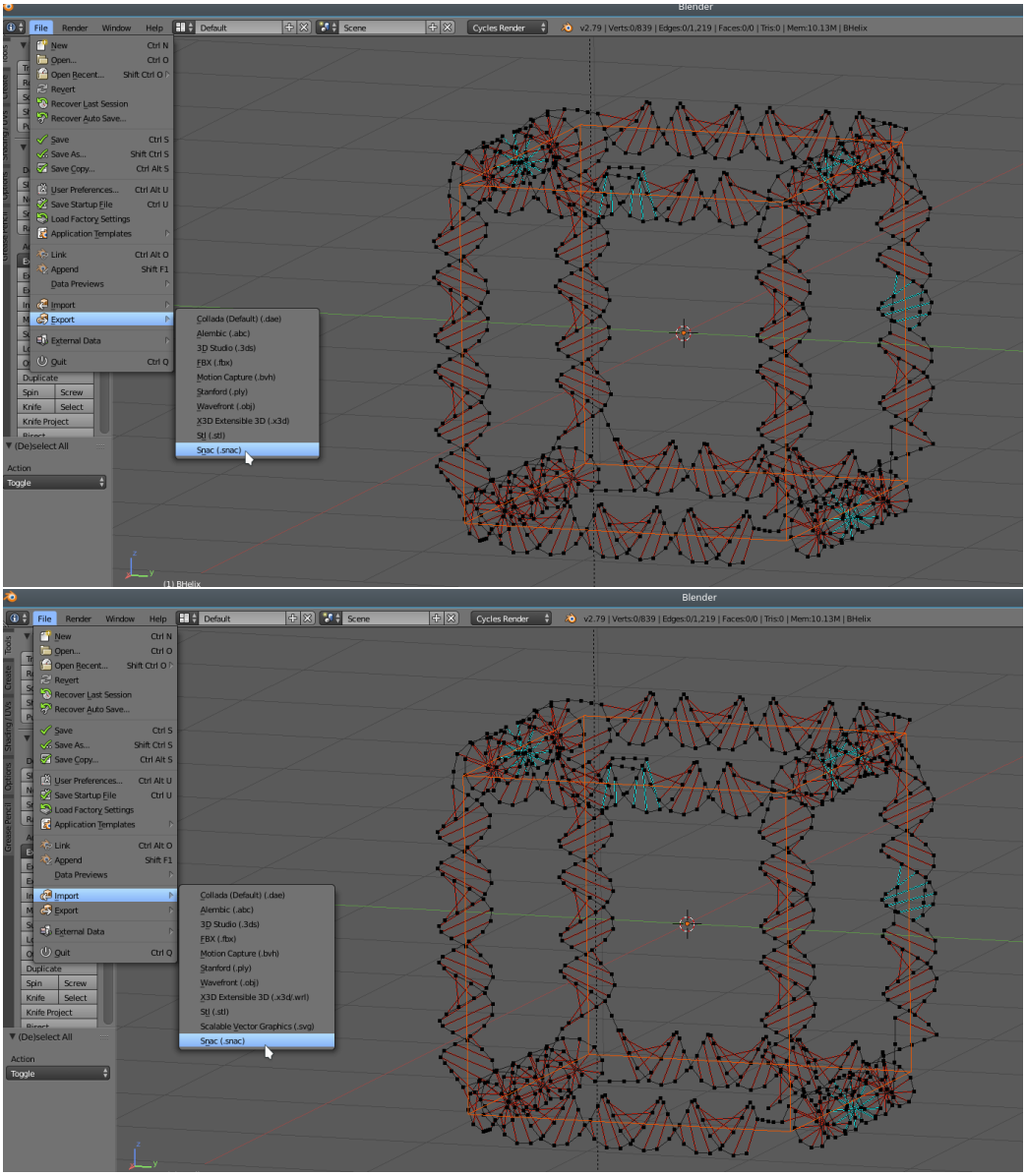
Next, navigate to the modifiers tab. If you wish to use a random spanning tree, tick "Use random spanning tree" box. By default, one blender unit is equal to 1 nm. This can be changed by modifying the scale parameter. The scale parameters defines how many nanometers each blender unit is. If the scale is too small, Sterna will be unable to fit a strand around the model. If you wish the edges to have integer number of turns, tick "Try to force integer number of turns", and select a number. This changes the scale in such a way that the edges of the object contain an integer multiple of 11 bases, which is the wavelength of an RNA helix. The results might be unsatisfactory for asymmetrical structures. Finally, click "Generate Helix" button to generate the RNA strand. If unsuccessful, try tweaking the parameters or the object. Usually a failure is due to too small a scale or faulty object. If tweaking the scale does not help, make sure the object has no disconnected components and that its normal vectors are properly defined.

