

incaRNAfbinv: a web server for the fragment-based design of RNA sequences

Matan Drory retwitzer, Vladimir Reinharz, Yann Ponty, Jérôme Waldispühl,
Danny Barash

► **To cite this version:**

Matan Drory retwitzer, Vladimir Reinharz, Yann Ponty, Jérôme Waldispühl, Danny Barash. incaRNAfbinv: a web server for the fragment-based design of RNA sequences. *Nucleic Acids Research*, Oxford University Press, 2016, 44 (W1), pp.W308 - W314. 10.1093/nar/gkw440 . hal-01319682

HAL Id: hal-01319682

<https://hal.inria.fr/hal-01319682>

Submitted on 22 May 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

incaRNAfbinv: a webserver for the fragment-based design of RNA sequences

Matan Drory Retwitzer¹, Vladmir Reinhartz², Yann Ponty^{3,4}, Jérôme Waldispühl², Danny Barash^{1*}

¹ Department of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel; ² School of Computer Science & McGill Centre for Bioinformatics, McGill University, Montréal, QC H3A 0E9, Canada; ³ Laboratoire d'Informatique (LIX) – CNRS UMR 7161, École Polytechnique, 91128 Palaiseau, France; and ⁴ AMIB team/project, INRIA Saclay, Batiment Alan Turing, 91128 Palaiseau, France

Received ??; Revised ??; Accepted ??

ABSTRACT

In recent years, new methods for computational RNA design have been developed and applied to various problems in synthetic biology and nanotechnology. Lately, there is considerable interest in incorporating essential biological information when solving the inverse RNA folding problem. Correspondingly, RNAfbinv aims at including biologically meaningful constraints and is the only program to-date that performs a fragment-based design of RNA sequences. In doing so it allows the design of sequences that do not necessarily exactly fold into the target, as long as the overall coarse-grained tree graph shape is preserved. Augmented by the weighted sampling algorithm of incaRNAtion, our web server called incaRNAfbinv implements the method devised in RNAfbinv and offers an interactive environment for the inverse folding of RNA using a fragment-based design approach. It takes as input: a target RNA secondary structure; optional sequence and motif constraints; optional target minimum free energy, neutrality, and GC content. In addition to the design

of synthetic regulatory sequences, it can be used as a pre-processing step for the detection of novel natural occurring RNAs. The two complementary methodologies RNAfbinv and incaRNAtion are merged together and fully implemented in our web server incaRNAfbinv, available at <http://www.cs.bgu.ac.il/incaRNAfbinv>.

Keywords: RNA inverse folding, RNA fragment-based design, RNA folding prediction

INTRODUCTION

The design of RNAs with favorable traits is a promising endeavor that can be viewed as part of growing efforts in synthetic biology (1), as well as other applications. For example, it can be used to enhance the search for particular RNAs such as ribozymes and riboswitches in sequenced genomes (2), as well as other non-coding RNAs that may act as regulators of disease (3) or participate in catalysis (4). For riboswitches (5, 6), aside of the classical problem of computationally designing transcription regulators and validating them experimentally (7, 8) to complement pure experimental designs (9–11), the inverse RNA folding problem that was initially formulated

*To whom correspondence should be addressed. Email: dbarash@cs.bgu.ac.il

© ?? The Author(s)

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/2.0/uk/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

and addressed in (12) can be used as a pre-processing step before BLAST for riboswitch identification (13). This recent use was also worked out for IRES-like structural subdomain identification in (14). It has potential to advance the field described in (15) for conserved RNAs in general.

Thus, computational RNA design is of increasing biological importance. Since the first program for solving the inverse RNA folding problem (or RNA design) called *RNAinverse* was put forth in (12), several other programs were developed. The approach to solve it by stochastic optimization relies on the solution of the direct problem using software available in RNA folding prediction web servers, e.g. the *RNAfold* server (16) or *mfold* (17), by performing energy minimization with thermodynamic parameters (18). Initially, a seed sequence is chosen, after which a local search strategy is used to mutate the seed and apply repeatedly the direct problem of RNA folding prediction by energy minimization. Then, in the vicinity of the seed sequence, a designed sequence is found with desired folding properties according to the objective function in the optimization problem formulation.

