RAFFT: Efficient prediction of RNA folding pathways using the fast Fourier transform

Vaitea Opuu[a], Nono S. C. Merleau[a], and Matteo Smerlak[a]

[a]Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany

We propose a novel heuristic to predict RNA secondary structures. The algorithm is inspired by the kinetic partitioning mechanism, by which molecules follow alternative folding pathways to their native structure, some much faster than others. Similarly, our algorithm `RAFFT` generates an ensemble of concurrent folding pathways ending in multiple metastable structures for each given sequence; this is in contrast with traditional thermodynamic approaches, which are based on aim to find single structures with minimal free energies. When analyzing 50 predicted folds per sequence, we found near-native predictions (79% PPV and 81% sensitivity) for RNAs of length ≤ 200 nucleotides, matching the performance of recent deep-learning-based structure prediction methods. Our algorithm also acts as a folding kinetic ansatz, which we tested on two RNAs: the coronavirus frameshifting stimulation element (CFSE) and a classic bi-stable sequence. For the CFSE, an ensemble of 68 distinct structures computed by `RAFFT` allowed us to produce complete folding kinetic trajectories, whereas known methods require evaluating millions of sub-optimal structures to achieve this result. For the second application, only 46 distinct structures were required to reproduce the kinetics, whereas known methods required a sample of $20,000$ structures. Thanks to the efficiency of the fast Fourier transform on which `RAFFT` is based, these computations are efficient, with complexity $\mathcal{O}(L^2 \log L)$.

RNA folding | Fourier transform | RNA kinetics | Fast-folding paths

Correspondence: ***vopuu@mis.mpg.de***

DRAFT

Introduction

The function of noncoding RNAs is largely determined by their three-dimensional structure (1). For instance, the catalytic function of ribozymes can often be analyzed in terms of basic structural motifs, e.g. hammerhead or hairpin structures (2). Other RNAs, like riboswitches, involve changes between alternative structures (3). Understanding the relation sequence and structure is therefore a central challenge in molecular biology. Because measuring the structure of RNAs through X-ray crystallography or NMR is difficult and expensive, computational approaches have played a central role in the analysis of natural RNAs (4, 5), and also in the design of synthetic RNAs (6).

Three levels of structures are used to describe RNA molecules: (1) the primary structure, that is, the nucleotide sequence itself; (2) the secondary structure formed by Watson-Crick (or wobble) base pairings; (3) the tertiary structure represents the molecule shape in three-dimensional space. Unlike proteins, RNA structures are usually formed hierarchically: the secondary structure is formed first, followed by the tertiary structure (7). This separation of time scales justifies focusing on the prediction of secondary structures; evidence suggests that the resulting tertiary structures (as well as the kinetic bottlenecks towards their formation) are indeed largely determined by the RNA's secondary structure. Although base pairs can be formed with various configurations (8), we only consider here the canonical edge-to-edge interactions: G-C, A-U, and G-U. Moreover, while various subtleties are involved in the definition of the secondary structure, we use here the formal definition called pseudoknot-free (9). In the rest of this work, 'structure' refers specifically to this notion of RNA secondary structure.

The thermodynamic stability ΔG_s of a structure s is the free energy difference with respect to the completely unfolded state. To predict biologically relevant structures, most computational methods search for structures that minimize this free energy. Computing the free energies, structures are decomposed into components called loops. According to the additivity principle (10), the free energy of a structure can be approximated by the sum of its constituent loops free energies. Many models allow to compute the free energies of those constituent loops, but the dominant one is the nearest-neighbor loop energy model (11). This model associates tabulated free energy values to loop types and nucleotide compositions; the Turner2004 (12) is one of the most widely used parameter sets. This structure decomposition allows an efficient dynamic programming algorithm that can determine the minimum free energy (MFE) structure of a sequence in the entire structure space. The gold standard for free-energy-based predictions is usually the MFE; however, it represents one structural estimate among many others, such as the maximum expected accuracy (MEA).

Several existing tools implement the Zucker dynamic programming algorithm (13), e.g. `RNAfold` (14), `Mfold` (15), or `RNAstructure` (16). While these methods were found to predict RNA structures accurately, as shown in recent benchmarks (17, 18), the additive principle is expected to break down when structures are too large. Moreover, thermodynamic models tend to ignore pseudoknot loops, which can sometimes limit their biological relevance.

Recently, machine learning (ML) approaches were investigated and seemed to overcome some of these shortcomings. ML-based structure prediction tools provide substantial improvements (17, 19). However, in addition to some overfitting concerns (20), these approaches cannot give dynamical information, as few data are available on structural dynamics. In addition, ML methods do not follow from first principles: structural training data are mostly obtained through phylo-

genetic analyses. Consequently, the predictions from those methods may contain biases, e.g. due to *in vivo* third-party elements.

From the dynamical standpoint, the RNA molecule navigates its structure space by following a free energy landscape. Three rate models describing elementary steps in the structure space are currently used to study RNAs folding dynamics: (1) the base stack model uses base stacks formations and breaking as elementary moves (21); (2) the base pair model as implemented in `kinfold` (22) gives the finest resolution with base pair steps, but at the cost of computation time; (3) the stem model (23) provides a coarse-grained description of the dynamics, where free energy changes due to stem formation guide steps. Although none of these models were definitively rejected nor accepted, the latter makes a notable assumption: transition states (or saddle points) involved in the formation of a stem are not considered (24). An alternative approach, implemented in `kinwalker` (25), used the observation that folded intermediates are generally locally optimal conformations.

In folding experiments, Pan and coworkers observed two kinds of pathways in the free energy landscape of a natural ribozyme (26). Firstly, the experiments revealed fast-folding pathways, in which a sub-population of RNAs folded rapidly into the native state. The second population, however, quickly reached metastable misfolded states, then slowly folded into the native structure. In some cases, these metastable states are functional. These phenomena are direct consequences of the rugged nature of the RNA folding landscape (27). The experiments performed by Russell and coworkers also revealed the presence of multiple deep channels separated by high energy barriers on the folding landscape, leading to fast and slow folding pathways (28). The formal description of the above mechanism, called kinetic partitioning mechanism, was first introduced by Guo and Thirumalai in the context of protein folding (29). In the free energy landscape, these metastable conformations form competing attraction basins in which RNA molecules are temporarily trapped. However, *in vivo*, folding into the native states can be promoted by molecular chaperones (30), which means that the active structure depends on factors other than the sequence. This may rise some discrepancy when comparing thermodynamic modeling to real data.

Here, we propose a novel approach to RNA structure prediction and dynamics, inspired by the kinetic partitioning mechanism. Our method has two components: (1) a folding algorithm that models the fast-folding pathways and (2) a kinetic ansatz that models the slow-folding pathways.

The basic idea of the folding algorithm is to construct multiple parallel folding pathways by sequentially forming stems. This procedure yield an ensemble of structures modeling the complete folding process, from the unfolded state to multiple folded states. To build this ensemble, we designed an algorithm called `RAFFT` that uses the fast Fourier transform (FFT) on RNA sequences to quickly find stems, and store multiple folding pathways. The FFT was also used when designing MAFFT (31), a well-known multiple-sequence-

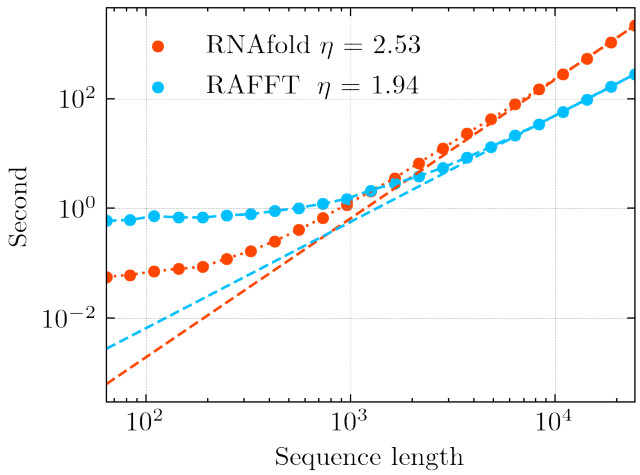

Fig. 1. Execution time comparison between `RAFFT` and `RNAfold`. For a corresponding length, we generated 30 random sequences, and we averaged their execution times. The average execution times obtained are displayed in log-log scale for `RAFFT` and `RNAfold`. The dashed lines show their estimated time complexity $O(L^\eta)$ where η is obtained with a non linear regression on these average time executions.

alignment tool which uses signal processing techniques to analyze nucleotide sequences (32, 33). To assess the quality of the structures predicted, we compared its performance for the folding task on a well-curated dataset, `ArchiveII` (34). The results were compared to two structure estimates: the MFE structure computed with `RNAfold`, and the ML structure computed with `MxFold2` (17). Our folding algorithm achieved good predictive accuracy in the folding task, comparable to ML predictions, while only using physical principles.

