**Rosetta comparative homology modeling (RosettaCM), command lines and code**

**Code and data availability:**

All code, input files, and finalized homology models are available at https://github.com/glasgowlab/Interface\_Energy\_nsp1.

**Rosetta version:**

2020.11.post.dev+202.master.0840a913640 0840a913640fb154357316971a8f4b0903f2659d

**RosettaCM reference:**

“High-resolution comparative modeling with RosettaCM.” Song Y, DiMaio F, Wang RY, Kim D, Miles C, Brunette T, Thompson J, Baker D., *Structure*. 2013 Oct 8;21(10):1735-42. doi:10.1016/j.str.2013.08.005. Epub 2013 Sep 12.

**Relax:**

Relaxing prepares high-resolution experimentally-solved ribosome-Nsp1 structures as templates for comparative homology modeling in Rosetta. We relaxed eight structures with PDB codes: 6ZLW, 6ZM7, 6ZME, 6ZMI, 6ZMO, 6ZMT, 6ZN5, 6ZOJ.

Example command line to relax the whole structure, PDB ID 6zlw:

/Users/anumglasgow/Rosetta/source/bin/relax.macosclangrelease -in:file:s 6zlw\_chain-i.pdb -database /Users/anumglasgow/Rosetta/database -relax:constrain\_relax\_to\_start\_coords -out:suffix \_relax -beta\_nov16 -corrections::beta\_nov16

Example command line to relax the ribosome-Nsp1 complex in a 150 Å radius around the Nsp1 C-terminal fragment, PDB ID 6zlw:

/Users/anumglasgow/Rosetta/source/bin/relax.macosclangrelease -in:file:s 6zlw\_small.pdb -database /Users/anumglasgow/Rosetta/database -relax:constrain\_relax\_to\_start\_coords -out:suffix \_relax -beta\_nov16 -corrections::beta\_nov16

**Partial threading:**

To guide the folding of the five evolutionarily-related Nsp1 C-terminal domain sequences, we threaded them on the eight ribosome-Nsp1 templates using the partial thread application in Rosetta. We also performed this step on the SARS-CoV-2 Nsp1 sequence. As input files for each model, we provided fasta files, alignment files in the Grishin format (one for every template), and the relaxed PDB template. Partial threading generates one model PDB file for each template, for each target sequence.

Example command line to build one threaded model:

/Users/anumglasgow/Rosetta/source/bin/partial\_thread.macosclangrelease -in:file:fasta ../fasta\_files/CoV1\_Nsp1.fasta -in:file:alignment ../grishin\_files/Cov1\_grishin/Cov1\_6zlw\_thread.grishin -in:file:template\_pdb ../minimize\_small\_pdbs/6zlw\_small\_relax.pdb

Fasta files:

>SARS-CoV-1\_Nsp1

ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

>SARS-CoV-2\_Nsp1

ELGTDPYEDFQENWNTKHSSGVTRELMRELNGG

>HKU1\_Nsp1

DAYAEVHAEPKGKYSQKAYALLRQYRG

>MERS\_Nsp1

EWMDDFEADPKGKYAQNLLKKLIGG

>MHV\_Nsp1.fasta

DACEEVHLNPKGKYSCKAYALLKGYRG

>OC43\_Nsp1

DAYDQVHDEPKGKFSKKAYALIRGYRG

Example alignment files for SARS-CoV-1 Nsp1 in the Grishin format:

## CoV1\_Nsp1 6zlw\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ELGTDPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zm7\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ----DPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zme\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ----DPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zmi\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ELGTDPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zmo\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ELGTDPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zmt\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ----DPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zn5\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ----DPYEDFQENWNTKHSSGVTRELMRELNGG

--

## CoV1\_Nsp1 6zoj\_small\_thread

#

scores\_from\_program: 0

0 ELGTDPIEDYEQNWNTKHGSGALRELTRELNGG

0 ELGTDPYEDFQENWNTKHSSGVTRELMRELNGG

--

**Hybridization:**

In this step, we wrote a RosettaScript to build a homology model for each Nsp1 C-terminal domain sequence as it may look when bound to the human ribosome. The inputs files were: the RosettaScript XML files, the Nsp1 fasta files, and the threaded template PDB models from the previous step. We generated 1500-2000 models for each Nsp1 sequence from the threaded structures.