In recent years, several programs for RNA design have been developed with the goal of offering added features with respect to the original *RNAinverse*, most of which are centralized at the level of RNA secondary structure (19–27) and a few at other hierarchical levels in structure (28, 29). Recently, an extension to the problem was gradually developed (30–32) that allows designing sequences that fold into a prescribed shape, leaving some flexibility in the secondary structure of RNA motifs that do not necessarily possess a known functional role. This extension, when offering a fragment selection to the user, is called "fragment-based" design because it is based on a user-selected secondary structure motif (the fragment) that possesses a functional role and is therefore inserted as a "fragment-based" constraint to the design problem.

The shape of the RNA can be represented as a tree-graph (33) that groups together a family of RNA secondary structures, all belonging to the same coarse grained graphical representation.

The aforementioned extension led to a unique inverse RNA folding program called *RNAfbinv* (32) that **to the best of our knowledge** is more general in scope than any existing program in its shape-based approach. In that regard a shape-based approach is more general than a structure-based approach by allowing more designed sequences as solutions to the design problem, although other generalizations like pseudoknot inclusion that can be found in several other programs have not yet been implemented in our program. In parallel, controlling nucleotide distribution in RNA design problems was addressed in our presented webserver in a unique way by a weighted sampling approach (34). The approach is of general importance for the future of inverse RNA folding because instead of a random start performing a local search, the initial sequence for performing the iterative procedure of solving the inverse problem is carefully picked by using global considerations in a guided manner in search space. The program for the weighted sampling approach called *incaRNA*-ation has so far only been exemplified in (34) for RNA design. The *incaRNAfbinv* web server described herein is a merge between *RNAfbinv* (32) and *incaRNA*ation (34). It offers design solutions that **to the best of our knowledge** are not available in neither the most recent programs for RNA design that are considered state-of-the art, namely *antaRNA* (26) and *RNAiFold* (27), nor any other such program that was devised since the seminal program called *RNAinverse* from the Vienna RNA package (12) was put forth. It should be noted that *incaRNAfbinv* relies on other programs aside of *RNAinverse* that are available in the Vienna RNA package such as *RNAfold* that solves a direct problem at each iteration and *RNAdistance*.

Moreover, pseudoknots have not yet been implemented in our designed program, as well as some experimental constraints, such as avoiding transcription slippage in the case of consecutive G nucleotides. In the future, it will be desired to add these features and others of experimental type to our program, as more experimental results with designed sequences obtained from our program are accumulated.

In the following sections, the *incaRNAfbinv* web server and its method are described.

WEBSERVER

The *incaRNAfbinv* webserver is available at <http://www.cs.bgu.ac.il/incaRNAfbinv>. It runs on a Unix Lenovo system x3650 M5 server with Dual Intel(R) Xeon(R) CPU E5-2620 v3 2.4GHz processors containing six logical cores and 15MB L3 cache each.

The backend is written in Java EE and run on Tomcat 8. It dispatches design tasks responsible for running *incaRNAtion* (34) and *RNAfbinv* (32).

Every design task runs on up to four cores depending on load. The server runs up to ten simultaneous design tasks while the rest wait in a queue.

The frontend is designed using the Bootstrap css framework. Webpages are generated using JSP and JSTL. They utilize JavaScript, JQuery, JSON and ajax.

