RS3D

Version 1.1

Yuba Bhandari, George Zaki and Yun-Xing Wang
National Cancer Institute, Frederick, MD 21702

April 10, 2018

National Institutes of Health

SOFTWARE TRANSFER AGREEMENT

Provider: National Cancer Institute ("Provider")

WHEREAS, Provider has certain proprietary software and associated material described below (hereinafter, collectively referred to as "Software"):

RS3D is a program used to calculate three dimensional topological structures of RNA by providing secondary structure information, SAXS data, and any readily available long-range interaction information as input. This work is published in Journal of Molecular Biology, 2017 [1].

and Provider agrees to transfer such Software to Recipient for non-commercial research purposes only.

NOW, THEREFORE, in consideration of the premises and mutual covenants contained herein, the Provider and Recipient agree as follows:

1. SOFTWARE SHALL NOT BE USED FOR TREATING OR DIAGNOSING HUMAN SUBJECTS. Recipient will not license or sell or use Software for commercial purposes or applications. Recipient Investigator shall retain control over Software and further will not transfer the Software to individuals not under Recipient Investigator's direct supervision without the advance written approval of Provider. Recipient agrees to comply with all regulations applicable to the Project and the use of the Software.

2. Recipient agrees not to copy Software, in whole or in part, except as required for use by Recipient Investigator for the conduct of the Project. If source code is not provided to Recipient as part of Software, Recipient shall not modify, extend, decompile, make derivatives of or reverse engineer the Software without written permission from Provider.

3. In all oral presentations or written publications concerning the Project, Recipient will acknowledge Provider's contribution of Software unless requested otherwise. Recipient may publish or otherwise publicly disclose the results of the Project, but if Provider has given Confidential Information to Recipient, such public disclosure may be made only after Provider has had 30 days to review the proposed disclosure, except when a shortened time period under court order or the Freedom of Information Act pertains.

4. Title in the Software shall remain with the Provider. It is understood that nothing herein shall be deemed to constitute, by implication or otherwise, the grant to either Party by the other of any license or other rights under any patent, patent application or other intellectual property right or interest. Provider reserves the right to distribute Software to others and to use it for Provider's own purposes.

5. When the Project is terminated, completed or when three (3) years have elapsed, whichever occurs first, Recipient will destroy all copies of Software and Provider's Confidential Information unless directed otherwise by Provider in writing.

6. This Agreement may be terminated by either Recipient or Provider by providing 30 days advance notice.

7. The Provider and Recipient each shall retain title to any patent or other intellectual property of their respective employees developed or created in the course of the Project defined in this Agreement. Neither Provider nor Recipient promise rights in advance for inventions developed under this Agreement.

8. No indemnification for any loss, claim, damage, or liability is intended or provided by any party under this agreement. Each party shall be liable for any loss, claim, damage, or liability that said party incurs as a result of said party's activities under this agreement, except that the NIH, as an agency of the United States, assumes liability only to the extent as provided under the United States Federal Tort Claims Act (28 U.S.C. Chapter 171).

9. Software is supplied AS IS, without any accompanying services or improvements from Provider. SOFTWARE IS SUPPLIED TO RECIPIENT WITH NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Provider makes no representations that the use of Software will not infringe any patent or proprietary rights of third parties.

Provider: National Cancer Institute
Provider Investigator: Yun-Xing Wang, Ph.D.

Any questions concerning legal issues should be addressed to:
Jeffrey W. Thomas, Ph.D.
Unit Supervisor
Technology Transfer Center
National Cancer Institute
8490 Progress Drive, Suite 400
Frederick, MD 21701
Ph: (301) 624-1251

1. System Requirements

Singularity: Singularity is a container technology that allows the distribution of a program including all dependencies and requirements so that the user can simply use the program out of the box without the need to install or compile it. Instructions on installing singularity on Linux system can be obtained from http://singularity.lbl.gov/install-linux.

The RS3D program requires a Linux cluster consisting of several hundred or more CPUs to compute structural models. For instance, an RNA of size 71 nt needs approx. 70 K core hours to sample 20000 conformations using a GCC compiler in NIH Biowulf cluster. Use of the intel compiler speeds up the computational efficiency by three time as much in the same HPC platform. It is recommended to generate a minimum pool of 10000 to 20000 conformations to explore the folding landscape.