The ensemble of structures constructed by `RAFFT` is then fed to a kinetic ansatz, which models the folding process as a continuous-time Markov chain. This technique allows studying dynamical effects such as kinetic traps. To illustrate the relevance of this kinetic ansatz we examined the dynamics of two RNAs: the Coronavirus frameshifting stimulation element (CFSE) (35), and a classic bi-stable sequence. For both RNAs, the ansatz produced results qualitatively similar to the state-of-the-art barrier kinetics (22). However, this procedure requires many fewer structures and models the complete folding process from the unfolded state. For the CFSE, our kinetic modeling revealed that the native structure is found as a kinetic trap, and the MFE structure only appears after a long time.

Material and Methods

Folding algorithm. We now describe the folding algorithm starting from a sequence of nucleotides $S = (S_1 \dots S_L)$ of length L, and its associated unfolded structure. We first create a numerical representation of S where each nucleotide is replaced by a unit vector of 4 components:

$$A \to \begin{pmatrix}1\\0\\0\\0\end{pmatrix}, U \to \begin{pmatrix}0\\0\\0\\1\end{pmatrix}, C \to \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, G \to \begin{pmatrix}0\\0\\1\\0\end{pmatrix}. \quad \textbf{(1)}$$

DRAFT

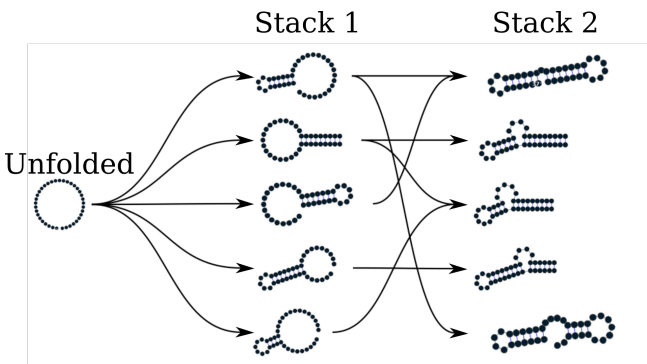

Fig. 2. Fast folding graph constructed using RAFFT. In this example, the sequence is folded in two steps. The algorithm starts with the unfolded structure on the left. The $N = 5$ best stems are stored in stack 1. From stack 1, multiple stems formation are considered, but only the $N = 5$ best are stored in stack 2. Structures are ordered (from top to bottom) by energy in each stack. All secondary structure visualizations were obtained using VARNA (36).

This encoding gives us a $(4 \times L)$-matrix we call X, where each row corresponds to a nucleotide as shown below:

$$X = \begin{pmatrix} X^A \\ X^C \\ X^G \\ X^U \end{pmatrix} = \begin{pmatrix} X^A(1) & X^A(2) & \dots & X^A(L) \\ X^C(1) & X^C(2) & \dots & X^C(L) \\ X^G(1) & X^G(2) & \dots & X^G(L) \\ X^U(1) & X^U(2) & \dots & X^U(L) \end{pmatrix} \quad \textbf{(2)}$$

For example, $X^A(i) = 1$ if $S_i = A$. Next, we create a second copy $\bar{S} = (\bar{S_L} \dots \bar{S_1})$ for which we reversed the sequence order. Then, each nucleotide of $\bar{S}$ is replaced by one of the following unit vectors:

$$\bar{A} \to \begin{pmatrix} 0 \\ 0 \\ 0 \\ w_{\text{AU}} \end{pmatrix}, \bar{U} \to \begin{pmatrix} w_{\text{AU}} \\ w_{\text{GU}} \\ 0 \\ 0 \end{pmatrix}, \bar{C} \to \begin{pmatrix} 0 \\ 0 \\ w_{\text{GC}} \\ 0 \end{pmatrix}, \bar{G} \to \begin{pmatrix} 0 \\ w_{\text{GC}} \\ 0 \\ w_{\text{GU}} \end{pmatrix}. \quad \textbf{(3)}$$

$\bar{A}$ (respectively $\bar{U}, \bar{C}, \bar{G}$) is the complementary of A (respectively U, C, G). w_{AU}, w_{GC}, w_{GU} represent the weights associated with each canonical base pair, and they are chosen empirically. We call this complementary copy $\bar{X}$, the mirror of X.

To search for stems, we use the complementary relation between X and $\bar{X}$ with the correlation function $\text{cor}(k)$. This correlation is defined as the sum of individual X and $\bar{X}$ row correlations:

$$\text{cor}(k) = \sum_{\alpha \in \{A,U,C,G\}} c_{X^\alpha, \bar{X}^\alpha}(k), \quad \textbf{(4)}$$

where a row correlation between X and $\bar{X}$ is given by:

$$c_{X^\alpha, \bar{X}^\alpha}(k) = \sum_{\substack{1 \le i \le L \\ 1 \le i+k \le L}} \frac{X^\alpha(i) \bar{X}^\alpha(i+k)}{\min(k, 2L-k)}. \quad \textbf{(5)}$$

For each $\alpha \in \{A,U,C,G\}$, $X^\alpha(i) \times \bar{X}^\alpha(i+k)$ is non zero if sites i and $i+k$ can form a base pair, and will have the value of the chosen weight as described above. If all the weights are set to 1, $\text{cor}(k)$ gives the frequency of base pairs for a positional lag k. Although the correlation naively requires $O(L^2)$ operations, it can take advantage of the FFT which reduces its complexity to $\mathcal{O}(L \log(L))$.

Large $\text{cor}(k)$ values between the two copies indicate positional lags k where the frequency of base pairs is likely to be high. However, this does not allow to determine the exact stem positions. Hence, we use a sliding window strategy to search for the largest stem within the positional lag (since the copies are symmetrical, we only need to slide over one-half of the positional lag). Once the largest stem is identified, we compute the free energy change associated with the formation of that stem. Next, we perform the same search for the n highest correlation values, which gives us n potential stems. Then, we define as the current structure the stem with the lowest free energy. Here, free energies were computed using Turner2004 energy parameters through ViennaRNA package API (37).

We are now left with two independent parts, the interior and the exterior of the newly formed stem. If the exterior part is composed of two fragments, they are concatenated into one. Then, we apply recursively the same procedure on the two parts independently in a "Breadth-First" fashion to form new consecutive base pairs. The procedure stops when no base pair formation can improve the energy. When multiple stems can be formed in these independent fragments, we combine all of them and pick the composition with the best overall stability. If too many compositions can be formed, we restrict this to the 10^4 bests in terms of energy. Figure 3 shows an example of execution to illustrate the procedure.

The complexity of this algorithm depends on the number and size of the stems formed. The main operations performed for each stem formed are: (1) the evaluation of the correlation function $\text{cor}(k)$, (2) the sliding-window search for stems, and (3) the energy evaluation. We based our approximate complexity on the correlation evaluation since it is the more computationally demanding step; the other operations only contribute a multiplicative constant at most. The best case is the trivial structure composed of one large stem where the algorithm stops after evaluating the correlation on the complete sequence. At the other extreme, the worst case is one where at most $L/2$ stems of size 1 (exactly one base pair peer stems) can be formed. The approximate complexity therefore depends on $\sum_{i=0}^{L/2}(L-2i)\log(L-2i) = \mathcal{O}(L^2 \log L)$. Figure 1 plots the execution time of a naive implementation of RAFFT and that of RNAfold for 20 random sequences of various lengths, showing a substantial speed-up for larger sequences.

