Redesigning the Eterna100 for the Vienna 2 folding engine

Rohan V. Koodli[1,2,*], Boris Rudolfs[3,*], Jonathan Romano[4,5,*], Hannah K. Wayment-Steele[6], William A. Dunlap IV[4], Eterna Participants[γ], Rhiju Das[2,4,5,7,8,^]

[1] Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA 94720
[2] Biomedical Informatics Program, Stanford University, Stanford, CA 94305
[3] Department of Chemistry and Biochemistry, University of California, San Diego, La Jolla, California, 92110
[4] Eterna Massive Open Laboratory, Stanford, CA 94305
[5] Howard Hughes Medical Institute
[6] Department of Chemistry, Stanford University, Stanford, CA 94305
[7] Department of Biochemistry, Stanford University School of Medicine, Stanford, CA 94305
[8] Department of Physics, Stanford University, Stanford, CA 94305

[*]Indicates equal contribution
[γ]Group Author: Membership of Eterna participants and list of player names is provided in Author Contributions Statement
[^]Corresponding Author: rhiju@stanford.edu

ORCID: Rohan V. Koodli, 0009-0005-4044-8664; Boris Rudolfs, 0000-0002-8791-9487; Jonathan Romano, 0000-0003-4031-0102; Hannah K. Wayment-Steele, 0000-0003-4949-2010; Eterna participants, 0000-0002-7508-6705; Rhiju Das, 0000-0001-7497-0972.

Abstract

The rational design of RNA is becoming important for rapidly developing technologies in medicine and biochemistry, spurring development of numerous RNA secondary structure design algorithms and benchmarks to evaluate their performance. However, the problem of RNA design is dependent upon the reverse problem of RNA structure prediction through "folding engines" that predict structure from sequence. We hypothesized that differences in RNA folding engines could impact design algorithms, and recruited an online community of RNA design experts to modify the widely-used RNA secondary structure design benchmark, Eterna100, to address unsolvability of some cases when changing the folding engine used (Vienna 1.8 updated to Vienna 2.6). We tested this new Eterna100-V2 benchmark with five RNA design algorithms, and found that while overall rankings remained similar, the performance of RNA design algorithms that depended on folding engines in their training did indeed depend on which underlying parameter set was used in training. This work demonstrates that the design "difficulty" of RNA structures is intrinsically linked to thermodynamic models, and suggests that future RNA design algorithms that are agnostic to thermodynamic models will result in optimal performance and development. Eterna100-V1 and Eterna100-V2 benchmarks and example solutions are freely available at https://github.com/eternagame/eterna100-benchmarking.

Keywords: RNA design, secondary structure, benchmark, inverse folding, citizen science

Author Summary

Designing RNA molecules that fold to a desired target structure is an algorithmic problem gathering increasing attention due to the emergence of RNA-based therapies and the need for rational design of RNAs. The Eterna100 dataset, a collection of target structures with increasing design difficulty, designed and selected by players of the crowdsourced RNA game Eterna, has been widely used to benchmark RNA design algorithms. However, these puzzles were originally developed using the now-deprecated version 1 of the ViennaRNA folding engine. In this manuscript, we introduce an updated benchmark, called the *Eterna100-V2*. We found that nineteen puzzles using Vienna 1 were unsolvable in Vienna 2, but that Eterna participants were able to re-design the puzzles with minimal modifications to make them solvable in Vienna 2. We confirmed that the rankings of 5 RNA design algorithms remained consistent between Eterna100-V1 and -V2. However, discrepancies in performance from algorithms that relied on thermodynamic models in their training suggest that algorithms will benefit from being agnostic to thermodynamic models as these models continue to improve.

Introduction

RNA has significantly expanded past its original proposed role as an intermediate in the genetic code and as a catalytic scaffold for protein synthesis. RNA has been observed to act as a genetic expression regulator [1], perform catalysis [2], be a scaffold for complex formation [3,4], and be used as a guide by several ribonucleoprotein complexes [5–7]. This increased appreciation for the versatile activity of RNA has led to the recent development of several RNA therapies that include the control of pre-mRNA splicing [8], gene editing and expression [6], and aptamers for binding and sequestering target molecules [9]. It is increasingly understood that RNAs obtain some of their function via structures largely determined by Watson-Crick base-pairing. Structured synthetic RNAs underlie methods to regulate gene expression in bacteria [10] and in eukaryotes [11], sense analytes [12], enhance gene editing systems via synthetic CRISPR guide RNAs [13], and design RNA therapeutics with increased stability [14].

The RNA secondary structure design problem, also known as the inverse folding problem, involves designing an RNA sequence that folds into a target secondary structure given an energy function [15]. Classic RNA inverse folding algorithms used cost function minimization through adaptive random walk [16], structure decomposition [17], minimization of the ensemble defect [18], or a genetic algorithm [19]. Performance of these older algorithms was not well characterized, as benchmarking occurred internally and was performed on well-characterized biological RNA or computationally predicted secondary structures from RNA sequences without selection for complexity or difficulty [17,20,21]. Furthermore, as RNA length and complexity increases, the number of asymmetric and symmetric elements increases, thereby increasing the difficulty of sequence design for these molecules [22].

To address the need for a community-wide standard benchmark for RNA design, Anderson-Lee et al. developed a set of 100 secondary structures published on the Eterna website using RNAfold 1.8.5 [17] from the ViennaRNA package (henceforth referred to as “Vienna 1”). This Eterna100 benchmark (henceforth referred to as Eterna100-V1) was chosen to showcase secondary structure motifs that were identified as being difficult to design, and the best-performing algorithm at the time [19] solved 54/100. Since the benchmark was published, several algorithms have surpassed this mark using convolutional neural networks [23], reinforcement learning [24,25], or a Monte Carlo search optimized for game theory [26]. However, algorithms have been inconsistent in which secondary structure prediction algorithm, or “folding engine”, they use. For example, EternaBrain [23] used Vienna 1.8.5, but Meta-LEARNA [25] used updated thermodynamic parameters in RNAfold 2.1.9.

Despite the fundamental link between which folding engine is used to train and which folding engine is used to evaluate design algorithms, there has been no systematic evaluation of the effect of folding engines in the training and performance of design algorithms. We challenged Eterna participants to determine if all the puzzles in the original Eterna100 could still be solved using an updated set of ViennaRNA parameters (the parameters from RNAfold 2.1.9, henceforth referred to as “Vienna 2”). Indeed, participants identified 19 of the 100 structures that were

deemed to be unsolvable in Vienna 2. We then challenged the community to adapt these secondary structures to the Vienna 2 parameter set using a minimal number of insertions and deletions, resulting in the Eterna100-V2 benchmark. We discuss key structural motifs that are intractable in one set of thermodynamic parameters but solvable in the other. We evaluated several state-of-the-art inverse folding algorithms and determined that while their relative performance is unchanged, neural-network-based methods would benefit from re-training with Vienna 2 parameters. Taken together, this work indicates that consideration of which folding engine is used in operating inverse folding algorithms is critical in their evaluation, even in determining the scope of what structures are fundamentally solvable.