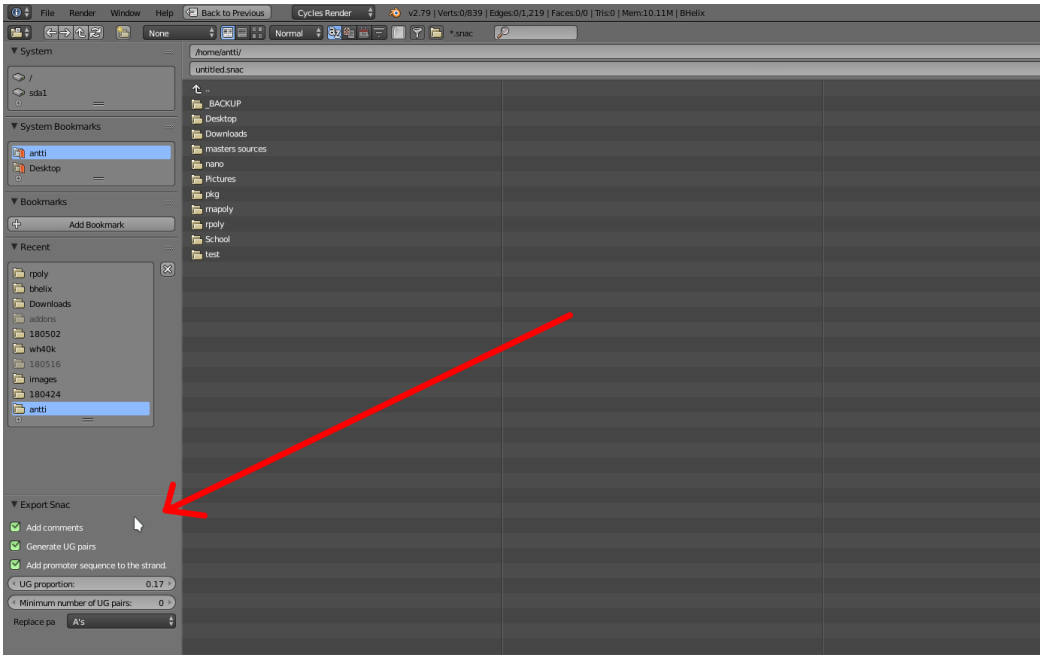




1.1.3 Input / Output

Sterna can import and export RNA strands in SNAC fileformat. To export a SNAC file, navigate to `File -> Exprot -> Snac`. Some attention should be paid to the options of the Sterna exporter. Generating UG pairs and their proportion in a helix can be toggled and changed, descriptive comments can be added and linkers can be replaced by any bases. If you wish to simulate the RNA strand with OxRNA, for instance, make sure "Add promoter sequences" is not ticked, since it adds 35 extra bases at the origin.


Blender
File Render Window Help
Default
Scene
Cycles Render
v2.79 | Verts:0/839 | Edges:0/1,219 | Faces:0/0 | Tris:0 | Mem:10.13M | BHelix
New Ctrl N
Open... Ctrl O
Open Recent... Shift Ctrl O
Revert
Recover Last Session
Recover Auto Save...
Save Ctrl S
Save As... Shift Ctrl S
Save Copy... Ctrl Alt S
User Preferences... Ctrl Alt U
Save Startup File Ctrl U
Load Factory Settings
Application Templates
Link Ctrl Alt O
Append Shift F1
Data Previews
Import
Export
External Data
Quit Ctrl Q
Collada (Default) (.dae)
Alembic (.abc)
3D Studio (.3ds)
FBX (.fbx)
Motion Capture (.bvh)
Stanford (.ply)
Wavefront (.obj)
X3D Extensible 3D (.x3d)
Stl (.stl)
Snac (.snac)
Duplicate
Spin Screw
Knife Select
Knife Project
(De)select All
Action
Toggle
(1) BHelix
Blender
File Render Window Help
Default
Scene
Cycles Render
v2.79 | Verts:0/839 | Edges:0/1,219 | Faces:0/0 | Tris:0 | Mem:10.13M | BHelix
New Ctrl N
Open... Ctrl O
Open Recent... Shift Ctrl O
Revert
Recover Last Session
Recover Auto Save...
Save Ctrl S
Save As... Shift Ctrl S
Save Copy... Ctrl Alt S
User Preferences... Ctrl Alt U
Save Startup File Ctrl U
Load Factory Settings
Application Templates
Link Ctrl Alt O
Append Shift F1
Data Previews
Import
Export
External Data
Quit Ctrl Q
Collada (Default) (.dae)
Alembic (.abc)
3D Studio (.3ds)
FBX (.fbx)
Motion Capture (.bvh)
Stanford (.ply)
Wavefront (.obj)
X3D Extensible 3D (.x3d/.wrl)
Stl (.stl)
Scalable Vector Graphics (.svg)
Snac (.snac)
Duplicate
Spin Screw
Knife Select
Knife Project
(De)select All
Action
Toggle



