IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 January 2024, accepted 21 January 2024, date of publication 25 January 2024, date of current version 1 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3358426

RESEARCH ARTICLE

Evolutionary Strategy to Enhance an RNA Design Tool Performance

ÁLVARO RUBIO-LARGO[1], LAURA ESCOBAR-ENCINAS[1], NURIA LOZANO-GARCÍA[1], AND JOSÉ MARIA GRANADO-CRIADO[2]

[1]Department of Computers and Telematics Systems Engineering, Escuela Politécnica, University of Extremadura, 10003 Cáceres, Spain
[2]Department of Computer and Communications Technologies, Escuela Politécnica, University of Extremadura, 10003 Cáceres, Spain

Corresponding author: Álvaro Rubio-Largo (arl@unex.es)

This work was supported in part by the Junta de Extremadura, Spain, funded by the European Regional Development Fund (ERDF) through the European Union (EU) under Contract IB20047 (iARN Project); in part by the Ministry of Science, Innovation and Universities (MCIU), Spain; in part by the State Research Agency (AEI), Spain; and in part by the ERDF through EU funded by MCIN/AEI/10.13039/501100011033/FEDER, EU under Project PID2022-137275NA-I00 (Multi-HPC-Bio Project).

ABSTRACT At present, designing an RNA sequence that folds into a specific secondary structure is a problem that is not fully solved, due to its exponentially increasing complexity. To address this matter, many computational methods have been developed, but none of them has been able to completely and in an affordable time solve Eterna100, a widely recognized benchmark used to test the performance of RNA inverse folding algorithms. In previous publications we presented the m2dRNAs tool, a Multiobjective Evolutionary Algorithm, and its extension eM2dRNAs, which added a recursive decomposition of the target structure, thus simplifying the problem. At that time they successfully improved the ability to solve the RNA inverse folding problem, but were still unable to complete the Eterna100 benchmark. Here we introduce ES+eM2dRNAs, an improvement of eM2dRNAs that optimizes the decomposition process, as a drawback in its nature was identified.A comparative study of this new tool against its predecessors and other RNA design methods was performed using the two current versions of the Eterna100 benchmark. ES+eM2dRNAs was shown to be the best in all performance indicators considered (number of structures solved, success rate, and total run time). Moreover, it is able to solve two Eterna100 structures for which none of the compared methods had ever found a solution.

INDEX TERMS Bioinformatics, evolutionary strategy, genetic algorithm, multiobjective evolutionary algorithm, multiobjective optimization problem, ncRNA design, RNA inverse folding, RNA secondary structure decomposition strategy.

I. INTRODUCTION

The RNA inverse folding problem consists in discovering a nucleotide sequence that folds into a specific secondary structure, known as target structure [1]. In this way, since the 3D structure determines the function, it is possible to design a non-coding RNA (ncRNA) molecule that fulfills a particular application. The nucleotide sequence is known as primary RNA structure and is usually coded as a sequence of four nucleotides: Adenine (A), Guanine (G), Cytosine (C), and Uracil (U). From this, nucleotides in the sequence interact to establish hydrogen bonds between two specific nucleotides, giving rise to canonical Watson-Crick base-pairs (AU, UA, GC, CG) [2], [3] or less commonly to wobble base pairs (UG/GU) [4]. Those base pairing interactions fold the RNA molecule, resulting first in the secondary structure, and then in the 3D structure (tertiary structure) [5].

Functional non-coding RNAs (ncRNAs) play essential roles in various biological processes, including splicing, regulation of gene expression, inactivation of human X-chromosome, translation, control of chromatin in epigenetic processes or mRNA stability [6]. The ease of constructing synthetic RNAs [7] has attracted significant attention, considering their potential as a powerful biotechnological

The associate editor coordinating the review of this manuscript and approving it for publication was Qiang Li.

tool. At present, these synthetic RNAs find applications in diverse domains such as constructing ribozymes and riboswitches to be used as drug or therapeutic agents [8], building self-assembling structures utilizing RNA molecules (nano-biotechnology) [9], and advancing the field of synthetic biology [10]. For the purpose of designing an ncRNA molecule that can perform a specific function, it is necessary to develop computational methods that successfully solve the RNA inverse folding problem in a reasonable time.

Brute force can be used to solve this problem [11], however with a complexity that exponentially increases as a function of the length of the target structure. Even considering the main feature of the RNA molecule folding, which is that the paired positions have to form the valid base pairs mentioned above, complexity and in consequence time remain exponential and consequently not affordable. To circumvent this problem, many alternative approaches have been developed.

A. MOTIVATION

Nowadays there is no computational method capable of designing any possible RNA target structure in an affordable time. To compare the capabilities of the methods being designed it is common practice to test them against a standard benchmark. Currently, no method has yet been able to completely solve the Eterna100 benchmark, so it is an open challenge in this field.

In previous publications we have developed m2dRNAs and its improvement eM2dRNAs, two RNA Design Tools based on evolutionary algorithms. As we are going to see, m2dRNAs does not include any pre-treatment of the target structure to ease its processing, so it has to handle the whole structure at once. This matter was tackled by eM2dRNAs, which incorporates a recursive decomposition strategy of the target structure at the beginning and creates a dependency graph. But the decomposition strategy is based on a greedy procedure, which does not necessarily result in an optimal dependency graph. In this paper we incorporate an Evolutionary Strategy (ES) that optimizes the eM2dRNAS decomposition process, in order to improve the performance of the algorithm both in quality and execution time.

II. STATE OF THE ART

In 1994 was presented the first RNA inverse folding tool. Since then, several methods have been introduced to tackle the RNA design problem from various perspectives. In [12] tools published during the period from 1994 to 2016 were summarized. Here we offer a condensed version of all the tools present in that summary, to show the evolution of this topic and the wide range of approaches that have been applied: RNAinverse [1] utilizes an adaptive random walk to minimize the difference between the Minimum Free Energy (MFE) secondary structure of the ongoing RNA sequence and the target structure; RNA-SSD [13] employs a stochastic local search after an initialization process to select an initial RNA sequence in a greedy manner; INFO-RNA [8] involves two steps: generating good initial sequences by means of a dynamic programming method and a refined stochastic local search; MODENA [14], [15], [16] follows the fast Non-Dominated Sorting Genetic Algorithm (NSGA-II), using two objective functions involving structure: 1) stability and 2) similarity; The NUPACK suite [17], [18] contains an RNA designer that employs an algorithm similar to RNA-SSD; The genetic algorithm fRNAkenstein [19] solves the multitarget version of the RNA design problem, where multiple target structures need to be found simultaneously; DSS-Opt [20] makes use of Newtonian dynamics in sequence space, simulated annealing and a negative design term; RNAiFOLD [21], [22] addresses the problem by means of constraint programming; EteRNA ensemble algorithm [23] is a folding procedure that combines strategies from EteRNA players and other RNA design software; ERD [24], [25] starts with pools of different components to create an initial sequence consistent with the target structure. The quality of this subsequences are then enhanced by an evolutionary algorithm; Lastly, the ant colony optimization method antaRNA [26], [27] manages multiple constraints to design RNA structures.

Methods published after that interval are reviewed in [28]. We summarize here the most relevant ones to this work, which are those that used the same benchmark as us (Eterna100, as we will see) and provide the full list of solved structures.

SentRNA [29] propose an initial solution sequence by means of a fully-connected neural network trained on solutions of the RNA design game EteRNA, submitted by human players. Then an adaptive walk algorithm, which integrates simple human design strategies, refines it.

In [30], an agent for RNA design (a neural network made of convolutional layers) was trained using Reinforcement Learning (RL) and a collection of randomly generated secondary structures to choose actions (modifying a single or two paired bases) that ultimately lead to a sequence with the specified target structure.

LEARNA [31] utilizes deep RL to train a policy network to sequentially create a RNA sequence. Next it locally adapts it with the help of the Hamming Distance (HD) between the candidate and the target structure as a measure of error. Furthermore, by meta-learning on a great compilation of biological sequences, the Meta-LEARNA extension develops a unique RNA design policy.