Results

Thermodynamic differences in motifs result in puzzle unsolvability

Eterna participants identified 19 secondary structures of the original 100 that were 'unsolvable' in Vienna 2. We verified that the stochastic algorithm NEMO [26], which is currently state-of-the-art in the Eterna100 with the original Vienna 1 folding engine, was also unable to find solutions within 24-hour timeframes for these 19 problems with the Vienna 2 folding engine.

These 19 puzzles varied in both length and relative complexity, but shared several structural features that gave rise to this discrepancy (**Fig. 1a**). The change in the relative free energies of internal and stem loops between Vienna 1 and Vienna 2 led to several structures with isolated base pairs no longer being solvable in the Vienna 2 parameters. Multi-helix junctions were also found to have lower free energies in Vienna 2 than Vienna 1, which significantly altered puzzles that feature these such as "Kyurem 5" and "Multilooping 6" (**Fig. 1b**).

Large internal loops increased in free energy in Vienna 2 as compared to Vienna 1, which made puzzles containing these such as "[RNA] Repetitious Sequences 8/10" unsolvable. For example, EternaBrain's solution to [RNA] Repetitious Sequences 8/10's Eterna100-V1 structure creates two large internal loops both with free energies of -1.1 kcal/mol. However, if the same structure and solution are used in Vienna 2, the free energies of both internal loops increases to 2.1 kcal/mol (**Fig. 1a**, **top panels**).

These results motivated us to develop a new set of puzzles for these 19 problems that would comprise a new Eterna100-V2 benchmark to be used with the Vienna 2 folding engine, which has largely displaced the Vienna 1 folding engine for wide use. We solicited these new problems from Eterna participants with a news post (**Supp. Fig. S1**) and provided the Vienna 2 folding engine in the Eterna "puzzlemaker" interface (**Fig. 1c**). These puzzles and example solutions are compiled in **Supplemental Table S1** as well as https://github.com/eternagame/eterna100-benchmarking.

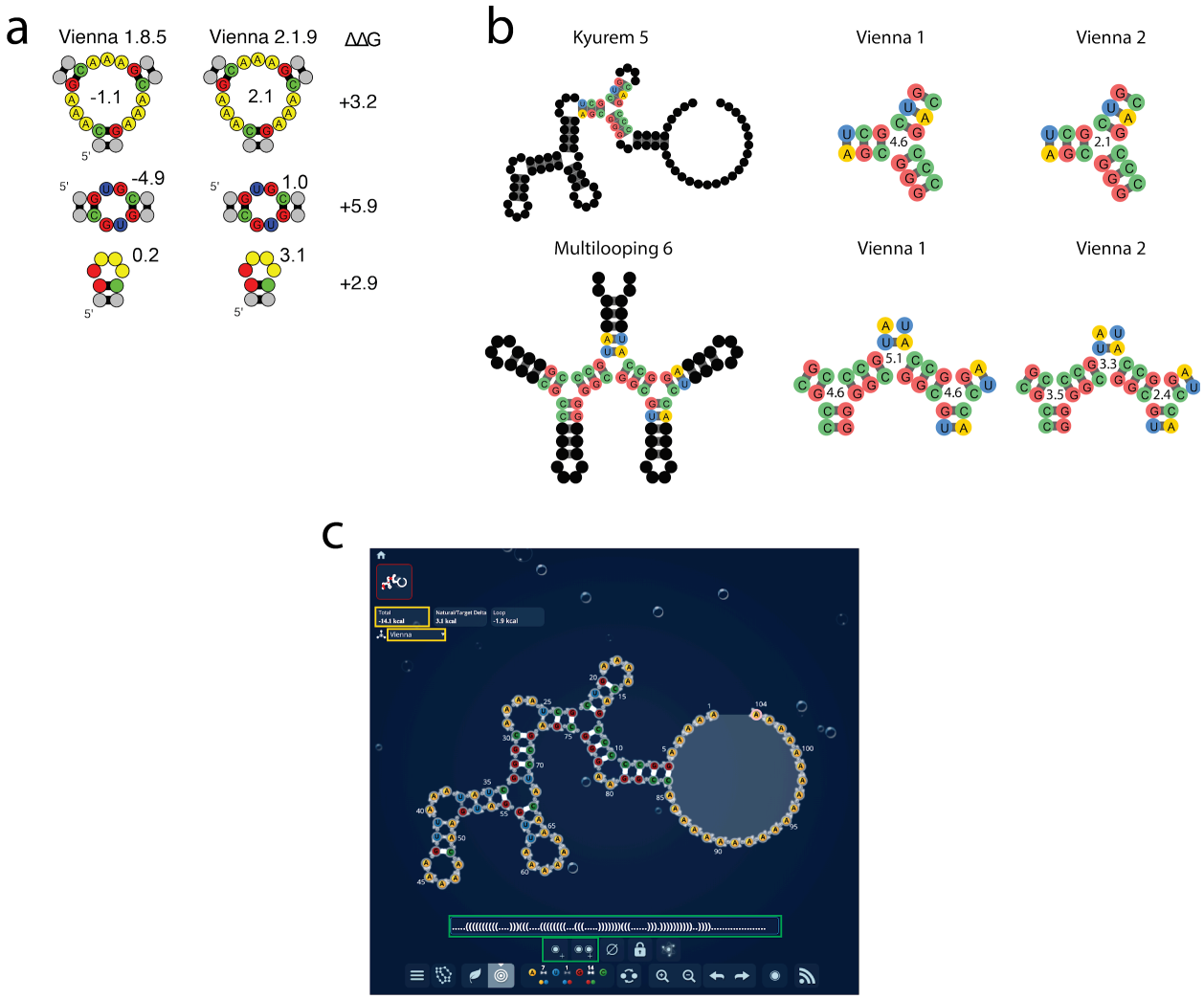

Figure 1. **(a)** Example free energy differences (ΔΔG, kcal/mol) between identical structures in Vienna 1.8 and Vienna 2.4. **(b)** Puzzle-specific differences in free energies of structures. The internal tri-junction in Kyurem 5 has a more stable free energy in Vienna 2. Similar repeated tri-junctions are highlighted in Multilooping 6, with similar changes in free energy from Vienna 1 to 2. **(c)** A screenshot of Eterna's "Puzzlemaker" interface. Players can see the free energy and folding engine in the top left corner (yellow rectangles). Players can add bases, add base pairs, and interact with the dot-bracket representation of the structure directly (green rectangles).