1.2 Settings

Sterna strand generation is affected by numerous input parameters and switches. What follows is a list explaining all of them.

- Scale parameter defines the length of Blender units in nanometers. A scale of 10 would mean that an edge of width 3.0 would be 30 nanometers long.
- Radius is a P-sticks model parameter, which defines the radius of the cylinder upon which bases are placed. A-form helix radius is 0.87 nm.
- Twist is a P-sticks model parameter, which defines the angle between consecutive bases along the helical axis. A-form helix twist is 0.5712 radians.
- Axis is a P-sticks model parameter, which defines the angle between bases from 5'-3' and from 3'-5' directions. A-form helix axis is 2.4417 radians.
- Rise is a P-sticks model parameter, which defines the distance between two consecutive bases along the helical axis. A-form helix rise is 0.2810 nm.

- Inclination is a P-sticks model parameter which defines the distance between two paired bases along the helical axis. A-form helix inclination is -0.745 nm, 5'-side ahead of 3'-side.
- Corner offset multiplier defines how close two helices are allowed to come to each other at vertices. A value of 0 would allow full overlap, a value of 1.0 zero overlap.
- Base distance defines the distance between two consecutive bases. It affects the density of linkers between helices.
- Min padding defines the minimum number of linkers between two helices.
- Max padding defines tha maximum number of linkers between two helices.
- Use adaptive offset switch forces the helices to get as close to each other as possible at vertices. If not toggled, all helices will be equidistant from the vertex.
- Use random spanning tree switch routes the helix around a random spanning tree instead of a user-defined spanning tree.
- Try to force integer number of turns switch tries to maximize the number of edges with integer number of full turns by modifying the scale parameter.
- Number of fulls defines how many turns the integer length edges should have.
- Spring relax switch tries to orientate the helices in a way which minimizes the distances/number of linkers between helices.
- Spring order defines the order of the spring strength. Larger values weigh long distance more strongly.
- Relaxation steps defines how many iterations the spring relaxation algorithm should perform. Larger values take longer but produce better results.

2 Snac fileformat

Snac is a fileformat for nucleic acids based on the YAML markup language. It stands for simple nucleic acid conformation.

2.1 Specification

A snac file consists of keys and values separated by colons. Values can either be strings or comma-separated lists. All key and value pairs are single-lined, unless the value is enclosed within angle-brackets. Comments start with a hashtag and they mark the rest of the line as a comment.

2.2 Example

```
# The coordinates are represented as comma-separated three-
positions: [2.5304946899414062 -3.349562168121338 -5.601208
-3.130314588546753 -4.694784164428711, 1.9479056596755981 -
-4.089140892028809, 2.2102911472320557 -1.6827961206436157
2.164266347885132 -0.9340331554412842 -4.964865684509277, 1
-0.9045512676239014 -5.748674392700195, 0.8564323782920837
-5.600904941558838, -0.037134621292352676 -1.14846348762512
-0.23751431703567505 -0.5540553331375122 -4.285967350006103
...
...
]
# domains separated by |:
# ..................... | ((((((((((((((((((((((((((((((
secondary_structure: .....................((((((((((((((((
# the following numbers are assigned to pseudoknot complexe
pseudoknot_numbering: 0 0
# Nucleic acid notation  '|':
# GACUAAUACGACUCACUAUA | GGGKNNNNNKNNNNNKNNNNNKNNNNNKNNNNN
primary_structure: GACUAAUACGACUCACUAUAGGGKNNNNNKNNNNNKNNNN
```

