Multiobjective Metaheuristic to Design RNA Sequences

Álvaro Rubio-Largo

NOVA Information Management School, Universidade Nova de Lisboa, Lisbon, Portugal

Leonardo Vanneschi

NOVA Information Management School, Universidade Nova de Lisboa, Lisbon, Portugal

Mauro Castelli

NOVA Information Management School, Universidade Nova de Lisboa, Lisbon, Portugal

Rubio-Largo, A., Vanneschi, L., Castelli, M., & Vega-Rodriguez, M. A. (2019). Multiobjective Metaheuristic to Design RNA Sequences. IEEE Transactions on Evolutionary Computation, 23(1). DOI: 10.1109/TEVC.2018.2844116

Multiobjective Metaheuristic to Design RNA Sequences

Álvaro Rubio-Largo, Leonardo Vanneschi, Mauro Castelli, Miguel A. Vega-Rodríguez

***Abstract*—RNA inverse folding problem is a bioinformatics problem where the objective is to find an RNA sequence that folds into a given target secondary structure. In this work, we use Evolutionary Computation to solve a new and innovative multiobjective definition of this problem. In this new multiobjective definition of the problem, we have considered the similarity between target and predicted structures as a constraint, and three objective functions: (i) Partition Function (free energy of the ensemble), (ii) Ensemble Diversity and (iii) Nucleotides Composition. The Multiobjective Metaheuristic To Design RNA Sequences (m2dRNAs) proposed in this paper is compared against other RNA inverse folding methods published in the literature, such as RNAinverse, RNA-SSD, INFO-RNA, MODENA, NUPACK, fRNAkenstein, DSS-Opt, RNAiFOLD, antaRNA, ERD, and Eterna players. After a comprehensive comparative study on two well-known benchmarks (Rfam and Eterna100), we conclude that m2dRNAs is capable of obtaining very promising results in terms of both quality of RNA designs and required runtime. The source code of m2dRNAs is available at http://arco.unex.es/arl/m2dRNAs-source_code.zip.**

***Index Terms*—Multiobjective optimization, metaheuristic, RNA, inverse folding.**

I. Introduction

Given a particular RNA secondary structure (target structure), finding a sequence of base pairs that would fold into this structure is known as the RNA inverse folding problem. It is a common practice to make use of this bioinformatics problem for designing non-coding RNAs, which are involved in gene regulation, chromosome replication and/or RNA modification [1], [2]. In the drug and therapeutic agents area, the design of RNA sequences is also applicable to the construction of ribozymes and riboswitches [3]. We can also find applications in nano-biotechnology in the context of building self-assembling structures from RNA molecules [4].

According to [5], the RNA inverse folding problem can be solved by using brute force. Unfortunately, its complexity grows exponentially as 4^n, where n is the length of the target structure. This upper bound may be refined if we take into account that paired positions have to form valid base pairs, such as: AU, UA, GC, CG, GU or UG. Therefore, given a target structure with u unpaired nucleotides and p paired nucleotides, the number of valid sequences will be $6^{p/2}4^u$. For example, see the following secondary target structure:

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

where the length is 30-nucleotides-long and its composition includes p=14 paired-nucleotides and u=16 unpaired nucleotides. For this small structure, the number of RNA compatible sequences is approximately 10^{15} sequences.

Theoretical evolutionary studies (the study of genotype/phenotype maps, [6]) and RNA inverse folding problem are also deeply connected. As an example, the detection of undesignable motifs [7] in empirical design studies implies that only a small number of secondary structures can be designed. In the other way, neutral evolution theories state that, for the practice of RNA design, a number of feasible designs can be obtained within secondary structures of identical length [8].

In the literature, we find several RNA inverse folding methods, the vast majority of them starts with an initial sequence that is iteratively modified until it either folds into the target structure or some stopping criterion is reached.

The first algorithm developed for RNA design was RNAinverse [9]. It is contained in the Vienna Package [10], [11]. RNAinverse divides the input target structure into smaller structures, then, by using an adaptive random walk procedure, it tries to minimize the base pair distance. In RNAinverse, the objective is to minimize the Hamming distance between the Minimum Free Energy (MFE) secondary structure of the current sequence and the target structure. The algorithm may return a successful RNA sequence, an approximate RNA sequence, or no RNA sequence at all.

In 2004, Andronescu et al. presented RNA Secondary Structure Designer (RNA-SSD) [12], a different and more efficient algorithm for RNA inverse folding problem. Similarly to RNAinverse, RNA-SSD uses a divide-and-conquer approach by hierarchically decomposing the input target structure. In contrast, it makes use of a greedy initialization procedure to select an initial RNA sequence and applies a stochastic local search instead of using a random adaptive walking (RNAinverse). According to [12], RNA-SSD is able to solve structures, consistently, for which RNAinverse is unable to find solutions.

Busch and Backofen [3] introduced an algorithm for the INverse FOlding of RNA (INFO-RNA). Their approach (INFO-RNA) consists of two basic steps, a dynamic programming method for good initial sequences and a following improved

This work was partially funded by the AEI (State Research Agency, Spain) and the ERDF (European Regional Development Fund, EU), under the contract TIN2016-76259-P (PROTEIN project). Álvaro Rubio-Largo is supported by the post-doctoral fellowship SFRH/BPD/100872/2014 granted by Fundação para a Ciência e a Tecnologia (FCT), Portugal.

Álvaro Rubio-Largo, Leonardo Vanneschi, and Mauro Castelli are with the NOVA Information Management School (NOVA IMS), Universidade Nova de Lisboa, Campus de Campolide, 1070-312 Lisboa, Portugal (email: arl@unex.es, lvanneschi@novaims.unl.pt, mcastelli@novaims.unl.pt).

Miguel A. Vega-Rodríguez is with the Department of Computer and Communications Technologies, University of Extremadura, Spain (e-mail: mavega@unex.es).

stochastic local search that uses an effective neighbor selection method. In the initialization, by using dynamic programming they generate a set of RNA sequences that fold into the input target structure; however, only the sequence with the lower possible energy is selected. For the selection of neighbors during the local search, they employ a kind of look-ahead of one selection step applying an additional energy-based criterion. After a comprehensive comparative study, the authors conclude that INFO-RNA performed better than RNAinverse and in most cases, better than RNA-SSD. The main disadvantage of INFO-RNA is that the returned RNA sequences present a high GC-content and tend to have little resemblance with biologically active RNA.

In [13], A. Taneda applied the well-known Fast Non-Dominated Sorting Genetic Algorithm (NSGA-II) [14] to this problem: MODENA (Multi-Objective DEsign of Nucleic Acids). This multiobjective approach explores the approximate set of weak Pareto optimal solutions in the objective function space of two objective functions: (i) a structure stability (energy of the MFE structure of the proposed sequence) and (ii) structure similarity (distance between the MFE structure for the candidate sequence and the target structure). In [15], the author extends MODENA with pseudoknot prediction methods, such as IPknot [16] and HotKnots [17]. A new version of MODENA appears in [18]. This new version allows MODENA to design RNA sequences which fold into multiple target secondary structures.

NUPACK is a suite of programs for computational nucleic acid analysis and includes an RNA designer [19]. NUPACK uses a similar method to that of RNA-SSD, but, the goal of NUPACK is to find an RNA sequence that minimizes the ensemble defect instead of the energy of the MFE structure. The ensemble defect is the average number of nucleotides that are incorrectly paired at equilibrium relative to the specified target structure, evaluated over the Boltzmann-weighted ensemble of secondary structure [20].

In [21], the authors presents a genetic algorithm for the design of RNA sequences: fRNAkenstein. The authors developed fRNAkenstein with the aim of finding one or more target structures at the same time, that is, to solve the multi-target inverse folding version of the problem. In comparison with other evolutionary techniques (such as MODENA). According to [21], fRNAkenstein was designed with little focus on running time, choosing Python as implementation language for the ease of development and flexibility it offers. In addition, its mutation and recombination processes provide additional computational burden.

Matthies et al. [22] proposed the Dynamics in Sequence Space Optimization (DSS-Opt) algorithm. DSS-Opt uses Newtonian dynamics in the sequence space, with a negative design term and simulated annealing to optimize a sequence such that it folds into the desired secondary structure. DSS-Opt makes use of multiple scoring functions, such as (i) Nearest-Neighbor Model Energies for Sequence Compositions, (ii) Negative Design, (iii) Sequence Heterogeneity and (iv) Composition Constraint. A detailed explanation of these scoring functions appears in [22].

RNAiFOLD is a constraint programming approach developed by J. A. García-Martín et al. [23], [24]. It allows the user to specify one of three different optimization criteria: MFE, the free energy or the ensemble defect. Furthermore, it provides a wide variety of design constraints, such as base-pairs bounds, specified motifs, etc.

The EteRNA Ensemble Algorithm (also known as EteRNA Bot) is a folding approach based on strategies made by tens of thousand EteRNA players and other RNA design software (RNAInverse, INFO-RNA, and RNA-SSD) [25]. It first creates an ensemble classifier from all player strategies in the market[1]. Then, the ensemble classifier is passed over to a sequence designer which first creates a sequence with 60% GC pairs and then keeps changing bases at random positions until it finds a sequence that gets high ensemble classifier score. Unfortunately, the EteRNA Bot webpage[2] is no longer available. From this project, the authors introduce the Eterna100 benchmark [26], a standard set of structures to be used for challenging and evaluating the next generation of automated RNA design algorithms.

In [27], Esmaili-Taheri et al. introduced an evolutionary algorithm for the RNA inverse folding problem: Evolutionary RNA Design (ERD). This algorithm starts reconstructing RNA sub-sequences (pools) which corresponds to different components with different lengths. Using these pools, it builds an initial RNA sequence that is compatible with the given target structure. Then it makes use of an evolutionary algorithm to improve the quality of the sub-sequences corresponding to the components. In [28], the authors present a new version of ERD and extend it to handle some sequence and energy constraints. In the sequence constraints, one can restrict sequence positions to a fixed nucleotide or to a subset of nucleotides. As for the energy constraint, one can specify an interval for the free energy ranges of the designed sequences.

In 2015, Kleinkauf et al. proposed an ant colony optimization algorithm to design RNA structures (antaRNA) [29], [30]. Besides the structural constraint, antaRNA realizes the usage of sequence constraints and provides the user to specify a GC value constraint. In [30], the authors presents antaRNA as multi-objective; however, antaRNA only handles multiple constraints but not multiple objectives.

In this work, we follow the research line opened by A. Taneda (MODENA) [13], [15], i.e. Multiobjective Optimization and Evolutionary Algorithms (MOEA) for solving the RNA inverse folding problem. We refer to our approach as Multiobjective Metaheuristic To Design RNA Sequences (m2dRNAs).

As we have mentioned before, MODENA focuses on optimizing the structure stability and the structure similarity of the RNA sequences. However, m2dRNAs is constrained by the structural similarity and optimizes the structural stability, the prediction reliability, and the nucleotide composition. This problem formulation allows m2dRNAs not only to provide a set of stable RNA sequences, but also ensuring reliability in the prediction of structures as well as avoiding strong biases in their composition. In m2dRNAs, the objective functions

[1] http://www.eternagame.org/

[2] http://eternabot.org.

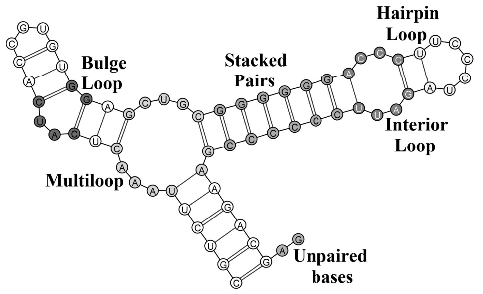


Fig. 1. Illustrative representation of the different RNA structure elements (hairpin loop, stacked pairs, internal loops, bulge loops, multi loop and unpaired bases). Illustration made by using [42].

capture certain aspects of a good solution; thus, it should be expected that the desired solutions will score relatively highly under all of the 'proxy' objectives. In the literature, we find a wide variety of problems successfully solved by using multiobjective optimization, e.g. [34], [35], [36], [37], [38].

The rest of the paper is organized as follows. Section II formulates the new multiobjective formulation of the RNA Inverse Folding problem. A detailed description of the m2dRNAs algorithm is presented in Section III. Section IV is devoted to the analysis of the experiments carried out and also a comparison with other heuristics and metaheuristics published in the literature. Finally, Section V summarizes the conclusions of the paper and discusses possible lines of future work.