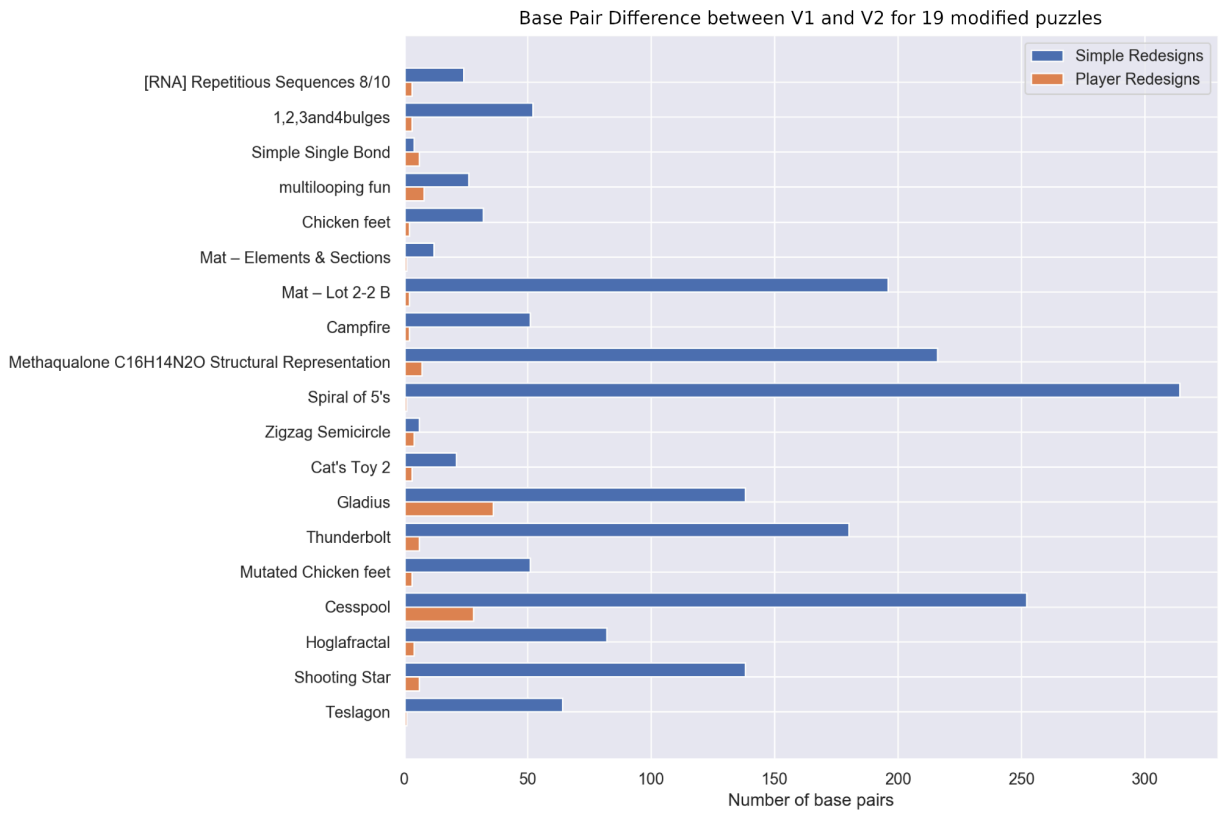

Figure 2. Eterna participant target structure re-designs result in fewer number of structure modifications than a naive re-design approach, quantified by string edit distance in RNAdistance (18).

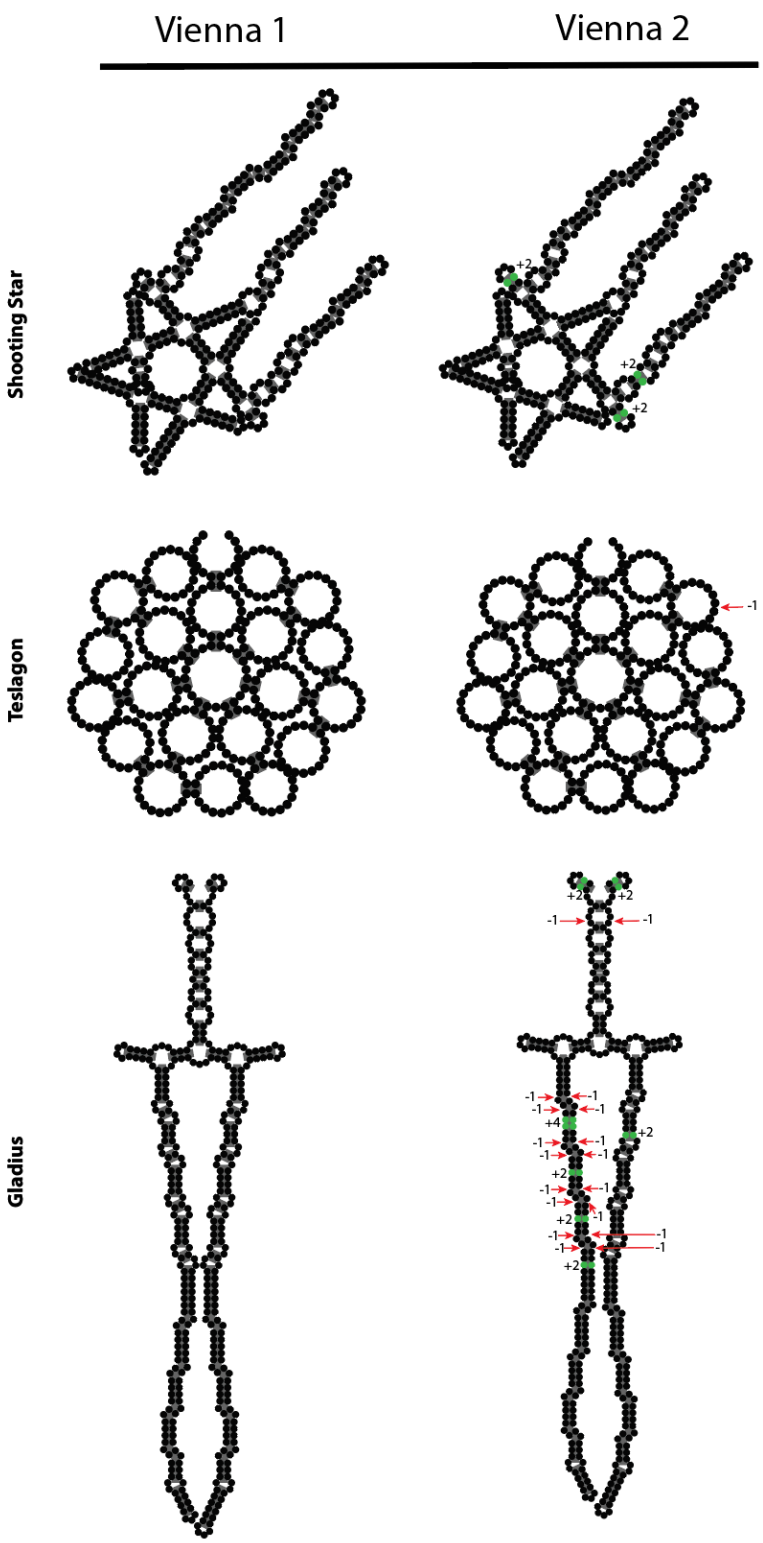

Figure 3. Base pair changes in "Shooting Star" from Vienna 1 to Vienna 2. The green nucleotides indicate the bases that were added to make the structure stable in Vienna 2, and the red indicate deletions.