The program has been tested on RHEL 6.9 and Ubuntu 14.04, which are common Linux operating systems.

Python 2.7 with packages Numpy and Pandas

2. Software Installation and Setup

RS3D computes topological folds of RNAs given the secondary structural information, small angle X-ray scattering and any additional experimental restraints. Once you download the RS3D package, please refer to the “README” file in the RS3D directory for set up.

RS3D evokes several accessory programs to facilitate coordinate conversion, all-atom refinement and display. These are RNA2D3D, G2G, PyMOL and XPLOR-NIH. The first two programs, RNA2D3D (singularity container) and G2G, are included in the RS3D package. Educational PyMOL is a freeware and can be downloaded from the website “https://pymol.org/2/”; XPLOR-NIH can be downloaded from the website “https://nmr.cit.nih.gov/xplor-nih/”. You should be able to call “pymol” and “xplor” command from your terminal once you install both the programs.

3. Workflow

Command *RS3D.py input_RNA* initiates calculation by RS3D. The overall workflow is the following: the program takes initial open structures, which are generated based on secondary structures, and converts those open structures into "glob" structures, where each type of residue is represented by a corresponding glob with a defined form factor. The secondary structure is defined in the input file called "*input_RNA*". This file also contains the sequence and tertiary contact information if available. In addition, the user also needs to provide approximate maximum dimension of the molecule, junction type and SAXS data. The program then performs simulated annealing calculations under the SAXS restraint as well as any other user input restraints such as residue-to-residue distances. The best glob conformations are converted to explicit all atom models and then undergo an all-atom refinement against the SAXS data. During the whole process of calculation, RS3D makes use of several accessory programs for coordinate conversion, area calculation and refinement. These are RNA2D3D [2], G2G [3], PyMOL [4] and XPLOR-NIH [5]. Those programs are evoked seamlessly without user intervention.

We have provided three example directories corresponding to three RNAs in the folder "RS3D/tests". Each of these examples contain two input files, "*input_RNA*" and "*saxs-file*", and a python script "*RS3D.py*". In order to set up a new run, specify the path to the job directory ("path to folder where you want your results saved/") in the python script file. Look for the place holder "self.job_dir" in the *RS3D.py* file. To run the complete workflow, simply run the command "*RS3D.py input_RNA*" in the terminal by sitting on the job directory. If you compile RS3D with intel compiler, uncomment the line [self.processing_script = "qsub-run-icpc"] in the "*RS3D.py*" file.

Note: The job submission command in "*RS3D.py"* is based on Slurm Job System. If your HPC cluster is based on "SLURM" job submission, it should work mostly as is, but you may still need to modify some platform dependent variables. Especially, look for the functions "generate_qsub_jobs()" and "generate_qsub_xplor_jobs()" in the "*RS3D.py"* file. You will need to figure out an equivalent batch job submission command in the line starting with "qsub_template = Template".

4. Input File Format

There is one input configuration file called “*input_RNA*” which contains information on sequence, base pairing, and long-range interactions information such as pseudoknots, helical stack and long-range contacts. The input configuration file is divided into three sections as described below for the ease of parsing. It is recommended to preserve the order of the sections for the sake of consistency in the file format. Please refer to the sample input file “*input_RNA*.pdf” while reading these instructions. Actual examples of three RNAs are included in the file “*input_examples*.pdf”. Numbering of the first residue at the 5’ end starts at 1.

Section 1:

This section is marked by the term [GENERAL_DESCRIPTION]. This contains the following sub-sections:

i) SEQUENCE

In this subsection, the user needs to specify the RNA sequence, without enclosing in any brackets.

ii) RNA

In this subsection, the user needs to specify the name of the RNA. Any space in the RNA name will be stripped and maximum length of the string is 80 characters.

iii) NPDBS

In this subsection, the user needs to specify the maximum number of structures (conformers) per helical stack that one wants to calculate. This has to be an integer between 0 and 20000. The number of structures to be computed heavily depends on computation resource (numbers of CPUs) available to users. One can set up a quick test run with a small number of structures, for example, 100, to get a peek into the initial structural model and identify any potential issues in the workflow that can be resolved before running a full-scale calculation. This value can be set between 5000 – 20000 for a production run.