II. Multiobjective RNA Inverse Folding Problem

Let x denote an RNA sequence of n nucleotides (A, C, G, U) [43]. We use $i \cdot j$ to define a base pair between bases x_i and x_j, where $1 \leq i < j \leq n$.

A secondary structure for an RNA sequence x is a set of base pairs $S = \{i \cdot j \mid 1 \leq i < j \leq n \wedge i < j - 3\}$. For all base pairs $i_1 \cdot j_1, i_2 \cdot j_2 \in S$ with $i_1 \cdot j_1 \neq i_2 \cdot j_2$:

i. $\{i_1, j_1\} \cap \{i_2, j_2\} = \emptyset$.
ii. $\{x_{i_1}, x_{j_1}\} \in \{\{A, U\}, \{U, A\}, \{C, G\}\{G, C\}, \{U, G\}, \{G, U\}\}$, i.e. only Watson-Crick and wobble base pairs are allowed.
iii. $i_1 < i_2 < j_1 < j_2$ (crossed base pairs are not allowed).

According to [43], the model of Gibbs free energy approximates the free energy by postulating that the energy of the full three dimensional structure only depends on the secondary structure, and that this turn can be broken into a sum of independent contributions from each loop in the secondary structure. In a secondary RNA structure, loops are classified by the number of interior base pairs they contain: Hairpin loop (no interior base pairs), Stacked Pairs (one interior base pair separated from the closing base pair on either side), Bulges loops (one interior base pair separated from the closing base pair on one side), Internal loops (one interior base pair separated from the closing base pair on both sides), Multiloops (have two or more interior base pairs). Note that bases not accessible from any base pair are called external or unpaired. In Figure 1, we can see an illustrative representation of the aforementioned RNA loops.

The RNA inverse folding problem consists in, given a particular RNA secondary structure S, finding an RNA sequence x of base pairs that would fold into S. In [44], Schnall-Levin et al. demonstrated that the RNA secondary structure design problem is NP-hard, please refer to [44] for more details about the NP-hardness of the problem.

In this work, we have formulated the RNA inverse folding problem as a Multiobjective Optimization Problem [45], where the final goal is to simultaneously optimize the following objective functions:

– Partition Function (f_1) [46]. Given an RNA sequence x, the partition function for the ensemble of all possible secondary structures can be calculated as follows:

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

where $-\Delta G$ is the Gibb's Free Energy change, R denotes the universal gas constant, T is the absolute temperature (37°C), and $S'(x)$, over which the summation is performed, is the set of all possible secondary structures. This partition function is determined by nonredundant and exhaustive recursions, a complete definition may be found in [47]. In this work, to calculate f_1, we have used the ViennaRNA Package 2 (v2.3.2) (C code library) [48].

– Ensemble Diversity (f_2). According to [49], an important approach to deal with uncertainty in prediction, is to provide reliability information that informs us how trustworthy a prediction is. A well-known reliability measure is known as ensemble diversity. The simplest distance measure between two structures is the *base pair distance* which counts the number of pairs present in one, but not both structures. Basically, the ensemble diversity represents the average base pair distance between all structures in the Boltzmann ensemble and can be expressed in terms of base pair probabilities p_{ij}:

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

In this work, the ViennaRNA Package 2 (v2.3.2) was used to compute f_2. For a detailed mathematical formulation of the Ensemble Diversity, please refer to [49].

– Nucleotides Composition (f_3). Given the designed RNA sequence x, we analyze its composition in terms of base-pairs percentages ($\%GC$: GC/CG, $\%AU$: AU/UA, $\%GU$: GU/UG)), unpaired bases ($\%uA$, $\%uC$, $\%uG$, $\%uU$), and total bases distribution ($\%A$, $\%C$, $\%G$, $\%U$). The first group shows the distribution over the three types of base pairs in paired positions in the target, the second group shows the nucleotide distribution for unpaired positions in the target, and the last group shows the overall nucleotide distribution in the sequence. In this way, the nucleotides composition objective function is calculated as follows:

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

As we can see, for each group (base-pairs, unpaired and total), we obtain the maximum percentage. Since f_3 needs to be minimized, we obtain well-balanced RNA sequences in terms of nucleotides composition.

Furthermore, every RNA sequence designed must satisfy the following constraint:

- Similarity (σ) [15]. It measures the similarity between the predicted structure of x and the input target structure:

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

where n is the total number of nucleotides in x, and d corresponds to the number of the nucleotide positions whose structure in the designed sequence are different from that in the target structure. If $\sigma(x)$=1, then σ indicates a perfect consensus between the predicted and the target structures.

In a more formal way, the RNA inverse folding problem may be formulated as a Multiobjective Optimization Problem:

$$\begin{aligned} \text{minimize} \quad & F(x) = (f_1(x), f_2(x), f_3(x)) \\ \text{subject to} \quad & \sigma(x) = 1 \\ & x \in \Omega \end{aligned}$$

where a solution x is a succession of letters within an RNA molecules (A, C, G, U): $x = \{x_1, \ldots, x_n\}$. The solution x must satisfy $\sigma(x)$=1. All the possible values (A, C, G, U) of each component of x constitute Ω (*decision space*), which contains all the RNA sequences of length n, i.e. $\mid \Omega \mid$=4^n sequences.

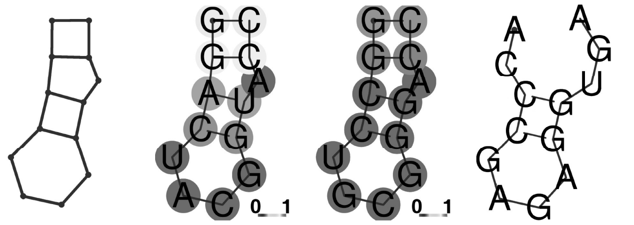
The three objective functions (f_1, f_2 and f_3) need to be minimized, constituting a multi-dimensional space $F : \Omega \to R^3$; which is commonly known as *objective space* (R^3). For each solution $x = \{x_1, \ldots, x_n\}$ in the decision space, there exists a point $z = \{z_1, z_2, z_3\}$ in the objective space.

In multiobjective optimization is necessary to compare solutions in order to decide which one is *better*. Thus, two solutions are compared on the basis of whether one *dominates* the other solution or not. Therefore, a solution x_1 is said to dominate the other solution x_2, if and only if the following conditions are satisfied:

i. The solution x_1 is no worse than x_2 in all objective functions, or $f_i(x_1) \leq f_i(x_2)$ for all $i \in \{1, 2, 3\}$.
ii. The solution x_1 is strictly better than x_2 in at least one objective function, or $f_i(x_1) < f_i(x_2)$ for at least one index $i \in \{1, 2, 3\}$.

Given a set of solutions P, we denominate a solution x^* as *Pareto-optimal* or *non-dominated solution* if no solution in P dominates x^*. The set of all non-dominated solutions in P is known as *Pareto set*, and its graphical representation as *Pareto front*.

MODENA optimizes two objective functions: (i) Minimum Free Energy (MFE), (ii) Similarity between the predicted and the target structures. Therefore, whereas MODENA minimized the dissimilarity between the predicted MFE structure (objective 1), our first objective focuses on optimizing the frequency of the target structure in the thermodynamic ensemble (free energy). According to [39], optimization via the Partition Function produces sequences with a very strong preference for the target structure. The second objective function (ensemble diversity) is also recommended in the literature [40] for long RNA sequences, mainly because MFE-prediction (MODENA) decreases with sequence length [41]. Our last objective function (Nucleotides Composition) tries to avoid strong biases in the composition of the designed sequences, allowing m2dRNAs to obtain diversity in the set of solutions returned.



(a) S (b) $x1$ (c) $x2$ (d) $x3$

Fig. 2. Target structure S and the predicted secondary structure for the RNA sequences: $x1$, $x2$, and $x3$. The color scheme represents the base pair probabilities (normalized in the range 0-1).

An illustrative example will help to understand the aforementioned definition of the problem. Let us consider the following target structure (in dot-bracket notation):

$$S = \texttt{((((....)).))}$$

and the following three candidate RNA sequences:

$$\begin{aligned} x1 &= \texttt{GGACUACGGUACC} \\ x2 &= \texttt{GGCCUGCGGGACC} \\ x3 &= \texttt{ACCCGAGAGGUGA} \end{aligned}$$

The first step is to obtain their predicted secondary structures and measure its similarity against S:

$$\begin{aligned} \sigma(x1) &= \frac{(13-0)}{13} \to \texttt{((((....)).))} \\ \sigma(x2) &= \frac{(13-0)}{13} \to \texttt{((((....)).))} \\ \sigma(x3) &= \frac{(13-4)}{13} \to \texttt{..((....))...} \end{aligned}$$

As we can see, only $x1$ and $x2$ satisfy the similarity constraint, therefore, we continue only with them.

By using the ViennaRNA package, we compute f_1 and f_2 for the two feasible solutions ($x1$ and $x2$):

$$\begin{aligned} f_1(x1) = \text{-1.48 and } f_2(x1) = 1.22 \\ f_1(x2) = \text{-3.47 and } f_2(x2) = 0.42 \end{aligned}$$

Then, we analyze the composition of x_1 and x_2 in order to compute the third objective function (f_3). In the following, we have highlighted the maximum percentage (%) within each group:

$$x1 = \begin{cases} \text{Pair} \begin{cases} GC = \mathbf{75} \\ AU = 25 \\ GU = 0 \end{cases} \\ \text{Unpair} \begin{cases} uA = \mathbf{40} \\ uC = 20 \\ uG = 20 \\ uU = 20 \end{cases} \\ \text{Total} \begin{cases} A = 23.07 \\ C = \mathbf{30.77} \\ G = 30.76 \\ U = 15.38 \end{cases} \end{cases} \qquad x2 = \begin{cases} \text{Pair} \begin{cases} GC = \mathbf{100} \\ AU = 0 \\ GU = 0 \end{cases} \\ \text{Unpair} \begin{cases} uA = 20 \\ uC = 20 \\ uG = \mathbf{40} \\ uU = 20 \end{cases} \\ \text{Total} \begin{cases} A = 7.77 \\ C = 38.46 \\ G = \mathbf{46.15} \\ U = 7.77 \end{cases} \end{cases}$$

The f_3 value of x_1 and x_2 is:

$$f_3(x1) = 75 + 40 + 30.77 = 145.77$$
$$f_3(x2) = 100 + 40 + 46.15 = 186.15$$

As we can see, $x2$ is a more stable RNA sequence because their values in f_1 and f_2 are lower than the values of $x1$; however, the nucleotides composition of x_1 (f_3) is less biased than the composition of $x2$. In this particular case, the solutions $x1$ and $x2$ are non-dominated solutions. Finally, in Figure 2, we illustrate the target structure S and the predicted secondary structures for the RNA sequences $x1$, $x2$, and $x3$.

III. Multiobjective Metaheuristic To Desgin RNA Sequences (m2dRNAs)

In this section, we describe the Multiobjective Metaheuristic To Design RNA Sequences (m2dRNAs) in detail. First, we briefly describe the multiobjective approach used within m2dRNAs: Fast Non-dominated Sorting Genetic Algorithm (NSGA-II). Then, we detail the chromosome encoding of the solutions and the genetic operators used within NSGA-II for solving this problem.

A. Fast Non-dominated Sorting Genetic Algorithm (NSGA-II)

As a revised version of NSGA [50], the Fast Non-Dominated Sorting Genetic Algorithm (NSGA-II) is a well-known MOEA created by Deb et al. [14], the source code is available at the Kanpur Genetic Algorithms Laboratory's webpage[3].

NSGA-II tries to obtain a new population (offspring population) from an original one (parent population) by applying classical genetic operators, such as selection, crossover and mutation. Then, both populations, offspring and parent, are mixed into a new population. This new population is sorted into categories (ranks) according to their relationship of dominance. After that, the best individuals are selected to create a new parent population for the next generation. In the case of having to choose among individuals with the same rank, the crowding distance of the individuals belonging to the same rank is calculated, in order to decide which are the best individuals. In NSGA-II, the diversity among non-dominated solutions is introduced by using the crowding comparison procedure, which is used in the tournament selection and during the population reduction phase.