The algorithm described so far tends to be stuck in the first local minima found along the folding trajectory. To alleviate this, we implemented a stacking procedure where the N best trajectories are stored in a stack and evolved in parallel. Figure 2 illustrates this modified procedure. Like the initial version, the algorithm starts with the unfolded structure; then, the $N = 5$ best potential stems are stored in the first stack. From these N structures, the procedure tries to add stems in the unpaired regions left and saves the N best structures

DRAFT

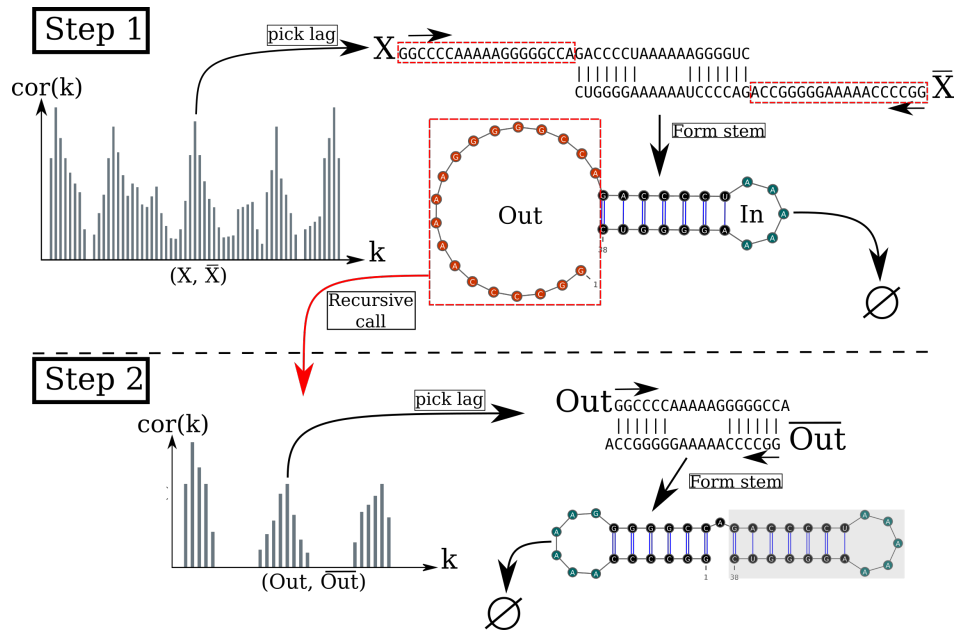

Fig. 3. Algorithm execution for one example sequence which requires two steps. (Step 1) From the correlation $cor(k)$, we select one peak which corresponds to a position lag k. Then, we search for the largest stem and form it. Two fragments, "In" (the interior part of the stem) and "Out" (the exterior part of the stem), are left, but only the "Out" may contain a new stem to add. (Step 2) The procedure is called recursively on the "Out" sequence fragment only. The correlation $cor(k)$ between the "Out" fragment and its mirror is then computed and analyzing the k positional lags allows to form a new stem. Finally, no more stem can be formed on the fragment left (colored in blue), so the procedure stops.

formed. Once no stem can be formed, the algorithm stops and output the structure with the best energy found among the structures stored in the last stack. This algorithm leads to the construction of a graph we call a *fast-folding graph*. In this graph, two structures are connected if the transition from one to another corresponds to the formation of a stem or if the two structures are identical.

Kinetic ansatz. The folding kinetic ansatz used here is derived from the fast-folding graph and allows us to model the slow processes in RNA folding. As described in Figure 2, transitions can occur from left to right (and right to left) but not vertically. The fast-folding graph follows the idea that parallel pathways quickly reach their endpoints; however, when the endpoints are non-native states, this ansatz allows slowly folding back into the native state (26).
As usually done, the kinetics is modelled as a continuous-time Markov chain (38), where populations of structure evolve according to transition rates. In this context, an Arrhenius formulation is commonly used to derive transition rates $r(x \to y) \propto \exp(-\beta E^{\ddagger})$, where $E^{\ddagger}$ is the activation energy separating x from y, and β is the inverse thermal energy (mol/kcal). In contrast, our kinetic ansatz uses transition rates $r(x \to y)$ based on the Metropolis scheme already used in (39), and defined as

$$r(x \to y) = k_0 \times \min(1, \exp(-\beta \Delta\Delta G(x \to y))), \qquad \textbf{(6)}$$

where $\Delta\Delta G(x \to y)$ is the stability change between structure x and y. Here k_0 is a conversion constant that we set to 1 for the sake of simplicity. These transitions are only allowed if y is connected to x in the graph (i.e. y is in the neighborhood of x, $y \in \mathcal{X}$). Here, we initialize the population $p_x(0)$ with only unfolded structures; therefore, the trajectory represents a complete folding process. The frequency of a structure x evolves according to the master equation

$$\frac{\mathrm{d}p_x(t)}{\mathrm{d}t} = \sum_{y \in \mathcal{X}} r(y \to x) p_y(t) - r(x \to y) p_x(t), \qquad \textbf{(7)}$$

where the sum runs over the neighborhood $\mathcal{X}$ of x.
The traditional kinetic approach starts by enumerating the whole space (or a carefully chosen subspace) of structures using `RNAsubopt`. Next, this ensemble is divided into local attraction basins separated from one another by energy barriers. This coarsening is usually done with the tool `barriers`. Then, following the Arrhenius formulation, one simulates a coarse grained kinetics between basins. In contrast, the Metropolis scheme used in our kinetic ansatz is based on the stability difference between structures, which may hide energy barriers. Due to this approximation, we referred to our approach as a 'kinetic ansatz'.

Benchmark dataset. To build the dataset for the folding task application, we started from the `ArchiveII` dataset derived from multiple sources (40–56). We first removed all the structures with pseudoknots, since the tools considered here do not handle these loops. Next, using the Turner2004 energy parameters, we evaluated the structures' energies and removed all the unstable structures: structures with energies $\Delta G_s > 0$. This dataset is composed of $2,698$ sequences with their corresponding known structures. 240 sequences were found multiple times (from 2 to 8 times); 19 of them were mapped to different structures. For the sequences that appeared with different structures, we picked the structure with the lowest energy. In the end we arrived at a dataset with

DRAFT

Table 1. Average performance displayed in terms of PPV and sensitivity. The metrics were first averaged at fixed sequence length, limiting the over-representation of shorter sequences. The first two rows show the average performance for all the sequences for each method. The bottom two rows correspond to the performances for the sequences of length ≤ 200 nucleotides. For the ML and MFE only one prediction per sequence and for RAFFT 50 predictions per sequence were used. Here RAFFT (respectively RAFFT*) refers to the case when the lowest free energy (resp. highest PPV) out of the 50 predictions is selected.

	RAFFT	RAFFT*	MFE	ML
	All sequences			
PPV	47.7	60.0	55.9	70.4
Sensitivity	52.8	62.8	63.3	77.1
	Sequences with lengths ≤ 200			
PPV	57.9	79.4	59.5	76.7
Sensitivity	63.2	81.2	65.5	82.9

$2,296$ sequences-structures.

Structure prediction protocols for benchmarks. To evaluate the structure prediction accuracy of the proposed method, we compared it to two structure estimates: the MFE structure and the ML structure. To compute the MFE structure, we used `RNAfold 2.4.13` with the default parameters and the Turner2004 set of energy parameters. We computed the prediction using `Mxfold2 0.1.1` with the default parameters for the ML structure. Therefore, only one structure prediction per sequence for those two methods was used for the statistics.