Eterna participants redesign target structures with minimal alterations

Given that the original Eterna100-V1 benchmark was designed by players, we hypothesized that players might be able to design modified structures for these puzzles using a minimum number of structure modifications, while maintaining the constraints of the Eterna software platform. Strategies included adding or removing base pairs to maintain the overall puzzle topology, reducing the size of large loops to mitigate Vienna 2's increased penalty on large loops. We compared Eterna participant-designed solutions to a "simple" redesign method: taking a set of known player puzzle solutions in Vienna 1 and calculating their minimum free energy structures in Vienna 2. However, these structures did not exhibit similar shapes or similarly difficult features as the original structures posed in the Eterna100 benchmark, quantified by string edit distance [16] (**Fig. 2**). In all 19 puzzles, this calculated difference compared to the V1 structures was much larger than the difference in the secondary structures that players created in parallel. Strikingly, some of the most structurally complex puzzles required minimal alterations."Shooting Star" consists of several multi-helix junctions and long helices that contain 29 isolated base pairs and was made solvable through only 3 base pair additions (**Fig. 3, top**). "Teslagon" consists of a series of loops around a core 7-way junction and was made solvable through the deletion of a single internal loop base (**Fig. 3, middle**). However, some puzzles like "Gladius" require more extensive changes, including breaking structural symmetry, to become solvable in Vienna 2 (**Fig. 3, bottom**).

Redesigned target structures are largely still solvable using Vienna 1

All structures which were modified in the Eterna100-V2 for solvability in Vienna 2 were tested for solvability under Vienna 1 through puzzles on Eterna. We found that all except for puzzle 97, "Hoglafractal", were solvable in both folding engines. This remaining puzzle appears to be unsolvable due to a single base pair at the beginning of the puzzle which is incompatible with Vienna 1 energetic parameters for internal loops and dangling ends. The observation that 99/100 Eterna100-V2 puzzles are solvable in both folding engines, compared to 81/100 for the original Eterna100-V1, suggested that the newer benchmark is somewhat easier than the original one, presumably due to its origin as a fix for puzzles that were first proposed by Eterna players to display and teach idiosyncrasies of the original Vienna 1 folding engine.

	Vienna 1 folding engine		Vienna 2 folding engine	
Method	**Eterna100-V1**	**Eterna100-V2**	**Eterna100-V1**	**Eterna100-V2**
RNAinverse	18	19	20	20
EternaBrain	60	65	49	49
LEARNA	67	72	66	69
SentRNA	69	72	63	68
NEMO	92	93	77	86
Eterna (humans)	100	81	99	100

Table 1. Number of solutions of the Eterna100-V1 and Eterna100-V2 benchmarks by five automated algorithms and human players of Eterna.

RNA design algorithm rankings are consistent in Eterna100-V2, but performance changes reflect modeling priors

After players modified the secondary structures for the 19 Eterna100 puzzles, we assessed the performance of 5 RNA inverse folding algorithms on this updated benchmark (Fig 5): two algorithms relying on Monte Carlo methods, RNAinverse and NEMO; and three algorithms that also included deep neural networks to guide search, EternaBrain, SentRNA, and LEARNA (**Fig. 1** and **Table 1**; more detailed results in **Supplemental Tables S2** and **S3**).

To evaluate the difficulty of the new Eterna100-V2 benchmark, we tested them with the Vienna 1 folding engine, which was also the engine used during development of most of the algorithms. For all of the algorithms, performance on Eterna100-V2 improved compared to Eterna100-V1 when retaining the Vienna 1 folding engine (**Fig. 4** and **Table 1**; RNAInverse, 18 to 19; EternaBrain, 60 to 65; Learna, 67 to 72; SentRNA, 69 to 72; and NEMO, 92 to 93). These observations support the results above suggesting that the new Eterna100-V2 benchmark is somewhat easier than the original Eterna100-V1 benchmark. Inspection of particularly difficult puzzles further supports this picture. For example, the 19 structures from Eterna100-V1 that were deemed unsolvable in Vienna 2 were of particular interest (boldface in Figure 5) as they were some of the hardest secondary structures: in the original Eterna100-V1 benchmark tested in the Vienna 1 folding engine, EternaBrain and SentRNA solved 3 (16%) and 6 (32%) out of the 19, respectively, lower than their average across all Eterna100-V1 puzzles (60% and 69%, respectively). All design methods demonstrated performance improvement in going from the 19 puzzles in Eterna100-V1 to Eterna100-V2 (RNAinverse, 0 to 1; EternaBrain and LEARNA, both 3 to 8; SentRNA, 6 to 9; and NEMO, 14 to 15; **Fig. 4** and **Supplemental Tables S2** and **S3**). Again this was expected given the increased solution space afforded by the redesigned structures for these 19 puzzles in Eterna100-V2 compared to Eterna100-V1 (for example, through smaller internal loops creating more favorable energetic conditions; **Fig. 1a-b** and **Fig. 3**).

The intended use case for the Eterna100-V2 is to benchmark RNA design algorithms with the more recent Vienna 2 folding engine. Unexpectedly, when comparing algorithm performance on the Eterna100-V1 when run with Vienna 1 versus performance on the Eterna100-V2 when run with Vienna 2 as the internal folding engine, performance of algorithms generally got worse, with some exceptions (**Fig. 4** and **Table 1**). RNAinverse solved 18 puzzles under Eterna100-V1/Vienna 1, which increased to 20 under the Eterna100-V2/Vienna 2; but EternaBrain's performance decreased dramatically, from 60 to 49 solved puzzles. LEARNA's performance increased slightly from 67 to 69, SentRNA decreased slightly from 69 to 68, and NEMO decreased from 92 to 86. Despite these changes, the rankings of the algorithms were nearly the same in Eterna100-V2/Vienna 2 compared to Eterna100-V1/Vienna 1 – only LEARNA and SentRNA switch places, but they were effectively tied in both benchmarks, with the number of solved puzzles by the two algorithms within 2. The three algorithms whose success rates dropped (EternaBrain, LEARNA, NEMO) included heuristics and/or neural network training examples that were derived from Eterna players' efforts with the deprecated Vienna 1 folding engine, suggesting that those sequence design methods are overly dependent on details of folding engine. The LEARNA reinforcement learning design method, which instead used the Vienna 2 folding engine during its training, showed slightly improved performance in Eterna100-V2 (**Fig. 4** and **Table 1**).