The goal in [32] is to prove how strategies of Eterna participants can refine automated computational RNA design. A Convolutional Neural Network was trained from a repository of top players' moves in advanced puzzles and subsequently extended with a Single-Action-Playout of six canonical human strategies, giving rise to the EternaBrain-SAP method.

RNAPOND [33] couples both positive and negative design objectives. It employs a Fixed-Parameter Tractable algorithm that adapts its sampled sequence distribution to prioritize solutions by identifying and eluding Disruptive Base Pairs.

MoiRNAiFold [34] is an upgrade of RNAiFold whose goal is to engineer complex functional RNAs, focusing on the control of gene expression via RNA-RNA interactions.

aRNAque [35] is an evolutionary algorithm with local mutations that minimizes three objective functions: HD, normalized energy distance and ensemble defect, which are used individually at different levels. It was updated in [36] by incorporating a Lévy-flight mutation scheme (local search integrated with infrequent significant leaps).

Next we present our previous contributions in this field: m2dRNAs and eM2dRNAs, on which ES+eM2dRNAs is based.

A. M2DRNAS ALGORITHM

The m2dRNAs algorithm is a Multiobjective Evolutionary Metaheuristic developed by Rubio-Largo et al. [12] to address the RNA inverse folding problem. This algorithm focuses on finding RNA sequences that minimize three objectives in the designed sequence:

- *Partition Function* [37]: To determine the partition function for the complete set of potential secondary structures of a given RNA sequence x, equation 1 can be employed:

$$f_1(x) = \sum_{S \epsilon S'(x)} e^{\frac{-\Delta G(S)}{RT}} \tag{1}$$

  where $-\Delta G$ means the Gibbs' free energy change, R indicates the universal gas constant, T symbolizes the absolute temperature (37 °C), and $S'(x)$ represents the collection of all potential secondary structures. The summation is carried out over this set.
- *Ensemble Diversity* [38]: This measure fundamentally represents the average base pair distance (number of base pairs present in one structure but not in the other), which is the more straightforward measure of distance between two structures. This calculation is performed for all structures in the Boltzmann ensemble and can be defined with respect to base pair probabilities p_{ij}, as in equation 2.

$$f_2(x) = \sum_{(i,j) \epsilon x} p_{ij} \cdot (1 - p_{ij}) \tag{2}$$

- *Nucleotides Composition*: This function is intended to promote diversity within the solution set, thus mitigating any significant bias that may arise in the composition of the designed sequences. The designed RNA sequence x is studied with respect to: Base-pairs percentages → $\%GC$: GC/CG, $\%AU$: AU/UA, and $\%GU$: GU/UG (distribution of the three types of base pairs along the paired positions in the target structure), unpaired bases percentages → $\%uA$, $\%uC$, $\%uG$, and $\%uU$ (nucleotides distribution in unpaired positions), and total bases distribution → $\%A$, $\%C$, $\%G$, and $\%U$ (total nucleotides distribution in the entire designed sequence). In consequence, the nucleotides composition objective function is calculated as equation 3:

$$\begin{aligned} f_3(x) = &\max\{\%GC, \%AU, \%UG\} \\ &+ \max\{\%uA, \%uC, \%uG, \%uU\} \\ &+ \max\{\%A, \%C, \%G, \%U\} \end{aligned} \tag{3}$$

In addition, similarity between the target structure and the secondary structure of the designed sequence is proposed as a constraint to be met. Similarity (σ) [15] is defined as equation 4:

$$\sigma(x) = \frac{n - d}{n} \tag{4}$$

where n represents the length of x, and d the number of nucleotide positions whose structure in the designed sequence is not the same as the corresponding one in the target structure.

If $\sigma(x) = 1$ the predicted and the target structures are exactly the same. Examples of this calculation can be found in [12].

The Multiobjective Evolutionary Algorithm (MOEA) used in m2dRNAs, known as *Fast Non-Dominated Sorting Genetic Algorithm* [39], is widely known in the field of multiobjective optimization. In [12], the RNA inverse folding problem is formulated as a multiobjective optimization problem with continuous variables, which allows the use of genetic operators suitable for this type of problem. From an input target structure S in dot-bracket notation, m2dRNAs starts processing it to create the sets of base pairs (B) and unpaired bases (U) positions. The encoding of the chromosome is a real-valued vector of length $|B| + |U|$,

$$X = \{\rho_1, ..., \rho_{|B|}, \rho_{|B|+1}, ..., \rho_{|B|+|U|}\}$$

where ρ is a real value in the range [0,1] that represents the type of base pair or unpaired nucleotide. The $|B|$ first elements save the ρ values of the base-pairs, while from $|B| + 1$ to $|B| + |U|$ ρ values of unpaired positions are stored.

There is a procedure to translate an input chromosome into an RNA sequence. The resulting RNA sequence will be evaluated to determine if it satisfies the similarity constraint and to calculate its objective functions. To initialize individuals, m2dRNAs randomly assign values for paired positions (B set). However, when selecting each unpaired position (U set), we consider its potential related base pairs. This approach minimizes the likelihood of generating unnecessary loops in the structure of the resulting RNA sequence.

Crossover and mutation operators utilized are those suggested by Deb et al. [39]: simulated binary crossover (SBX) and polynomial mutation. The crossover probability (p_c) and mutation probability (p_m) are set to 0.9 and $1/n'$ respectively, where $n = |B| + |U|$.

B. EM2DRNAS ALGORITHM

eM2dRNAs [28] is an improved version of m2dRNAs. This algorithm begins with the recursive decomposition of the input target structure, which simplifies the problem to be solved. The results obtained, compared to several published methods using the Eterna100 dataset, show that this proposal obtains significantly better results.

We summarize here the process followed by eM2dRNAs:

1) Given an input target structure, it is decomposed into one or more substructures using a recursive procedure. Each substructure thus becomes an RNA inverse problem (*RNAinv*) to be solved. If a repeated substructure is found, it is referenced instead of creating a new problem. This generates a directed acyclic graph G representing the dependencies between *RNAinv* problems.
2) Removal of those *RNAinv* problems that meet one of the following criteria
   - The outcome of multiplying the size of the substructure by the number of times it is referenced is less than 20. For example, if a substructure of size 19 occurs only once, it would be removed. However, a different substructure of size 10 arising three times in the target structure would be retained.
   - Each substructure that contains only one substructure will be removed, regardless of its size.
   - If a substructure is the sole child of the target structure, then it will be deleted.
3) Provided a global stopping criterion determined by execution time, it is allocated proportionally to the size of each substructure. To achieve this, we consider the *participation* of each problem. Initially, we sum up the sizes of all the structures, which we'll refer to as the "total" size. Next, we calculate the *participation* of each problem by dividing its structure size by the total size. Finally, we multiply the execution time by the *participation* of each problem, resulting in a specific runtime assigned to each *RNAinv* problem.
4) A topological ordering of the *RNAinv* problems that need to be solved is performed This ordering provides a linear sequence of all nodes (*RNAinv*-problems) in the directed acyclic graph G. In G, an edge uv indicates that problem u must be solved before problem v. This graph must be acyclic, as is unfeasible to obtain a topological ordering of the graphs containing cycles.
5) Find the solution of each *RNAinv* problem utilizing an adapted variant of the m2dRNAs algorithm. m2dRNAs algorithm receives as input parameters the substructure to be solved and the stopping criterion (max. execution time). Two groups are created: base-pairs (B) and unpaired (U). The input substructure is processed, storing in B and U the positions of the base-pairs or unpaired nucleotides respectively. Also, the subproblems positions (P) of the input structure are available from previous steps, so they are used to encode the input structure as a chromosome (X) of length $|B| + |U| + |P|$. The output will consist of a collection of all the discovered RNA sequences, which will be saved. Once each problem is solved, it is necessary to recalculate the *participation* of each remaining problem. This is important because it is possible that the assigned time for solving the problems is not fully utilized, due to m2dRNAs' stagnation detection mechanism. The assigned time will be redistributed in consequence. In this manner, simpler problems do not consume time that might be required for solving more intricate problems. If m2dRNAs do not solve a particular subproblem, eM2dRNAs will eliminate that subproblem and reconfigure the graph.
6) The final problem to be addressed will be the one encompassing the target structure. After solving it the first valid solution found or a set of non-dominated RNA sequences will be reported.