iv) SAXS_FILE

In this subsection, the user needs to specify the name of SAXS data file. Conventional format with three columns “q_value”, “intensity”, and “error” is expected. The error column is optional. RS3D makes a default upper cutoff at q = 0.5 A^{-1}.

v) DMAX

In this subsection, the user needs to specify the maximum particle dimension derived from the SAXS data. This can be an integer or a decimal number.

vi) JUNCTION_TYPE

In this subsection, the user needs to specify the type of RNA junction as an integer number. In case of multiple junctions, choose the largest junction type.

vii) GENERAL_PROPERTIES

In this subsection, the user needs to specify some basic molecule information and RS3D parameters. All the specifier terms should be self-explanatory and are also introduced below.

PDBFILE: name of input pdb file at glob level. This field should not be changed as the input pdb file is generated by the workflow, unless you want to provide a custom input file.

NRESIDUES: size of RNA in terms of total residues

VDWEXPT: specify the number of bond length exceptions to be ignored [options: integer]. This will allow some unusually long or short bond lengths in the single stranded region. Default value is 0. It is recommended not to change unless necessary.

VDW: enable optimum distance of separation between nucleotides [options: 1, 0]. Setting this to 0 may cause artificial overlaps.

CHI: enable SAXS χ^2 fit [options: 1,0] for conformational sampling

MOVESCALE: scale factor for the size of moves. Can be left to default value of 1.

ITERATIONS: number of steps for each temperature step. Typical value of 20,000

CYCLES: at the end of every cycle, the annealing temperature is reduced by a factor of 0.9. Typical value is 60.

TEMP: annealing temperature (options: 0-1000, float type)

Qmin: minimum value of the momentum transfer vector. Default is minimum Q-value in data.

Qmax: maximum value of the momentum transfer vector. Typical value of 0.3 for glob sampling

NGrids: number of grid points for interpolating the scattering intensity based on q-range.

The long-range contact information can be specified under the heading "PROXIMITY_CONSTRAINT". This will calculate the distance of separation between the centers of geometry represented by the two residue groups: GROUPA and GROUPB. Each group may contain a single residue or a group of residues. The approximate distance (Å) information between the two groups is specified in the field "CONSTRAINT_VALUE", and the program will try to converge to this distance during the conformational sampling. There can be more than one "PROXIMITY_CONSTRAINT" field in the input file (see example RNA 1Y26).

viii) NONPAIRED

In this section, the user specifies information on the single stranded region such as terminal loop, internal loop, and junction.

a) TERMINAL_LOOP and INTERNAL_LOOP

The terms TERMINAL_LOOP and INTERNAL_LOOP are immediately followed by another term "UNIT HUNIT_#", where the user specifies which helical unit the loop is a part of. A helical unit is considered to be a group of duplexes and loops forming a single unit. The number assigned to the helical unit should be an integer and starts at 1. The user is expected to enter the same "UNIT HUNIT_#" information while writing the corresponding duplex in the "PAIRED" subsection. The information on helical unit is utilized while carrying out the large-scale moves, that transform or rotate a complete helical segment as a rigid body, which consists of internal and terminal loops as well as the duplexes.

Long internal and terminal loops may be given a separate helical unit id (see example RNAs 2M58 and 2N8V). This will help translate and rotate the long loop as a rigid body, resulting in faster sampling of the conformational space.

b) JUNCTIONS

Internal and terminal loops are treated equally while carrying out the moves, but junction moves are relatively more frequent as they play an important role in the assembly of the helical segments for achieving the tertiary fold. Junction does not belong to any helical unit.

c) MY_SEGMENT

Also, user can specify a fragment of nucleotides under the heading MY_SEGMENT. This should be associated with a unique helical unit id specified by UNIT HUNIT_#. This feature is especially helpful, if you are using a fragment template from the database and want to translate or rotate the whole segment as a rigid unit. This feature is also useful in moving a complete domain or subdomain while working with multi-domain RNAs.

ix) PAIRED

In this section, the user needs to specify the base paring information for each duplex separately under the heading "HELICAL_SEGMENT". The helical unit information "UNIT HUNIT_#" should be specified immediately below, that represents the helical unit to which the duplex belongs. The base pair information is represented by writing individual, space delimited base pair in each line. Several duplexes may belong to the same helical unit represented by "UNIT

HUNIT_#”. If a short duplex (~ 3bp) is connected to a longer duplex through one/two nucleotide internal loop, all of them can be kept under the same helical unit (see example RNA 2M58).

