Consistent Consideration of RNA Structural Alignments Improves Prediction Accuracy of RNA Secondary Structures

Masaki Tagashira [1,2,*]

[1]Department of Computational Biology and Medical Sciences, University of Tokyo, Chiba, 277-8561, Japan and [2]Artificial Intelligence Research Center, AIST, Tokyo, 135-0064, Japan.

ABSTRACT

The probabilistic consideration of the global pairwise sequence alignment of two RNAs tied with their global single secondary structures, or global pairwise structural alignment, is known to predict more accurately global single secondary structures of unaligned homologs by discriminating between conserved local single secondary structures and those not conserved. However, conducting rigorously this consideration is computationally impractical and thus has been done to decompose global pairwise structural alignments into their independent components, i.e. global pairwise sequence alignments and single secondary structures, by conventional methods. ConsHomfold and ConsAlifold, which predict the global single and consensus secondary structures of unaligned and aligned homologs considering consistently preferable (or sparse) global pairwise structural alignments on probability respectively, were developed and implemented in this study. These methods demonstrate the best trade-off of prediction accuracy while exhibiting comparable running time compared to conventional methods. ConsHomfold and ConsAlifold optionally report novel types of loop accessibility, which are useful for the analysis of sequences and secondary structures. These accessibilities are average on sparse global pairwise structural alignment and can be computed to extend the novel inside-outside algorithm proposed in this study that computes pair alignment probabilities on this alignment.

INTRODUCTION

Investigating the (global secondary) structures of potentially functional *ncRNAs* is an important key to discover functional ncRNAs and uncover their functional details, because single structures are often conserved among homologs even in the case where the sequences of these homologs are nonconserved. (1, 2, 3) Methods that predict a structure from only a sequence (or *mono-foldings*), such as RNAfold (4), Mfold (5), Pfold (6), Sfold (7), CONTRAfold (8), and CentroidFold (9), have aided experimental single structure probing methods, such as Cryo-EM (10), icSHAPE (11), SHAPE-MaP (12), PARS (13), and PARIS (14), to provide input model single structures and predict single structures constrained by single structure probing data (15, 16, 17).

Mono-foldings suffer from the limited accuracy of predicted single structures, because there is no room for exploiting information other than single sequences. (Figure 1c) In order to predict more accurate structures, other sequences that share homology with a sequence are used. Predicting the (global) sequence alignment of RNAs tied with their single structures, or (global) *structural alignment* (18), is one of the most effective ways to predict more precise structures of homologs, which are informed by (base-)pairing substitutions (1, 19), though many methods that predict structural alignments report *consensus structure*s, rather than single structures. Consensus structures are a set of column pairs on sequence alignment that contain pairings conserved among homologs. The serious problem of predicting rigorous structural alignments is the impractical running time and memory usage of this prediction, even if this prediction is pairwise, which are bounded by $O(L^{3R})$ and $O(L^{2R})$, respectively, where L and R are the maximum length and the number of sequences, respectively. (18) This problem is solved by utilizing the progressive alignment (20) and *sparsification*, which filters out candidates that do not satisfy certain thresholds from consideration to set the scores of these candidates to $-\infty$, by many popular methods that predict structural alignments, such as PMcomp (21), Foldalign (22, 23), Murlet (24), MXSCARNA (25), LocARNA (26), RAF (27), DAFS (28), and SPARSE (29).

As other ways to predict more precise structures from homologs, the following methods are available:

- *hom-folding*: predicting the single structure of each unaligned homolog considering pairwise structural alignments between this homolog and the remaining unaligned homologs on the probability distributions of these alignments as CentroidHomfold (30) and TurboFold (31)
- *ali-folding*: predicting the consensus structure of aligned homologs as RNAalifold (32), PETfold (33), and CentroidAlifold (34).