The success of eM2dRNAs in solving or not the target structure (as well as the time invested), lies fundamentally in the decomposition of the target structure.

III. METHODS

The main goal of this paper is the implementation of an ES that optimizes the decomposition process of a target RNA structure, in order to simplify the problem to be solved. The multi-objective metaheuristic m2dRNAs is the core that finds suitable solutions of the substructures, and eM2dRNAs is responsible for the initial decomposition and the managing of the substructures.

Another objective is to test the performance of ES+eM2dRNAs by developing a comparative study between our proposal and other algorithms published in the literature to address the RNA inverse folding problem.

Before explaining the methodology of both goals, we state the formal definition of the problem and explain the chromosome encoding, since it is a key feature of the algorithm

A. FORMAL DEFINITION OF THE PROBLEM

The problem addressed here can be formulated as a binary optimization problem.

$$\begin{aligned} &\text{Maximize:} \quad \sigma(\text{eM2dRNAs}(\mathbf{x})) \\ &\text{subject to} \quad x_k \in \{0,1\}, \quad k = 1,2,\ldots,|B| \end{aligned}$$

where:

- $\mathbf{x} = (x_1, x_2, \ldots, x_{|B|})$ is a binary vector with $|B|$ decision variables.
- Similarity$\rightarrow \sigma(\text{eM2dRNAs}(\mathbf{x}))$ is the objective function to maximize. It will be calculated between the predicted secondary structure of the RNA sequence resulting from calling a modified version of the eM2dRNAs algorithm

and the target structure. Similarity (σ) is defined as in equation 4.
- $|B|$ is the number of base-pairs in the target structure.

B. CHROMOSOME DEFINITION

In the ES, the individual's chromosome (x) has been defined as a binary vector $\mathbf{x} = (x_1, x_2, \ldots, x_{|B|})$, where $|B|$ is the number of paired bases in the target structure. In this encoding $x_k = 1$ means that the base-pair is the limit of a substructure, and $x_k = 0$ that is not. This binary representation is called the guide and will be received by the eM2dRNAs algorithm for the dependency graph creation process and subsequently solve the different subproblems to address the main one.

To clarify this definition, we provide some examples below, including what the dependency graph will look like when the guide is processed by eM2dRNAs (Recall that when encountering a duplicated substructure, it is referenced rather than generating a new problem). Consider the following target structure represented in dot-bracket notation:

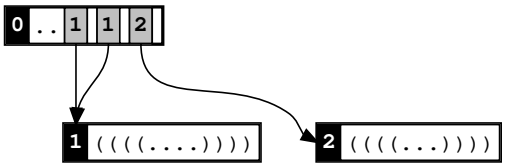
S : . . ((((. . . .)))) ((((. . . .)))) ((((. . .))))
($n = 37$)

As we can see, there are 12 base-pair ($|B| = 12$). Therefore, 3 example chromosomes could be:

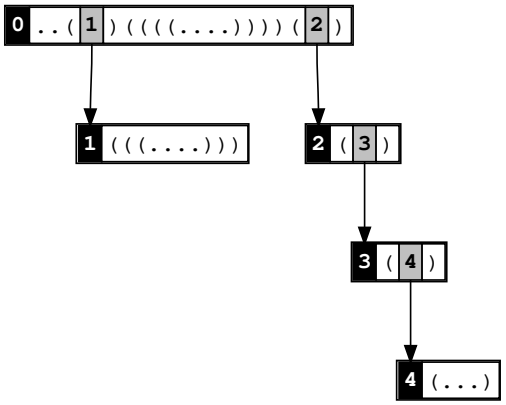
x_1 : 100010001000
x_2 : 010000000111
x_3 : 010101010101

As established before, a value of 1 means that the base-pair is the limit of a substructure, marking thus the limits for the decomposition. In consequence the decomposition will be:

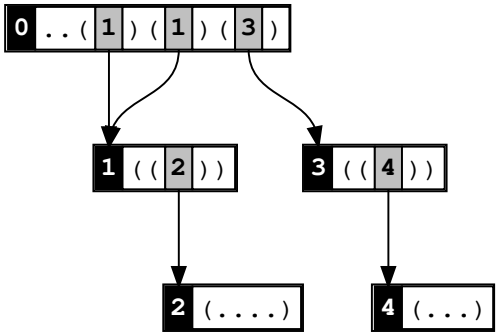
x_1 : . . ((((. . . .)))) ((((. . . .)))) ((((. . .))))



x_2 : . . ((((. . . .)))) ((((. . . .)))) ((((. . .))))



x_3 : . . ((((. . . .)))) ((((. . . .)))) ((((. . .))))



The output of eM2dRNAs with each of the guides will be an RNA sequence and its predicted secondary structure, with which the similarity to the target structure will be calculated.

C. ES+EM2DRNAS ALGORITHM

We analyze here the process followed by ES+eM2dRNAs (see Algorithm 1) that aims to find an RNA sequence that folds into a given target structure. It receives as input the target structure in dot-bracket notation (S) and the stopping criterion: maximum execution time of the algorithm (t_g). Its output will be the RNA sequence (X_x) found with the highest similarity to the target structure (S).

This process can be explained as a sequence of steps:

1) It starts by dividing the execution time t_g into two equal parts. In order to be able to perform at least 50 generations in the ES, the first of these halves, which will be the stopping criterion of the ES, is divided into 50 time intervals: t_l. These intervals will be used to control the execution time of the eM2dRNAs algorithm (as eM2dRNAs incorporates a stopping mechanism that can cause t_l not to time out). As an example, suppose t_g is 100 seconds. It will be divided into two parts of 50 s (amount of time that will be the stopping criterion) each. One is divided into 50 t_l intervals, so $t_l = 1$ s. The remaining part of 50 s is kept for later.
2) Next, a greedy initialization is applied (see line 2 in Algorithm 1). This initialization is the one exposed in the first step of eM2dRNAs, and leads to a initial guide x with a high probability of solving the problem. After this, modified eM2dRNAs is called to evaluate the initial guide x (see line 3 in Algorithm 1). It receives as input the target structure S, the guide x and the maximum execution time t_l. As a result eM2dRNAs will return a candidate RNA sequence (X_x) and its predicted secondary structure (S^x). To show this step, we are going to use the following structure as target, that corresponds to an RNA sequence of 129 nucleotides, identifying a total of 39 base pairs:

S : . (((. (((. . . (((. (((. . .))) . (((. .
. .))) .))) . . . (((. (((. . . .))) . ((((. .