3 Snacseq

Snacseq is a commandline tool used to generate the primary structure of an RNA strand based on its secondary structure and various other constraints. It uses Pkgen to select pseudoknots, and it generates the primary structure with NUPACK. The input is a snac-file, and the output either modifies the input file or generates a new snac-file. Snacseq generates the input for NUPACK by completing a premade template file, which contains fields such as prevented sequences, temperature and sodium content. Snacseq also allows the user to define the maximum G-C-content, to enforce certain pseudoknots to be selected or to define the minimum energy between mismatching pseudoknots. Snacseq can also be used to generate only pseudoknots, if the user wishes to generate the rest of the sequence by other means.

3.1 Usage

```
usage: snacseq.py [-h] [-p PAIR_LIST] [-e ENFORCED_PAIRS]
                  [-o OUTPUT] [-r] [-c] [-i] [-gc GC_CONTENT]
                  [-t THRESHOLD] [-x] [-PPRIMARY_GENERATOR] snac
```

- -h; show help message and exit
- snac; The input file in snac format.
- -p PAIR_LIST; Select pseudoknots from this list.
- -e ENFORCED_PAIRS; Enforce the selection of these pseudoknots.
- -o OUTPUT; The output file name. If not specified, output is input file + _seq.
- -r; Replace the input file with the output.
- -c; Ignore conflicts with prevented sequences.
- -i; Ignore the existing primary structure. Otherwise use it as a restriction in generation.
- -gc GC_CONTENT; The maximum gc content of a pseudoknot in percents.
- -t THRESHOLD; The minimum threshold between mismatching pseudoknots. Default = 3.
- -x; Don't run primary structure design. Only find pseudoknots.

- -P PRIMARY_GENERATOR; The executable for the primary structure generator

3.2 Example

The simplest use-case is to generate the primary structure using default settings.

```
./snacseq.py cube.snac
```

Generating only pseudoknots and replacing the input file:

```
./snacseq.py -xr cube.snac
```

Generating pseudoknots from a file and enforcing some pseudoknots from another file:

```
./snacseq.py  -p resources/pairs/default.txt
              -e resources/pairs/ibuki.txt cube.snac
```

4 Pkgen

Pkgen is a tool used to generate and select pseudoknots. When used as a commandline program, it generates all possible combinations of two strands of length N and calculates the nearest-neighbor-energy of them using NUPACK. It takes N as an input and outputs a sorted list of the strands and their energies. Pkgen also implements an API of Python functions that can be called directly from another Python script to select pseudoknot strands from a list based on restrictions such as G-C content, prevented sequences or the energy between mismatching strands.

4.1 Usage

```
usage: pkgen.py [-h] [-l LENGTH] [-c COUNT] output
```

- -h; show help message and exit
- output; The output path
- -l LENGTH; Pseudoknot length
- -c COUNT; Number of pseudoknots to generate

4.2 Example

Generating all length 5-pseudoknots:

```
./pkgen.py -l 5 5pk.txt
```

Generating 5000 random length 7 pseudoknots:

```
./pkgen.py -l 7 -c 5000
```

5 Snac2ox

Snac2ox is a converter used to convert snac-files to the input format of OxRNA, i.e., a topology file, a configuration file and a simulation setup file. It also allows the user to generate a file describing kinetic traps between either pseudoknot base pairs or all base pairs.

5.1 Usage

```
usage: snac2ox.py [-h] [-o OUTPUT] [-x] [-r] [-c] [-t] [-p] [-g] input
```

- -h; show help message and exit
- input; The name of the snac file
- -o OUTPUT; The output file name.
- -x; Do not run the relaxation procedure.
- -r; Replace the initial configuration with the relaxed one.
- -c; Do not clean up the temporary files.
- -t; Do not add traps to the relaxation procedure.
- -p; Add traps only to the pseudoknots.
- -g; Also generate an input file.

5.2 Example

Converting a snac-file without relaxatation.

```
./snac2ox.py -x cube.snac
```

Converting a snac file and generating an OxRNA input file without relaxation:

```
./snac2ox.py -xg cube.snac
```

Converting a snac file with relaxation but without any traps:

```
./snac2ox.py -t cube.snac
```