The results above suggested that, while the Eterna100-V2 benchmark is easier (since all methods solved more puzzles than with Eterna100-V1 when retaining the Vienna 1 folding engine, or even while maintaining the Vienna 2 folding engine; **Table 1**), the switch from the Vienna 1 foldine engine to the Vienna 2 folding engine makes all puzzles more challenging for the methods developed with Vienna 1. As a result, depending on the puzzle and the algorithm, performance worsened, was sustained, or improved in going from the original Eterna100-V1/Vienna 1 benchmark to the new Eterna100-V2/Vienna 2 benchmark. To further test this picture, we retrained EternaBrain and SentRNA with both the Vienna 1 and Vienna 2 folding engines (for LEARNA, we could not reproduce the performance level of the published pre-trained design model). As a confirmation of the validity of our retraining protocol, we reproduced and improved upon the performance of the published versions of EternaBrain and SentRNA, respectively, on both Eterna100-V1 and Eterna100-V2 benchmarks when Vienna 1 was used during the retraining, when compared with the original studies reporting these methods **(Supplemental Table S2)**. Using Vienna 2 during re-training, however, changed the number of solved puzzles by 1 or less for both EternaBrain and SentRNA. Both algorithms remained poorer with the Vienna 2 folding engine than with Vienna 1 folding engine even after retraining their neual network components with the respective folding engine. This result, along with the large drop in NEMO performance noted above, suggests that the dependence these algorithms' performance on folding engine depends not so much on the neural network components of the models but instead on other aspects that were biased by use of the Vienna 1 folding engine. Such aspects include heuristic rules for initialization and design refinement developed during Eterna

gameplay and/or the Eterna players' moves and solutions used for training, which both occurred in the context of the Vienna 1 folding engine.

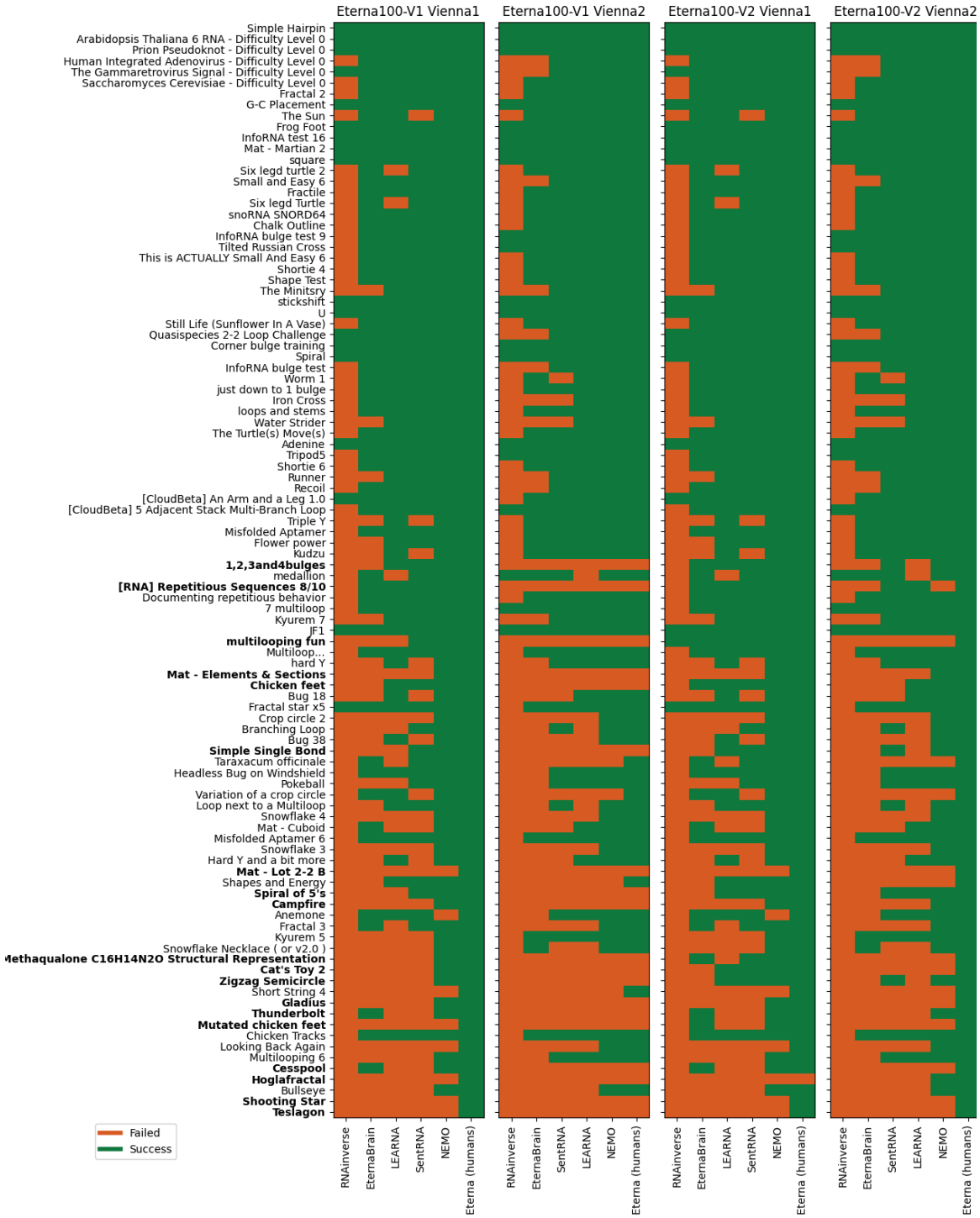

Figure 4. Performance of five algorithms for RNA secondary structure design on the Eterna100-V1 and Eterna100-V2 benchmarks, using either Vienna 1 or Vienna 2 as the folding

engine at runtime. Green: Solved; Orange: Unsolved. Bolded names indicate puzzles changed between Eterna100-V1 and Eterna100-V2 benchmarks.