Algorithm 1 ES+eM2dRNAs

Input: − A target structure in dot-bracket notation (S)
− Execution time (stopping-criterion) (t_g)
Output: − RNA sequence with maximum similarity (X_x)

1: $t_l \leftarrow (t_g/2)/50$; /* Execution time eM2dRNAs in ES */
2: $x \leftarrow$ GreedyInitialization(S);
/* eM2dRNAs returns, for a guide x, a RNA sequence (X_x) and its predicted secondary structure (S^x) */
3: $\{X_x, S^x\} \leftarrow$ eM2dRNAs (S, x, t_l);
4: $g_e \leftarrow 0$; /* Stagnation control */
5: $p_m \leftarrow 0.01$; /* Mutation factor */
/* Evolutionary strategy (ES) */
6: **while** $\neg stopping-criterion(t_g * 0.5)$ & $\sigma(S^x, S) \neq 1$ **do**
7: $y \leftarrow$ Mutation (x, p_m);
8: $\{X_y, S^y\} \leftarrow$ eM2dRNAs (S, y, t_l);
9: **if** $\sigma(S^y, S) > \sigma(S^x, S)$ **then**
10: $x \leftarrow y$;
11: $\{X_x, S^x\} \leftarrow \{X_y, S^y\}$;
12: $g_e \leftarrow 0$;
13: $p_m \leftarrow 0,01$;
14: **else**
15: $g_e \leftarrow g_e + 1$;
16: **if** $g_e = 5$ **then**
17: $p_m \leftarrow p_m * 1.2$;
18: $g_e \leftarrow 0$;
19: **end if**
20: **end if**
21: **end while**
/* In case the ES does not find a solution, use the best guide (x) until the execution time t_g is exhausted */
22: **if** $\sigma(S^x, S) \neq 1$ **then**
23: $t_l \leftarrow$ calculateRemainingTime(); /* At least: $t_g * 0.5$ */
24: $\{X_x, S^x\} \leftarrow$ eM2dRNAs(S, x, t_l);
25: **end if**

```
.))))..)))...(((.(((....))).((((..
.))))..)))...)))..((((...)))).))))..
```

The greedy initialization decomposes this structure as shown in Fig. 1.

The chromosome representing this decomposition would be a vector (x) of size 39, which is the initial guide:

$$x: \texttt{100100100100100100}$$
$$\texttt{001000100100100010000}$$

The target structure S, the obtained initial guide x and $t_l = 1$ are used as input for eM2dRNAs, which returns a candidate RNA sequence (X_x) and its predicted secondary structure (S^x).

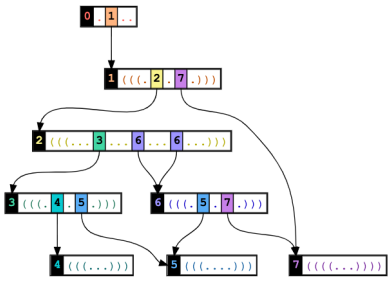


FIGURE 1. Graph obtained by the greedy initialization for the target structure S.

3) In lines 4 and 5 of Algorithm 1, the stagnation counter g_e is initialized to 0 and the mutation factor p_m to 0.01 (1%). Then, the ES is started (see line 6 in Algorithm 1), which will continue as long as none of these conditions are met: that the stopping criterion (which is related to the execution time) is reached and that the similarity between X_x and the target structure S is equal to 1. As calculated before, in our example the stopping criterion is 50 s.
4) At each iteration of the loop, the guide resulting from mutating the best guide so far (x) is stored in y (see line 7 in Algorithm 1). The mutation process is very simple: the chromosome of x is traversed and, with a probability of p_m, the binary value 0 or 1 is interchanged.
With this new guide (y), a call is made to the modified eM2dRNAs algorithm to obtain the generated candidate RNA sequence (X_y) and its predicted secondary structure (S^y) (see line 8 in Algorithm 1). If the similarity obtained from the y guide is better than that obtained from the x guide (see line 9 in Algorithm 1), the x guide is replaced by y, as well as the candidate RNA sequence and secondary structure (see lines 10 and 11 in Algorithm 1). The stagnation counter g_e and mutation factor p_m are reinitialized to 0 and 0.01 respectively (see lines 12 and 13 in Algorithm 1). ES+eM2dRNAs has a stagnation control (see lines 14-18 in Algorithm 1), if the guide does not improve within 5 generations, the mutation factor p_m is increased by 20%, thus trying to avoid possible local optima.
We are going to assume that with S^y, which is the structure predicted from the new guide y, the similarity between S^y and the target structure S is 0.9. If the similarity between S^x, the structure predicted from the initial guide x were, for example, 0.95, since it is greater (better) than 0.9, x would be maintained and g_e would increase by 1. On the other hand, if it were 0.8, y would replace x, X_y and S^y would replace X_x and S^x, as well as g_e and p_m would be reset to their original values.
5) Once the ES is finished, it is checked whether an RNA sequence identical to S has been found (see line 22 in Algorithm 1). If no such RNA sequence has been

found, the remaining execution time is calculated (see line 23 in Algorithm 1) and the best guide found during the whole evolutionary process of the ES is used in a last call to eM2dRNAs, assigning it the remaining execution time (see line 24 in Algorithm 1). In our example, we suppose that the ES has finished once its stopping criterion (50 s) has been reached, having therefore not found an RNA sequence whose similarity to the target structure is 1. Therefore the remaining time (which will be at least the second part of 50 s that was saved at the beginning) is calculated, and will be used as the time limit for the last call to eM2dRNAs with the guide stored in x (the best one) instead of $t_l = 1$ s as the previous ones.

D. COMPARATIVE STUDY

The performance evaluation of ES+eM2dRNAs and various RNA inverse folding methods utilized the broadly recognized benchmark Eterna100. This benchmark consists of 100 RNA secondary structures obtained from the EteRNA puzzle game, encompassing a diverse range of design complexities, ranging from simple hairpins to intricate 400-nucleotide designs.

Initially, the Eterna100 dataset (Eterna100-V1) was assembled using ViennaRNA 1.8.5, which relied on the Turner1999 (T99) energy parameters [40]. However, upon identifying 19 targets that were insoluble by ViennaRNA 2.4, which employs Turner2004 (T04) energy parameters [41], these structures were reengineered. The updated dataset, known as Eterna100-V2 [42], includes the modified 19 structures, aiming to be solvable by any inverse folding algorithm using default ViennaRNA 2.4 or any version utilizing Turner2004 parameters. Both versions of Eterna100 were utilized in the evaluation of eM2dRNAs, along with its corresponding set of Turner energy parameters.

The parameter configuration for ES+eM2dRNAs consisted of the following settings: a population size of 52 individuals, with a stopping criterion determined by time, specifically 24 hours. The genetic operators employed were SBX (Simulated Binary Crossover) with parameters $\eta_c = 10$ and $p_c = 0.9$, along with polynomial mutation using $\eta_m = 5$ and $p_m = 10/n'$, where n' represents the total number of base pairs and unpaired positions. The compilation of the program was done using g++ (GCC) version 7.5.0.

For each combination of Eterna100-version and Turner-version (Eterna100-V1T99 and Eterna100-V2T04), all structures were attempted ten times during the evaluation process.

The methods selected to participate in this comparative study were those for which Eterna100-V1T99 and/or Eterna100-V2T04 results are available, including a list detailing the solved structures (LEARNA and the method presented by Eastman et al. in 2018 were discarded because they used Eterna100-V1T04). The results of RNAinverse, RNA-SSD, INFO-RNA, MODENA, DSS-Opt, and m2dRNAs were collected from the source [12]. As for the remaining methods, the data was obtained directly from their respective sources.

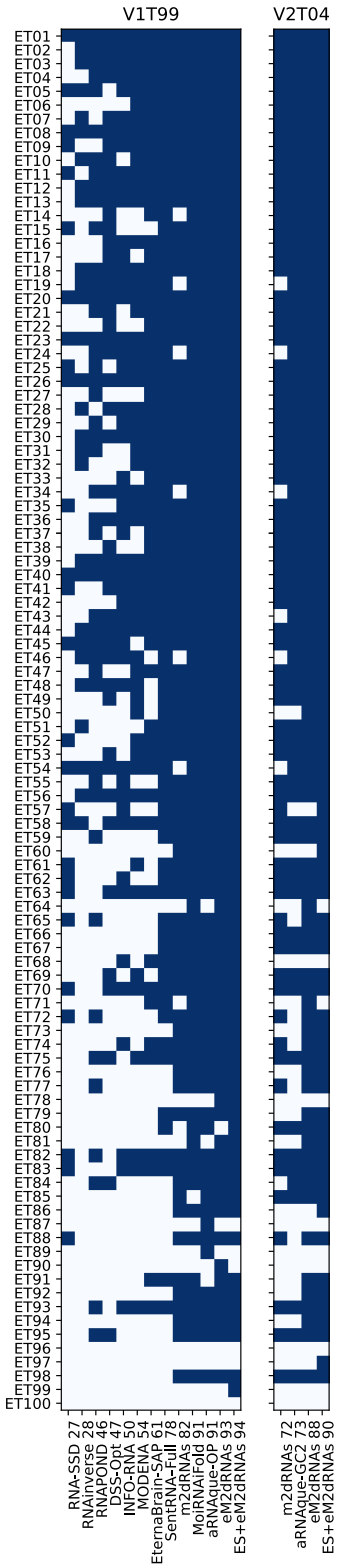


FIGURE 2. Comparison among diverse methods when solving the Eterna100 (Version1/Version2, V1/V2) benchmark for their corresponding energy parameters Turner1999 (T99) and Turner2004 (T04). The successes (■) and failures ( ) are shown for each algorithm. The number of solved structures is shown next to the name of each method.