Recently, a new version of this approach (NSGA-III) has been proposed by Deb and Jain [51] to handle many-objective optimization problems. Ishibuchi et al. [52] compare both versions (NSGA-II and NSGA-III) on various many-objective test problems. For three-objective optimization problems (like the RNA inverse folding problem), the authors conclude that similar solution sets are obtained by the two algorithms. The main disadvantage of NSGA-III is the ρ parameter. It is not clear how ρ is calculated and this is missing from the methodology [51]. On the other hand, NSGA-II is extremely well established and has proven across an enormous number of problems. As NSGA-II and NSGA-III have been shown to be comparable for related problems [52], in this work, we have applied the NSGA-II to solve this bioinformatics problem.

B. Chromosome Encoding and Genetic Operators

The chromosome encoding of the individual determines how the RNA inverse folding problem is structured in the proposed algorithm. It gives us the necessary knowledge to understand the behaviour of the evolutionary algorithm.

For example, MODENA [13] encodes the solutions as a string of characters defined over the RNA alphabet, making use of one crossover operator (structural n-point crossover) and two mutation operators (point accepted mutation and error diagnosis mutation). In MODENA, the three genetic operators are invoked with an equal probability.

In this work, given a target structure S in dot-bracket notation, the first step is to create two sets: base-pairs (B) and unpaired (U). As we can see in Algorithm 1, we process the target structure, storing in B and U the positions of the base-pairs or unpaired nucleotides, respectively. In Algorithm 1, the notation $|.|$ indicates the number of elements in a set.

Note that, although Algorithm 1 does not significantly affect the overall running time of m2dRNAs, it is possible to use a stack to extract the required information in $O(n)$ time complexity.

A chromosome (X) is then encoded as a real-valued vector of length $|B|$+$|U|$:

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

where ρ is a real value in the range [0,1] and indicates the type of base-pair or unpaired nucleotides, depending on the position. In Algorithm 2, we present the procedure to convert an input chromosome (X) to an RNA sequence (x) that will be evaluated to check if it satisfies the similarity constraint and to compute their values of objective functions.

The initialization of individuals is performed randomly for paired positions (B set) but for selecting each unpaired position (U set), we take into account its possible related base-pairs, so, we minimize the probability of creating unnecessary loops in the structure of the new RNA sequence.

The main advantage of this chromosome encoding is to ensure a certain level of greed, allowing m2dRNAs to find sequences that fold into the target structure in a faster way.

For example, given the following target structure:

$$\begin{array}{ccccccccccccccc} & & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\ S & = & (& (& (& (& . & . & . & . &) &) & . &) &) \end{array}$$

[3] https://www.iitk.ac.in/kangal/codes.shtml

Algorithm 1: Process target structure

```
Input : a target structure in dot-bracket notation ($S$) ;
$B \leftarrow \emptyset$;
$U \leftarrow \emptyset$;
for $i \leftarrow 1$ to $|S|$) do
    if $S_i =$ '.' then $U \leftarrow U \cup i$ ;
    if $S_i =$ '(' then
        $c \leftarrow 0$;
        for $j \leftarrow i$ to $|S|$) do
            if $S_j =$ '(' then $c \leftarrow c + 1$ ;
            if $S_j =$ ')' then
                $c \leftarrow c - 1$;
                if $c < 0$ then $B \leftarrow B \cup (i, j)$ ;
```

Algorithm 2: Chromosome encoding to RNA sequence

```
Input : a real-valued vector ($X$) ;
Output: an RNA sequence ($x$);
$x \leftarrow \emptyset$;
foreach $(i, j)$ in $B$ do
    $x_i \leftarrow U$;    $x_j \leftarrow G$;
    if $X_i < 1/6$ then       $x_i \leftarrow G$;   $x_j \leftarrow C$ ;
    else if $X_i < 2/6$ then  $x_i \leftarrow C$;   $x_j \leftarrow G$ ;
    else if $X_i < 3/6$ then  $x_i \leftarrow A$;   $x_j \leftarrow U$ ;
    else if $X_i < 4/6$ then  $x_i \leftarrow U$;   $x_j \leftarrow A$ ;
    else if $X_i < 5/6$ then  $x_i \leftarrow G$;   $x_j \leftarrow U$ ;
foreach $i$ in $U$ do
    $x_i \leftarrow U$;
    if $X_i < 1/4$ then       $x_i \leftarrow C$ ;
    else if $X_i < 2/4$ then  $x_i \leftarrow G$ ;
    else if $X_i < 3/4$ then  $x_i \leftarrow A$ ;
return $x$
```

after processing the structure S (see Algorithm 1), we obtain the following two sets:

B={ (1,13), (2,12), (3,10), (4,9) }

U={ 5, 6, 7, 8, 11 }

As we can see, the chromosome will be a vector of 9 real-values in the range [0, 1]. For example, if we consider the following two chromosomes:

$X1$={ 0.12, 0.09, 0.43, 0.23, 0.81, 0.62, 0.14, 0.31, 0.55 }

$X2$={ 0.15, 0.12, 0.31, 0.27, 0.79, 0.31, 0.05, 0.29, 0.66 }

then, we could evaluate them after a proper conversion into an RNA sequence (see Algorithm 2):

$$\begin{array}{rccccccccccccc} & {\scriptstyle 1} & {\scriptstyle 2} & {\scriptstyle 3} & {\scriptstyle 4} & {\scriptstyle 5} & {\scriptstyle 6} & {\scriptstyle 7} & {\scriptstyle 8} & {\scriptstyle 9} & {\scriptstyle 10} & {\scriptstyle 11} & {\scriptstyle 12} & {\scriptstyle 13} \\ x1 = & G & G & A & C & U & A & C & G & G & U & A & C & C \\ x2 = & G & G & C & C & U & G & C & G & G & G & A & C & C \end{array}$$

With the proposed chromosome encoding, any crossover and mutation operators for continuous optimization problems may be applied within m2dRNAs. In this work, we have selected the crossover and mutation operators proposed by Deb et al. [14]: simulated binary crossover (SBX) and polynomial mutation.

IV. Experimental Results

This section is devoted to present a comparative study of our proposal (m2dRNAs) and well-known approaches for the RNA inverse folding problem. In the first place, we describe the methodology followed in the study. Then, we compare the performance of m2dRNAs against other approaches published in the literature.

A. Methodology

We have evaluated the performance of m2dRNAs and other RNA inverse folding approaches with two benchmarks: Rfam [13] and Eterna100 [26]. The first one is a benchmark created from the seed alignments of Rfam 9.0.21, containing a total of 29 target structures. The second benchmark (Eterna100) contains a total of 100 secondary structure design challenges that span a large range in design difficulty, from short hairpins to complex 400-nucleotide designs.

The parameter configuration of m2dRNAs is: population size of 52 individuals (m2dRNAs requires a population size multiple of 4 to perform the binary tournament), stopping criterion based on iterations (50 iterations), simulated binary crossover (η_c=10, p_c=0.9), polynomial mutation (η_m=5, p_m=$10/n'$, where n' is the total number of base pairs and unpaired positions). Therefore, after 50 iterations, m2dRNAs outputs up to 52 RNA sequences; however, to perform a fair comparative study, we only select 50 sequences by using the crowding-distance procedure. It was compiled by using g++ (GCC) 4.9.3.

The methods involved in the comparative study are: RNAinverse v2.3.3 [9], RNA-SSD v02/2004 [12], INFO-RNA v2.1.2 [3], MODENA v0.0.67 [13], [15], NUPACK v3.0.6 [19], fRNAkenstein v1.22/07/12 [21], DSS-Opt v01/2014 [22], RNAiFOLD v3.1 [23], [24], antaRNA v114 [29],[30], ERD v2.0.0 [28], and Eterna players [26].

In Rfam benchmark, to make a fair comparison, for each structure, all the methods involved in the comparison were configured to return a total of 50 RNA designs. Some of these methods do not allow user to define a number of output RNA designs (RNAinverse, INFO-RNA, NUPACK, DSS-Opt), therefore, they were run 50 times to obtain a set of 50 RNA designs (like the other methods). In addition, all the methods were run with default parameters configuration and with no total time limit. To evaluate the quality of the output RNA designs obtained by the approaches in terms of partition function and ensemble diversity, we have used two well-known multiobjective metrics: the Hypervolume (HV) [53] quality indicator and the Set Coverage [54] indicator. In order to extract some useful conclusions with a certain level of statistical confidence (a significance level of 0.5%), 31 independent runs were performed per experiment by each method (therefore, 31 sets of 50 RNA designs were output by each approach), and the output set of RNA designs with the

TABLE I
MEDIAN HYPERVOLUME (HV, IN %) OBTAINED BY THE TEN APPROACHES WHEN SOLVING THE RFAM STRUCTURES.

	Len.	m2dRNAs	ERD	RNAiFOLD	DSS-Opt	MODENA	INFO-RNA	fRNAkenstein	antaRNA	NUPACK	RNAinverse
RF00001.121.ss	118	**86.74**	75.12	51.99	0	67.99	77.78	25.88	34.63	0	0
RF00002.2.ss	152	**91.41**	67.75	53.90	85.12	69.17	0	0	0	0	0
RF00003.94.ss	162	**90.28**	54.85	0	66.88	0	0	0	0	0	0
RF00004.126.ss	194	**90.32***	67.50	74.04	**91.18***	84.41	79.17	18.80	57.43	57.05	28.91
RF00005.1.ss	75	**96.58***	76.82	76.29	**96.59***	85.17	89.51	14.39	54.60	55.62	42.53
RF00006.1.ss	90	89.17	72.23	66.13	**90.45**	77.40	73.77	13.19	54.46	61.25	27.59
RF00007.20.ss	155	**92.52***	72.87	69.87	**92.99***	84.42	73.17	18.44	49.99	63.92	25.61
RF00008.11.ss	55	**96.39**	78.85	73.45	95.84	88.43	94.82	13.82	54.27	66.30	43.83
RF00009.115.ss	349	**93.92**	66.62	69.20	90.58	80.46	9.06	10.98	0	0	0
RF00010.253.ss	358	0	0	0	0	0	0	0	0	0	0
RF00011.18.ss	383	0	0	0	0	0	0	0	0	0	0
RF00012.15.ss	216	**94.51***	72.10	73.87	**95.04***	88.49	45.84	27.73	62.63	62.83	0
RF00013.139.ss	186	90.61	67.67	71.77	**92.38**	84.00	84.90	18.45	55.13	59.11	30.23
RF00014.2.ss	88	**91.09***	73.21	69.94	89.98	85.86	**91.57***	24.74	52.47	53.64	47.94
RF00015.101.ss	141	**91.67***	77.88	67.68	**91.51***	78.44	66.04	21.13	51.84	58.87	21.62
RF00016.15.ss	130	0	0	0	0	0	0	0	0	0	0
RF00017.90.ss	302	**90.05**	65.70	70.32	84.51	81.07	80.06	15.38	48.15	63.47	24.49
RF00018.2.ss	361	**81.41**	0	45.33	0	0	0	0	0	0	0
RF00019.115.ss	84	**92.41**	74.25	65.38	90.45	79.94	87.78	18.28	52.43	58.33	37.52
RF00020.107.ss	120	0	0	0	0	0	0	0	0	0	0
RF00021.10.ss	119	**94.43**	71.98	76.31	91.18	91.16	87.77	22.28	57.74	66.08	48.06
RF00022.1.ss	149	**92.44**	68.07	75.64	92.01	85.52	69.64	20.80	59.57	60.25	29.23
RF00024.16.ss	452	0	0	0	0	0	0	0	0	0	0
RF00025.12.ss	211	87.06	63.39	64.10	**92.04**	75.94	37.69	0	50.85	0	0
RF00026.1.ss	103	**86.50**	73.89	47.61	72.00	65.09	23.22	15.05	67.68	56.67	14.04
RF00027.7.ss	80	**97.71**	75.24	82.87	89.37	95.25	95.99	37.43	61.59	68.25	55.28
RF00028.1.ss	345	**86.96**	62.94	68.57	0	0	0	0	0	0	0
RF00029.107.ss	74	**91.14**	76.16	66.21	88.19	78.48	65.10	19.42	48.26	55.68	24.05
RF00030.30.ss	341	**91.74**	63.13	71.48	89.70	81.78	48.95	14.08	51.50	63.50	0
avg. HV		**75.42**	55.80	53.52	64.41	58.91	47.65	12.77	35.35	35.55	17.27
Rfam datasets solved		24	23	23	21	21	20	19	19	17	15

* no statistically significant differences

median value of HV was used. All the approaches were run on an Intel Xeon E5-2630 v3 2.4GHz and 40GB RAM with Ubuntu 14.04.