Input

The input screen of the *incaRNAfbinv* webserver is shown in Figure 1. Initially, the user provides a query combining an input RNA secondary structure in dot-bracket notation, along with optional constraints and parameters. Sequence constraints are expressed using the IUPAC sequence notation. The motif selection constraint, one of the unique features that sets *incaRNAfbinv* apart from other inverse RNA folding programs, is presented on the right with a question mark until the user inserts the input RNA secondary structure. Once the input structure is available, the question mark is replaced by the secondary structure drawing, and in the motif selection box the user can specify which motif should be preserved exactly. For illustration, in the guanine-binding riboswitch aptamer example, taken from RFAM (35) and used for sequence design in (13), the *multi-loop M13* may be selected from the drop-down menu since the ligand is known to bind within the multi-loop motif. *incaRNAtion* is the default option for the seed generation method, and leads to the execution of the weighted sampling algorithm (34). Alternative options include a random initial guess or a user defined sequence, as described in the original *RNAfbinv* (32). In combination

with the *incaRNAtion* option, the user is allowed to specify a targeted GC content (default=50%). Further options can be accessed by checking the *Show Advanced Options* box, where the target minimum free energy (kcal/mol) and target mutational robustness (neutrality between 0 and 1) can be specified. Optional parameters include the number of simulated annealing iterations (default value is set to 1000) and the number of output sequences (default value is set to 20), along with an email address and query name which can be specified to locate the job later on. The user validates the task by clicking the *Submit job* button when the form is complete.

Results are sent by email if specified, otherwise the results are available upon completion in an interactive *job* mode. Aside of the *Design Form* page, a *Search Result* page is available in the top menu, should the user wish to re-analyze a previously-computed result, using its corresponding query name or identification. A general *Help* page is also available, as well as contextual tooltips that provide brief explanations for each field.

Output

The results can be accessed through the web link provided to the user, and are guaranteed to be accessible for at least a week following their generation. In addition to keeping the web link for later use, the user has an option to download the results in excel format for further analysis.

After the example parameters in the input screen of Figure 1 are inserted and the form is submitted, the main results screen appearing in Figure 2 is obtained. The query structure and associated sequence constraint appears at the top of the page. Below it are filtering options of the results displayed and further below is a table with a list of results. The table contains all the designed sequences that were generated. Each row provides a designed sequence result and its folded predicted structure in dot-bracket notation (12), its Shapiro tree-graph representation (33), minimum free energy in kcal/mol (calculated using *RNAeval* from the Vienna RNA package, according to the Turner energy model, 2004 (18)),

The neutrality of an RNA sequence of length L is calculated by the formula $\langle (L - d)/L \rangle$, where d is the base-pair distance between the secondary structure of the original sequence and the secondary structure of the mutant, averaged over all $3L$ one-mutant neighbors. The base-pair distance is evaluated by the `RNAdistance` program from the Vienna RNA package.

The minimum free energy ΔG is for measuring thermodynamic stability in kcal/mol, therefore a unity weight is assigned. All distances between secondary structures are calculated using `RNAdistance` in the Vienna RNA package (12) (supporting both the coarse-grain tree graphs called the Shapiro representation (33), and the dot bracket representation of the secondary structure). For the tree edit distance between Shapiro representations, a relatively large weight of 100 is chosen for shape preservation, while for the base pair distance in the last term, a very small weight of 10^{-2} is assigned. This last term is the one used in the original `RNAinverse` (12) for preserving the exact secondary structure and its purpose is to protect the solutions from being over-dominated by shape. As explained in (31, 32), shape preservation that is controlled by the term with the weight of 100 (minimizing distances between shapes) is an important aspect of our method, allowed with a flexible RNA inverse folding solver. `RNAfbinv` uses simulated annealing with a four-nucleotide look ahead local search function.

IncaRNAtion

`incaRNAtion` (34) addresses RNA design in a complementary way. Rather than preventing the formation of alternative secondary structures (negative design principle), it stochastically produces sequences having high affinity towards the target structure S^* , as measured by its free-energy (positive design principle). To that purpose, a *pseudo-Boltzmann distribution* is postulated on the set of sequences compatible with S^* , where the probability of emitting an RNA w for a given pseudo-temperature T is proportional to $e^{-E_w(S^*)/kT}$, where $E_w(S^*)$ is the free-energy of S^* upon an RNA sequence w , and k is the suitably-dimensioned Boltzmann