Two parameters are critical for RAFFT, the number of positional lags in which stems are searched, and the number of structures stored in the stack. For our computational experiments, we searched for stems in the $n = 100$ best positional lags and stored $N = 50$ structures. The correlation function $\mathrm{cor}(k)$ which allows to choose the positional lags is computed using the weights $w_{GC} = 3$, $w_{AU} = 2$, and $w_{GU} = 1$.

To assess the performance of RAFFT, we analyzed the output in two different ways. First, we considered only the structure with the lowest energy found for each sequence. This procedure allows us to assess RAFFT performance in search of low energy structure only. Second, we computed the accuracy of all $N = 50$ structures saved in the last stack for each sequence and displayed only the best structure in terms of accuracy. As mentioned above, the lowest energy structure found may not be the active structure. Therefore, this second assessment procedure allows us to show whether one of the pathways is biologically relevant.

We used two metrics to measure the prediction accuracy: the positive predictive value (PPV) and the sensitivity. The PPV measures the fraction of correct base pairs in the predicted structure, while the sensitivity measure the fraction of base pairs in the accepted structure that are predicted. These metrics are defined as follows:

$$PPV = \frac{TP}{TP+FP}, \quad \text{Sensitivity} = \frac{TP}{TP+FN}, \qquad \textbf{(8)}$$

where TP, FN, and FP stand respectively for the number of correctly predicted base pairs (true positives), the number of base pairs not detected (false negatives), and the number of wrongly predicted base pairs (false positives). To be consistent with previous studies, we computed these metrics using the `scorer` tool provided by Matthews *et al.* (34), which also provides a more flexible estimate where shifts are allowed.

Structure space visualization. We used a Principal Component Analysis (PCA) to visualize the loop diversity in the datasets considered here. To extract the weights associated with each structure loop from the dataset, we first converted the structures into weighted coarse-grained tree representation (57). In the tree representation, the nodes are generally labelled as E (exterior loop), I (interior loop), H (hairpin), B (bulge), S (stacks or stem-loop), M (multi-loop) and R (root node). We separately extracted the corresponding weights for each node, and the weights are summed up and then normalized. Excluding the root node, we obtained a table of 6 features and n entries. This allows us to compute a 6×6 correlation matrix that we diagonalize using the `eigen` routine implemented in the `scipy` package. For visual convenience, the structure compositions were projected onto the first two Principal Components (PC). Figures 4C and 4D show the two principal components of the benchmark dataset, the predicted structure using RAFFT, `RNAfold` (MFE-structure) and `MxFold2` (ML-structure), where the arrows represent the direction of each feature in the PC space.

Results

Application to the folding task. We started by analyzing the prediction performances with respect to sequence lengths: we averaged the performances at fixed sequence length. Figure 4A shows the performance in predicted positive values (PPV) and sensitivity for the three methods. It shows that the ML method consistently outperformed RAFFT and MFE predictions. The t-test between the ML and the MFE predictions revealed not only a significant difference (p-value $\approx 10^{-12}$) but also a substantial improvement of 14.5% in PPV. RAFFT showed performances similar to the MFE predictions; however, RAFFT is significantly less accurate (p-value ≈ 0.0002), with a drastic loss of performance for sequences of length greater than 300 nucleotides. In contrast, when only the most accurate predicted structure among the 50 recorded structures per sequence was considered, we obtained 57.9% of PPV and 63.2% of sensitivity on average. The gain of performance was even more substantial for sequences of length below 200 nucleotides. The PPV was 79.4%, and the sensitivity was 81.2%. In contrast, longer sequences did not dis-

DRAFT

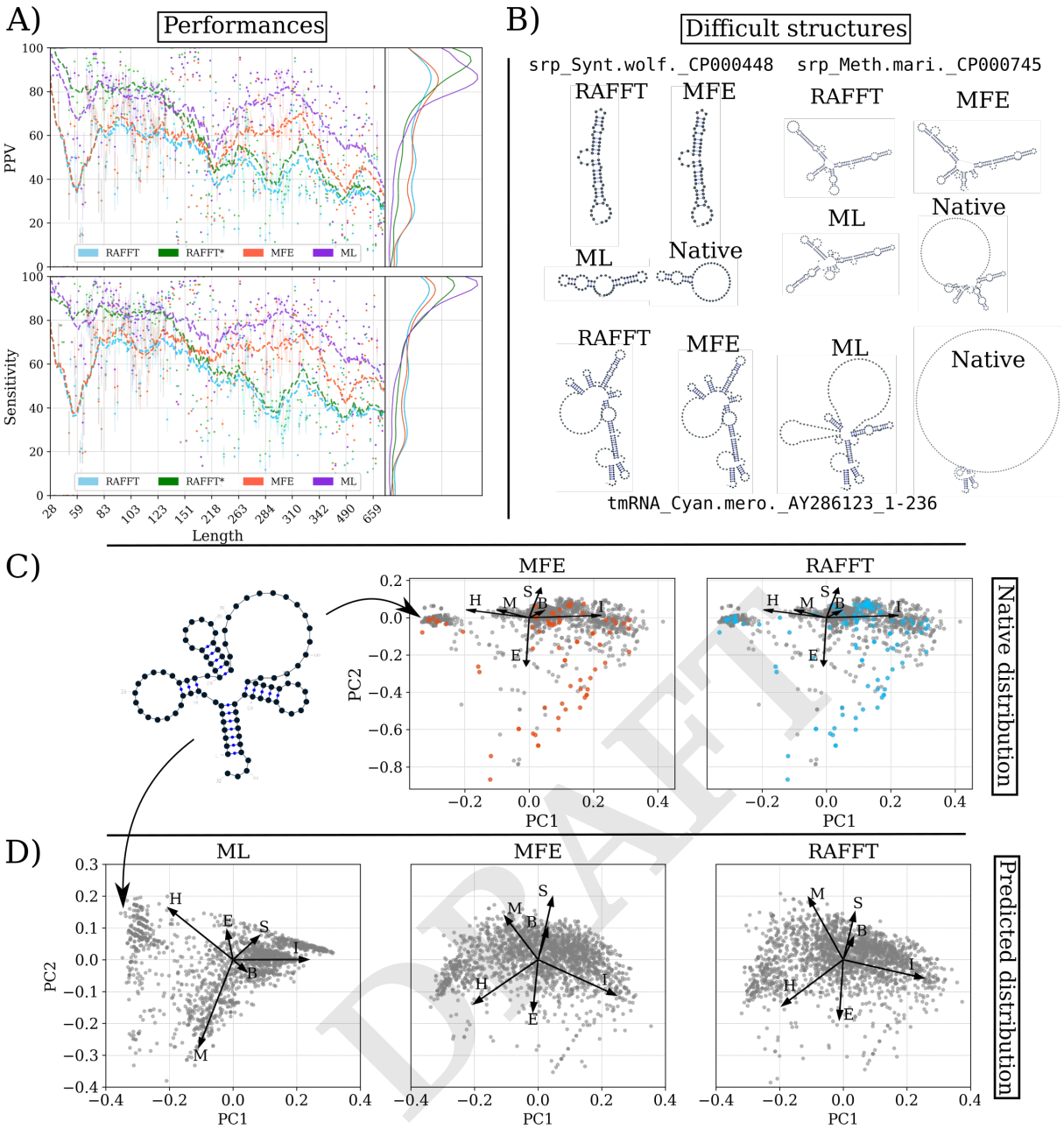

Fig. 4. RAFFT's performance on folding task. (A) PPV and sensitivity *vs* sequence length. In the left panels, RAFFT (in blue) shows the scores when for the structure (out of $N = 50$ predictions) with the lowest free energy, whereas RAFFT* (in green) shows the best PPV score in that ensemble. Each dot corresponds to the mean performance for a given sequence length, and vertical lines display their standard deviation. The right panels of both figures show the distribution of PPV and sensitivity sequence-wise. (B) List of structures that are challenging to predict using the thermodynamic model. The sequence names are from the dataset. The native structures have large regions with unpaired nucleotides. (C) PCA for structures in the dataset. An example of structures with a large hairpin (**H**) is shown on the left. The points marked in orange (resp. blue) are the MFE (resp. RAFFT) structures with PPV $\leq 10\%$. (D) PCA for the predicted structures. The MFE and RAFFT structure spaces look similar and more diverse than the ML structure space, closer to the native structure space.

play any gain. The average performances are shown in table 1. We also investigated the relation to the number of bases between paired bases (base pair spanning), but we found no striking effect, as already pointed out in one previous study (58).