For the Eterna100 benchmark, we have followed the same methodology as Anderson-Lee et al. [26], that is, all structures were attempted five times by each algorithm with a total time limit of twenty-four hours.

In our comparative study, we have performed a statistical analysis of the results obtained [55]. In the first place, a test for calculating residual normality is applied, in our case, Kolmogorov-Smirnov. The main objective of this test is to check whether the values of the results follow a gaussian distribution or not. For non-gaussian distributions, we perform a non-parametric analysis, such as Kruskal-Wallis. However, if the values follow a gaussian distribution, a test to check the homogeneity of the variances (Levene test) is also carried out. Finally, if this is positive, we apply an ANOVA analysis; otherwise, we perform the Kruskal-Wallis analysis.

B. Comparative study

This comparative study is divided into two parts. The first one compares the performance of m2dRNAs and other approaches when dealing with the Rfam benchmark in terms of multiobjective quality and required running time. The second one is focused on comparing the success of the approaches when solving the Eterna100 benchmark.

We start comparing our proposal (m2dRNAs) with ERD, RNAiFOLD, DSS-Opt, MODENA, INFO-RNA, fRNAkenstein, antaRNA, NUPACK and RNAinverse. Unfortunately, we cannot include RNA-SSD (it is only available to run via web-server) and Eterna (it is no longer available) in this first comparison.

As we can see in Table I, there exists a total of five Rfam structures (RF00010.253.ss, RF00011.18.ss, RF00016.15.ss, RF00020.107.ss, and RF00024.16.ss) in which no approach is able to obtain a successful RNA design (in median), that is a structure with similarity (σ) equal to one. According to a formal proof given in [7], these structures contain two kinds of motifs that are impossible to design for any algorithm based on the current thermodynamic model.

As we mentioned in the methodology, the HV metrics measures the quality (in terms of partition function and ensemble diversity) of the set of up to 50 successful RNA designs obtained by an algorithm. After performing 31 independent runs for each Rfam structure, in Table I, we present the median value of HV (in %) obtained by the ten approaches under study. In terms of median HV, m2dRNAs is able to obtain the highest values in 21 out of the 29 Rfam structures. If we compare m2dRNAs and MODENA (two multiobjective approaches), we can conclude that m2dRNAs overcomes the results obtained by MODENA in all feasible Rfam structures.

The three best approaches in terms of structures successfully solved are m2dRNAs, ERD and RNAiFOLD. As we can observe, only m2dRNAs is able to solve the remaining 24 feasible Rfam structures, whereas ERD fails in RF00018.2.ss and RNAiFOLD in RF00003.94.ss. The rest of approaches fails in three or more Rfam structures, highlighting the particularly poor performance of RNAinverse, which is only able to successfully solve 15 Rfam structures (in median).

In Figure S1, we present, for each Rfam structure, a box plot showing the distribution of HV obtained by each of the ten approaches under study (those structures considered impossible-to-design were excluded). As we may see, there are some RNA inverse folding methods, such as RNAiFOLD, NUPACK, or fRNAkenstein, with a high deviation of HV, which is translated in a non-reliable behaviour. For our proposal (m2dRNAs), we can observe that, in the vast majority of cases, a good distribution of HV in 31 independent runs is obtained.

TABLE II
SETS OF ALGORITHMS WHERE THE DIFFERENCES OF HV ARE STATISTICALLY NOT SIGNIFICANT (P-VALUE > 0.005). NOTE THAT, WE HAVE HIGHLIGHTED (IN BOLD) THOSE SETS OF ALGORITHMS IN WHICH M2DRNAS APPEARS.

RF00001.121.ss	(b,j), (f,g,i)
RF00002.2.ss	(a,b,f,g,h), (c,j)
RF00003.94.ss	(a,b,c,e,f,g,h)
RF00004.126.ss	(a,g), **(d,i)**
RF00005.1.ss	(a,g), **(d,i)**
RF00006.1.ss	-
RF00007.20.ss	(b,e,g), **(d,i)**
RF00008.11.ss	(e,g)
RF00009.115.ss	(a,f,g), (b,h)
RF00012.15.ss	(a,g), **(d,i)**, (e,j)
RF00013.139.ss	(b,c), (e,j)
RF00014.2.ss	(a,g), **(b,d)**
RF00015.101.ss	(a,g), (b,e), (c,j), **(d,i)**, (f,h)
RF00017.90.ss	(b,c)
RF00018.2.ss	(a,b,c,f,g,h,i,j)
RF00019.115.ss	(c,j), (e,g)
RF00021.10.ss	(c,i), (e,g,j)
RF00022.1.ss	(a,g), (b,j)
RF00025.12.ss	(e,j), (f,g,h)
RF00026.1.ss	(a,c), (b,f), (e,g), (f,h)
RF00027.7.ss	(b,c)
RF00028.1.ss	(a,b,c,f,g,h,i), (e,j)
RF00029.107.ss	(a,g), (b,e), (c,j), (f,h)
RF00030.30.ss	(a,b), (g,j)
Notation:	(a) antaRNA, (b) INFO-RNA, (c) MODENA, (d) m2dRNAs, (e) RNAiFOLD, (f) RNAinverse, (g) NUPACK, (h) fRNAkenstein, (i) DSS-Opt, and (j) ERD.

Due to the stochastic behaviour of the RNA inverse folding methods considered in this study, we have performed a statistical analysis (see Section IV-A) to demonstrate that the differences of HV among the methods are statistically significant. The confidence level considered in this work is always 99.5% in the statistical tests (a significance level of 0.5% or p-value under 0.005). A summary of the statistical analysis is presented in Table II (note that the five impossible to design structures were not taken into account). More precisely, for each RFam structure, we present the sets of algorithms in which the differences of HV in 31 independent runs are not statistically significant. As we can see, there is no significant difference between m2dRNAs and INFO-RNA in one structure (RF00012.15.ss). Compared with DSS-Opt, we can see that their differences of HV are not significant in five structures: RF00004.126.ss, RF00005.1.ss, RF00007.20.ss, RF00012.15.ss, and RF00015.101.ss.

We continue comparing the methods by using the Set Coverage (SC) metrics. If we suppose two sets of non-dominated solutions A and B, the SC(A,B) indicates the fraction of non-dominated solutions in B; which are covered by the non-dominated solutions in A. To make a fair comparison among the approaches, we compare the set of non-dominated solutions (RNA designs) with median value of HV in 31 independent runs. In Table III, we present the coverage relation between every pair of methods involved in this study. As we may see, the non-dominated solutions obtained by m2dRNAs are poorly dominated by the non-dominated solutions obtained by the other methods. More precisely, we find that only DSS-Opt and INFO-RNA dominates a low percentage of solutions of m2dRNAs. In contrast, m2dRNAs dominates a large fraction of non-dominated solutions obtained by the other approaches, 68.61% in average.

TABLE III
AVERAGE COVERAGE RELATION AMONG THE APPROACHES WHEN SOLVING THE RFAM STRUCTURES.

A	B	SC(A, B) (in %)	A	B	SC(A, B) (in %)
m2dRNAs	antaRNA	82.07	NUPACK	antaRNA	13.19
	INFO-RNA	41.10		INFO-RNA	0.00
	MODENA	75.12		MODENA	0.69
	RNAiFOLD	70.16		m2dRNAs	0.00
	RNAinverse	82.76		RNAiFOLD	1.15
	NUPACK	82.76		RNAinverse	35.63
	fRNAkenstein	82.76		fRNAkenstein	58.62
	DSS-Opt	21.26		DSS-Opt	0.00
	ERD	79.54		ERD	5.89

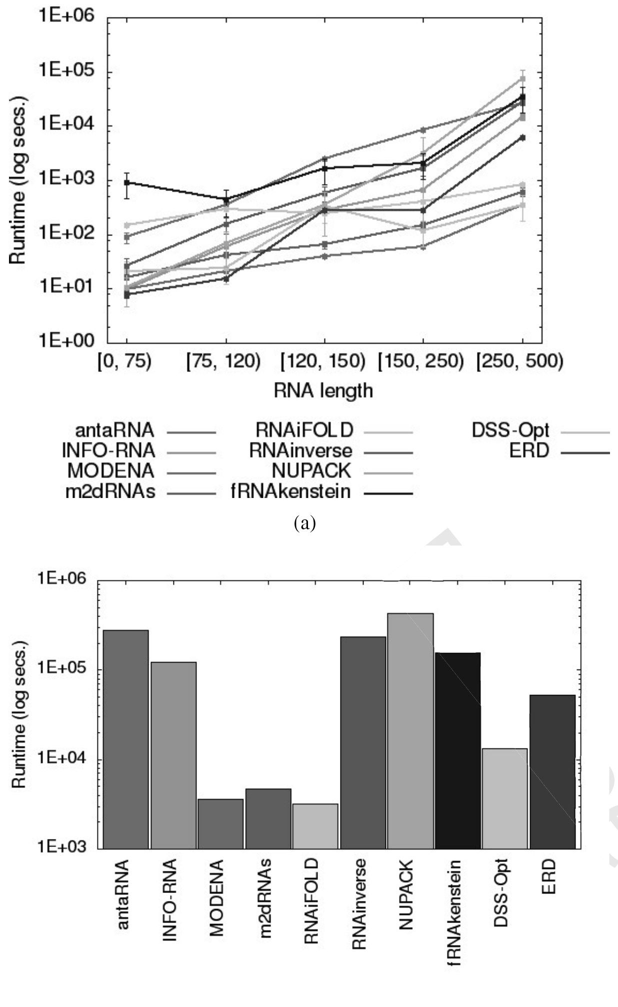
A	B	SC(A, B) (in %)	A	B	SC(A, B) (in %)
DSS-Opt	antaRNA	55.71	MODENA	antaRNA	66.63
	INFO-RNA	25.00		INFO-RNA	10.06
	MODENA	39.77		m2dRNAs	0.34
	m2dRNAs	12.75		RNAiFOLD	13.88
	RNAiFOLD	42.51		RNAinverse	72.41
	RNAinverse	68.10		NUPACK	58.62
	NUPACK	72.41		fRNAkenstein	72.41
	fRNAkenstein	72.41		DSS-Opt	3.45
	ERD	45.80		ERD	48.34

A	B	SC(A, B) (in %)	A	B	SC(A, B) (in %)
RNAiFOLD	antaRNA	65.05	INFO-RNA	antaRNA	22.53
	INFO-RNA	10.34		MODENA	8.55
	MODENA	13.37		m2dRNAs	2.79
	m2dRNAs	0.00		RNAiFOLD	2.30
	RNAinverse	73.28		RNAinverse	64.66
	NUPACK	75.86		NUPACK	13.79
	fRNAkenstein	79.31		fRNAkenstein	58.62
	DSS-Opt	10.34		DSS-Opt	4.14
	ERD	25.83		ERD	12.85

A	B	SC(A, B) (in %)	A	B	SC(A, B) (in %)
antaRNA	INFO-RNA	3.45	RNAinverse	antaRNA	0.00
	MODENA	0.00		INFO-RNA	0.00
	m2dRNAs	0.00		modena	0.00
	RNAiFOLD	0.00		m2dRNAs	0.00
	RNAinverse	60.34		RNAiFOLD	0.00
	NUPACK	6.90		NUPACK	0.00
	fRNAkenstein	65.52		fRNAkenstein	41.38
	DSS-Opt	3.45		DSS-Opt	0.00
	ERD	0.69		ERD	0.00

A	B	SC(A, B) (in %)	A	B	SC(A, B) (in %)
fRNAkenstein	antaRNA	3.45	ERD	antaRNA	58.68
	INFO-RNA	0.00		INFO-RNA	18.39
	MODENA	0.00		MODENA	9.41
	m2dRNAs	0.00		m2dRNAs	0.16
	RNAiFOLD	0.00		RNAiFOLD	8.43
	RNAinverse	18.97		RNAinverse	78.45
	NUPACK	6.90		NUPACK	27.59
	DSS-Opt	3.45		fRNAkenstein	79.31
	ERD	0.00		DSS-Opt	6.90

In Figure 3, we present a runtime comparison among the different methods involved in the study. Given an input target structure, the running time of each method corresponds with the time spent by each method in obtaining 50 RNA sequences. First, we analyze the required runtime of each methods depending on the length of the target sequences, see Figure 3(a). As we may observe, with small and medium length (lenght$<$120), all the methods except antaRNA and NUPACK require a similar amount of time, highlighting the two fastest approaches: ERD and MODENA. With medium-large target structures (length$\geq$150), the fastest approaches resulted to be



(a)

(b)

Fig. 3. Runtime Analysis. In (a), we show the runtime spent by each method depending on the length of the RNA structure. In (b), we present the total runtime required by each method in designing all the Rfam structures (in seconds).