The above foldings are more reasonable than structural alignment predictions, because the progressive alignment, which is not required for these foldings, is computationally complex and heavy. The conventional hom-foldings (CentroidHomfold and TurboFold) decompose pairwise structural alignments on probability into their independent components, i.e. pairwise sequence alignments and single

[*] To whom correspondence should be addressed. Emails : tagashira_masaki_17@stu-cbms.k.u-tokyo.ac.jp (primary) and heartsh@heartsh.io (second)

structures. (Figure 1d) Also, the conventional ali-foldings (RNAalifold, PETfold, and CentroidAlifold) do not consider pairwise structural alignments on probability, which will improve the prediction accuracy of these ali-foldings. (Figure 1c) There is possibly room for improving the prediction accuracy of hom-foldings and ali-foldings to exploit more accurately the homology among homologs.

Contributions of this study

A hom-folding and an ali-folding that consider consistently likely (or sparse) pairwise structural alignments on probability were proposed in this study as ConsHomfold and ConsAlifold (Figure 1a), respectively. (Figure 1d) These foldings are of the maximum expected accuracy principle whose prediction is in general more accurate than that of the maximum likelihood principle, because the number of all possible references grow exponentially to sequence length and thus results in the extremely high dimension of predictive space (35, 36). (Figure 1c) ConsHomfold and ConsAlifold compute *pair alignment probabilities* (or *pair match probabilities*) on sparse pairwise structural alignment that were proposed including the computation of them in LocARNA-P (37). The scores to calculate these probabilities in these foldings are based on *Turner's (nearest neighbor physics) model*, which scores single structures in terms of the free energy of their *loops* (38), and are expected to be more suitable to score structural alignments than those that are used in LocARNA-P. An algorithm to compute the probabilities was developed in this study, though this algorithm shares the framework of inside-outside algorithm with LocARNA-P. The novel algorithm requires (quadratic) less computational complexities than those quartic of LocARNA-P to impose more hard sparsity.

Also, novel types of the *loop accessibility* (= *structural profile* = *structural context*) proposed in CapR (39), which is useful for the analysis of RNAs and their structures, and an algorithm to compute the novel accessibilities were proposed in this study to extend that of the novel alignment probabilities. These novel accessibilities are based on sparse pairwise structural alignment, whereas those in CapR are based on local single structure. ConsHomfold and ConsAlifold output optionally the novel accessibilities.

MATERIALS AND METHODS

Pairwise structural alignment

Let $\boldsymbol{S}_{\boldsymbol{R}}$, $\boldsymbol{A}_{\mathbb{R}}$, and $\mathbb{A}_{\mathbb{R}}$ be the structure of the sequence $\boldsymbol{R}$, the sequence alignment between the pair of sequences $\mathbb{R}$, and the structural alignment between the pair $\mathbb{R}$, respectively. An alignment $\mathbb{A}_{\mathbb{R}}$ is composed of the alignment $\boldsymbol{A}_{\mathbb{R}}$ and the structures $\boldsymbol{S}_{\boldsymbol{R}}$ and $\boldsymbol{S}_{\boldsymbol{R}'}$, i.e. $\mathbb{A}_{\mathbb{R}} = (\boldsymbol{S}_{\boldsymbol{R}}, \boldsymbol{S}_{\boldsymbol{R}'}, \boldsymbol{A}_{\mathbb{R}})$ where $\mathbb{R} = (\boldsymbol{R}, \boldsymbol{R}')$. (Figure 1b) Assume *collinear* pairwise sequence alignments (40) and single structures without any *pseudoknot*s (41) to avoid larger computational complexities.

A position u is said to be *accessible from* (or *closed by*) pairing positions i and j if $i < u < j$ and the position pair (i,j) is the closest to the position u of all pairing position pairs. A position set about pairing positions i and j (= $L_{ij} = \{u | I^{\text{loop}}_{ij}(u) = 1\}$) is the loop of the positions i and j where $I^{\text{loop}}_{ij}(u)$ is 1 if the position u is accessible from the positions i and j and 0 otherwise. A loop L_{ij} is a *b-loop* if the loop L_{ij} contains $b-1$ pairing position pairs accessible from the positions i and j. A loop L_{ij} is said to be *internal* and *external* (or *outer*) if the positions i and j are accessible from the loops of other positions and the pseudo-positions (= the virtual positions closing the both ends of the sequence $\boldsymbol{R}$), respectively. (Figure 2) (Internal) b-loop is divided into three classes on Turner's model, which approximates the free energy of structure on thermodynamics (38): 1-loop (= *hairpin loop*), 2-loop (= *stacking (loop), bulge loop, and interior loop*), and $(b > 2)$-loop (= *multi-loop*).

