

TUTORIAL

ARTEMIS version 1.51

Bohdan, D. R., Bujnicki, J. M., & Baulin, E. F. (2024). **ARTEMIS: a method for topology-independent superposition of RNA 3D structures and structure-based sequence alignment.** *Nucleic Acids Research*, gkae758. DOI: [10.1093/nar/gkae758](https://doi.org/10.1093/nar/gkae758)

TABLE OF CONTENT

INTRODUCTION	1
Installation	1
Usage	2
Quick Start	2
USAGE EXAMPLES	4
1. Residue specifications	4
2. Superposition-only mode	5
3. Sub-optimal matches	6
4. Structure search	7
5. PDB/CIF conversion, formatting & cutting-out	8
ADVANCED USAGE	8
6. Modified nucleic acid residues & residue representations	8
7. Adjusting the algorithm parameters	9
SEE ALSO	10

INTRODUCTION

Installation

ARTEMIS is written in Python3 and implemented as a command-line application. To install ARTEMIS, download the repository from GitHub and install the dependencies:

```
git clone https://github.com/david-bogdan-r/ARTEMIS.git
```

```
cd ARTEMIS
```

```
pip install -r requirements.txt
```

ARTEMIS requires five libraries to be installed: *numpy*, *pandas*, *scipy*, *matplotlib*, *requests*. It was tested with two different Python3 environments:

1) Ubuntu 20.04

Distance table:

1ivs	dist	4p5j
1.C.G.901.	5.50	1.A.U.65.
1.C.G.902.	4.91	1.A.C.66.
1.C.G.903.	4.01	1.A.C.67.
1.C.C.904.	2.24	1.A.G.68.
1.C.G.905.	0.74	1.A.A.69.
1.C.G.906.	1.59	1.A.G.70.
1.C.C.907.	2.03	1.A.G.71.
1.C.A.909.	3.56	1.A.G.2.
1.C.G.910.	6.38	1.A.A.1.
1.C.C.911.	3.39	1.A.C.3.
1.C.U.912.	3.45	1.A.U.4.
1.C.C.913.	2.12	1.A.C.5.
1.C.A.914.	1.30	1.A.G.6.
1.C.G.915.	2.04	1.A.C.7.
1.C.C.916.	5.45	1.A.C.8.
1.C.G.917.	3.33	1.A.A.9.
1.C.G.918.	4.25	1.A.G.10.
1.C.A.919.	6.81	1.A.U.11.
1.C.A.920.	1.98	1.A.G.14.
1.C.G.921.	3.96	1.A.G.16.
1.C.A.922.	2.34	1.A.A.17.
1.C.G.923.	0.60	1.A.G.18.
1.C.C.924.	1.25	1.A.G.19.
1.C.G.925.	0.67	1.A.U.20.
1.C.C.926.	0.64	1.A.C.21.
1.C.U.927.	1.14	1.A.U.22.
1.C.C.928.	1.04	1.A.G.23.
1.C.G.929.	0.97	1.A.U.24.
1.C.C.930.	2.22	1.A.C.25.
1.C.A.936.	2.02	1.A.C.32.
1.C.G.938.	1.82	1.A.G.33.
1.C.C.939.	1.40	1.A.A.34.
1.C.G.940.	1.57	1.A.C.35.
1.C.A.941.	1.68	1.A.A.36.
1.C.G.942.	0.25	1.A.G.37.
1.C.A.943.	1.82	1.A.A.38.
1.C.G.944.	2.83	1.A.U.39.
1.C.G.945.	2.96	1.A.A.40.
1.C.U.946.	2.85	1.A.A.41.
1.C.C.947.	6.81	1.A.U.42.
1.C.G.948.	1.40	1.A.G.72.
1.C.U.949.	2.72	1.A.C.43.
1.C.G.951.	3.13	1.A.G.44.
1.C.G.952.	3.36	1.A.G.45.
1.C.U.953.	3.74	1.A.U.47.
1.C.U.954.	2.25	1.A.G.48.
1.C.C.955.	1.80	1.A.C.49.
1.C.A.956.	1.69	1.A.A.50.
1.C.A.957.	0.86	1.A.A.51.
1.C.G.958.	1.28	1.A.C.52.
1.C.U.959.	1.47	1.A.U.53.
1.C.C.960.	2.92	1.A.C.54.
1.C.C.961.	3.76	1.A.C.55.
1.C.U.962.	3.44	1.A.C.56.
1.C.A.963.	2.02	1.A.G.57.
1.C.C.964.	1.38	1.A.C.58.
1.C.G.965.	1.94	1.A.C.59.
1.C.C.966.	2.51	1.A.C.60.
1.C.C.967.	3.46	1.A.C.61.
1.C.G.968.	4.44	1.A.C.77.
1.C.C.969.	3.91	1.A.G.79.
1.C.C.970.	4.82	1.A.A.80.
1.C.C.971.	7.60	1.A.A.81.