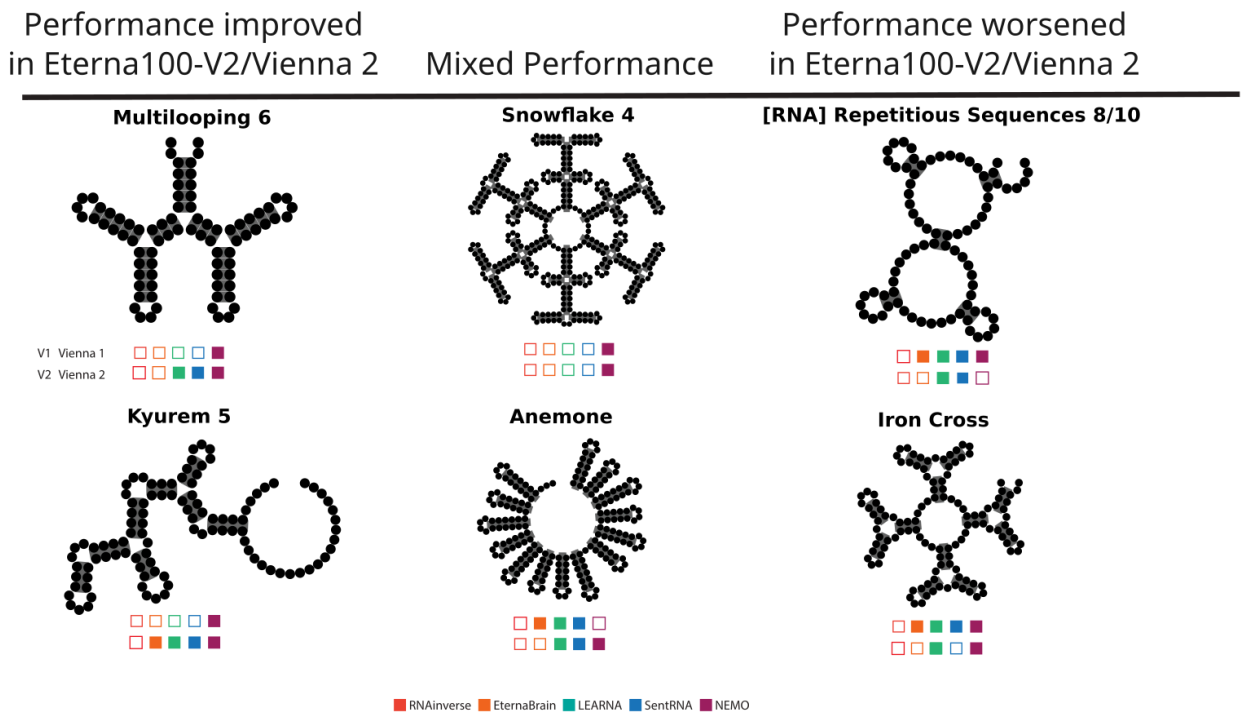

Figure 5. Selected Eterna100-V1 and -V2 puzzles that demonstrate differences in algorithms' puzzle-solving ability. Open squares indicate that a given algorithm was unable to solve that puzzle; filled squares indicate that the algorithm did solve the puzzle. The first row of squares indicates algorithms' performance on Eterna100-V1 using Vienna 1, and the second row indicates the performance on Eterna100-V2 using Vienna 2. Red: RNAinverse; Orange: EternaBrain; Teal: LEARNA; Blue: SentRNA; Purple: NEMO.

Discussion

In this work, we have reported that 19 of the 100 structures in the widely used Eterna100 benchmark for evaluating RNA inverse design were unsolvable when the thermodynamic parameters of Vienna 1 were substituted with those of Vienna 2. This presents a problem for evaluating inverse folding algorithms, since their performance is intrinsically limited by the thermodynamic model chosen to develop and test each algorithm. To amend this problem, we asked Eterna participants to redesign the 19 puzzles. Participants found strategies to do so that they proposed would largely preserve the original challenges of the puzzles. Unexpectedly, the 19 new puzzles appear to be solvable in both the new Vienna 2 folding engine and the original Vienna 1 folding engine. Indeed, the final new benchmark, termed Eterna100-V2, has 100 of 100

puzzles solvable in Vienna 2 but also 99 of 100 puzzles solvable in Vienna 1, suggesting that it is easier than the original Eterna100-V1 benchmark.

We next evaluated state-of-the-art algorithms on this updated Eterna100-V2 benchmark. We found that changes in performance appear to reflect model priors. For example, EternaBrain, which was developed with player solutions and heuristic strategies generated under Vienna 1, exhibited notably worse performance with the Vienna 2 folding engine. Additional tests were performed to retrain neural network components of the design methods using folding engines "matching" the respective benchmark but identified minimal impact on original model quality. Instead, other components of the EternaBrain and NEMO models besides their neural networks, such as heuristic design rules based on Vienna 1 gameplay, appear to reduce their ability to generalize to the Vienna 2 folding engine.

Given the number of secondary structures that were unsolvable in the original Eterna100 benchmark with the Vienna 2 parameters, it seems likely this benchmark will need to be continuously updated as RNA structure prediction becomes more accurate. Folding engines like CONTRAfold [27] and EternaFold [28] demonstrate improved performance over Vienna folding engines in predicting experimental thermodynamic ensembles, but generally predict less stable structures than the Vienna folding engines for the same sequences. Furthermore, metrics beyond minimum free energy structure are potentially better suited [29] for predicting *in vitro* and *in vivo* structural viability, in addition to other desired characteristics which are not reflected by the dominant folded structure alone, such as stability and protein expression [30]. The recent report of a replica Monte Carlo algorithm desiRNA solving all the Eterna100-V1 also suggests that a more difficult benchmark is now needed to drive progress in RNA design [31]. Taken together, this work indicates that future RNA inverse folding algorithms should strive to be folding-engine independent, or at least rapidly retrainable with new folding engines and 'win conditions'. We hope that open availability of Eterna100-V1 and Eterna100-V2 will enable the testing of such algorithms.

Methods

Design challenges on Eterna platform

Through iterative manual design, we identified 19 secondary structures in the original Eterna100 (Eterna100-V1) benchmark that we hypothesized were not solvable using Vienna 1.8.5. We asked the Eterna community to redesign these 19 puzzles to be compliant with the thermodynamic parameters of the ViennaRNA 2.1.9 software package as implemented in Eterna. Players achieved this through the Eterna "Puzzlemaker" interface, whereby individual base pairs and bases may be deleted or added to a structure. Players submitted 52 total puzzles as modifications of the unsolvable 19. Nineteen of those submissions were chosen by the authors to both maintain the constraints of the Eterna platform and the identity of unique structural elements that existed in the original puzzles.

During this process we also discovered that for one of the puzzles (puzzle 5, The Gammaretrovirus Signal - Difficulty Level 0) in the Eterna100-V1, the structure did not match that of the Eterna puzzle it was derived from (and so neither did the sample solutions solve the structure). This was due to an error which caused a single nucleotide to be omitted - we have treated the Eterna100-V1 structure as correct and provided an updated Eterna puzzle ID and sample sequence for Vienna 1 in **Supplemental Table S2**.