Assume structural alignments without any *indels of 2-loops* (Figure 1b), which are required to align *stem (loops)* (= lines of successive stackings) of different lengths keeping pairings (18, 29), to prevent scoring structural alignments from becoming further complicated though scoring these indels does not increase computational complexities (29). Two sets of position pairs are said to be *pair-aligned* if these sets are pairing and aligned. Two positions are said to be *loop-aligned* if these positions are unpaired and aligned.

Posterior pair alignment probability matrix

Let $\mathcal{A}_{\mathbb{R}}$ and $s_{\mathbb{A}}$ be a set of all possible alignments $\mathbb{A}_{\mathbb{R}}$ and the score of the alignment $\mathbb{A}$, respectively. Assume that the probability of any alignment $\mathbb{A} \in \mathcal{A}_{\mathbb{R}}$ (= $p_{\mathbb{A}}$) obeys a *Boltzmann (probability) distribution*, i.e. $p_{\mathbb{A}} = \frac{\exp(s_{\mathbb{A}})}{Z}$ where $Z = \sum_{\mathbb{A}} \exp(s_{\mathbb{A}})$. Z is called a *partition function*.

Let $\boldsymbol{P}^{\text{PA}}_{\mathbb{R}}$ be the *pair alignment probability* matrix given the pair $\mathbb{R}$. Let $I^{\text{PA}}_{ijkl}(\mathbb{A})$ be 1 if the pairs (i,j) and (k,l) are pair-aligned in the alignment $\mathbb{A}$ and 0 otherwise, where i and j are two positions in the sequence $\boldsymbol{R}$, k and l are two positions in the other sequence $\boldsymbol{R}'$, $i < j$, and $k < l$. The matrix $\boldsymbol{P}^{\text{PA}}_{\mathbb{R}}$ can be written by the probabilities $p_{\mathbb{A}}$: $\boldsymbol{P}^{\text{PA}}_{\mathbb{R}} = (p^{\text{PA}}_{ijkl})$ where $p^{\text{PA}}_{ijkl} = p^{\text{PA}}_{ijkl}(\mathbb{R}) = \sum_{\mathbb{A} | I^{\text{PA}}_{ijkl}(\mathbb{A}) = 1} p_{\mathbb{A}}$. p^{PA}_{ijkl} is the probability that the pairs (i,j) and (k,l) are pair-aligned.

Composition of pairwise structural alignment score $s_{\mathbb{A}}$

Let $e_{\boldsymbol{S}}$ be the free energy of the structure $\boldsymbol{S}$. Score $s_{\mathbb{A}_{\mathbb{R}}}$ is decomposed into additional components: $s_{\mathbb{A}_{\mathbb{R}}} = -e_{\boldsymbol{S}_{\boldsymbol{R}}} - e_{\boldsymbol{S}_{\boldsymbol{R}'}} + s_{\boldsymbol{A}_{\mathbb{R}}} + s^{\text{PA}} + s^{\text{LA}}$. Here, $s_{\boldsymbol{A}_{\mathbb{R}}}$, s^{PA}, and s^{LA} are the score of the alignment $\boldsymbol{A}_{\mathbb{R}}$, the sum score of the pair alignments in the alignment $\mathbb{A}_{\mathbb{R}}$, and the sum score of the loop alignments in the alignment $\mathbb{A}_{\mathbb{R}}$, respectively.