*NOTE: Lines starting with “NAME” key word are optional and can be used for ease to keep track of the loops and duplexes (see example RNA 2M58).

Section 2:

This section is marked by the term [PSEUDO_DUP]. The pseudoknot base pair information can be specified in this section under “PSEUDO_DUP#=” heading (see example RNA 2M58). There can be more than one pseudoknots specified, and the program will try to converge the pseudoknot pairs in order to achieve duplex like base pairing between them. The pseudoknot pairs are expressed like base pairs in the duplex.

Section 3:

This section is marked by the term [HELICAL_STACKS]. All possible helical stacks across the junctions can be specified under separate “HELICL_STACK_#=” headings. The first helical stack type is numbered at 1. Under each helical stack type, stacking across a junction as well as additional stacking across internal or terminal loop can also be specified, with additional lines of stacking residues (see example RNA 2M58). There will be separate pool of conformational sampling for each helical stack type. It is recommended that the users be prudent on deciding how many helical stack types they want to run their calculation on. Mainly because, helical stacks across junction plays a crucial role in determining long range interactions and overall topological fold. Additionally, increase in the number of helical stack type also adds significant computational overload. The best structures will be derived from a combined pool of all different types of helical stacks.

Here is a description on how to write the helical stack information. For example, if the RNA has 3-Way junction with three helical segments H1, H2, and H3 across the junctions. There are three unique possibilities of stacking: between H1 and H2 (H1:H2 stack), between H2 and H3 (H2:H3 stack), and between H3 and H1 (H1:H3 stack). This translates to three kinds of helical stack, namely three headings HELICAL_STACK_1, HELICAL_STACK_2, and HELICAL_STACK_3 (see example RNA 2N8V). Under each heading, you will need to write all the nucleotides that are supposed to participate in the helical stacking. You may write in an increasing or decreasing order of nucleotides, but they have to be sequential and it is important that the first three nucleotides in the list should belong to a duplex. Helical stacking is implemented in RS3D based on distance of nucleotides relative to the position of the first three nucleotides lying in a duplex.

Information on long range interactions available from various biophysical/biochemical data may be helpful in excluding a specific helical stack across the junction. For example, in case of adenine riboswitch RNA, kissing loop interaction between the terminal loops of helix H2 and H3 rules out the possibility of H2:H3 stacking.

5. Analyzing the Results:

After the calculation is complete, the program should create a number of files and six directories inside the given job directory, namely, /best_pdbs, /best_allatoms, /processing, /rna2d3d, /xplor-processing, and /xplor_runs. Best 10 glob conformations are present in the folder /best_pdbs. A complete list of all glob conformations properly ranked based on total score can be found in the file “all_stats1_norm_final.txt, and these conformations are present in the folder “/processing”. Top 20 all-atom conformations with actual file path are listed in the file “top20-filter.txt” along with the SAXS χ^2 value.

References

[1] Bhandari YR, Fan L, Fang X, Zaki GF, Stahlberg EA, Jiang W, et al. Topological Structure Determination of RNA Using Small-Angle X-Ray Scattering. J Mol Biol. 2017;429:3635-49.
[2] Martinez HM, Maizel JV, Jr., Shapiro BA. RNA2D3D: a program for generating, viewing, and comparing 3-dimensional models of RNA. J Biomol Struct Dyn. 2008;25:669-83.
[3] Wang J, Zuo X, Yu P, Xu H, Starich MR, Tiede DM, et al. A method for helical RNA global structure determination in solution using small-angle x-ray scattering and NMR measurements. J Mol Biol. 2009;393:717-34.
[4] The PyMOL Molecular Graphics System, Version 15 Schrödinger, LLC.
[5] Schwieters CD, Kuszewski JJ, Tjandra N, Clore GM. The Xplor-NIH NMR molecular structure determination package. J Magn Reson. 2003;160:65-73.