All methods performed poorly on two groups of sequences: one group of 80 nucleotides long RNAs, and the second group of around 200 nucleotides. Figure 4B shows three examples of such sequences. Both groups have large unpaired regions, which for the first group lead to structures with average free energies 9.8 kcal/mol according to our dataset. The PCA analysis of the native structure space, shown in Figure 4C, reveals a propensity for interior loops and the presence of large unpaired regions like hairpins or external loops. Figure 4D shows the structure space produced by the ML predictions, which seems close to the native structure space. In contrast, the structure spaces produced by RAFFT and RNAfold (MFE) are similar and more diverse.

Selected applications of the kinetic ansatz. We started with the CFSE, a natural RNA sequence of 82 nucleotides

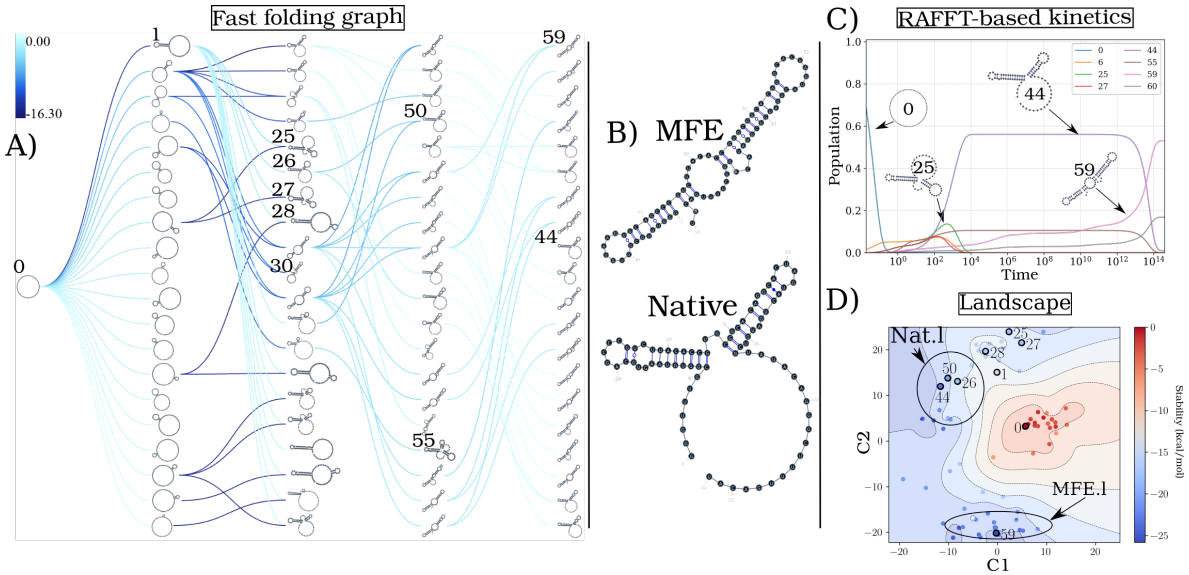

Fig. 5. Application of the folding kinetic ansatz on CFSE. (A) Fast-folding graph in four steps and $N = 20$ structures stored in a stack at each step. The edges are coloured according to $\Delta\Delta G$. At each step, the structures are ordered by their free energy from top to bottom. The minimum free energy structure found is at the top left of the graph. A unique ID annotates visited structures in the kinetics. For example, "59" is the ID of the MFE structure. (B) MFE (computed with `RNAfold`) and the native CFSE structure. (C)The change in structure frequencies over time. The simulation starts with the whole population in the open-chain or unfolded structure (ID 0). The native structure (**Nat.I**) is trapped for a long time before the MFE structure (**MFE.I**) takes over the population. (D) Folding landscape derived from the 68 distinct structures predicted using `RAFFT`. The axes are the components optimized by the MDS algorithm, so the base pair distances are mostly preserved. Observed structures are also annotated using the unique ID. MFE-like structures (**MFE.I**) are at the bottom of the figure, while native-like (**Nat.I**) are at the top.

with a structure determined by sequence analysis and obtained from the RFAM database. This structure has a pseudoknot which is not taken into account here.

Figures 5A and 5B show respectively the fast-folding graph constructed using `RAFFT`, and the MFE and native structures for the CFSE. The fast-folding graph is computed in four steps. At each step, stems are constructed by searching for $n = 100$ positional lags and, a set of $N = 20$ structures (selected according to their free energies) are stored in a stack. The resulting fast-folding graph consists of 68 distinct structures, each of which is labelled by a number. Among the structures in the graph, 6 were found similar to the native structure (16/19 base pairs differences). The structure labelled "29" in the graph leading to the MFE structure "59" is the 9th in the second stack. When storing less than 9 structures in the stack at each step, we cannot obtain the MFE structure using `RAFFT`; this is a direct consequence of the greediness of the proposed method. To visualize the energy landscape drawn by `RAFFT`, we arranged the structures in the fast-folding graph onto a surface according to their base-pair distances; for this we used the multidimensional scaling algorithm implemented in the `scipy` package. Figure 5D shows the landscape interpolated with all the structures found; this landscape illustrates the bi-stability of the CFSE, where the native and MFE structures are in distinct regions of the structure space.

From the fast-folding graph produced using `RAFFT`, the transition rates from one structure in the graph to another are computed using the formula given in Eq 6. Starting from a population of unfolded structure and using the computed transition rates, the native of structures is calculated using Eq 7. Figure 5C shows the frequency of each structure; as the frequency of the unfolded structure decreases to 0, the frequency of other structures increases. Gradually, the structure labelled "44", which represents the CFSE native structure, takes over the population and gets trapped for a long time, before the MFE structure (labelled "59") eventually becomes dominant. Even though the fast-folding graph does not allow computing energy landscape properties (saddle, basin, etc.), the kinetics built on it reveals a high barrier separating the two meta-stable structures (MFE and native).

Our kinetic simulation was then compared to `Treekin` (59). First, we generated 1.5×10^6 sub-optimal structures up to 15 kcal/mol above the MFE structure using `RNAsubopt` (37). Since the MFE is $\Delta G_s = -25.8$ kcal/mol, the unfolded structure could not be sampled. Second, the ensemble of structures is coarse-grained into 40 competing basins using the tool `barriers` (59), with the connectivity between basins represented as a barrier tree (see Figure 6A). When using `Treekin`, the choice of the initial population is not straightforward. Therefore we resorted to two initial structures I_1 and I_2 (see Figure 6B and 6C, respectively). In Figure 6B, the trajectories show that only the kinetics initialized in the structure I_2 can capture the complete folding dynamics of CFSE, in which the two metastable structures are visible. Thus, in order to produce a folding kinetics in which the native and the MFE structures are visible, the kinetic simulation performed using `Treekin` required a particular initial condition and a barrier tree representation of the energy landscape built from a set of 1.5×10^6 structures. By contrast, using the fast-folding graph produced by `RAFFT`, which consists only of 68 distinct structures, our kinetic simulation produces complete folding dynamics starting from a population of unfolded structure.