MODENA, RNAiFOLD and m2dRNAs. These conclusions are corroborated with the plot shown in Figure 3(b), where it is clear that RNAiFOLD, MODENA and m2dRNAs are clearly the three fastest RNA inverse folding methods.

In Figure 4, we analyze the RNA sequences designed for the RF00009.115.ss structure. This specific structure was selected after considering the five longest Rfam sequences, which are: RF00024.16.ss (452 nucleotides), RF00011.18.ss (383 nucleotides), RF00018.2.ss (361 nucleotides), RF00010.253.ss (358 nucleotides), and RF00009.115.ss (349 nucleotides). Discarding those impossible-to-design Rfam structures (RF00024.16.ss, RF00011.18.ss, and RF00010.253.ss), only two remains: RF00018.2.ss and RF00009.115.ss. Taking into account that both structures are of a very similar length (361 and 349 nucleotides, respectively), we decided to choose RF00009.115.ss because it was successfully solved by 7 out of the 10 algorithms under study, whereas RF00018.2.ss was only successfully solved by 2 algorithms (m2dRNAs and RNAiFOLD).

TABLE IV
AVERAGE NUCLEOTIDE DISTRIBUTION IN THE GENERATED SEQUENCES FOUND BY EACH METHOD IN RFAM STRUCTURES.

	Paired (%)			*
	GC	*AU*	*GU*	
m2dRNAs	65.01	28.91	6.08	0.1064
fRNAkenstein	46.44	38.03	15.53	0.1350
ERD	68.54	23.51	7.95	0.1418
RNAinverse	43.46	38.17	18.37	0.1669
antaRNA	59.11	40.89	0.00	0.1709
NUPACK	74.24	25.76	0.00	0.2200
MODENA	76.18	23.82	0.00	0.2398
DSS-Opt	92.29	7.71	0.00	0.4372
RNAiFOLD	91.85	5.01	3.15	0.4400
INFO-RNA	94.18	4.21	1.61	0.4666
Natural [21]	57.00	30.00	13.00	

* Euclidean distance to the Natural RNA sequence reported in [21]

	Unpaired (%)				*
	A	*C*	*G*	*U*	
RNAinverse	29.46	23.77	20.59	26.19	0.0458
fRNAkenstein	31.74	25.93	17.44	24.89	0.0858
INFO-RNA	36.58	21.10	23.93	18.40	0.1093
m2dRNAs	38.60	19.26	14.22	27.92	0.1235
antaRNA	34.30	29.65	12.80	23.24	0.1516
ERD	47.29	13.13	20.34	19.24	0.2034
NUPACK	42.76	27.46	12.30	17.48	0.2058
MODENA	79.43	7.48	6.37	6.73	0.5734
DSS-Opt	89.43	1.31	9.24	0.01	0.6927
RNAiFOLD	95.89	1.18	1.23	1.69	0.7623
Natural [21]	30.00	20.00	23.00	27.00	

* Euclidean distance to the Natural RNA sequence reported in [21]

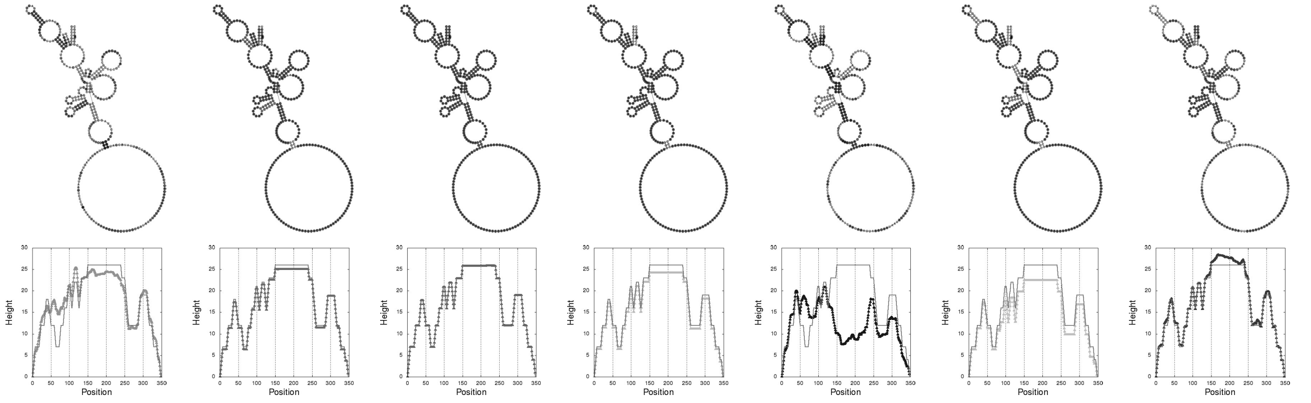
	Total (%)				*
	A	*C*	*G*	*U*	
RNAinverse	23.17	22.65	26.57	27.61	0.0411
fRNAkenstein	25.65	24.12	24.06	26.17	0.0522
m2dRNAs	26.21	26.25	24.37	23.16	0.0541
ERD	29.80	23.34	29.36	17.49	0.0953
antaRNA	27.90	29.63	20.31	22.15	0.1088
NUPACK	28.02	32.38	24.24	15.35	0.1358
INFO-RNA	17.02	36.08	37.28	9.62	0.2178
MODENA	44.40	23.41	22.93	9.26	0.2648
DSS-Opt	44.19	25.40	28.20	2.22	0.3042
RNAiFOLD	46.57	24.76	25.63	3.04	0.3164
Natural [21]	23.00	24.00	28.00	24.00	

* Euclidean distance to the Natural RNA sequence reported in [21]

Figures 4(a) to 4(g) show the predicted structure of a selected RNA design found by the aforementioned methods, where the color scheme represents the base pair probabilities and the mountain plot represents a secondary structure in a plot of height versus position, where the height $m(k)$ is given by the number of base pairs enclosing the base at position k. As we can see, the top-3 methods for RF00009.115.ss are: m2dRNAs, RNAiFOLD, and MODENA – This highlights the particularly good performance of our approach.

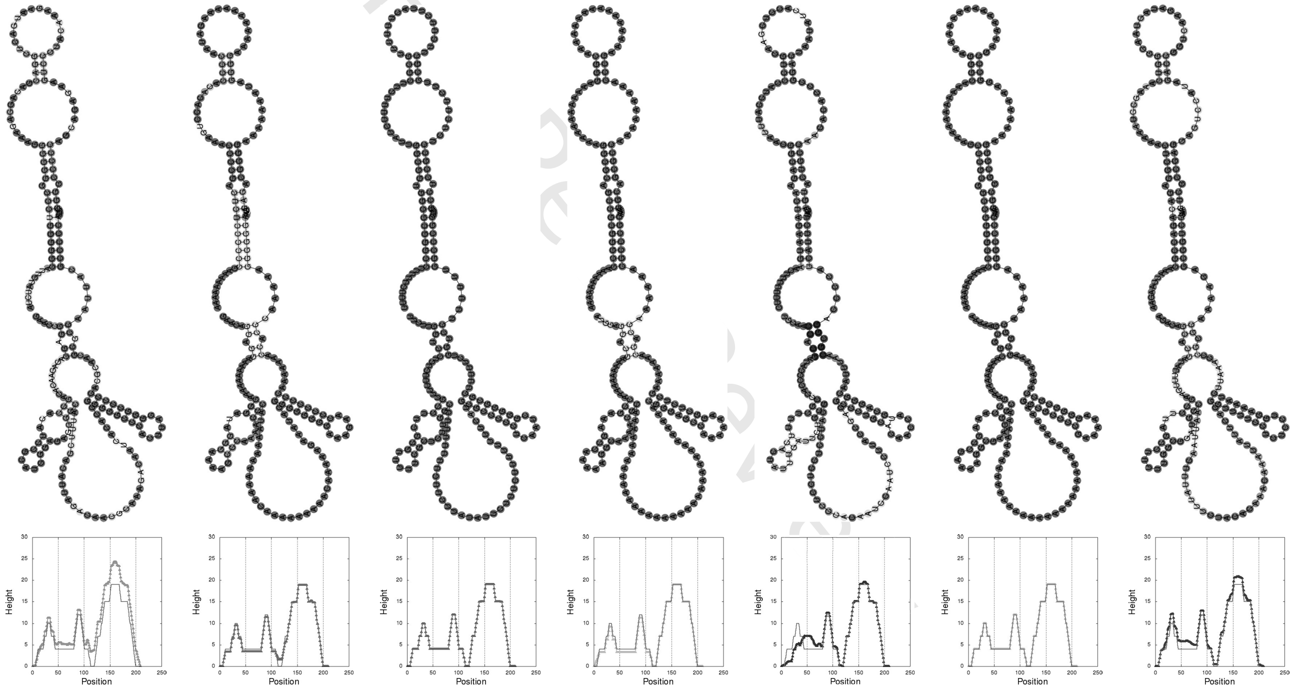
In Figure 5, we study the RNA sequences found by the different approaches for the RF00025.12.ss structure. According to Table I, it is the least favourable structure for our method. As we may observe in Figures 5(a) to 5(g), the two best methods for RF00025.12.ss are DSS-Opt and m2dRNAs, which their quality is far from the quality of the other ones.

We have analyzed so far the quality of the different methods in terms of partition function and ensemble diversity (HV and SC metrics); however, the nucleotides distribution of base pairs and nucleotides in the successful designs for each method needs to be studied in order to determine which method produce RNA sequences with distribution closer to the natural distribution. Like other authors [21], we present in Table IV



(a) INFO-RNA (b) MODENA (c) m2dRNAs (d) RNAiFOLD (e) fRNAkenstein (f) DSS-Opt (g) ERD

Fig. 4. Analysis of the RNA sequences designed for the RF00009.115.ss structure. Figures (a) to (g) show the structure of the RNA sequence found by INFO-RNA, MODENA, m2dRNAs, RNAiFOLD, fRNAkenstein, DSS-Opt, and ERD, where the color scheme represents the base pair probabilities and the mountain plot represents a secondary structure in a plot of height versus position, where the height $m(k)$ is given by the number of base pairs enclosing the base at position k. Note that antaRNA, RNAinverse, and NUPACK did not find any successful RNA design for RF00009.115.ss.



(a) INFO-RNA (b) MODENA (c) m2dRNAs (d) RNAiFOLD (e) antaRNA (f) DSS-Opt (g) ERD

Fig. 5. Analysis of the RNA sequences designed for the RF00025.12.ss structure. Figures (a) to (g) show the structure of the RNA sequence found by INFO-RNA, MODENA, m2dRNAs, RNAiFOLD, antaRNA, DSS-Opt, and ERD, where the color scheme represents the base pair probabilities and the mountain plot represents a secondary structure in a plot of height versus position, where the height $m(k)$ is given by the number of base pairs enclosing the base at position k. Note that fRNAkenstein, RNAinverse, and NUPACK did not find any successful RNA design for RF00025.12.ss.

the average nucleotide distribution in their designed RNA sequences. In Table IV, the first group indicates the distribution over the three types of base pairs in paired positions, the second group shows the nucleotide distribution for unpaired positions, and the last group shows the overall nucleotide distribution in the sequences. For each of the aforementioned groups we calculate the Euclidean distance between the natural distribution [21] and each method involved in this study. As we may observe, INFO-RNA, DSS-Opt, and RNAiFOLD are heavily biased towards GC base pairs, whereas m2dRNAs, fRNAkenstein, and ERD are closer methods to the natural distribution of base pairs. If we focus on the second group (unpaired), we find that RNAiFOLD, DSS-Opt, and MODENA are heavily biased towards A nucleotide. In contrast, we can see that our proposal (m2dRNAs) is not biased towards any nucleotide, showing a distribution reasonably close to the natural distribution. Finally, if we compare the overall nucleotide distributions, we conclude that the sequences obtained by RNAinverse, fRNAkenstein, and m2dRNAs are very well-distributed among the four bases, and their distributions are close to the natural one.

The last comparison shows the reliability of each method.