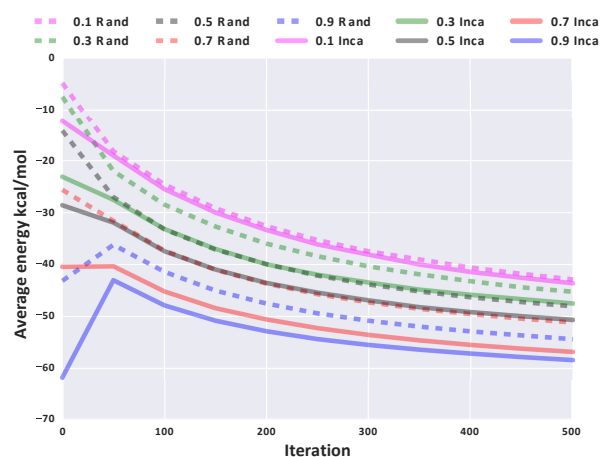


Figure 4. Comparison of the free-energies of candidate solutions along the execution of `RNAfbinv`, for targeted GC contents from 0.1 to 0.9 and using `incaRNAtion` (solid lines) and the default random initialization of `RNAfbinv` (dashed lines), for the design of a guanine-binding riboswitch aptamer. Values averaged over 1000 runs.

constant. A linear-time dynamic programming algorithm is then used to generate sequences at random exactly from the pseudo-Boltzmann distribution, resulting in candidate designs whose affinity towards S^* ranges from extreme to reasonable, depending on the value of T . Further terms can be incorporated in the free-energy function, and combined with a provably-efficient rejection step, to control the GC-content of produced sequences.

Preliminary analyses (34) revealed that `incaRNAtion` produces sequences that are more diverse than those obtained using competing algorithms. Furthermore, it was shown that sequences designed by `incaRNAtion` could be used as seeds for algorithms implementing negative design principles, increasing the diversity of their final output, while generally retaining the general properties (high-affinity, prescribed GC-content...) enforced by `incaRNAtion` in its initial generation. Figure 4 illustrates the general robustness of this good behavior in combination with `RNAfbinv`, by showing that sequences generated by `incaRNAtion` not only have lower free-energy than those produced by the default initialization of `RNAfbinv`, but that this higher stability persists across its iterative improvements.

CONCLUSION

When solving the inverse RNA folding problem, it is important to be able to address biological constraints in the forms of structural constraints, as well as physical observables and sequence constraints. New programs that were recently developed such as antaRNA (26) and RNAiFold (27) attempt to address these constraints but they are yet limited in their scope and cannot handle fragment-based constraints like the ones handled in RNAfbinv (32) or GC-content like in the structured and efficient way it is handled in incaRNAtion (34). These types of constraints can substantially improve targeted design of RNA sequences in the cases when such a biological-driven constraint is known in advance. The uniqueness of the fragment-based design approach together with the weighted sampling approach that traverses the search space in a guided manner merits a user-friendly webserver that can accommodate practitioners of various backgrounds. We present a new webserver called incaRNAfbinv that fulfills this need. It is based on the methodologies implemented in the programs (none of which is available as a webserver) called RNAfbinv (32) and incaRNAtion (34). It offers a significant extension to programs performing RNA design that do not consider the aforementioned advanced constraints and are limited to strictly obeying the RNA secondary structure of the input as in the original and well-accustomed formulation of RNAinverse (12), even if a small deviation from it can produce a designed sequence that can much better accommodate the biological constraint imposed based on prior knowledge.

It should be noted that the allowed flexibility of the fragment-based design approach may also introduce spurious solutions that could be more noticeable in specific cases. Some of these issues could be remedied in the future, especially when more practical experience is gained on biologically-driven problems. As a consequence, the user should not get disappointed in special cases when the imposed constraints do not seem to lead to the desired outcome from the biological standpoint. For example, the fragment-based constraint and the sequence constraints are not fully compatible, and this could introduce designed sequences in which the sequence constraints that

were meant to appear inside a certain selected motif appear outside it in adjacent motifs. Such compatibility issues could be alleviated in future versions of our approach by enforcing links between the different types of constraints that are beyond the scope of the present work. At present, non-desired results as a consequence of these issues could be neglected or filtered out in a suitable post-processing step. Extensions for pseudoknot consideration and additional biologically-driven constraints, including varied-length designed sequences, are also left as prospects for future work.