As a second illustrative example, we applied both kinetic models to the classic bi-stable sequence

DRAFT

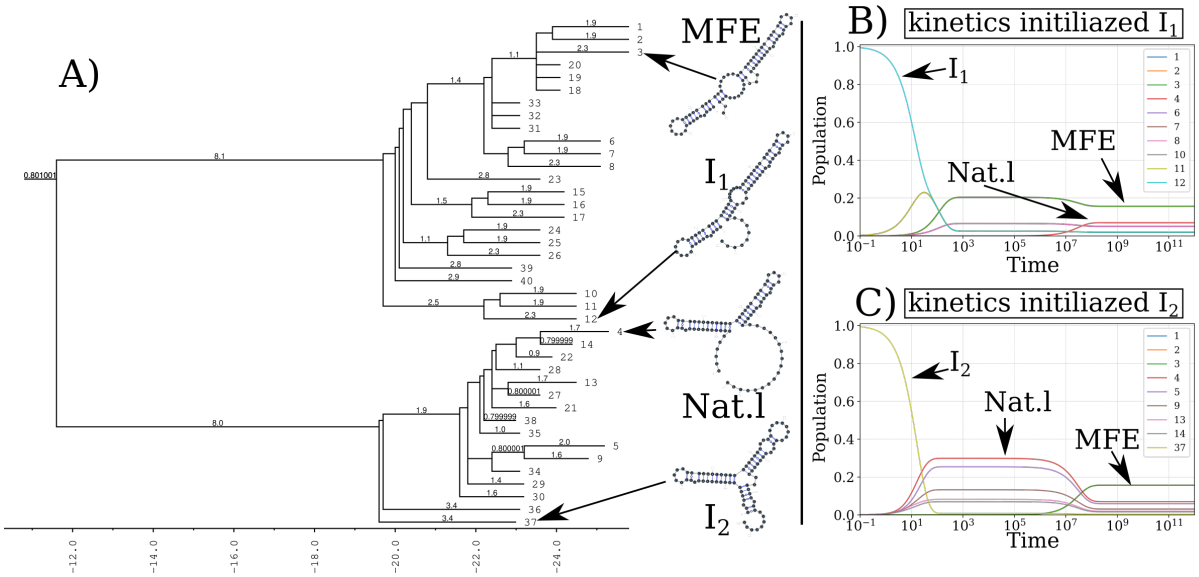

Fig. 6. Folding kinetics of CFSE using `Treekin`. A) Barrier tree of the CFSE. From a set of 1.5×10^6 sub-optimal structures, 40 local minima were found, connected through saddle points. The tree shows two alternative structures separated by a high barrier with the global minimum (MFE structure) on the right side. (B) Folding kinetics with initial population I_1. Starting from an initial population of I_1, as the initial frequency decreases, the others increase, and gradually the MFE structure is the only one populated. (C) Folding kinetics with initial population I_2. When starting with a population of I_2, the native structure (labelled **Nat.1**) is observable, and gets kinetically trapped for a long time due to the high energy barrier separating it from the MFE structure.

GGCCCCUUUGGGGGCCAGACCCCUAAAGGGGUC. For `Treekin`, we first sampled the whole space of 20×10^3 sub-optimal structures from the unfolded state to the MFE structure, and from that set, 40 basins were also computed using `barriers`. The barrier tree in Figure 7 shows the bi-stable landscape, where the two deepest minima are denoted S_A and S_B. As in the first application, we also chose two initializations with the structures denoted I_1 and I_2 in Figure 7A and 7B. Secondly, we simulate the kinetics starting from the two initial conditions (See Figure 7B). When starting from I_2, the slow-folding dynamics is visible: S_B first gets kinetically trapped, and the MFE structure (S_A) only takes over later on. For our kinetic ansatz, we started by constructing the fast-folding graph using `RAFFT`, consisting of only 46 distinct structures. The resulting kinetics, shown in Figure 7B' was found qualitatively close to the barrier kinetics initialized with structure I_2. Once again, with few as 48 structures, our proposed kinetic ansatz can produce complete folding dynamics starting from a population of unfolded structure.

Discussion

We have proposed a method for RNA structure and dynamics predictions called `RAFFT`. Our method was inspired by the experimental observation of parallel fast-folding pathways. We designed an algorithm that produces parallel folding pathways, in which stems are formed sequentially, to mimic this observation. Then, to model the slow part of the folding process, we proposed a kinetic ansatz that exploits the parallel fast-folding pathways predicted.

First, we compared the algorithm performance for the folding task. Two structure estimates were compared with our method: the MFE structure computed using `RNAfold`, and the ML estimate using `MxFold2`. Other thermodynamic-based and ML-based tools were investigated but not shown here because their performances were found to be very similar to the one of `MxFold2` and `RNAfold` (See SI for the complete benchmark). When we considered the lowest energy structure, the comparison of `RAFFT` to existing tools confirmed the overall validity of our approach. In more detail, comparison with thermodynamic/ML models yielded the following results. First, the ML predictions performed consistently better than both `RAFFT` and the MFE approaches, where the PPV $= 70.4\%$ and sensitivity $= 77.1\%$ on average. Second, the ML methods produced loops, such as long hairpins or external loops. We argue that the density of those loops correlate with the ones in the benchmark dataset, which a PCA analysis revealed too. In contrast, the density of loops was lower in the structure spaces produced by `RAFFT` and MFE, implying some over-fitting in the ML model. Finally, known structures obtained through covariation analysis reflect structures *in vivo* conditions. Therefore, the structures predicted by ML methods may not only result from their sequences alone but also from their molecular environment, e.g. chaperones. We expect the thermodynamic methods to provide a more robust framework for the study of sequence-to-structure relations.

With respect to MFE-based tools, we obtained a substantial gain of performance when analyzing $N = 50$ predicted structures per sequence and not only the lowest energy one. This gain was even more remarkable for sequences with fewer than 200 nucleotides, reaching the accuracy of ML predictions. So how does `RAFFT` produce better structures, although these structures are less thermodynamically stable? The interplay of three effects may explain this finding. First, the MFE structure may not be relevant because active structures can be in kinetic traps. Second, `RAFFT` forms a set of pathways that cover the free energy landscape until they