These methods include SentRNA-Full, EternaBrain-SAP, RNAPOND, MoiRNAiFold, aRNAque, and eM2dRNAs.

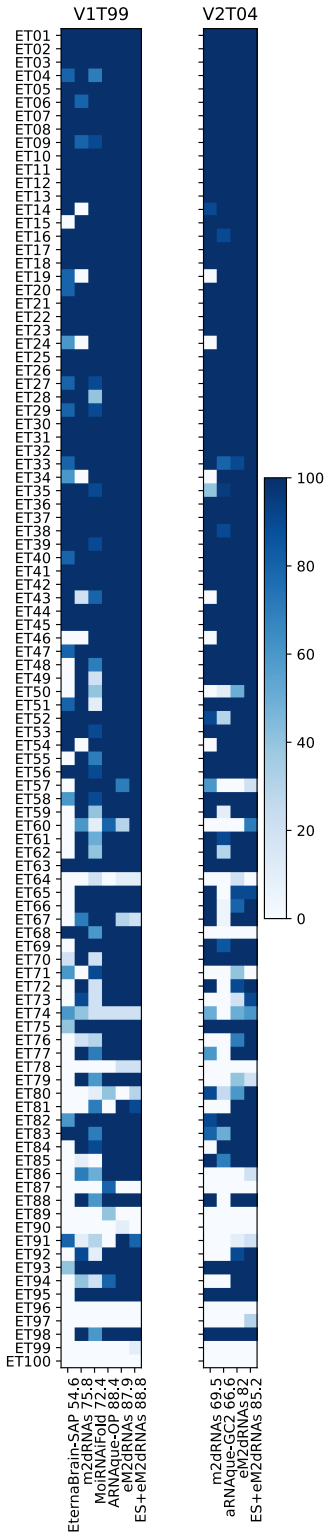


FIGURE 3. Comparison, in percentage of executions solving each structure, among diverse methods when solving the Eterna100 benchmark. All of them (100%) is represented by (■) and none (0%) as ( ). The overall percentage of solved structures is shown next to the name of each method.

IV. EXPERIMENTAL RESULTS

The comparative study is divided into two parts, depending on the version of Eterna100 and Turner energy parameters used

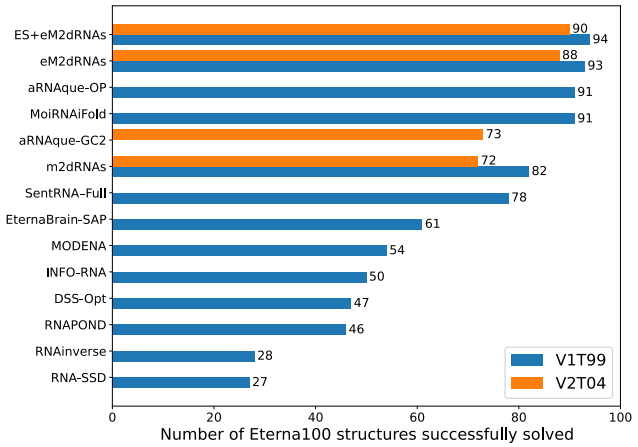


FIGURE 4. Comparison, in number of successfully solved structures, among diverse methods when solving the Eterna100 (Version1/Version2, V1/V2) benchmark for their corresponding energy parameters Turner1999 (T99) and Turner2004 (T04).

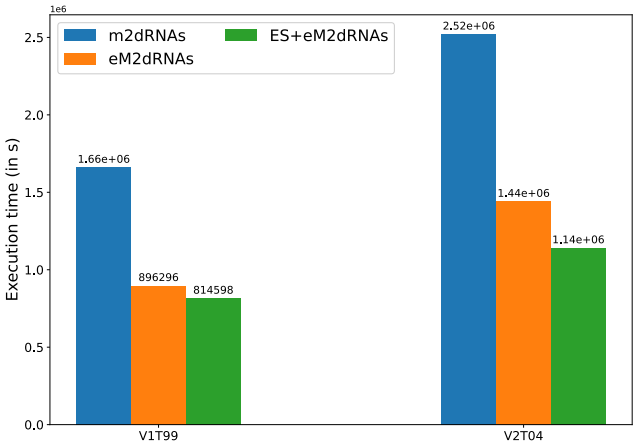


FIGURE 5. Total execution time required by m2dRNAs, eM2dRNAs and ES+eM2dRNAs in designing all the Eterna100 structures (in seconds).

(Eterna100-V1T99 and Eterna100-V2T04). Fig. 2 shows the Eterna100 structures solved by each individual method, whereas the heatmap in Fig. 3 represents the percentage of executions that solve each structure, when the data to perform this calculation are available. Also, the counts of number of successfully solved structures by each method can be found in Fig. 4. We then proceed to evaluate the effectiveness of various methods as success rate, defined as the percentage of successful runs in relation to the total number of executions. Due to data availability, this calculation is limited to Meta-LEARNA, EternaBrain-SAP, MoiRNAiFold, aRNAque (−OP and −GC2), m2dRNA, eM2dRNAs, and of course ES+eM2dRNAs. Therefore, this comparison will be conducted exclusively among these competitors (Table 1). Finally, we compare execution times of m2dRNAs, eM2dRNAs and ES+eM2dRNAs, since they were run in the same machine (Fig. 5).

As we can observe, for Eterna100-V1T99, ES+eM2dRNAs solves 94 of the 100 structures, improving

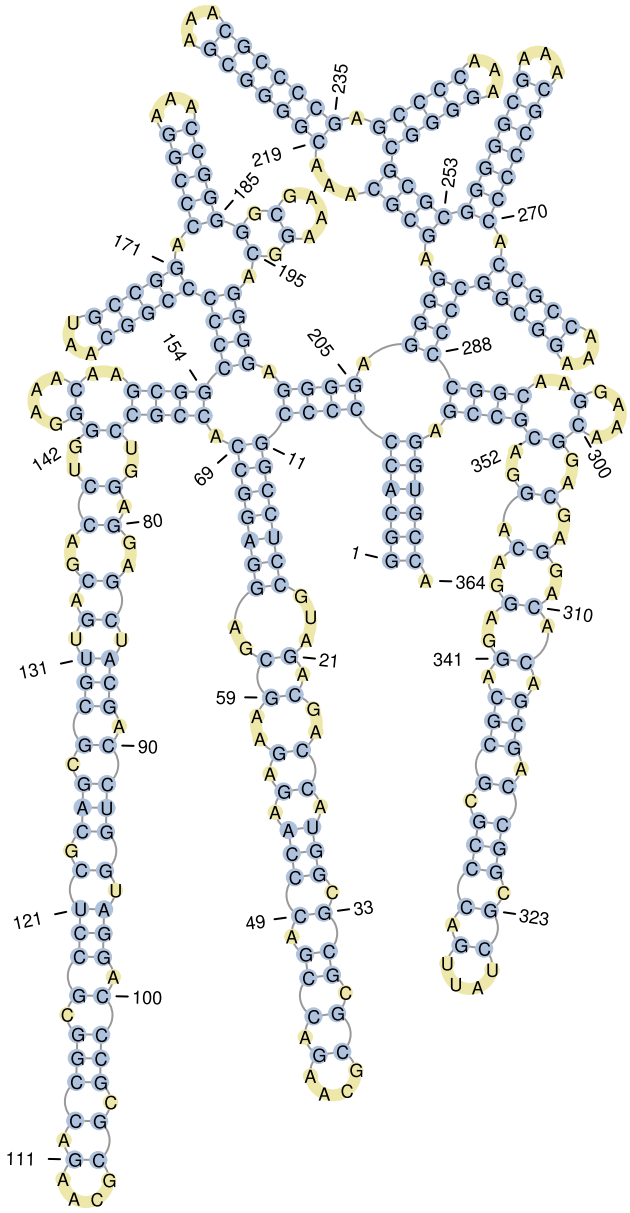


FIGURE 6. Solution found for ET99 structure (V1T99).