The incaRNAfbinv webserver was developed with the goal of making the unique methods of fragment-based design with RNAfbinv and targeted weighted sampling with incaRNAtion available for the entire biological community. The webserver is user-friendly and accessible to practitioners, both in terms of ease of use and simplification of the output. We believe that it will serve experimental groups for improving their capability to perform RNA sequence design.

ACKNOWLEDGEMENTS

We thank Arik Goldfeld and Vitaly Shapira from the computer science laboratory at Ben-Gurion University for their help with our webserver.

FUNDING

This work was supported by ISF within the ISF-UGC joint research program framework (grant No. 9/14).

REFERENCES

- [1] Isaacs, F. J., Dwyer, D. J., and Collins, J. J. (May, 2006) RNA synthetic biology. *Nat Biotechnol*, **24**(5), 545–554.
- [2] Hammann, C. and Westhof, E. (2007) Searching genomes for ribozymes and riboswitches. *Genome Biol*, **8**(4), 210 [PubMed:17472738] [PubMed Central:PMC1895996] [doi:10.1186/gb-2007-8-4-210].
- [3] Taft, R. J., Pang, K. C., Mercer, T. R., Dinger, M., and Mattick, J. S. (Jan, 2010) Non-coding RNAs: regulators of disease. *J Pathol*, **220**(2), 126–139 [PubMed:19882673].
- [4] Strobel, S. A. and Cochrane, J. C. (Dec, 2007) RNA catalysis: ribozymes, ribosomes, and riboswitches. *Curr Opin Chem Biol*,