Automated tests of RNA secondary structure design algorithms

We evaluated the performance of five algorithms (EternaBrain, SentRNA, RNAinverse, LEARNA, and NEMO) on the Eterna100-V1 and on the Eterna100-V2 with the 19 modified puzzles, using ViennaRNA 1.8.5 as well as the more recent ViennaRNA 2.6.4 as the folding engine. For each puzzle, algorithm, and Vienna version combination (with the exclusion of the 19 unsolvable Eterna100-V1 puzzles and one unsolvable Eterna100-V2 puzzle under Vienna1), algorithms were run for a maximum of 24 hours with a single AMD 7502 core on the Stanford University Sherlock cluster. (We note that use of the ViennaRNA 2 package with Turner99 parameters, as carried out recently [31], can be used to emulate ViennaRNA 1 but should use the setting "--dangles=2" to fully reproduce the behavior of the Vienna 1 folding engine. Additionally, where multiple structures have the same free energy for a given sequence, Vienna 2 run in this way may select a different structure as the MFE structure than Vienna 1, resulting in a sequence purported to solve a puzzle correctly evaluated as having failed to solve the puzzle)

EternaBrain uses a combination of a convolutional neural network trained on Eterna player moves and a Single Action Playout (SAP) [23], a depth-1 Monte Carlo Search using Eterna player strategies. In addition to the pretrained CNN trained with features generated by Vienna 1, the CNN was retrained using features generated by Vienna 1 and Vienna 2. Variants were tested using combinations of the Vienna 1 or Vienna 2 trained CNN, using Vienna 1 or Vienna 2 during inference to generate features passed to the CNN, and using either Vienna 1 or Vienna 2 during the SAP step. Variants where the folding engine in the SAP step was different from the engine used to generate features for the CNN inference step (termed "flipped", eg vienna1-flipsap run in "vienna1" mode uses the CNN trained using Vienna 1 features, with Vienna 1 for the CNN inference features, and Vienna 2 for the SAP) were tested as the player moves used to train the CNN were all using Vienna 1 originally, and insufficient data exist to train a neural network with Vienna 2 player moves. The "extended" 92-puzzle dataset from the latest version of the EternaBrain repository was used, rather than the published 78-puzzle dataset. Training was run until completion.

SentRNA, unlike EternaBrain, does not rely on player moves to train its deep neural networks; instead, the authors define their own set of features in the Methods section of the paper (though it does rely on player solutions generated under Vienna 1) [24]. In addition to using the pretrained models (selecting the 20 models trained using 20 structural features), under both Vienna 1.8.5 and 2.6.4, we re-trained an ensemble of 20 networks (each using 20 structural features) with 300 "solution trajectories" [24]. Training was run until completion. For LEARNA,

in addition to the pretrained model, we attempted to similarly retrain the reinforcement learning models using Vienna 1.8.5 and 2.6.4, keeping the original hyperparameter and architecture values (i.e., without performing a new joint architecture and hyperparameter search). Training was run for one hour on 20 CPUs on the Stanford University Sherlock cluster, but did not lead to performance competitive with the pre-trained LEARNA models made available with the publication.

RNAinverse and NEMO, which are both stochastic methods, were able to be run without modification using either Vienna 1 or Vienna 2 as the internal folding engine. RNAinverse was benchmarked using standard settings, allowing either Vienna 1.8 or Vienna 2.6.4 to be used as the folding algorithm. NEMO was run with a maximum of 2500 Monte Carlo iterations. To more stringently test the unsolvability of the 19 “unsolvable” puzzles, i.e. the 19 original puzzles run in Vienna 2, NEMO was given 10 independent 24-hour periods to attempt to solve using 16 (unspecified) cores with a maximum of 10^4 Monte Carlo iterations. No solutions were found.

Acknowledgements

Calculations and model training were performed on the Stanford Sherlock cluster and Berkeley CSUA cluster. We acknowledge funding from the National Science Foundation (GRFP to H.K.W.S.), the National Institutes of Health (R35 GM122579 to R.D.), and the Howard Hughes Medical Institute (R.D.).

Author Contributions Statement

B.R. and R.D. formulated the problem. R.V.K., J.A.R., and H.K.W.S. evaluated RNA design algorithms. R.V.K., B.R., and J.A.R. performed analyses. Eterna Participants designed the secondary structures and generated additional solutions, as follows: B.R. (Brourd), Tesla’sDisciple, W.A.D. (Astromon), c-quence, JR, Jieux, jnicol, clollin created the updated 19 puzzles in Eterna100-V2 with minimal changes; and wawan151, Jieux, lroppy (Leonard Oppenheimer), jnicol (John Nicol), JSci solved puzzles without algorithmic solutions. W.A.D. performed checks on solutions, including solvability of Eterna100-V2. R.V.K., B.R., J.A.R., H.K.W.S., and R.D. wrote the manuscript.

Legends

Figure 1. (A) Free energy differences ($\Delta\Delta G$, kcal/mol) between identical structures in Vienna 1.8 and Vienna 2.4. Topmost: an internal triloop; middle: U-G-U-G superboost with G-C end pairs; bottom: tetraloop hairpin with G-boost and G-C end pairs. (B) Puzzle-specific differences in free energies of structures. The internal tri-junction in Kyurem 5 has a more stable free energy in Vienna 2. Similar repeated tri-junctions are highlighted in Multilooping 6, with similar

changes in free energy from Vienna 1 to 2.

Figure 2. Number of nucleotide modifications for each of the 19 puzzles that were changed to be made solvable in Vienna 2. Blue denotes RNAdistance calculated distance metric between the simple design algorithm involving calculating structure differences between Vienna 1 and Vienna 2. Orange denotes differences in player designs between Vienna 1 and Vienna 2.

Figure 3. Base pair changes in Shooting Star from Vienna 1 to Vienna 2. The neon yellow nucleotides indicate the bases that were added to make the structure stable in Vienna 2.

Figure 4. Performance of five algorithms for RNA secondary structure design on the Eterna100-V1 and Eterna100-V2 benchmarks, using either Vienna 1 or Vienna 2 as the folding engine at runtime. Green: Solved; Orange: Unsolved. Bolded names indicate puzzles changed between Eterna100-V1 and Eterna100-V2 benchmarks.
Figure 5. Selected Eterna100-V1 and -V2 puzzles that demonstrate differences in algorithms' puzzle-solving ability. Open squares indicate that a given algorithm was unable to solve that puzzle; filled squares indicate that the algorithm did solve the puzzle. The first row of squares indicates algorithms' performance on Eterna100-V1 using Vienna 1, and the second row indicates the performance on Eterna100-V2 using Vienna 2. Red: RNAinverse; Orange: EternaBrain; Teal: LEARNA; Blue: SentRNA; Purple: NEMO.

Supplemental Figure S1. Eterna announcement asking players to attempt to solve the nineteen modified puzzles in Vienna 1 and Vienna 2.