DRAFT

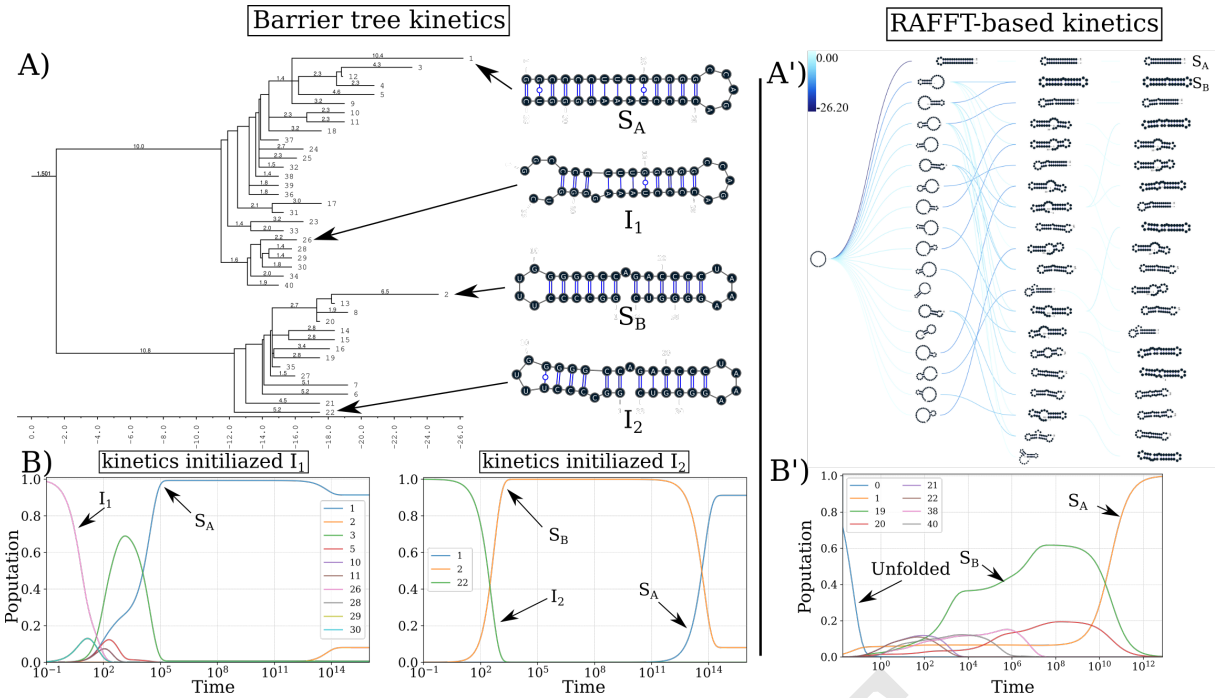

Fig. 7. `RAFFT` *vs* `Treekin`: folding kinetics of a bi-stable RNA sequence. (A) Barrier tree for the bi-stable example sequence. The local minima and the corresponding barriers are computed from the complete enumeration of the structure space. The bi-stability is visible on the barrier tree through the two branches separated by a high barrier. (B) Folding kinetics trajectories. The left plot shows the folding dynamics starting from a population with I_1, and the right size is the kinetics when the population is initialized in structure I_2. When starting from I_1, S_A is quickly populated; starting from I_2, the bi-stability is more apparent. (A') Fast-folding graph using `RAFFT`. A maximum of $N = 20$ structures are stored in a stack at each step and overall 46 distinct structures are visited. (B') Folding kinetics trajectory obtained from the fast-folding graph (indices are different from the barrier tree indices). The dynamics starts with a population with only unfolded structure, and slowly, S_B is populated and gets trapped for a long time before the MFE structure S_A becomes populated.

reach local minima, yielding multiple long-lived structures accessible from the unfolded state. Third, the energy function is not perfect, so that the MFE structures computed by minimizing it may not in fact be the most stable.

We also showed that the fast-folding graph produced by `RAFFT` can be used to reproduce state-of-the-art kinetics, at least qualitatively. Our method demonstrated three main benefits. First, the kinetics can be drawn from as few as 68 structures, whereas the barrier tree may require millions. Second, the kinetics ansatz describes the complete folding mechanism starting from the unfolded state. Third, for the length range tested here, the procedure did not require any additional coarse-graining into basins. (Longer RNAs might require such a coarse-graining step, in which structures connected in the fast-folding graph are merged together).

Based on our results, we believe that the proposed method is a robust heuristic for structure prediction and folding dynamics. The folding landscape depicted by `RAFFT` was designed to follow the kinetic partitioning mechanism, where multiple folding pathways span the folding landscape. This approach has shown good predictive potential. Furthermore, we derived a kinetic ansatz from the fast-folding graph to model the slow part of the folding dynamics. It was shown to approximate the usual kinetics framework qualitatively, albeit using many fewer structures.

However, further improvements and extensions of the algorithm may be investigated. For starters, the choice of stems is limited to the largest in each positional lag, a greedy choice which may not be optimal. Furthermore, we have constructed parallel pathways leading to a diversity of accessible structures, but we have not given any thermodynamic-based criterion to identify which are more likely to resemble the native structure. We suggest using an ML-optimized score to this effect. Our method can also find applications in RNA design, where the design procedure could start with the identification of long-lived intermediates and use them as target structures. We also believe that mirror encoding can be helpful in phylogenetic analysis. Indeed, the correlation spectra $\mathrm{cor}(k)$ computed here contained global information of base-pairing that can be used as a similarity measure.

DATA AVAILABILITY

The implementation in `python3.0` of `RAFFT` and the benchmark data used in this manuscript are available at `https://github.com/strevol-mpi-mis/RAFFT`. We also provide the scripts used for the figures and kinetic analyses.

ACKNOWLEDGEMENTS
We thank Onofrio Mazzarisi for helpful discussions and Peter F. Stadler for insightful comments. Funding for this work was provided by the Alexander von Humboldt Foundation in the framework of the Sofja Kovalevskaja Award endowed by the German Federal Ministry of Education and Research, and by the Human Science Frontier Program Organization through a Young Investigator Award.

Bibliography

1. Thomas R Cech and Joan A Steitz. The noncoding RNA revolution—trashing old rules to forge new ones. *Cell*, 157(1):77–94, 2014.
2. Elizabeth A Doherty and Jennifer A Doudna. Ribozyme structures and mechanisms. *Annual Review of Biophysics and Biomolecular Structure*, 30(1):457–475, 2001.