#Total CPU time is 2.86 seconds

Additionally, ARTEMIS should save six files to the working directory (saveto=.):

4p5j_to_1ivs.pdb	- the query structure in PDB format superimposed to the reference according to the sequentially-ordered alignment;
4p5j_to_1ivs.png	- 2D plot of the sequentially-ordered alignment;
4p5j_to_1ivs.tsv	- list of the matched residues according to the sequentially-ordered alignment;
4p5j_to_1ivs_ti.pdb	- the query structure in PDB format superimposed to the reference according to the topology-independent alignment;
4p5j_to_1ivs_ti.png	- 2D plot of the topology-independent alignment;
4p5j_to_1ivs_ti.tsv	- list of the matched residues according to the topology-independent alignment;

USAGE EXAMPLES

1. Residue specifications

ARTEMIS allows users to specify which residues should be considered as part of input structures, which residues should be ignored, and which residues should be saved as the superimposed query structure and in what format:

```
python artemis.py r=6ugg rresneg=#1/B q=1ivs qres=/C  
saveres=/A saveformat=pdb saveto=result
```

In this example, ARTEMIS superimposes one of the two tRNA(Val) chains from PDB entry 1IVS (q=1ivs qres=/C, chain C is considered as the query structure, and chain D is ignored) to one of the two tRNA(Asp) chains from PDB entry 6UGG (r=6ugg rresneg=#1/B, chain A is considered as the reference structure, and chain B is ignored). Then, ARTEMIS saves only the tRNA(Val) synthetase protein molecule from 1IVS (saveres=/A) as resulted from the alignment of the two RNA molecules. The superimposed protein molecule is saved in .pdb format under the “result” subfolder of the working directory (saveformat=pdb saveto=result).

The list of residue specification parameters and their example values:

rres=/A	- reference residues to select, chain A;
rresneg=#2	- reference residues to ignore, model 2;

qres=#1:_1_100 - query residues to select, model 1, residues 1 to 100;
qresneg=":G :A" - query residues to ignore, all guanines and adenines;
saveres=:_10B - query residues to save, residue 10 with ins.code B.

The details of the ChimeraX-like residue specification format:

[#[INT]][/[STRING]][:[STRING][_INT[CHAR|_INT]]]

This is the regex pattern of the residue specification keyword. Each specification parameter can be expressed as a quoted space-separated list of keywords (e.g., **qresneg=":G :A"**). Insertion code can be specified only for a single residue, not for a range. The components:

#[INT]	- Model number
/[STRING]	- Chain identifier
:[STRING][_INT[CHAR _INT]]	- Residue(s) specification:
:STRING	- Residue type
:_INT[CHAR]	- Residue number [with insertion code]
:_INT_INT	- Range of residue numbers

2. Superposition-only mode

When the two input structures have identical lengths and the perfect sequence alignment should be assumed between them (for example when comparing two models of the same nucleic acid molecule), the user can run ARTEMIS in the superposition-only mode (-superonly):

```
python artemis.py r=7sam q=7scq -superonly
```

The output of this command should look like this:

```
*****
* ARTEMIS (Version 1.5) *
* using ARTEM to Infer Sequence alignment *
* Reference: 10.1093/nar/gkx758 *
* Please email comments and suggestions to dav.bog.rom@gmail.com *
*****

Name of structure r: 7sam:A
Name of structure q: 7scq:C (to be superimposed onto structure r)
Length of structure r: 169 residues
Length of structure q: 169 residues
```

Aligned length= 169, RMSD= 14.07, Seq_ID=n_identical/n_aligned= 1.000
 TM-score= 0.63014 (normalized by length of structure r: L=169, d0=5.29)
 TM-score= 0.63014 (normalized by length of structure q: L=169, d0=5.29)

(":" denotes residue pairs of $d < 5.0$ Angstrom, "." denotes other aligned residues)

```
CGUGGUUGACACGCAGACCUCUACAAGAGUGUCUAGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAA
GUCUUAGAAUGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGACCA
.....
:.....
CGUGGUUGACACGCAGACCUCUACAAGAGUGUCUAGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAA
GUCUUAGAAUGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGACCA
```