Example command line for hybridization to build the SARS-CoV1 Nsp1 models:

/Users/anumglasgow/Rosetta/source/bin/rosetta\_scripts.macosclangrelease -database /Users/anumglasgow/Rosetta/database -in:file:fasta CoV1\_Nsp1.fasta -parser:protocol CoV1\_hybridize.xml -nstruct 10000 -relax:jump\_move true -relax:dualspace -out:suffix \_hyb -beta -default\_max\_cycles 2000

Example RosettaScript for hybridization to build the SARS-CoV1 Nsp1 models:

<ROSETTASCRIPTS>

<TASKOPERATIONS>

</TASKOPERATIONS>

<SCOREFXNS>

<ScoreFunction name="stage1"

weights="score3" symmetric="1">

<Reweight scoretype="atom\_pair\_constraint" weight="0.1"/>

</ScoreFunction>

<ScoreFunction name="stage2"

weights="score4\_smooth\_cart" symmetric="1">

<Reweight scoretype="atom\_pair\_constraint" weight="0.1"/>

</ScoreFunction>

<ScoreFunction name="fullatom"

weights="beta\_cart" symmetric="1">

<Reweight scoretype="atom\_pair\_constraint" weight="0.1"/>

</ScoreFunction>

</SCOREFXNS>

<FILTERS>

</FILTERS>

<MOVERS>

<Hybridize name="hybridize"

stage1\_scorefxn="stage1" stage2\_scorefxn="stage2"

fa\_scorefxn="fullatom" batch="1">

<Template pdb="6zlw\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zm7\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zme\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zmi\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zmo\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zmt\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zn5\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

<Template pdb="6zoj\_small\_thread.pdb"

weight="1.0" cst\_file="AUTO"/>

</Hybridize>

</MOVERS>

<PROTOCOLS>

<Add mover="hybridize"/>

</PROTOCOLS>

</ROSETTASCRIPTS>

**Modeling Nsp1-ribosome complexes:**

The 1500-2000 models for each Nsp1 were ranked according to total score using the Rosetta energy function. We selected the four models with the lowest total energies for each sequence. These Nsp1 models were then aligned to the experimentally-solved Nsp1 C-terminal domain fragment in the SARS-CoV-2 high-resolution PDB structure 6LZW using the *align* command in the protein modeling software PyMOL (Schrödinger). We then relaxed and scored the ribosome-Nsp1 complexes in Rosetta with backbone constraints, as described in the previous section, for a total of four models for each Nsp1-ribosome complex.

**Scoring Nsp1-ribosome complex.**

Each nsp1-ribosome complex model was scored using the Rosetta energy function with the *rna\_res\_level\_energy4* weights settings. We divided the scoring process into per residue energies total score, interface energy, and score type of the following rna terms: *fa\_elec\_rna\_phos\_phos, rna\_torsion, fa\_stac rna\_sugar\_close, hbond\_sr\_bb\_sc.* Additionally, we also scored the experimentally-solved SARS-CoV-2 high resolution PDB structure 6LZW.

Example Command for scoring nsp1-ribosome complex and interface.

python InterfaceAnalaysis\_run.py --pdb\_path inputs/COV1\_01.pdb --output\_dir output --target\_chain A

**Analysis:**

Characterizing the Nsp1 interaction.

We calculated interface energies of Nsp1 and the ribosomal proteins and RNA

Determining which Nsp1 residues contribute *most strongly* to the interaction with the ribosome.

Determining which Nsp1 residues make *similar or conserved* ribosomal interactions.

Validating modeled RNA interactions in the ribosome.