- 11(6), 636–643 [PubMed:17981494] [PubMed Central:PMC2184879] [doi:10.1016/j.cbpa.2007.09.010].
- [5] Breaker, R. R. (Sep, 2011) Prospects for riboswitch discovery and analysis. *Mol Cell*, **43**(6), 867–879 [PubMed:21925376] [PubMed Central:PMC4140403] [doi:10.1016/j.molcel.2011.08.024].
- [6] Serganov, A. and Nudler, E. (Jan, 2013) A decade of riboswitches. *Cell*, **152**(1-2), 17–24 [PubMed:23332744] [PubMed Central:PMC4215550] [doi:10.1016/j.cell.2012.12.024].
- [7] Findeiß, S., Wachsmuth, M., Mörl, M., and Stadler, P. F. (2015) Design of transcription regulating riboswitches. *Methods Enzymol*, **550**, 1–22 [PubMed:25605378] [doi:10.1016/bs.mie.2014.10.029].
- [8] Wachsmuth, M., Domin, G., Lorenz, R., Serfling, R., Findeiß, S., Stadler, P. F., and Mörl, M. (2015) Design criteria for synthetic riboswitches acting on transcription. *RNA Biol*, **12**(2), 221–231 [PubMed:25826571] [PubMed Central:PMC4615730] [doi:10.1080/15476286.2015.1017235].
- [9] Soukup, G. A. and Breaker, R. R. (Dec, 1999) Nucleic acid molecular switches. *Trends Biotechnol*, **17**(12), 469–476 [PubMed:10557159].
- [10] Chang, A. L., Wolf, J. J., and Smolke, C. D. (Oct, 2012) Synthetic RNA switches as a tool for temporal and spatial control over gene expression. *Curr Opin Biotechnol*, **23**(5), 679–688 [PubMed:22305712] [PubMed Central:PMC3354030] [doi:10.1016/j.copbio.2012.01.005].
- [11] Berens, C. and Suess, B. (Feb, 2015) Riboswitch engineering - making the all-important second and third steps. *Curr Opin Biotechnol*, **31**, 10–15 [PubMed:25137633] [doi:10.1016/j.copbio.2014.07.014].
- [12] Hofacker, I. L., Fontana, W., Stadler, P. F., Bonhoeffer, L. S., Tacker, M., and Schuster, P. (1994) Fast folding and comparison of RNA secondary structures. *Monatshfte für Chemie / Chemical Monthly*, **125**(2), 167–188.
- [13] Drory Retwitzer, M., Kifer, I., Sengupta, S., Yakhini, Z., and Barash, D. (2015) An Efficient Minimum Free Energy Structure-Based Search Method for Riboswitch Identification Based on Inverse RNA Folding. *PLoS One*, **10**(7), e0134262 [PubMed:26230932] [PubMed Central:PMC4521916] [doi:10.1371/journal.pone.0134262].
- [14] Dotu, I., Lozano, G., Clote, P., and Martínez-Salas, E. (Dec, 2013) Using RNA inverse folding to identify IRES-like structural subdomains. *RNA Biol*, **10**(12), 1842–1852 [PubMed:24253111] [PubMed Central:PMC3917987] [doi:10.4161/rna.26994].
- [15] Ruzzo, W. L. and Gorodkin, J. (2014) De novo discovery of structured ncRNA motifs in genomic sequences. *Methods Mol Biol*, **1097**, 303–318 [PubMed:24639166] [doi:10.1007/978-1-62703-709-9_15].
- [16] Hofacker, I. L. (Jul, 2003) Vienna RNA secondary structure server. *Nucleic Acids Res*, **31**(13), 3429–3431 [PubMed:12824340] [PubMed Central:PMC169005].
- [17] Zuker, M. (Jul, 2003) Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res*, **31**(13), 3406–3415 [PubMed:12824337] [PubMed Central:PMC169194].
- [18] Mathews, D. H., Disney, M. D., Childs, J. L., Schroeder, S. J., Zuker, M., and Turner, D. H. (May, 2004) Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc Natl Acad Sci U S A*, **101**(19), 7287–7292 [PubMed:15123812] [PubMed Central:PMC409911] [doi:10.1073/pnas.0401799101].
- [19] Busch, A. and Backofen, R. (Aug, 2006) INFO-RNA—a fast approach to inverse RNA folding. *Bioinformatics*, **22**(15), 1823–1831 [PubMed:16709587] [doi:10.1093/bioinformatics/btl194].
- [20] Aguirre-Hernández, R., Hoos, H. H., and Condon, A. (2007) Computational RNA secondary structure design: empirical complexity and improved methods. *BMC Bioinformatics*, **8**(1), 34 [PubMed:17266771] [PubMed Central:PMC1808480] [doi:10.1186/1471-2105-8-34].
- [21] Zadeh, J. N., Wolfe, B. R., and Pierce, N. A. (aug, 2011) Nucleic acid sequence design via efficient ensemble defect optimization. *J. Comput. Chem.*, **32**(3), 439–452 [PubMed:20717905] [doi:10.1002/jcc.21633].
- [22] Lyngsø, R. B., Anderson, J. W., Sizikova, E., Badugu, A., Hyland, T., and Hein, J. (2012) Frnakenstein: multiple target inverse RNA folding. *BMC Bioinformatics*, **13**(1), 260 [PubMed:23043260] [PubMed Central:PMC3534541] [doi:10.1186/1471-2105-13-260].
- [23] Cohen, B. and Skiena, S. (2003) Natural selection and algorithmic design of mRNA. *J Comput Biol*, **10**(3-4), 419–432 [PubMed:12935336] [doi:10.1089/10665270360688101].
- [24] Taneda, A. (2012) Multi-Objective Genetic Algorithm for Pseudoknotted RNA Sequence Design. *Front. Gene.*, **3**, 36 [PubMed:22558001] [PubMed Central:PMC3337422] [doi:10.3389/fgene.2012.00036].
- [25] Esmaili-Taheri, A. and Ganjtabesh, M. (jan, 2015) ERD: a fast and reliable tool for RNA design including constraints. *BMC Bioinformatics*, **16**(1), 20 [PubMed:25626878] [PubMed Central:PMC4384295] [doi:10.1186/s12859-014-0444-5].
- [26] Kleinkauf, R., Mann, M., and Backofen, R. (Oct, 2015) antaRNA: ant colony-based RNA sequence design. *Bioinformatics*, **31**(19), 3114–3121 [PubMed:26023105] [PubMed Central:PMC4576691] [doi:10.1093/bioinformatics/btv319].
- [27] Garcia-Martin, J. A., Dotu, I., and Clote, P. (Jul, 2015) RNAiFold 2.0: a web server and software to design custom and Rfam-based RNA molecules. *Nucleic Acids Res*, **43**(W1), W513–W521 [PubMed:26019176] [PubMed Central:PMC4489274] [doi:10.1093/nar/gkv460].
- [28] Bindewald, E., Afonin, K., Jaeger, L., and Shapiro, B. A. (Dec, 2011) Multistrand RNA Secondary Structure Prediction and Nanostructure Design Including Pseudoknots. *ACS Nano*, **5**(12), 9542–9551 [PubMed:22067111] [PubMed Central:PMC3263976] [doi:10.1021/nn202666w].
- [29] Yesselman, J. D. and Das, R. (Jul, 2015) RNA-Redesign: a web server for fixed-backbone 3D design of RNA. *Nucleic Acids Res*, **43**(W1), W498–W501 [PubMed:25964298] [PubMed Central:PMC4489241] [doi:10.1093/nar/gkv465].
- [30] Dromi, N., Avihoo, A., and Barash, D. (Aug, 2008) Reconstruction of natural RNA sequences from RNA shape, thermodynamic stability,