#Total CPU time is 0.06 seconds

3. Sub-optimal matches

ARTEMIS can report non-overlapping alternative matches of the smaller query structure in the larger reference structure:

```
python artemis.py r=486d q=1euy addhits=0.5
```

In this example, ARTEMIS is asked to report all matches of the tRNA molecule from PDB entry 1EUY, chain B within the reference structure from PDB entry 486D that have the query TM-score ≥ 0.5 (addhits=0.5). The entry 486D contains three tRNA chains: A, C, and E. If $0 < \text{addhits} \leq 1$, the parameter is treated as the minimum query TM-score threshold. If $\text{addhits} > 1$, the parameter is treated as the maximum number of alternative matches to report. In this example $\text{addhits}=0.5$ and $\text{addhits}=2$ will produce identical results: one optimal match of 66 residues with chain C and two sub-optimal matches of 65 and 62 residues with chains E and A respectively:

R	Q	ID	RL	QL	TYPE	LALI	RMSD	RTM	QTM	MATCH
486d	1euy	0	332	73	SEQORD	66	2.61	0.18408	0.60728	
1.C.C.2.=1.B.G.902.,1.C.G.3.=1.B.G.903.,1.C.G.4.=1.B.G.904.,1.C.A.5.=1.B.G.905.,1.C.U.6.=1.B.U.906.,1.C.U.7.=1.B.A.907.,1.C.U.8.=1.B.U.908.,1.C.A.9.=1.B.C.909.,1.C.2MG.10.=1.B.G.910.,1.C.C.11.=1.B.C.911.,1.C.U.12.=1.B.C.912.,1.C.C.13.=1.B.A.913.,1.C.A.14.=1.B.A.914.,1.C.G.15.=1.B.G.915.,1.C.G.18.=1.B.G.918.,1.C.G.19.=1.B.G.919.,1.C.G.20.=1.B.U.920.,1.C.A.21.=1.B.A.921.,1.C.G.22.=1.B.A.922.,1.C.A.23.=1.B.G.923.,1.C.G.24.=1.B.G.924.,1.C.C.25.=1.B.C.925.,1.C.C.27.=1.B.C.927.,1.C.C.28.=1.B.C.928.,1.C.C.29.=1.B.G.929.,1.C.G.30.=1.B.G.930.,1.C.C.31.=1.B.A.931.,1.C.PSU.32.=1.B.U.932.,1.C.U.33.=1.B.U.933.,1.C.G.34.=1.B.C.934.,1.C.U.35.=1.B.U.935.,1.C.C.36.=1.B.G.936.,1.C.C.38.=1.B.A.937.,1.C.G.39.=1.B.U.938.,1.C.U.40.=1.B.U.939.,1.C.G.41.=1.B.C.940.,1.C.G.42.=1.B.G.942.,1.C.G.43.=1.B.G.943.,1.C.A.44.=1.B.C.944.,1.C.G.45.=1.B.A.945.,1.C.7MG.46.=1.B.A.947.,1.C.C.48.=1.B.G.948.,1.C.5MC.49.=1.B.C.949.,1.C.U.50.=1.B.G.950.,1.C.G.51.