TABLE 1. Comparison among diverse methods in terms of success rate (in %).

Version	Success Rate (in %)	
V1T99	ES+eM2dRNAs	88.8
	aRNAque-OP	88.4
	eM2dRNAs	87.9
	m2dRNAs	75.8
	MoiRNAiFold	72.4
	EternaBrain-SAP	54.6
V2T04	ES+eM2dRNAs	85.2
	eM2dRNAs	82
	m2dRNAs	69.5
	aRNAque-GC2	66.6

the results of eM2dRNAs by one, which was already superior to the rest of the algorithms shown. We can highlight that for the structure **ET99**, ES+eM2dRNAs is the only algorithm able to solve it in the scientific literature. Its structure with the RNA sequence found embedded is shown in Fig. 6, represented by PseudoViewer3 web server [43] (http://pseudoviewer.inha.ac.kr/). In terms of success rate, ES+eM2dRNAs is the winner with 88.8%, followed closely by aRNAque-OP (88.4%) and eM2dRNAs

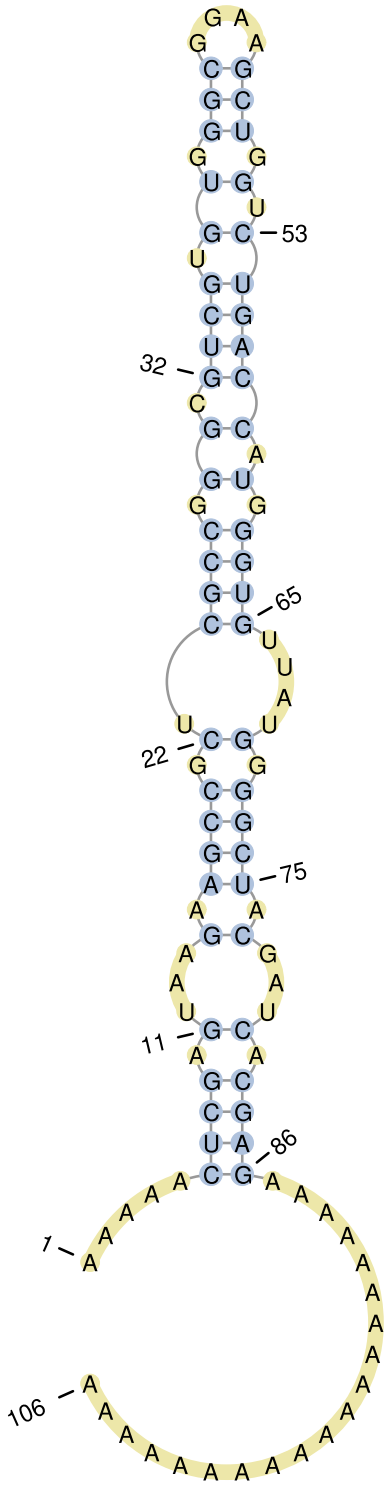


FIGURE 7. Solution found for ET60 structure (V2T04).

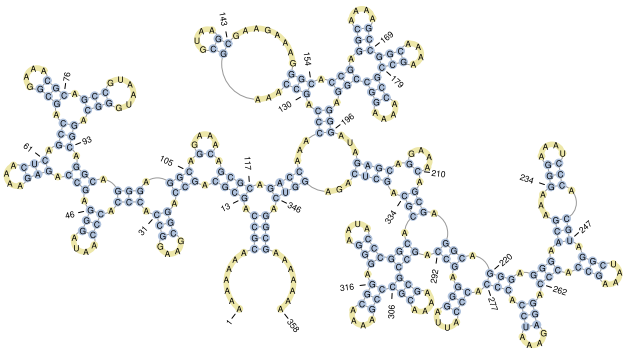


FIGURE 8. Solution found for ET86 structure (V2T04).

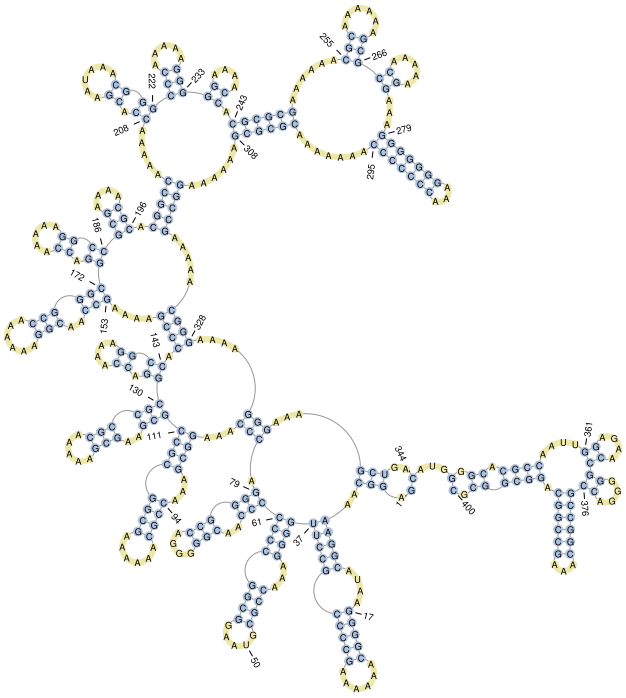


FIGURE 9. Solution found for ET97 structure (V2T04).

(87.9%). Moreover, execution time of ES+eM2dRNAs is better than eM2dRNAs, which was already significantly better than m2dRNAs.

For Eterna100-V2T04, ES+eM2dRNAs solves 90 of the 100 structures. Apart from our previously published methods, to our knowledge the only alternative method tested with Eterna100-V2T04 is aRNAque. To use it in our comparative study, we collected the data of its best configuration (aRNAque-GC2, Lévy flight version) from their GitHub repository. As we can see in Fig. 4, ES+eM2dRNAs beats all contenders, wining by two structures over the second (eM2dRNAs), and both methods improving quite a lot the results of the remaining algorithms (aRNAque-GC2 and m2dRNAs, with 73 and 72 respectively). ES+eM2dRNAs is the only method within this category that solves ET60 (Fig. 7), ET86 (Fig. 8) and **ET97** (Fig. 9). Remarkably, this last structure has also not been solved by any published method. About success rate, ES+eM2dRNAs is the best (85.2%) and only eM2dRNAs is not far behind (82%). As before, ES+eM2dRNAs is also the best in execution time.

These results demonstrate the better performance of ES+eM2dRNAs compared to the other methods. Given that this is the method that solves the most structures and has the best success rate, and that the Eterna100 benchmark includes a wide variety of structure types, it is the most versatile. This is also supported by its ability to find solutions to structures that no other tool has been able to achieve. Thus, this tool would be the most reliable for trying to find RNA sequences that fold into a desired target structure.

As discussed in the introduction, time is also an important feature to consider here, as it is necessary to be able to obtain the solution RNA sequences in affordable times. Although not many tools are compared with this metric (due to the need to be run under the same conditions), the fact that it is the fastest is a further point in favor of ES+eM2dRNAs.

V. CONCLUSION

We have developed a new enhancement to our previously presented m2dRNAs RNA design tool, which was first improved by eM2dRNAs adding a recursive decomposition of the target structure. Since the greedy procedure used by eM2dRNAs does not necessarily result in an optimal dependency graph, we have incorporated an ES to optimize this decomposition process and improve the performance of the core MOEA algorithm. In consequence, this new tool is called ES+eM2dRNAs.