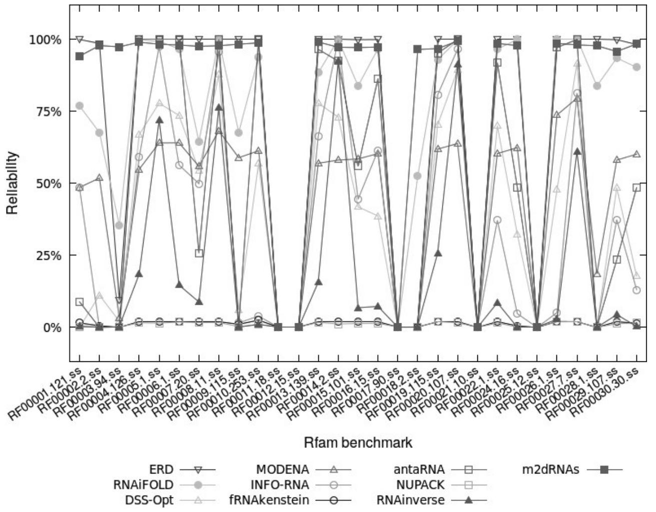


Fig. 6. Reliability comparison among m2dRNAs and the contestant techniques (Rfam benchmark).

TABLE V
COMPARISON AMONG DIVERSE METHODS WHEN SOLVING THE RFAM BENCHMARK IN TERMS OF AVERAGE RELIABILITY (IN %).

Average Reliability (in %)			
m2dRNAs	77.42	INFO-RNA	37.72
ERD	71.17	fRNAkenstein	1.26
RNAiFOLD	67.23	antaRNA	50.10
DSS-Opt	38.80	NUPACK	0.88
MODENA	42.26	RNAinverse	18.34

As we mentioned in Section IV.A (Methodology), all the methods involved in the comparison were configured to return a total of 50 RNA designs; therefore, the reliability metric indicates how many RNA sequences were successfully designed by each method (in percentage).

In Figure 6, we present a comparison between m2dRNAs and the other contestant techniques in terms of reliability. As we can see, m2dRNAs and ERD are able to obtain 50 different RNA designs in almost all Rfam datasets (excluding the five impossible-to-design structures). antaRNA, RNAiFOLD, MODENA, DSS-Opt, and INFO-RNA output approximately half of the structures requested. Finally, RNAinverse, fRNAkenstein, and NUPACK are able to obtain a poor pool of different solutions. The average reliability of each method for the 29 Rfam datasets is presented in Table V.

All in all, we can say that our Multiobjective Metaheuristic To design RNA Sequences is not only able to obtain a reliable set of RNA designs in a single run but also the nucleotide distribution of its designed sequences is reasonably natural.

The second benchmark tested is Eterna100. In this case, we compare our proposal (m2dRNAs) with Eterna players, ERD, fRNAkenstein, MODENA, antaRNA, INFO-RNA, DSS-Opt, NUPACK, RNAiFOLD, RNA-SSD, and RNAinverse. As we mentioned in Section IV-A, we have followed the same methodology as Anderson-Lee et al. [26], that is, all structures of Eterna100 were attempted five times by each algorithm with a total time limit of twenty-four hours.

In Table VI, we present the number of Eterna100 structures successfully solved by each method under study. As

TABLE VI
COMPARISON AMONG DIVERSE METHODS WHEN SOLVING THE ETERNA100 BENCHMARK. THE SUCCESSES (■) AND FAILURES (□) ARE SHOWN FOR EACH ALGORITHM.

Structure	Len.	RNAinverse	RNA-SSD	RNAiFOLD	NUPACK	DSS-Opt	INFO-RNA	antaRNA	MODENA	fRNAkenstein	ERD	m2dRNAs	†	Players*
ET01	16	■	■	■	■	■	■	■	■	■	■	■	(0.37)	98.37
ET02	116	■	□	■	■	■	■	■	■	■	■	■	(1.89)	36.62
ET03	36	■	□	■	■	■	■	■	■	■	■	■	(0.39)	25.91
ET04	192	□	□	□	■	■	■	■	■	■	■	■	(2.88)	22.12
ET05	101	■	■	■	■	□	■	■	■	■	■	■	(1.57)	16.97
ET06	379	□	□	□	■	□	□	■	■	■	■	■	(9.73)	5.36
ET07	284	■	□	■	■	■	■	■	■	■	■	■	(7.12)	0.60
ET08	12	■	■	■	■	■	■	■	■	■	■	■	(0.36)	0.41
ET09	258	□	■	■	■	■	■	□	■	■	□	■	(8.03)	0.40
ET10	45	■	□	■	■	■	□	■	■	■	■	■	(0.39)	0.29
ET11	36	□	■	■	■	■	■	■	■	■	■	■	(0.38)	0.29
ET12	213	■	□	■	■	■	■	■	■	■	■	■	(6.08)	0.28
ET13	67	■	□	■	■	■	■	■	■	■	■	■	(0.88)	0.26
ET14	98	□	□	□	□	■	□	■	□	■	■	■	(0.38)	0.24
ET15	30	□	■	□	□	■	□	□	□	■	■	■	(0.15)	0.24
ET16	105	□	□	□	■	■	■	□	■	□	■	■	(0.43)	0.24
ET17	105	□	□	□	□	■	■	□	□	■	■	■	(0.35)	0.23
ET18	67	■	□	■	■	■	■	■	■	■	■	■	(0.46)	0.23
ET19	104	■	□	■	■	■	■	■	■	■	■	■	(0.89)	0.23
ET20	42	■	■	□	■	■	■	■	■	■	■	■	(0.29)	0.23
ET21	105	□	□	■	■	■	□	■	■	■	■	■	(0.65)	0.23
ET22	400	□	□	■	■	■	□	■	□	□	□	■	(11.97)	0.22
ET23	17	■	■	■	■	■	■	□	■	■	■	■	(0.13)	0.22
ET24	104	□	□	■	■	■	■	■	■	■	■	■	(0.71)	0.20
ET25	62	□	■	■	□	□	■	■	■	■	■	■	(0.23)	0.20
ET26	26	■	■	■	■	■	■	■	■	■	■	■	(0.25)	0.20
ET27	57	□	□	■	■	□	□	■	□	■	■	■	(0.37)	0.19
ET28	378	■	□	■	■	■	■	■	■	■	■	■	(15.48)	0.18
ET29	105	□	□	■	■	□	■	■	■	■	■	■	(1.32)	0.18
ET30	31	■	□	■	■	■	■	■	■	■	■	■	(0.31)	0.18
ET31	102	■	□	■	■	□	□	■	■	■	■	■	(1.37)	0.18
ET32	122	■	□	■	□	□	□	■	■	■	■	■	(0.86)	0.18
ET33	45	□	□	□	□	□	■	□	□	■	■	■	(1.81)	0.17
ET34	96	□	□	■	■	■	■	■	■	□	■	■	(0.58)	0.17
ET35	123	□	■	□	□	□	■	□	■	□	■	■	(8.24)	0.16
ET36	151	□	□	■	■	■	■	■	■	■	■	■	(1.38)	0.16
ET37	103	□	□	□	□	□	■	□	□	□	■	■	(0.4)	0.16
ET38	316	□	□	□	■	■	□	□	□	□	■	■	(4.95)	0.14
ET39	174	■	□	■	■	■	■	■	■	■	■	■	(4.72)	0.14
ET40	38	■	■	■	■	■	■	■	■	■	■	■	(0.27)	0.14
ET41	35	□	■	■	□	■	■	□	■	■	■	■	(0.47)	0.13
ET42	116	□	□	■	■	□	■	■	■	□	■	■	(0.77)	0.13
ET43	85	□	□	■	■	■	■	■	■	■	■	■	(0.41)	0.13
ET44	63	■	□	■	■	■	■	■	■	■	■	■	(0.38)	0.13
ET45	63	■	■	■	■	■	■	■	□	■	■	■	(0.5)	0.13
ET46	97	■	□	□	■	■	■	■	■	■	■	■	(0.49)	0.13
ET47	50	□	□	□	□	□	□	■	■	■	■	■	(0.19)	0.12
ET48	102	■	□	■	■	■	■	■	■	■	■	■	(0.59)	0.12
ET49	101	□	□	■	□	■	□	■	■	■	■	■	(0.47)	0.10
ET50	105	□	□	□	□	□	□	□	■	□	□	■	(640.21)	0.10
ET51	337	■	□	□	□	□	□	■	□	■	■	■	(11.01)	0.09
ET52	80	□	■	□	□	□	□	□	■	□	□	■	(174.38)	0.09
ET53	400	□	□	□	□	■	□	□	■	□	□	■	(7.54)	0.08
ET54	92	■	■	■	■	■	■	■	■	■	■	■	(0.99)	0.08
ET55	102	□	□	■	□	□	■	■	□	■	■	■	(0.59)	0.08
ET56	387	■	□	■	■	■	■	■	■	■	■	■	(24.98)	0.08
ET57	36	□	■	□	□	□	■	□	□	□	□	■	(64.43)	0.08
ET58	111	■	■	■	■	■	■	■	■	■	■	■	(0.95)	0.07
ET59	68	□	□	□	■	□	□	□	□	■	□	■	(251.39)	0.07
ET60	105	□	□	□	□	□	□	□	□	□	□	□		0.07
ET61	67	□	■	□	□	□	□	□	■	□	□	□		0.06
ET62	53	□	■	□	□	□	■	□	□	□	■	■	(110.48)	0.06
ET63	400	□	■	□	■	■	■	■	■	■	■	■	(14.29)	0.06
ET64	110	□	□	□	□	□	□	□	□	□	□	□		0.05
ET65	40	□	■	□	□	□	□	□	□	□	□	□		0.05
ET66	36	□	□	□	□	□	□	□	□	□	■	□		0.05
ET67	61	□	□	□	□	□	□	□	□	□	□	□		0.05
ET68	282	□	□	□	□	□	■	□	□	□	□	□		0.05
ET69	200	□	□	□	□	■	□	□	■	□	□	■	(6.79)	0.05
ET70	184	□	■	■	■	■	■	■	■	■	■	■	(1.95)	0.05
ET71	88	□	□	□	□	□	□	□	□	□	□	□		0.05
ET72	73	□	■	□	□	□	□	□	□	■	■	■	(521.34)	0.04
ET73	370	□	□	□	□	□	□	□	□	□	□	□		0.04
ET74	380	□	□	□	□	□	■	□	□	□	□	■	(2440.57)	0.04
ET75	75	□	□	■	■	■	□	□	■	■	■	■	(0.31)	0.04
ET76	393	□	□	□	□	□	□	□	□	□	□	□		0.04
ET77	106	□	□	□	□	□	□	□	□	□	□	□		0.04
ET78	284	□	□	□	□	□	□	□	□	□	□	□		0.04
ET79	101	□	□	□	■	□	□	□	□	□	□	□		0.04
ET80	397	□	□	□	□	□	□	□	□	□	□	□		0.03
ET81	212	□	□	□	□	□	□	□	□	□	□	□		0.03
ET82	214	□	■	□	■	□	■	■	■	■	■	■	(4.37)	0.03
ET83	119	□	■	□	□	□	■	□	■	□	■	■	(287.67)	0.03
ET84	104	□	□	□	□	■	□	■	□	■	■	■	(0.53)	0.03
ET85	389	□	□	□	□	□	□	□	□	□	■	■	(151.93)	0.02
ET86	355	□	□	□	□	□	□	□	□	□	□	□		0.02
ET87	97	□	□	□	□	□	□	□	□	□	□	□		0.02
ET88	34	□	■	□	□	□	□	□	□	□	□	□		0.02
ET89	97	□	□	□	□	□	□	□	□	□	□	□		0.02
ET90	400	□	□	□	□	□	□	□	□	□	□	□		0.02
ET91	392	□	□	□	□	□	□	□	□	□	□	□		0.02
ET92	100	□	□	□	□	□	□	□	□	□	□	□		0.01
ET93	116	□	□	■	■	□	■	■	■	■	■	■	(1.15)	0.01
ET94	389	□	□	□	□	□	□	□	□	□	■	□		0.01
ET95	82	□	□	□	□	■	□	■	□	■	■	■	(0.47)	0.01
ET96	358	□	□	□	□	□	□	□	□	□	□	□		0.01
ET97	400	□	□	□	□	□	□	□	□	□	□	□		0.01
ET98	202	□	□	□	□	□	□	□	□	■	■	■	(5.73)	0.01
ET99	364	□	□	□	□	□	□	□	□	□	□	□		0.01
ET100	382	□	□	□	□	□	□	□	□	□	□	□		0.00
		28	27	43	48	47	50	50	54	57	66	73		

[†] Average runtime (in seconds) spent by m2dRNAs in finding a successfully designed RNA sequence. For those dataset failed, the required runtime is always 24 hours (not shown).