=1.B.A.951.,1.C.U.52.=1.B.G.952.,1.C.G.53.=1.B.G.953.,1.C.5MU.54.=1.B.U.954.,1.C.PSU.55.=1.B.U.955.,1.C.C.56.=1.B.C.956.,1.C.G.57.=1.B.G.957.,1.C.1MA.58.=1.B.A.958.,1.C.U.59.=1.B.A.959.,1.C.C.60.=1.B.U.960.,1.C.C.61.=1.B.C.961.,1.C.A.62.=1.B.C.962.,1.C.C.63.=1.B.U.963.,1.C.A.64.=1.B.C.964.,1.C.G.65.=1.B.G.965.,1.C.A.66.=1.B.U.966.,1.C.A.67.=1.B.A.967.,1.C.U.68.=1.B.C.968.,1.C.U.69.=1.B.C.969.,1.C.C.70.=1.B.C.970.,1.C.G.71.=1.B.C.971.,1.C.C.72.=1.B.A.972.										
486d	1euy	1	266	73	SEQORD	65	2.92	0.21980	0.60861	
1.E.U.1.=1.B.G.902.,1.E.C.2.=1.B.G.903.,1.E.G.4.=1.B.G.904.,1.E.U.5.=1.B.G.905.,1.E.G.6.=1.B.U.906.,1.E.A.7.=1.B.A.907.,1.E.U.8.=1.B.U.908.,1.E.A.9.=1.B.C.909.,1.E.G.10.=1.B.G.910.,1.E.U.11.=1.B.C.911.,1.E.U.12.=1.B.C.912.,1.E.PSU.13.=1.B.A.913.,1.E.A.14.=1.B.A.914.,1.E.A.15.=1.B.G.915.,1.E.G.17.=1.B.G.918.,1.E.G.18.=1.B.G.919.,1.E.C.20.=1.B.U.920.,1.E.A.21.=1.B.A.921.,1.E.G.22.=1.B.A.922.,1.E.A.23.=1.B.G.923.,1.E.A.24.=1.B.G.924.,1.E.U.25.=1.B.C.925.,1.E.G.26.=1.B.A.926.,1.E.G.27.=1.B.C.927.,1.E.G.28.=1.B.C.928.,1.E.C.29.=1.B.G.929.,1.E.G.30.=1.B.G.930.,1.E.C.31.=1.B.A.931.,1.E.PSU.32.=1.B.U.932.,1.E.U.33.=1.B.U.933.,1.E.G.34.=1.B.C.934.,1.E.U.35.=1.B.U.935.,1.E.C.36.=1.B.G.936.,1.E.C.38.=1.B.U.938.,1.E.G.39.=1.B.U.939.,1.E.U.40.=1.B.C.940.,1.E.G.41.=1.B.C.941.,1.E.C.42.=1.B.G.942.,1.E.C.43.=1.B.G.943.,1.E.A.44.=1.B.C.944.,1.E.G.45.=1.B.A.945.,1.E.A.46.=1.B.A.947.,1.E.U.48.=1.B.G.948.,1.E.5MC.49.=1.B.C.949.,1.E.G.50.=1.B.G.950.,1.E.G.51.=1.B.A.951.,1.E.G.52.=1.B.G.953.,1.E.C.53.=1.B.G.954.,1.E.C.54.=1.B.U.955.,1.E.C.55.=1.B.U.956.,1.E.C.56.=1.B.C.957.,1.E.C.57.=1.B.G.958.,1.E.C.58.=1.B.A.959.,1.E.C.59.=1.B.U.960.,1.E.C.60.=1.B.C.961.,1.E.C.61.=1.B.C.962.,1.E.C.62.=1.B.U.963.,1.E.C.63.=1.B.C.964.,1.E.C.64.=1.B.A.965.,1.E.C.65.=1.B.G.966.,1.E.C.66.=1.B.U.967.,1.E.C.67.=1.B.A.968.,1.E.C.68.=1.B.C.969.,1.E.C.69.=1.B.G.970.,1.E.C.70.=1.B.C.971.,1.E.C.71.=1.B.C.972.,1.E.C.72.=1.B.A.973.										