We also present a comparative study of ES+eM2dRNAs against its predecessors and other tools at the scientific literature, to test the performance of this extension and show the progression of our algorithm as it is improved. The two available versions of the Eterna100 benchmark in combination with its corresponding versions of Turner energy parameters were used in this comparative. The selected alternative tools were: RNA-SSD, RNAinverse, RNAPOND, DSS-Opt, INFO-RNA, MODENA, EternaBrain-SAP, SentRNA-Full, MoiRNAiFold, and aRNAque. ES+eM2dRNAs outperforms all contenders in all categories considered (Number of structures solved, success rate, and total execution time) for both versions of Eterna100. Moreover, ES+eM2dRNAs is the first known method that solves ET97 and ET99.

To summarize, as main contributions of ES+eM2dRNAs we can mention:

- It is the first RNA design method capable of solving 94 of the 100 RNA secondary structures of the first version and/or 90 of the second version of the Eterna100 benchmark, widely used in this area.
- It manages to find solutions to two structures that had not been solved by any other computational method.
- It also has better success rate and total execution time than its predecessors.

- To the best of our knowledge, there is no other computational tool that applies an Evolutionary Strategy to decompose the RNA target structure at the beginning of the RNA design process.

RNA design tools allow prediction of RNA sequences that fold into a desired secondary structure, which determines function. As ES+eM2dRNAs is currently the RNA design tool that proves to obtain the best results, any decision on which RNA sequences should be constructed and experimentally tested to achieve a ncRNA with the desired function could be supported by computational design of candidate RNA sequences using this tool. This reduces time and funding efforts.

Future applications of this work include any field where it is necessary to find RNA sequences that fold into a desired target structure, thus being candidates to fulfill a desired function once synthesized. Synthetic RNAs are used in diverse biotechnological areas, such as nano-biotechnology, biomedical engineering and synthetic biology.

Future research efforts could be aimed at modifying this tool in order to improve its ability to solve structures or decrease execution time. Related actions that could be taken could be directed at:

- Modify the ES itself (changing the mutation operator or using a different ES).
- Completely change the early decomposition strategy to something other than the ES presented here or the recursive decomposition used in eM2dRNAs.
- Modify the core m2dRNAs algorithm to: Optimize other objective functions (such as those used by other published tools), use another existing multiobjective algorithm, or change the chromosome encoding, mutation operator, and/or crossover operator.

Another possible action to extend this work would be to repeat the comparative study utilizing other sets of structures, for example larger and more complex ones, to analyze the scalability of the algorithms included in it.

Finally, it would be interesting to widen the capabilities of ES+eM2dRNAs by adding desirable features in this field, such as the possibility of including design constraints (base-pairs bounds and specified motifs), allowing for pseudoknots, and multi-target RNA design.

From an algorithmic point of view, we are interested in studying whether existing genetic algorithms, differential evolution or particle swarm optimizers will perform well on the RNA design problem. In the future we will perform a comparison with state-of-the-art algorithms using the same initialization strategy.

REFERENCES

[1] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster, "Fast folding and comparison of RNA secondary structures," *Monatshefte für Chem. Chem. Monthly*, vol. 125, no. 2, pp. 167–188, Feb. 1994.

[2] N. C. Seeman, J. M. Rosenberg, F. L. Suddath, J. J. P. Kim, and A. Rich, "RNA double-helical fragments at atomic resolution: I. The crystal and molecular structure of sodium adenylyl-3′,5′-uridine hexahydrate," *J. Mol. Biol.*, vol. 104, no. 1, pp. 109–144, Jun. 1976.

[3] J. M. Rosenberg, N. C. Seeman, R. O. Day, and A. Rich, "RNA double-helical fragments at atomic resolution: II. the crystal structure of sodium guanylyl-3′,5′-cytidine nonahydrate," *J. Mol. Biol.*, vol. 104, no. 1, pp. 145–167, Jun. 1976.

[4] G. Varani and W. H. McClain, "The G·U wobble base pair," *EMBO Rep.*, vol. 1, no. 1, pp. 18–23, Jul. 2000.

[5] I. Tinoco and C. Bustamante, "How RNA folds," *J. Mol. Biol.*, vol. 293, no. 2, pp. 271–281, Oct. 1999.

[6] S. Hombach and M. Kretz, *Non-Coding RNAs: Classification, Biology and Functioning*. Cham, Switzerland: Springer, 2016, pp. 3–17.

[7] C. B. Reese, "Oligo- and poly-nucleotides: 50 years of chemical synthesis," *Organic Biomolecular Chem.*, vol. 3, no. 21, p. 3851, 2005.

[8] A. Busch and R. Backofen, "INFO-RNA—A fast approach to inverse RNA folding," *Bioinformatics*, vol. 22, pp. 1823–1831, May 2006.

[9] M. Qiu, E. Khisamutdinov, Z. Zhao, C. Pan, J.-W. Choi, N. B. Leontis, and P. Guo, "RNA nanotechnology for computer design and in vivo computation," *Phil. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 371, Oct. 2013, Art. no. 20120310.

[10] S. Meyer, J. Chappell, S. Sankar, R. Chew, and J. B. Lucks, "Improving fold activation of small transcription activating RNAs (STARs) with rational RNA engineering strategies," *Biotechnol. Bioengineering*, vol. 113, no. 1, pp. 216–225, Sep. 2015.

[11] A. Churkin, M. D. Retwitzer, V. Reinharz, Y. Ponty, J. Waldispühl, and D. Barash, "Design of RNAs: Comparing programs for inverse RNA folding," *Briefings Bioinf.*, vol. 19, Jan. 2017, Art. no. bbw120.

[12] Á. Rubio-Largo, L. Vanneschi, M. Castelli, and M. A. Vega-Rodríguez, "Multiobjective metaheuristic to design RNA sequences," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 156–169, Feb. 2019.

[13] M. Andronescu, A. P. Fejes, F. Hutter, H. H. Hoos, and A. Condon, "A new algorithm for RNA secondary structure design," *J. Mol. Biol.*, vol. 336, no. 3, pp. 607–624, Feb. 2004.

[14] A. Taneda, "MODENA: A multi-objective RNA inverse folding," *Adv. Appl. Bioinf. Chem.*, vol. 2011, pp. 1–12, Dec. 2010.

[15] A. Taneda, "Multi-objective genetic algorithm for pseudoknotted RNA sequence design," *Frontiers Genet.*, vol. 3, p. 36, Apr. 2012.

[16] A. Taneda, "Multi-objective optimization for RNA design with multiple target secondary structures," *BMC Bioinf.*, vol. 16, no. 1, p. 280, Sep. 2015.

[17] J. N. Zadeh, B. R. Wolfe, and N. A. Pierce, "Nucleic acid sequence design via efficient ensemble defect optimization," *J. Comput. Chem.*, vol. 32, no. 3, pp. 439–452, Aug. 2010.

[18] J. N. Zadeh, C. D. Steenberg, J. S. Bois, B. R. Wolfe, M. B. Pierce, A. R. Khan, R. M. Dirks, and N. A. Pierce, "NUPACK: Analysis and design of nucleic acid systems," *J. Comput. Chem.*, vol. 32, no. 1, pp. 170–173, Nov. 2010.

[19] R. B. Lyngsø, J. W. Anderson, E. Sizikova, A. Badugu, T. Hyland, and J. Hein, "Frnakenstein: Multiple target inverse RNA folding," *BMC Bioinf.*, vol. 13, no. 1, p. 260, Oct. 2012.

[20] M. C. Matthies, S. Bienert, and A. E. Torda, "Dynamics in sequence space for RNA secondary structure design," *J. Chem. Theory Comput.*, vol. 8, no. 10, pp. 3663–3670, Aug. 2012.