Table 1. Number of solutions of the Eterna100-V1 and Eterna100-V2 benchmarks by five automated algorithms and human players of Eterna.

References

1. Serganov A, Nudler E. A decade of riboswitches. Cell. 2013;152: 17–24. doi:10.1016/j.cell.2012.12.024
2. Jo B-S, Choi SS. Introns: the functional benefits of introns in genomes. Genomics Inform. 2015;13: 112–118. doi:10.5808/GI.2015.13.4.112
3. Will CL, Lührmann R. Spliceosome structure and function. Cold Spring Harb Perspect Biol. 2011;3. doi:10.1101/cshperspect.a003707
4. Haerty W, Ponting CP. Mutations within lncRNAs are effectively selected against in fruitfly but not in human. Genome Biol. 2013;14: R49. doi:10.1186/gb-2013-14-5-r49
5. Song M-S, Rossi JJ. Molecular mechanisms of Dicer: endonuclease and enzymatic

activity. Biochem J. 2017;474: 1603–1618. doi:10.1042/BCJ20160759

6. Doudna JA, Charpentier E. Genome editing. The new frontier of genome engineering with CRISPR-Cas9. Science. 2014;346: 1258096. doi:10.1126/science.1258096

7. Chen J-L, Greider CW. Telomerase RNA structure and function: implications for dyskeratosis congenita. Trends Biochem Sci. 2004;29: 183–192. doi:10.1016/j.tibs.2004.02.003

8. Singh RN, Singh NN. Mechanism of splicing regulation of spinal muscular atrophy genes. Adv Neurobiol. 2018;20: 31–61. doi:10.1007/978-3-319-89689-2_2

9. Ng EWM, Shima DT, Calias P, Cunningham ET, Guyer DR, Adamis AP. Pegaptanib, a targeted anti-VEGF aptamer for ocular vascular disease. Nat Rev Drug Discov. 2006;5: 123–132. doi:10.1038/nrd1955

10. Green AA, Silver PA, Collins JJ, Yin P. Toehold switches: de-novo-designed regulators of gene expression. Cell. 2014;159: 925–939. doi:10.1016/j.cell.2014.10.002

11. Zhao EM, Mao AS, de Puig H, Zhang K, Tippens ND, Tan X, et al. RNA-responsive elements for eukaryotic translational control. Nat Biotechnol. 2022;40: 539–545. doi:10.1038/s41587-021-01068-2

12. Jung JK, Alam KK, Verosloff MS, Capdevila DA, Desmau M, Clauer PR, et al. Cell-free biosensors for rapid detection of water contaminants. Nat Biotechnol. 2020;38: 1451–1459. doi:10.1038/s41587-020-0571-7

13. Allen D, Rosenberg M, Hendel A. Using synthetically engineered guide rnas to enhance CRISPR genome editing systems in mammalian cells. Front Genome Ed. 2020;2: 617910. doi:10.3389/fgeed.2020.617910

14. Wayment-Steele HK, Kim DS, Choe CA, Nicol JJ, Wellington-Oguri R, Watkins AM, et al. Theoretical basis for stabilizing messenger RNA through secondary structure design. Nucleic Acids Res. 2021;49: 10604–10617. doi:10.1093/nar/gkab764

15. Zuker M. On finding all suboptimal foldings of an RNA molecule. Science. 1989;244: 48–52. doi:10.1126/science.2468181

16. Hofacker IL. Vienna RNA secondary structure server. Nucleic Acids Res. 2003;31: 3429–3431. doi:10.1093/nar/gkg599

17. Andronescu M, Fejes AP, Hutter F, Hoos HH, Condon A. A new algorithm for RNA secondary structure design. J Mol Biol. 2004;336: 607–624. doi:10.1016/j.jmb.2003.12.041

18. Zadeh JN, Steenberg CD, Bois JS, Wolfe BR, Pierce MB, Khan AR, et al. NUPACK: Analysis and design of nucleic acid systems. J Comput Chem. 2011;32: 170–173. doi:10.1002/jcc.21596

19. Taneda A. MODENA: a multi-objective RNA inverse folding. Adv Appl Bioinform Chem.

2011;4: 1–12. doi:10.2147/aabc.s14335

20. Busch A, Backofen R. INFO-RNA--a server for fast inverse RNA folding satisfying sequence constraints. Nucleic Acids Res. 2007;35: W310-3. doi:10.1093/nar/gkm218

21. Griffiths-Jones S, Bateman A, Marshall M, Khanna A, Eddy SR. Rfam: an RNA family database. Nucleic Acids Res. 2003;31: 439–441. doi:10.1093/nar/gkg006

22. Anderson-Lee J, Fisker E, Kosaraju V, Wu M, Kong J, Lee J, et al. Principles for predicting RNA secondary structure design difficulty. J Mol Biol. 2016;428: 748–757. doi:10.1016/j.jmb.2015.11.013

23. Koodli RV, Keep B, Coppess KR, Portela F, Eterna participants, Das R. EternaBrain: Automated RNA design through move sets and strategies from an Internet-scale RNA videogame. PLoS Comput Biol. 2019;15: e1007059. doi:10.1371/journal.pcbi.1007059

24. Shi J, Das R, Pande VS. SentRNA: Improving computational RNA design by incorporating a prior of human design strategies. arXiv preprint arXiv: 180303146. 2018.

25. Runge F, Stoll D, Falkner S, Hutter F. Learning to Design RNA. ICLR 2019. 2018.

26. Portela F. An unexpectedly effective Monte Carlo technique for the RNA inverse folding problem. BioRxiv. 2018. doi:10.1101/345587

27. Do CB, Woods DA, Batzoglou S. CONTRAfold: RNA secondary structure prediction without physics-based models. Bioinformatics. 2006;22: e90-8. doi:10.1093/bioinformatics/btl246

28. Wayment-Steele HK, Kladwang W, Eterna Participants, Das R. RNA secondary structure packages ranked and improved by high-throughput experiments. BioRxiv. 2020. doi:10.1101/2020.05.29.124511

29. Zhou T, Dai N, Li S, Ward M, Mathews DH, Huang L. RNA design via structure-aware multifrontier ensemble optimization. Bioinformatics. 2023;39: i563–i571. doi:10.1093/bioinformatics/btad252

30. Leppek K, Byeon GW, Kladwang W, Wayment-Steele HK, Kerr CH, Xu AF, et al. Combinatorial optimization of mRNA structure, stability, and translation for RNA-based therapeutics. Nat Commun. 2022;13: 1536. doi:10.1038/s41467-022-28776-w

31. Wirecki TK, Lach G, Badepally NG, Moafinejad SN, Jaryani F, Klaudel G, et al. DesiRNA: structure-based design of RNA sequences with a replica exchange Monte Carlo approach. Nucleic Acids Res. 2025;53. doi:10.1093/nar/gkae1306