52., 1.E.G.53.=1.B.G.953., 1.E.5MU.54.=1.B.U.954., 1.E.PSU.55.=1.B.U.955., 1.E.C.56.=1.B.C.956., 1.E.A.57.=1.B.G.957., 1.E.A.58.=1.B.A.958., 1.E.U.59.=1.B.A.959., 1.E.U.60.=1.B.U.960., 1.E.C.61.=1.B.C.961., 1.E.C.62.=1.B.C.962., 1.E.C.63.=1.B.U.963., 1.E.C.64.=1.B.C.964., 1.E.G.65.=1.B.G.965., 1.E.U.66.=1.B.U.966., 1.E.C.67.=1.B.C.968., 1.E.G.68.=1.B.C.969., 1.E.C.69.=1.B.C.970., 1.E.G.70.=1.B.C.971.

486d	1euy	2	201	73	SEQORD	62	2.90	0.26461	0.55625																																																				
1.A.U.1.=1.B.G.902.,	1.A.C.3.=1.B.G.903.,	1.A.G.4.=1.B.G.904.,	1.A.U.5.=1.B.G.905.,	1.A.G.6.=1.B.U.906.,	1.A.A.7.=1.B.A.907.,	1.A.U.8.=1.B.U.908.,	1.A.A.9.=1.B.C.909.,	1.A.G.10.=1.B.G.910.,	1.A.U.11.=1.B.C.911.,	1.A.U.12.=1.B.C.912.,	1.A.PSU.13.=1.B.A.913.,	1.A.A.14.=1.B.A.914.,	1.A.A.15.=1.B.G.915.,	1.A.G.17.=1.B.G.918.,	1.A.G.18.=1.B.G.919.,	1.A.C.20.=1.B.U.920.,	1.A.A.21.=1.B.A.921.,	1.A.G.22.=1.B.A.922.,	1.A.A.23.=1.B.G.923.,	1.A.A.24.=1.B.G.924.,	1.A.U.25.=1.B.C.925.,	1.A.C.27.=1.B.C.927.,	1.A.C.28.=1.B.C.928.,	1.A.A.29.=1.B.G.929.,	1.A.G.30.=1.B.G.930.,	1.A.A.31.=1.B.U.932.,	1.A.OMC.32.=1.B.U.933.,	1.A.U.33.=1.B.U.935.,	1.A.A.36.=1.B.G.936.,	1.A.A.38.=1.B.A.937.,	1.A.U.39.=1.B.U.938.,	1.A.5MC.40.=1.B.U.939.,	1.A.U.41.=1.B.C.940.,	1.A.G.42.=1.B.C.941.,	1.A.G.43.=1.B.G.942.,	1.A.A.44.=1.B.G.943.,	1.A.G.45.=1.B.A.945.,	1.A.A.46.=1.B.A.947.,	1.A.U.48.=1.B.G.948.,	1.A.5MC.49.=1.B.C.949.,	1.A.G.50.=1.B.G.950.,	1.A.G.51.=1.B.A.951.,	1.A.G.52.=1.B.G.952.,	1.A.G.53.=1.B.G.953.,	1.A.5MU.54.=1.B.U.954.,	1.A.PSU.55.=1.B.U.955.,	1.A.C.56.=1.B.C.956.,	1.A.A.57.=1.B.G.957.,	1.A.A.58.=1.B.A.958.,	1.A.U.59.=1.B.A.959.,	1.A.U.60.=1.B.U.960.,	1.A.C.61.=1.B.C.961.,	1.A.C.62.=1.B.C.962.,	1.A.C.63.=1.B.U.963.,	1.A.C.64.=1.B.C.964.,	1.A.G.65.=1.B.G.965.,	1.A.U.66.=1.B.U.966.,	1.A.C.67.=1.B.A.967.,	1.A.G.68.=1.B.C.969.,	1.A.C.69.=1.B.C.970.,	1.A.G.70.=1.B.C.971.

Legend:

- R** - Reference structure file;
- Q** - Query structure file;
- ID** - Iteration ID (starting from 0);
- RL** - Reference structure length (number of residues);
- QL** - Query structure length (number of residues);
- TYPE** - Sequentially-ordered (*SEQORD*) / Topology-independent (*TOPIND*) alignment;
- LALI** - Match length L_{ali} (number of residues);
- RMSD** - Match RMSD (calculated for C3' atoms);
- RTM** - Reference structure TM-score_{RNA};
- QTM** - Query structure TM-score_{RNA};
- MATCH** - Comma-separated list of matching residues (<*r-residue*>=<*q-residue*>).

4. Structure search

ARTEMIS can be used for screening a structure of interest against a set of structures, for example for automated database searches for similar folds:

```
python artemis.py r=DBfolder/ q=struct_filename -silent -tsv
> matches.tsv
```

Here, ARTEMIS will run a search of <*struct_filename*> query structure (q=struct_filename) against all coordinate files within the <*DBfolder*> directory (r=DBfolder/). The found matches will be reported in the tsv format (-tsv, see Section 3 for the format details) and saved into a file (> matches.tsv). If ARTEMIS crashes on a particular pair of structures, it will print an error message into *stderr* and proceed to the next structure without stopping the entire search (-silent). If the query structure is

larger than the average structure in the database, it's advised to swap reference and query inputs to improve the screening time (q=DBfolder/ r=struct_filename).

5. PDB/CIF conversion, formatting & cutting-out

ARTEMIS is a handy tool to cut out residues of interest from the coordinate files, convert between PDB and mmCIF formats, and “fix” non-canonical PDB/CIF format variants. For example, ARTEMIS automatically turns “C3*”, “O1P”, and “O2P” nucleotide atoms into their more common variants: “C3”, “OP1”, and “OP2”. The PDB-formatted output provided by ARTEMIS is readily acceptable for CASP and RNA-Puzzles submissions. Furthermore, ARTEMIS automatically downloads and saves nucleic acid structures from PDB entries in the preferred format:

```
python artemis.py r=8uo6 q=8uo6 -superonly -notmopt saveto=.  
saveres=/B saveformat=cif
```

In this example ARTEMIS downloads the PDB entry 8UO6 as both the reference and the query structure (r=8uo6 q=8uo6), trivially superimposes it onto itself via the Kabsch algorithm, i.e. without trying to maximize the TM-score_{RNA} (-superonly -notmopt), and saves the chain B only (saveres=/B) in mmCIF format (saveformat=cif) under the working directory (saveto=.).