[21] J. A. García-Martín, P. Clote, and I. Dotú, "RNAiFold: A constraint programming algorithm for RNA inverse folding and molecular design," *J. Bioinf. Comput. Biol.*, vol. 11, no. 2, Apr. 2013, Art. no. 1350001.

[22] J. A. Garcia-Martin, I. Dotu, and P. Clote, "RNAiFold 2.0: A web server and software to design custom and Rfam-based RNA molecules," *Nucleic Acids Res.*, vol. 43, no. W1, pp. W513–W521, May 2015.

[23] J. Lee, W. Kladwang, M. Lee, D. Cantu, M. Azizyan, H. Kim, A. Limpaecher, S. Gaikwad, S. Yoon, A. Treuille, and R. Das, "RNA design rules from a massive open laboratory," *Proc. Nat. Acad. Sci. USA*, vol. 111, no. 6, pp. 2122–2127, Jan. 2014.

[24] A. Esmaili-Taheri, M. Ganjtabesh, and M. Mohammad-Noori, "Evolutionary solution for the RNA design problem," *Bioinformatics*, vol. 30, no. 9, pp. 1250–1258, Jan. 2014.

[25] A. Esmaili-Taheri and M. Ganjtabesh, "ERD: A fast and reliable tool for RNA design including constraints," *BMC Bioinf.*, vol. 16, no. 1, p. 20, Jan. 2015.

[26] R. Kleinkauf, M. Mann, and R. Backofen, "AntaRNA: Ant colony-based RNA sequence design," *Bioinformatics*, vol. 31, no. 19, pp. 3114–3121, May 2015.

[27] R. Kleinkauf, T. Houwaart, R. Backofen, and M. Mann, "antaRNA—Multi-objective inverse folding of pseudoknot RNA using ant-colony optimization," *BMC Bioinf.*, vol. 16, no. 1, p. 389, Nov. 2015.

[28] Á. Rubio-Largo, N. Lozano-García, J. M. Granado-Criado, and M. A. Vega-Rodríguez, "Solving the RNA inverse folding problem through target structure decomposition and multiobjective evolutionary computation," *Appl. Soft Comput.*, vol. 147, Nov. 2023, Art. no. 110779.
[29] J. Shi, R. Das, and V. S. Pande, "SentRNA: Improving computational RNA design by incorporating a prior of human design strategies," 2018, *arXiv:1803.03146*.
[30] P. Eastman, J. Shi, B. Ramsundar, and V. S. Pande, "Solving the RNA design problem with reinforcement learning," *PLOS Comput. Biol.*, vol. 14, no. 6, Jun. 2018, Art. no. e1006176.
[31] F. Runge, D. Stoll, S. Falkner, and F. Hutter, "Learning to design RNA," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–29.
[32] R. V. Koodli, B. Keep, K. R. Coppess, F. Portela, and R. Das, "EternaBrain: Automated RNA design through move sets and strategies from an internet-scale RNA videogame," *PLOS Comput. Biol.*, vol. 15, no. 6, Jun. 2019, Art. no. e1007059.
[33] H.-T. Yao, J. Waldispühl, Y. Ponty, and S. Will, "Taming disruptive base pairs to reconcile positive and negative structural design of RNA," in *Proc. 25th Int. Conf. Res. Comput. Mol. Biol. (RECOMB)*, Paris, France, Apr. 2021.
[34] G. Minuesa, C. Alsina, J. A. Garcia-Martin, J. C. Oliveros, and I. Dotu, "MoiRNAiFold: A novel tool for complex in silico RNA design," *Nucleic Acids Res.*, vol. 49, no. 9, pp. 4934–4943, May 2021.
[35] N. S. C. Merleau and M. Smerlak, "A simple evolutionary algorithm guided by local mutations for an efficient RNA design," in *Proc. Genetic Evol. Comput. Conf.*, Jun. 2021, pp. 1027–1034.
[36] N. S. C. Merleau and M. Smerlak, "ARNAque: An evolutionary algorithm for inverse pseudoknotted RNA folding inspired by Lévy flights," *BMC Bioinf.*, vol. 23, no. 1, p. 335, Aug. 2022.
[37] J. S. McCaskill, "The equilibrium partition function and base pair binding probabilities for RNA secondary structure," *Biopolymers*, vol. 29, nos. 6–7, pp. 1105–1119, May 1990.
[38] R. Lorenz, M. T. Wolfinger, A. Tanzer, and I. L. Hofacker, "Predicting rna secondary structures from sequence and probing data," *Methods*, vol. 103, pp. 86–98, 2016.
[39] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[40] D. H. Mathews, J. Sabina, M. Zuker, and D. H. Turner, "Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure," *J. Mol. Biol.*, vol. 288, no. 5, pp. 911–940, May 1999.
[41] D. H. Mathews, M. D. Disney, J. L. Childs, S. J. Schroeder, M. Zuker, and D. H. Turner, "Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure," *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 19, pp. 7287–7292, May 2004.
[42] R. V. Koodli, B. Rudolfs, H. K. Wayment-Steele, and R. Das, "Redesigning the EteRNA100 for the Vienna 2 folding engine," *BioRxiv*, pp. 2008–2021, 2021.
[43] Y. Byun and K. Han, "PseudoViewer3: Generating planar drawings of large-scale RNA structures with pseudoknots," *Bioinf.*, vol. 25, pp. 1435–1437, Apr. 2009.

ÁLVARO RUBIO-LARGO received the Ph.D. degree in computer engineering from the University of Extremadura, Cáceres, Spain, in 2013.

He is currently an Associate Professor in computer science with the Department of Computers and Telematics Systems Engineering, University of Extremadura. He has authored or coauthored over 70 publications, including more than 35 journal citation report (JCR) articles. His current research interests include big data, machine learning, and the use of evolutionary computation for solving real-world multiobjective optimization problems.

Dr. Rubio-Largo has co-organized several international workshops and formed part of technical program committee in a number of international conferences. He has also co-edited several special issues in diverse international JCR-indexed journals. In addition, he is an active reviewer of diverse international JCR-indexed journals.

LAURA ESCOBAR-ENCINAS received the bachelor's degree (Hons.) in software engineering from the University of Extremadura, Cáceres, in 2023, and the degree (Hons.) in the EBAU (University Entrance Examination). She is currently pursuing the master's degree in artificial intelligence with the International University of La Rioja (UNIR).

She is a member of the Department of Computers and Telematics Systems Engineering, University of Extremadura, in a Scientific and Research Staff position as a Bioinformatician. Her current research interest includes the use of high-performance artificial intelligence to optimize the design of RNA sequences.

NURIA LOZANO-GARCÍA received the integrated B.Sc. + M.Sc. degree in biology from the University of Valencia, Spain, in 2003, the Technical Engineering degree in computer systems from National Distance Education University (UNED), Spain, in 2012, and the M.Sc. degree in bioinformatics and computational biology from the Complutense University of Madrid, Spain, in 2012.

She is currently a member of the Department of Computers and Telematics Systems Engineering, University of Extremadura, in a Scientific and Research Staff position as a Bioinformatician, and she has been held similar positions in Universities and Research Centres, since 2012. She has coauthored or authored seven journal citation report (JCR) articles. Her research interests include the field of bioinformatics, as metagenomics, 16S rDNA, bacterial genomes and SNPs detection, chip-seq, and more recently evolutionary computation and multiobjective optimization applied to RNA design.

JOSÉ MARIA GRANADO-CRIADO received the Ph.D. degree in computer science from the University of Extremadura, Spain, in 2009.

Since 2006, he has been an Associate Professor in computer architecture with the Department of Computers and Communications Technologies, University of Extremadura. He has authored or coauthored more than 50 publications, including journal articles (17 JCR-indexed journal articles), book chapters, and peer-reviewed conference proceedings. His main research interests include reconfigurable and embedded computing, cryptography, parallel computing, evolutionary computation, and bioinformatics.

Dr. Granado-Criado has edited two special issues of international JCR-indexed journals and he has contributed to the organization of several international conferences and workshops.