[*] Percentage (%) of Eterna100 players that successfully solve the puzzle, a total of 60,710 players were considered in [26].

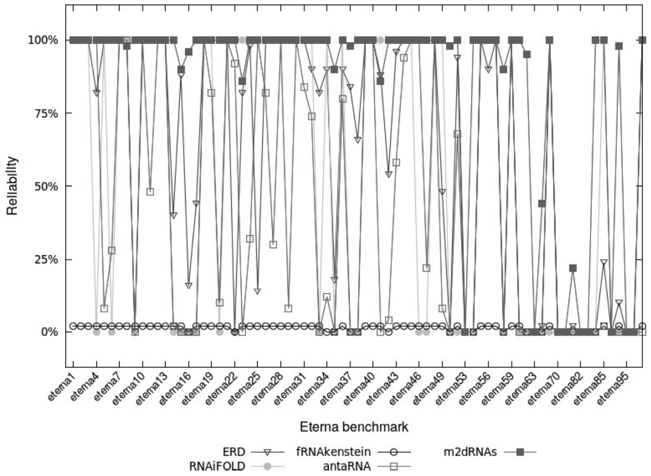


Fig. 7. Reliability comparison among m2dRNAs and the contestant techniques (Eterna benchmark).

TABLE VII

Comparison among diverse methods when solving the Eterna benchmark in terms of average reliability (in %).

Average Reliability (in %)	
m2dRNAs	69.91
ERD	52.18
RNAiFOLD	43.00
fRNAkenstein	1.14
antaRNA	36.40

we can see, the notation used for successes and failures is: '■' and '□', respectively. The Eterna players found at least one successful solution for almost every Eterna100 structures; however, the percentage of players capable of solving these structures is very low (<1% in 64 out of the 100 structures). As pointed by Anderson-Lee et al. [26], many Eterna100 structures were only solved by the creator. If we focus on the RNA inverse folding methods, we can see that m2dRNAs outperforms the other methods obtaining successful RNA designs in 73 structures, whereas the second and third best approaches (ERD and fRNAkenstein) solved 66 and 57 structures, respectively.

Next, we compare the required runtime of the different approaches. As we have mentioned before, for each Eterna dataset unsuccessfully solved, an algorithm spent twenty-four hours, therefore, the total runtime of the algorithms will be affected by the number of dataset failed. Note that, the results of Eterna players, MODENA, INFO-RNA, DSS-Opt, NUPACK, RNA-SSD, and RNAinverse were obtained from [26]; therefore, their runtime are not available. The total runtime, in hours:minutes:seconds format, spent by the remaining methods are: 1368:07:55 (RNAiFold), 1269:12:15 (antaRNA), 1042:00:28 (fRNAkenstein), 819:22:40 (ERD), and 649:20:27 (m2dRNAs). In Table VI, we present, for each Eterna dataset successfully solved, the computational time spent by m2dRNAs (in seconds).

As we did with the Rfam benchmark, we conclude by measuring the reliability of m2dRNAs and compare it against the available contestant techniques (ERD, RNAiFOLD, fRNAkenstein, and antaRNA). As we can see in Figure 7, m2dRNAs is more reliable than the other methods in the vast majority of Eterna datasets; therefore, it is not only able to solve more Eterna datasets (see Table VI), but it also outputs a higher number of solutions. The average reliability of each method for the 100 Eterna datasets is presented in Table VII.

V. Conclusions and Future Works

To find a sequence of base pairs that would fold into a given target structure is known as the RNA Inverse Folding problem.

In this work, a Multiobjective Evolutionary Algorithm (MOEA) has been applied for solving this problem, we refer to our approach as Multiobjective Metaheuristic To Design RNA Sequences (m2dRNAs). This multiobjective approach is focused on solving a new multiobjective definition of the RNA inverse folding problem. In this new multiobjective definition of the problem, we have considered the similarity between target and predicted structures as a constraint, and focused on optimizing the following three objective functions simultaneously: (i) Partition Function (free energy of the ensemble), (ii) Ensemble Diversity, (iii) Nucleotides Composition.

We have evaluated the performance of m2dRNAs and other RNA inverse folding approaches with two benchmarks: Rfam and Eterna100. The methods involved in the comparative study are: RNAinverse, RNA-SSD, INFO-RNA, MODENA, NUPACK, fRNAkenstein, DSS-Opt, RNAiFOLD, antaRNA, ERD, and Eterna players. From this comparison, we can conclude that m2dRNAs is capable of obtaining very promising results, outperforming the results obtained by other methods published in the literature. In addition, the required runtime of m2dRNAs is also an advantage when the length of the target structure starts to grow.

In the last years, several MOEAs have been proposed. We may classify them into three groups: (i) dominance-based, (ii) indicator-based and (iii) decomposition-based. Therefore, a straightforward line of future work is to analyze the performance of different MOEAs when solving the multiobjective version of the RNA inverse folding problem proposed in this work. Another interesting line of future work is to study the behaviour of m2dRNAs by using different chromosome encodings (e.g. an integer encoding), mutation and crossover operators, as well as a deeper statistical analysis focusing on different objectives optimized: not only the ones used by m2dRNAs but also those used by other well-known approaches. In addition, m2dRNAs will be extended for allowing multiple target structures, variety of design constraints (base-pairs bounds, specified motifs) and pseudoknot prediction methods, such as IPknot [16] and HotKnots [17]. Finally, another challenging future work is to analyze the scalability of diverse algorithms when they are applied to the RNA inverse folding problem using structures of increasing length.

References

[1] T. R. Cech, "RNA finds a simpler way," *Nature*, vol. 428, no. 6980, pp. 263–264, 2004.

[2] T. Ivry, S. Michal, A. Avihoo, G. Sapiro, and D. Barash, "An image processing approach to computing distances between RNA secondary structures dot plots," *Algorithms for Molecular Biology*, vol. 4, no. 1, 2009. [Online]. Available: https://doi.org/10.1186/1748-7188-4-4

[3] A. Busch and R. Backofen, “INFO-RNA – a fast approach to inverse RNA folding,” *Bioinformatics*, vol. 22, no. 15, pp. 1823–1831, 2006.

[4] 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,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 371, no. 2000, 2013. [Online]. Available: http://rsta.royalsocietypublishing.org/content/371/2000/20120310

[5] A. Churkin, M. D. Retwitzer, V. Reinharz, Y. Ponty, J. Waldispühl, and D. Barash, “Design of RNAs: Comparing Programs for inverse RNA folding,” *Briefings in Bioinformatics*, 2017. [Online]. Available: https://doi.org/10.1093/bib/bbw120

[6] S. F. Greenbury, S. Schaper, S. E. Ahnert, and A. A. Louis, “Genetic correlations greatly increase mutational robustness and can both reduce and enhance evolvability,” *PLOS Computational Biology*, vol. 12, no. 3, 2016. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1004773

[7] R. Aguirre-Hernández, H. H. Hoos, and A. Condon, “Computational RNA secondary structure design: empirical complexity and improved methods,” *BMC bioinformatics*, vol. 8, no. 1, 2007. [Online]. Available: https://doi.org/10.1186/1471-2105-8-34

[8] T. Jörg, O. C. Martin, and A. Wagner, “Neutral network sizes of biological RNA molecules can be computed and are not atypically small,” *BMC Bioinformatics*, vol. 9, no. 1, 2008. [Online]. Available: https://doi.org/10.1186/1471-2105-9-464

[9] 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 Chemie/Chemical Monthly*, vol. 125, no. 2, pp. 167–188, 1994.

[10] A. R. Gruber, R. Lorenz, S. H. Bernhart, R. Neuböck, and I. L. Hofacker, “The Vienna RNA Websuite,” *Nucleic acids research*, vol. 36, no. suppl 2, pp. W70–W74, 2008.

[11] 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 Chemie/Chemical Monthly*, vol. 125, no. 2, pp. 167–188, 1994.

[12] M. Andronescu, A. P. Fejes, F. Hutter, H. H. Hoos, and A. Condon, “A new algorithm for RNA secondary structure design,” *Journal of molecular biology*, vol. 336, no. 3, pp. 607–624, 2004.

[13] A. Taneda, “MODENA: a multi-objective RNA inverse folding,” *Adv. Appl. Bioinform. Chem*, vol. 4, pp. 1–12, 2011.

[14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A Fast Elitist Multi-Objective Genetic Algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2000.

[15] A. Taneda, “Multi-objective genetic algorithm for pseudoknotted RNA sequence design,” *Frontiers in Genetics*, vol. 3, 2012. [Online]. Available: https://doi.org/10.3389/fgene.2012.00036

[16] K. Sato, Y. Kato, M. Hamada, T. Akutsu, and K. Asai, “IPknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming,” *Bioinformatics*, vol. 27, no. 13, pp. i85–i93, jun 2011. [Online]. Available: https://doi.org/10.1093/bioinformatics/btr215

[17] J. Ren, B. Rastegari, A. Condon, and H. H. Hoos, “HotKnots: heuristic prediction of RNA secondary structures including pseudoknots,” *RNA*, vol. 11, no. 10, pp. 1494–1504, 2005.

[18] A. Taneda, “Multi-objective optimization for RNA design with multiple target secondary structures,” *BMC Bioinformatics*, vol. 16, no. 1, 2015. [Online]. Available: https://doi.org/10.1186/s12859-015-0706-x

[19] J. N. Zadeh, B. R. Wolfe, and N. A. Pierce, “Nucleic acid sequence design via efficient ensemble defect optimization,” *Journal of computational chemistry*, vol. 32, no. 3, pp. 439–452, 2011.

[20] R. M. Dirks, M. Lin, E. Winfree, and N. A. Pierce, “Paradigms for computational nucleic acid design,” *Nucleic Acids Research*, vol. 32, no. 4, pp. 1392–1403, 2004.

[21] R. B. Lyngsø, J. W. Anderson, E. Sizikova, A. Badugu, T. Hyland, and J. Hein, “fRNAnakenstein: multiple target inverse RNA folding,” *BMC Bioinformatics*, vol. 13, no. 1, 2012. [Online]. Available: https://doi.org/10.1186/1471-2105-13-260

[22] M. C. Matthies, S. Bienert, and A. E. Torda, “Dynamics in sequence space for RNA secondary structure design,” *Journal of chemical theory and computation*, vol. 8, no. 10, pp. 3663–3670, 2012.

[23] J. A. Garcia-Martin, P. Clote, and I. Dotu, “RNAiFold: a constraint programming algorithm for RNA inverse folding and molecular design,” *Journal of bioinformatics and computational biology*, vol. 11, no. 2, 2013. [Online]. Available: https://doi.org/10.1142/s0219720013500017

[24] 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 research*, vol. 43, no. W1, pp. W513–W521, 2015.

[25] J. Lee, W. Kladwang, M. Lee, D. Cantu, M. Azizyan, H. Kim, A. Limpaecher, S. Gaikwad, S. Yoon, A. Treuille, R. Das, and E. Participants, “RNA design rules from a massive open laboratory,” *Proceedings of the National Academy of Sciences*, vol. 111, no. 6, pp. 2122–2127, 2014.

[26] J. Anderson-Lee, E. Fisker, V. Kosaraju, M. Wu, J. Kong, J. Lee, M. Lee, M. Zada, A. Treuille, and R. Das, “Principles for predicting RNA secondary structure design difficulty,” *Journal of molecular biology*, vol. 428, no. 5, pp. 748–757, 2016.

[27] 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. [Online]. Available: https://doi.org/10.1093/bioinformatics/btu001

[28] A. Esmaili-Taheri and M. Ganjtabesh, “ERD: a fast and reliable tool for RNA design including constraints,” *BMC Bioinformatics*, vol. 16, no. 1, 2015. [Online]. Available: https://doi.org/10.1186/s12859-014-0444-5

[29] R. Kleinkauf, M. Mann, and R. Backofen, “antaRNA: ant colony-based RNA sequence design,” *Bioinformatics*, vol. 31, no. 19, pp. 3114–3121, 2015.