ADVANCED USAGE

6. Modified nucleic acid residues & residue representations

Currently, ARTEMIS recognizes the following standard and modified residues:

A, C, G, U;

DA, DC, DG, DT;

ADP, CDP, GDP, UDP, ATP, CTP, GTP, UTP;

1MA, 2MG, 3TD, 4AC, 4SU, 5MC, 5MU, 6MZ, 7MG, A2M, B8N, LV2, MA6, OMC, OMG, OMU, PSU, SAM, SSU, UR3, UY1, YYG.

To extend this list, users may add the representations of new residues into two JSON files:

1) **src/resrepr/artemis_1.json** - 3-atom residue representations used for seed superpositions. Example line:

"A": **["OP1 O5'", "C2' C4'", "N9"]** - By default, ARTEMIS uses a center of mass of OP1 and O5' atoms as a pseudo atom for phosphate, center of mass of C2' and C4' atoms as a pseudo atom for ribose, and a glycosidic base atom (N9 for purines, N1 for pyrimidines) as a base atom.

2) **src/resrepr/artemis_2.json** - 1-atom residue representations used for RMSD and TM-score_{RNA} calculations. Example line:

"A": **["C3"]** - By default, ARTEMIS uses C3' atoms as a 1-atom representation of residues.

The default residue representations may be changed to suit the users' needs. For example, it's possible to extend the representations with amino acids to make ARTEMIS applicable for the alignment of proteins and nucleic acid-protein complexes.

7. Adjusting the algorithm parameters

In some cases, it may be helpful to tweak the core parameters of the ARTEMIS algorithm. Here are these parameters and their default values when the query structure is less than 500 residues:

```
python artemis.py r=<ref> q=<query> matchrange=3.5 shift=3  
stepdiv=0 toplargest=<query_size> threads=CPU_COUNT
```

When the query structure is 500 residues or more:

```
python artemis.py r=<ref> q=<query> matchrange=3.5 shift=20  
stepdiv=100 toplargest=<2*threads> threads=CPU_COUNT
```

Details:

matchrange - a distance threshold in angstroms. A higher matchrange value will produce more "noisy" matchings but won't miss anything. A lower matchrange value will produce more "clean" matchings but can miss something.

shift - a distance threshold in angstroms. A higher value leads to a higher coverage of the sequentially-ordered alignment.

stepdiv - this parameter is used to speed up the procedure for large structures. If stepdiv = 0, ARTEMIS will try all reference residues as matching seeds, otherwise ARTEMIS will try only <stepdiv> reference residues as matching seeds. For example, stepdiv = 100 means that 100 residues (uniformly distributed over the reference sequence) will be tried as a matching seed. Thus, lower positive values lead to faster but possibly less accurate alignments.

toplargest - number of largest mutually closest residue sets (i.e. hits) for which ARTEMIS will try to build an alignment. Higher values lead to slower performance but a lower chance of missing the optimal alignment.

threads - number of CPUs to be used by ARTEMIS. By default, ARTEMIS uses all of the machine's CPUs.

SEE ALSO

[SQUARNA](#) - an RNA secondary structure prediction method based on a greedy stem formation model. DOI: [10.1101/2023.08.28.555103](https://doi.org/10.1101/2023.08.28.555103)

[ARTEM](#) - a tool for RNA tertiary motif search. DOI: [10.1093/nar/gkad605](https://doi.org/10.1093/nar/gkad605)

[urslib2](#) - a Python library for processing RNA structure data from PDB/mmCIF files & DSSR annotations. DOI: [10.1093/nar/gkad605](https://doi.org/10.1093/nar/gkad605)

[LORA](#) - a dataset of long-range RNA 3D modules. DOI: [10.1093/nar/gkad605](https://doi.org/10.1093/nar/gkad605)

[ARTEM-KT](#) - a dataset of kink-turn-like RNA 3D modules. DOI: [10.1101/2024.05.31.596898](https://doi.org/10.1101/2024.05.31.596898)