DRAFT

3. Alexey G Vitreschak, Dimitry A Rodionov, Andrey A Mironov, and Mikhail S Gelfand. Riboswitches: the oldest mechanism for the regulation of gene expression? *Trends in Genetics*, 20(1):44–50, 2004.
4. Matthew G Seetin and David H Mathews. RNA structure prediction: an overview of methods. *Bacterial Regulatory RNA*, pages 99–122, 2012.
5. Jörg Fallmann, Sebastian Will, Jan Engelhardt, Björn Grüning, Rolf Backofen, and Peter F Stadler. Recent advances in RNA folding. *Journal of Biotechnology*, 261:97–104, 2017.
6. Gesine Domin, Sven Findeiß, Manja Wachsmuth, Sebastian Will, Peter F Stadler, and Mario Mörl. Applicability of a computational design approach for synthetic riboswitches. *Nucleic Acids Research*, 45(7):4108–4119, 2017.
7. Ignacio Tinoco Jr and Carlos Bustamante. How RNA folds. *Journal of Molecular Biology*, 293(2):271–281, 1999.
8. Neocles B Leontis and Eric Westhof. Geometric nomenclature and classification of RNA base pairs. *RNA*, 7(4):499–512, 2001.
9. Michael S Waterman. Secondary structure of single-stranded nucleic acids. *Adv. math. suppl. studies*, 1:167–212, 1978.
10. Ken A Dill. Additivity principles in biochemistry. *Journal of Biological Chemistry*, 272(2): 701–704, 1997.
11. Douglas H Turner and David H Mathews. NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Research*, 38 (suppl_1):D280–D282, 2010.
12. David H Mathews, Matthew D Disney, Jessica L Childs, Susan J Schroeder, Michael Zuker, and Douglas H Turner. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proceedings of the National Academy of Sciences*, 101(19):7287–7292, 2004.
13. Michael Zuker and Patrick Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9(1):133–148, 1981. doi: 10.1093/nar/9.1.133.
14. I. L. Hofacker. Vienna RNA secondary structure server. *Nucleic Acids Research*, 31(13): 3429–3431, 2003. doi: 10.1093/nar/gkg599.
15. M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Research*, 31(13):3406–3415, 2003. doi: 10.1093/nar/gkg595.
16. Jessica S Reuter and David H Mathews. RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics*, 11(1):129, 2010. doi: 10.1186/1471-2105-11-129.
17. Kengo Sato, Manato Akiyama, and Yasubumi Sakakibara. RNA secondary structure prediction using deep learning with thermodynamic integration, 2021.
18. Liang Huang, He Zhang, Dezhong Deng, Kai Zhao, Kaibo Liu, David A Hendrix, and David H Mathews. Linearfold: linear-time approximate RNA folding by 5'-to-3'dynamic programming and beam search. *Bioinformatics*, 35(14):i295–i304, 2019.
19. Jaswinder Singh, Jack Hanson, Kuldip Paliwal, and Yaoqi Zhou. RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nature Communications*, 10(1):1–13, 2019.
20. Elena Rivas, Raymond Lang, and Sean R Eddy. A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more. *RNA*, 18(2):193–212, 2012.
21. Wenbing Zhang and Shi-Jie Chen. RNA hairpin-folding kinetics. *Proceedings of the National Academy of Sciences*, 99(4):1931–1936, 2002.
22. Christoph Flamm, Walter Fontana, Ivo L Hofacker, and Peter Schuster. RNA folding at elementary step resolution. *RNA*, 6(3):325–338, 2000.
23. Hugo M. Martinez. An RNA folding rule. *Nucleic Acids Research*, 12(1):323–334, 1984. doi: 10.1093/nar/12.1part1.323.
24. Wenbing Zhang and Shi-Jie Chen. Exploring the complex folding kinetics of RNA hairpins: I. general folding kinetics analysis. *Biophysical Journal*, 90(3):765–777, 2006.
25. Michael Geis, Christoph Flamm, Michael T Wolfinger, Andrea Tanzer, Ivo L Hofacker, Martin Middendorf, Christian Mandl, Peter F Stadler, and Caroline Thurner. Folding kinetics of large RNAs. *Journal of Molecular Biology*, 379(1):160–173, 2008.
26. Jie Pan, D. Thirumalai, and Sarah A. Woodson. Folding of RNA involves parallel pathways. *Journal of Molecular Biology*, 273(1):7–13, 1997. doi: 10.1006/jmbi.1997.1311.
27. Sergey V. Solomatin, Max Greenfeld, Steven Chu, and Daniel Herschlag. Multiple native states reveal persistent ruggedness of an RNA folding landscape. *Nature*, 463(7281):681–684, 2010. doi: 10.1038/nature08717.
28. Rick Russell, Xiaowei Zhuang, Hazen P Babcock, Ian S Millett, Sebastian Doniach, Steven Chu, and Daniel Herschlag. Exploring the folding landscape of a structured RNA. *Proceedings of the National Academy of Sciences*, 99(1):155–160, 2002.
29. Zhuyan Guo and D Thirumalai. Kinetics of protein folding: nucleation mechanism, time scales, and pathways. *Biopolymers: Original Research on Biomolecules*, 36(1):83–102, 1995.
30. Shaon Chakrabarti, Changbong Hyeon, Xiang Ye, George H Lorimer, and D Thirumalai. Molecular chaperones maximize the native state yield on biological times by driving substrates out of equilibrium. *Proceedings of the National Academy of Sciences*, 114(51): E10919–E10927, 2017.
31. Kazutaka Katoh, Kazuharu Misawa, Kei-ichi Kuma, and Takashi Miyata. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14):3059–3066, 2002.
32. Joseph Felsenstein, Stanley Sawyer, and Rochelle Kochin. An efficient method for matching nucleic acid sequences. *Nucleic Acids Research*, 10(1):133–139, 1982.
33. Donald C. Benson. Fourier methods for biosequence analysis. *Nucleic Acids Research*, 18 (21):6305–6310, 1990. doi: 10.1093/nar/18.21.6305.
34. David H. Mathews. How to benchmark RNA secondary structure prediction accuracy. *Methods*, 162-163(162):60–67, 2019. doi: 10.1016/j.ymeth.2019.04.003.
35. Pavel V Baranov, Clark M Henderson, Christine B Anderson, Raymond F Gesteland, John F Atkins, and Michael T Howard. Programmed ribosomal frameshifting in decoding the SARS-CoV genome. *Virology*, 332(2):498–510, 2005.
36. Kévin Darty, Alain Denise, and Yann Ponty. VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, 25(15):1974, 2009.
37. Ronny Lorenz, Stephan H Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. Viennarna Package 2.0. *Algorithms for Molecular Biology*, 6(1):26, 2011. doi: 10.1186/1748-7188-6-26.
38. Ronny Lorenz, Christoph Flamm, Ivo Hofacker, and Peter Stadler. Efficient computation of base-pairing probabilities in multi-strand RNA folding. In *Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies*, pages 23–31, - 2020. doi: 10.5220/0008916600230031.
39. Konstantin Klemm, Christoph Flamm, and Peter F Stadler. Funnels in energy landscapes. *The European Physical Journal B*, 63(3):387–391, 2008.
40. Mirela Andronescu, Vera Bereg, Holger H Hoos, and Anne Condon. RNA STRAND: the RNA secondary structure and statistical analysis database. *BMC Bioinformatics*, 9(1):1–10, 2008.
41. James W Brown. The ribonuclease P database. *Nucleic Acids Research*, 26(1):351–352, 1998.
42. S. Bellaousov and D. H. Mathews. Probknot: fast prediction of RNA secondary structure including pseudoknots. *RNA*, 16(10):1870–1880, 2010. doi: 10.1261/rna.2125310.
43. Jennifer Daub, Paul P Gardner, John Tate, Daniel Ramsköld, Magnus Manske, William G Scott, Zasha Weinberg, Sam Griffiths-Jones, and Alex Bateman. The RNA wikiproject: community annotation of RNA families. *RNA*, 14(12):2462–2464, 2008.
44. Simon H Damberger and Robin R Gutell. A comparative database of group I intron structures. *Nucleic Acids Research*, 22(17):3508–3510, 1994.
45. C. Zwieb. Tmrdb (tmRNA database). *Nucleic Acids Research*, 28(1):169–170, 2000. doi: 10.1093/nar/28.1.169.
46. C. Zwieb. Tmrdb (tmRNA database). *Nucleic Acids Research*, 31(1):446–447, 2003. doi: 10.1093/nar/gkg019.
47. Richard B Waring and R Wayne Davies. Assessment of a model for intron RNA secondary structure relevant to RNA self-splicing—a review. *Gene*, 28(3):277–291, 1984.
48. T. Specht, M. Szymanski, M. Z. Barciszewska, J. Barciszewski, and V. A. Erdmann. Compilation of 5s rRNA and 5s rRNA gene sequences. *Nucleic Acids Research*, 25(1):96–97, 1997. doi: 10.1093/nar/25.1.96.
49. Mathias Sprinzl, Carsten Horn, Melissa Brown, Anatoli Ioudovitch, and Sergey Steinberg. Compilation of tRNA sequences and sequences of tRNA genes. *Nucleic Acids Research*, 26(1):148–153, 1998.
50. Michael F Sloma and David H Mathews. Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA*, 22(12):1808–1818, 2016.
51. Murray N Schnare, Simon H Damberger, Michael W Gray, and Robin R Gutell. Comprehensive comparison of structural characteristics in eukaryotic cytoplasmic large subunit (23 s-like) ribosomal RNA. *Journal of Molecular Biology*, 256(4):701–719, 1996.
52. David H Mathews, Jeffrey Sabina, Michael Zuker, and Douglas H Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of Molecular Biology*, 288(5):911–940, 1999.
53. Tore Samuelsson and Christian Zwieb. The signal recognition particle database (srpdb). *Nucleic Acids Research*, 27(1):169–170, 1999.
54. Robin R Gutell, Michael W Gray, and Murray N Schnare. A compilation of large subunit (23s and 23s-like) ribosomal RNA structures: 1993. *Nucleic Acids Research*, 21(13):3055, 1993.
55. Robin R Gutell. Collection of small subunit (16s-and 16s-like) ribosomal RNA structures: 1994. *Nucleic Acids Research*, 22(17):3502–3507, 1994.
56. Paul P Gardner, Jennifer Daub, John G Tate, Eric P Nawrocki, Diana L Kolbe, Stinus Lindgreen, Adam C Wilkinson, Robert D Finn, Sam Griffiths-Jones, Sean R Eddy, et al. Rfam: updates to the RNA families database. *Nucleic Acids Research*, 37(suppl_1):D136–D140, 2009.
57. Bruce A Shapiro and Kaizhong Zhang. Comparing multiple RNA secondary structures using tree comparisons. *Bioinformatics*, 6(4):309–318, 1990.
58. Fabian Amman, Stephan H. Bernhart, Gero Doose, Ivo L. Hofacker, Jing Qin, Peter F. Stadler, and Sebastian Will. *The trouble with long-range base pairs in RNA folding*, pages 1–11. Advances in Bioinformatics and Computational Biology. Springer International Publishing, 2013. doi: 10.1007/978-3-319-02624-4_1.
59. Christoph Flamm, Ivo L. Hofacker, Peter F. Stadler, and Michael T. Wolfinger. Barrier trees of degenerate landscapes. *Zeitschrift für Physikalische Chemie*, 216(2):nil, 2002. doi: 10.1524/zpch.2002.216.2.155.

DRAFT