[30] R. Kleinkauf, T. Houwaart, R. Backofen, and M. Mann, “antaRNA – multi-objective inverse folding of pseudoknot RNA using ant-colony optimization,” *BMC Bioinformatics*, vol. 16, no. 1, 2015. [Online]. Available: https://doi.org/10.1186/s12859-015-0815-6

[31] J. D. Knowles, R. A. Watson, and D. W. Corne, “Reducing local optima in single-objective problems by multi-objectivization,” *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 269–283, 2001.

[32] S. Bleuler, J. Bader, and E. Zitzler, “Reducing bloat in GP with multiple objectives,” *Natural Computing Series*, pp. 177–200, 2008.

[33] S. A. Thomas and Y. Jin, “Single and multi-objective in silico evolution of tunable genetic oscillators,” *Lecture Notes in Computer Science*, vol. 7811, pp. 696–709, 2013.

[34] J. Handl, S. C. Lovell, and J. Knowles, “Investigations into the effect of multiobjectivization in protein structure prediction,” *Parallel Problem Solving from Nature – PPSN X*, pp. 702–711, 2008.

[35] J. Handl, D. B. Kell, and J. Knowles, “Multiobjective Optimization in Bioinformatics and Computational Biology,” *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 4, no. 2, pp. 279–292, 2007.

[36] A. Rubio-Largo, M. A. Vega-Rodríguez, and D. L. González-Álvarez, “Hybrid multiobjective artificial bee colony for multiple sequence alignment,” *Applied Soft Computing*, vol. 41, pp. 157–168, 2016.

[37] ——, “A Hybrid Multiobjective Memetic Metaheuristic for Multiple Sequence Alignment,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 499–514, 2016.

[38] ——, “A multiobjective approach based on artificial bee colony for the static routing and wavelength assignment problem,” *Soft Computing*, vol. 17, no. 2, pp. 199–211, 2013.

[39] I. L. Hofacker, “Vienna RNA secondary structure server,” *Nucleic acids research*, vol. 31, no. 13, pp. 3429–3431, 2003.

[40] A. Wilm, D. G. Higgins, and C. Notredame, “R-Coffee: a method for multiple alignment of non-coding RNA,” *Nucleic acids research*, vol. 36, no. 9, pp. e52–e52, 2008.

[41] K. J. Doshi, J. J. Cannone, C. W. Cobaugh, and R. R. Gutell, *BMC Bioinformatics*, vol. 5, no. 1, p. 105, 2004. [Online]. Available: https://doi.org/10.1186/1471-2105-5-105

[42] K. Darty, A. Denise, and Y. Ponty, “VARNA: Interactive drawing and editing of the RNA secondary structure,” *Bioinformatics*, vol. 25, no. 15, pp. 1974–1975, 2009. [Online]. Available: https://doi.org/10.1093/bioinformatics/btp250

[43] R. B. Lyngsø, “RNA secondary structure prediction by minimum free energy,” *Encyclopedia of Algorithms*, pp. 782–785, 2008. [Online]. Available: https://doi.org/10.1007/978-0-387-30162-4_347

[44] M. Schnall-Levin, L. Chindelevitch, and B. Berger, “Inverting the Viterbi algorithm: an abstract framework for structure design,” *Proceedings of the 25th international conference on Machine learning*, pp. 904–911, 2008.

[45] Y. Collette and P. Siarry, *Multiobjective optimization: principles and case studies*. Springer Science & Business Media, 2013.

[46] J. S. McCaskill, “The equilibrium partition function and base pair binding probabilities for RNA secondary structure,” *Biopolymers*, vol. 29, no. 6-7, pp. 1105–1119, 1990.

[47] R. B. Lyngsø, “RNA secondary structure boltzmann distribution,” *Encyclopedia of Algorithms*, pp. 777–779, 2008. [Online]. Available: https://doi.org/10.1007/978-0-387-30162-4_345

[48] R. Lorenz, S. H. Bernhart, C. H. zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker, “ViennaRNA package 2.0,” *Algorithms*

for Molecular Biology, vol. 6, no. 1, 2011. [Online]. Available: https://doi.org/10.1186/1748-7188-6-26

[49] 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. [Online]. Available: https://doi.org/10.1016/j.ymeth.2016.04.004

[50] N. Srinivas and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evolutionary Computation*, vol. 2, pp. 221–248, 1994.

[51] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints." *IEEE Trans. Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[52] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Performance comparison of NSGA-II and NSGA-III on various many-objective test problems," *IEEE Congress on Evolutionary Computation (CEC)*, pp. 3045–3052, 2016.

[53] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[54] E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evolutionary Computation*, vol. 8, pp. 173–195, 2000.

[55] D. J. Sheskin, "Handbook of Parametric and Nonparametric Statistical Procedures," *5th ed. New York: Chapman & Hall/CRC Press*, 2011.

Álvaro Rubio-Largo received the Ph.D. degree in Computer Engineering from the University of Extremadura, Extremadura, Spain, in 2013. He is currently working at NOVA Information Management School (NOVA IMS), Universidade Nova de Lisboa. He has authored or coauthored 60+ publications including 25+ Journal Citation Report (JCR) papers. His main research interests include big data, machine learning and the use of evolutionary computation for solving real-world multiobjective optimization problems He has co-organized several international workshops and formed part of technical programm committee in a number of international conferences. He has also co-edited a special issue of an international JCR-indexed journal. In addition, he is an active reviewer of diverse international JCR-indexed journals.

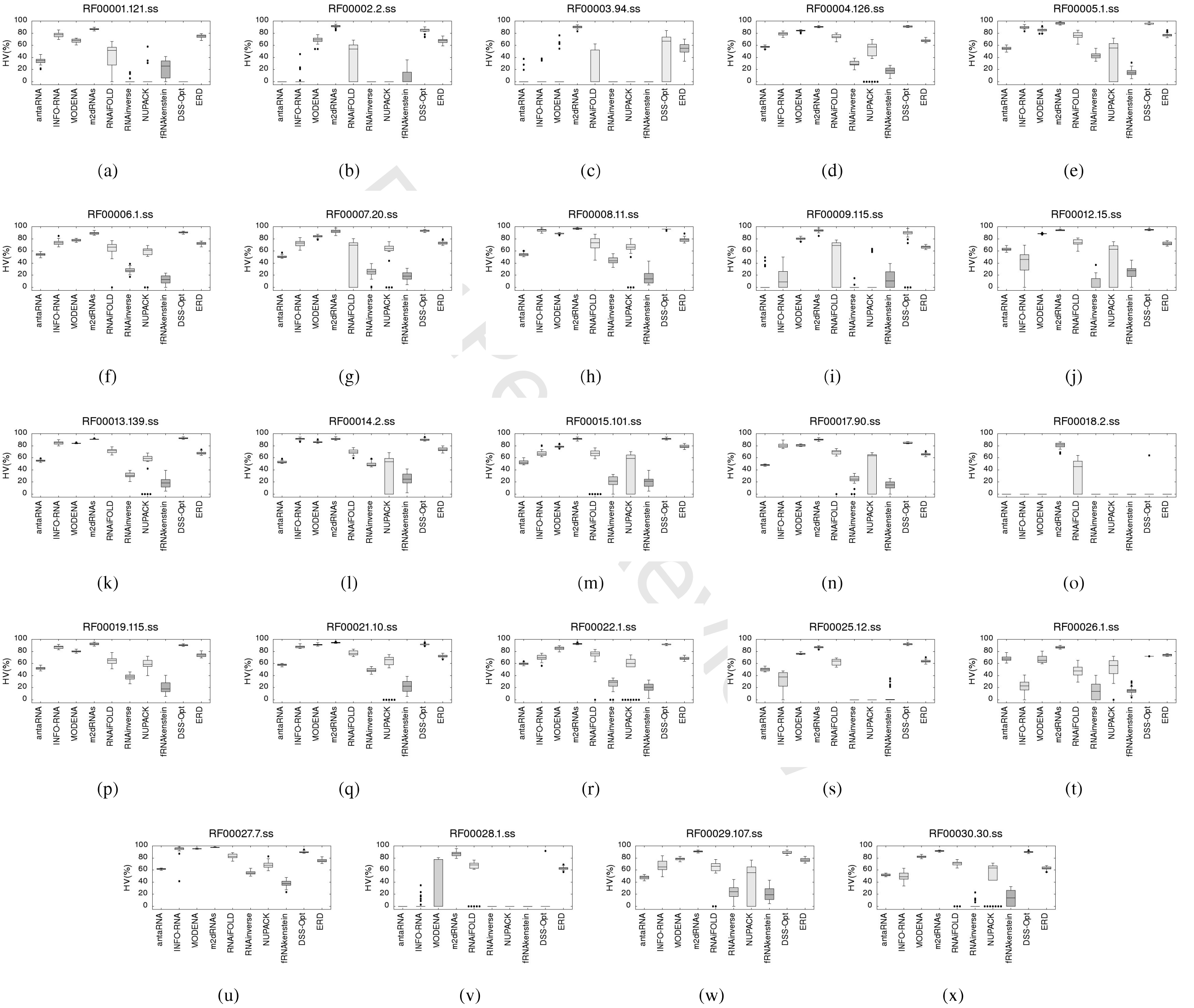
Leonardo Vanneschi took his University degree (Laurea) in Computer Science at the University of Studies of Pisa (Italy) in 1996 (110/110 summa cum Laude) and his PhD in Computer Science at the University of Lausanne (Switzerland) in 2004 (PhD thesis honoured with the Excellence Award of the Science Faculty of the University of Lausanne). He is currently an Associate Professor (Tenure) at NOVA Information Management School (Lisbon, Portugal). His main research interests concern Machine Learning, Complex Systems, Data Mining, and in particular Evolutionary Computation. He has published about 160 scientific contributions, among which 40 have appeared in top-ranked scientific journals, and 10 have been honoured with international awards. He is member of the editorial board of two internationally renown scientific journals. He is member of the steering committee and program committee of various international conferences. He has been the editor of several international conference proceedings and of two scientific journal special issues.

Mauro Castelli obtained his Master degree in Computer Science at the University of Milano Bicocca (Italy) in 2008 ("summa cum Laude"), and his PhD at the University of Milano Bicocca in 2012. He is assistant professor at NOVA IMS, Universidade Nova de Lisboa (Portugal). His main research interests are in the field of Artificial Intelligence (in particular, Evolutionary Computation and Genetic Programming) and in the application of Machine Learning techniques to solve complex real-life problems, especially in the field of biology and medicine.

Miguel A. Vega-Rodríguez received the PhD degree in computer engineering from the University of Extremadura, Spain, in 2003. He is currently Associate Professor (accredited as Full Professor) of computer architecture in the Department of Computer and Communications Technologies, University of Extremadura. He has authored or coauthored more than 580 publications including journal papers (more than 100 JCR-indexed journal papers), book chapters, and peer-reviewed conference proceedings, for which he got several awards - such as ISDA'11 Best Paper Award, IBERGRID'11 Best Paper Award, ICEC'09 Best Paper Award, and IEA-AIE'08 Best Paper Award. He has contributed to the organization of several international conferences and workshops, namely as general chair or co-chair. He has edited 10 special issues of international JCR-indexed journals. In addition, he is an editor and a reviewer of diverse international JCR-indexed journals. His main research interests include parallel and distributed computing, evolutionary computation, reconfigurable and embedded computing, and bioinformatics.

Supplementary Material for “Multiobjective Metaheuristic to Design RNA Sequences”

Álvaro Rubio-Largo, Leonardo Vanneschi, Mauro Castelli, Miguel A. Vega-Rodríguez



RF00001.121.ss (a)

RF00002.2.ss (b)

RF00003.94.ss (c)

RF00004.126.ss (d)

RF00005.1.ss (e)

RF00006.1.ss (f)

RF00007.20.ss (g)

RF00008.11.ss (h)

RF00009.115.ss (i)

RF00012.15.ss (j)

RF00013.139.ss (k)

RF00014.2.ss (l)

RF00015.101.ss (m)

RF00017.90.ss (n)

RF00018.2.ss (o)

RF00019.115.ss (p)

RF00021.10.ss (q)

RF00022.1.ss (r)

RF00025.12.ss (s)

RF00026.1.ss (t)

RF00027.7.ss (u)

RF00028.1.ss (v)

RF00029.107.ss (w)

RF00030.30.ss (x)

Fig. S1. Box plots with the 31 independent runs (in terms of % HV) for each of the ten approaches in Rfam structures. Note that, we have not included the impossible-to-design Rfam structures (RF000153.ss, RF00011.18.ss, RF00016.15.ss, RF00020.107.ss, and RF00024.16.ss).