- mutational robustness, and linguistic complexity by evolutionary computation. *J Biomol Struct Dyn*, **26**(1), 147–162 [PubMed:18533734] [doi:10.1080/07391102.2008.10507231].
- [31] Avihoo, A., Churkin, A., and Barash, D. (2011) RNAexinv: An extended inverse RNA folding from shape and physical attributes to sequences. *BMC Bioinformatics*, **12**(1), 319 [PubMed:21813013] [PubMed Central:PMC3176266] [doi:10.1186/1471-2105-12-319].
- [32] Weinbrand, L., Avihoo, A., and Barash, D. (Nov, 2013) RNAfbinv: an interactive Java application for fragment-based design of RNA sequences. *Bioinformatics*, **29**(22), 2938–2940 [PubMed:23975763] [doi:10.1093/bioinformatics/btt494].
- [33] Shapiro, B. A. (Aug, 1988) An algorithm for comparing multiple RNA secondary structures. *Comput Appl Biosci*, **4**(3), 387–393 [PubMed:2458170].
- [34] Reinharz, V., Ponty, Y., and Waldspühl, J. (Jul, 2013) A weighted sampling algorithm for the design of RNA sequences with targeted secondary structure and nucleotide distribution. *Bioinformatics*, **29**(13), i308–i315 [PubMed:23812999] [PubMed Central:PMC3694657] [doi:10.1093/bioinformatics/btt217].
- [35] Nawrocki, E. P., Burge, S. W., Bateman, A., Daub, J., Eberhardt, R. Y., Eddy, S. R., Floden, E. W., Gardner, P. P., Jones, T. A., Tate, J., and Finn, R. D. (Jan, 2015) Rfam 12.0: updates to the RNA families database. *Nucleic Acids Res*, **43**(Database issue), D130–D137 [PubMed:25392425] [PubMed Central:PMC4383904] [doi:10.1093/nar/gku1063].
- [36] Darty, K., Denise, A., and Ponty, Y. (Aug, 2009) VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, **25**(15), 1974–1975 [PubMed:19398448] [PubMed Central:PMC2712331] [doi:10.1093/bioinformatics/btp250].

Supplementary Data are available at NAR online: Supplementary methods