The components $e_{\boldsymbol{S}_{\boldsymbol{R}}}$ and $e_{\boldsymbol{S}_{\boldsymbol{R}'}}$ can be computed by the estimated parameters of Turner's model. On it, energy $e_{\boldsymbol{S}}$ is decomposed into four categories of additional component: $e_{\boldsymbol{S}} = \sum_{ij\lambda | I^{\text{pair}}_{ij}(\boldsymbol{S}) I^{\lambda}_{ij} = 1} e^{\lambda}_{ij}$ where $\lambda \in \{1, 2, \text{multi}, \text{outer}\}$ and $I^{\text{pair}}_{ij}(\boldsymbol{S})$ returns 1 if the positions i and j are pairing in the structure $\boldsymbol{S}$. Here, I^{λ}_{ij} is 1 if the loop L_{ij} is a λ-loop and e^{λ}_{ij} is the free energy of the loop L_{ij} when the loop L_{ij} is a λ-loop. Energy e^{2}_{ij} depends on the pairing positions m and n accessible from the positions i and j: $e^{2}_{ij} = e^{2}_{ijmn}$ where e^{2}_{ijmn} is the energy e^{2}_{ij} parameterized with the positions m and n. Turner's model restricts the number of unpaired positions of the 2-loop L_{ij}, $(m-i)+(j-n)+2$:

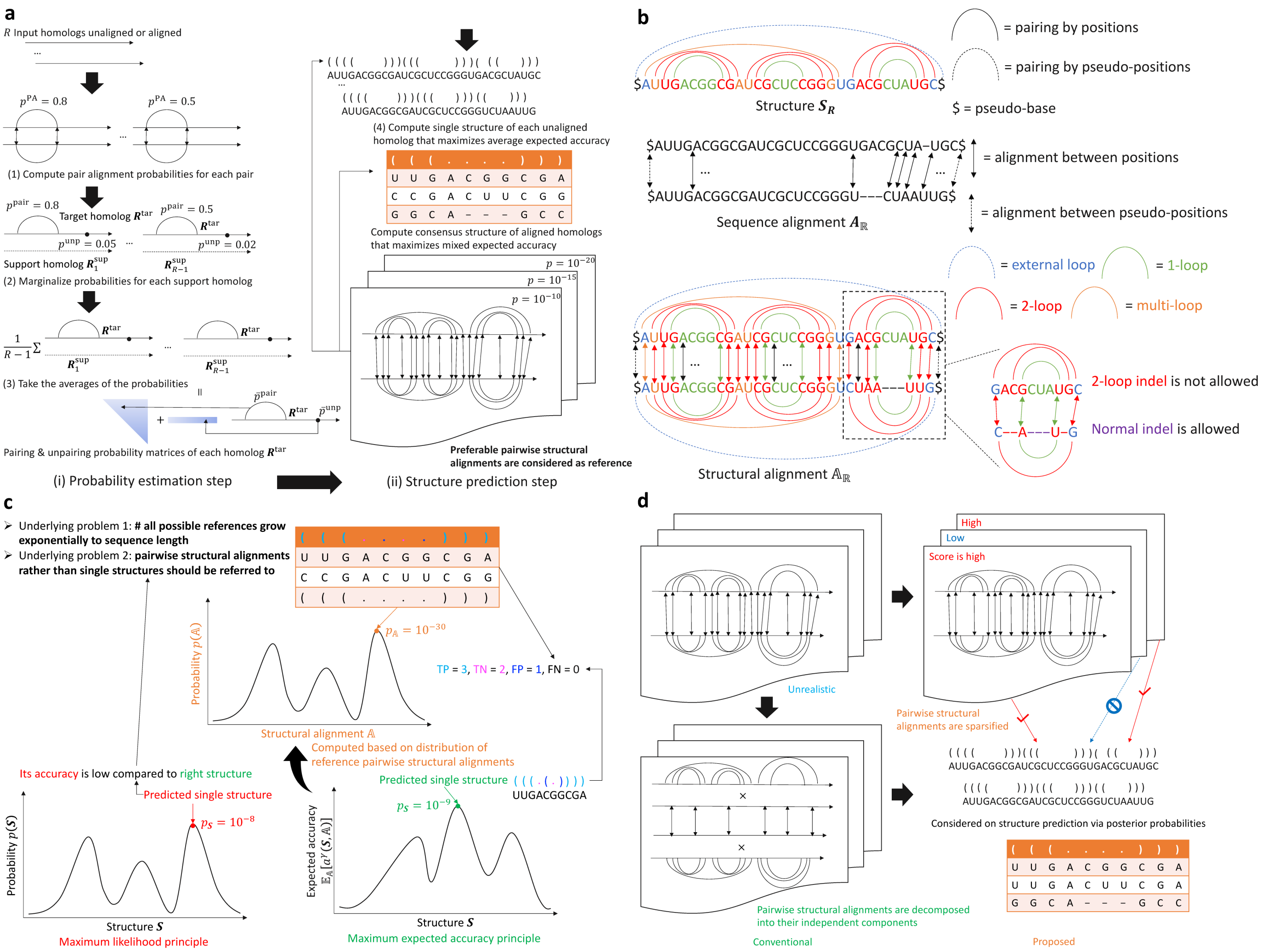

a
R Input homologs unaligned or aligned
$p^{\mathrm{PA}} = 0.8$
$p^{\mathrm{PA}} = 0.5$
(1) Compute pair alignment probabilities for each pair
$p^{\mathrm{pair}} = 0.8$
Target homolog $\boldsymbol{R}^{\mathrm{tar}}$
$p^{\mathrm{pair}} = 0.5$
$\boldsymbol{R}^{\mathrm{tar}}$
$p^{\mathrm{unp}} = 0.05$
$p^{\mathrm{unp}} = 0.02$
Support homolog $\boldsymbol{R}^{\mathrm{sup}}_1$
$\boldsymbol{R}^{\mathrm{sup}}_{R-1}$
(2) Marginalize probabilities for each support homolog
$\frac{1}{R-1}\Sigma$
(3) Take the averages of the probabilities
$\bar{p}^{\mathrm{pair}}$
$\bar{p}^{\mathrm{unp}}$
Pairing & unpairing probability matrices of each homolog $\boldsymbol{R}^{\mathrm{tar}}$
(i) Probability estimation step
((((   )))(((   )))(((   )))
AUUGACGGCGAUCGCUCCGGGUGACGCUAUGC
((((   )))(((   )))((   )))
AUUGACGGCGAUCGCUCCGGGUCUAAUUG
(4) Compute single structure of each unaligned homolog that maximizes average expected accuracy
(((. . . .)))
U U G A C G G C G A
C C G A C U U C G G
G G C A – – – G C C
Compute consensus structure of aligned homologs that maximizes mixed expected accuracy
$p = 10^{-20}$
$p = 10^{-15}$
$p = 10^{-10}$
Preferable pairwise structural alignments are considered as reference
(ii) Structure prediction step
b
= pairing by positions
= pairing by pseudo-positions
$AUUGACGGCGAUCGCUCCGGGUGACGCUAUGC$
Structure $\boldsymbol{S_R}$
$ = pseudo-base
$AUUGACGGCGAUCGCUCCGGGUGACGCUA–UGC$
$AUUGACGGCGAUCGCUCCGGGU–––CUAAUUG$
= alignment between positions
= alignment between pseudo-positions
Sequence alignment $\boldsymbol{A}_{\mathbb{R}}$
= external loop
= 1-loop
= 2-loop
= multi-loop
$AUUGACGGCGAUCGCUCCGGGUGACGCUAUGC$
$AUUGACGGCGAUCGCUCCGGGUCUAA–––UUG$
GACGCUAUGC
C––A–––U–G
2-loop indel is not allowed
Normal indel is allowed
Structural alignment $\mathbb{A}_{\mathbb{R}}$
c
➢ Underlying problem 1: # all possible references grow exponentially to sequence length
➢ Underlying problem 2: pairwise structural alignments rather than single structures should be referred to
(((. . . .)))
U U G A C G G C G A
C C G A C U U C G G
(((. . . .)))
$p_{\mathbb{A}} = 10^{-30}$
Probability $p(\mathbb{A})$
Structural alignment $\mathbb{A}$
TP = 3, TN = 2, FP = 1, FN = 0
Computed based on distribution of reference pairwise structural alignments
Its accuracy is low compared to right structure
Predicted single structure
$p_S = 10^{-8}$
Probability $p(\boldsymbol{S})$
Structure $\boldsymbol{S}$
Maximum likelihood principle
Predicted single structure (((.(.))))
UUGACGGCGA
$p_S = 10^{-9}$
Expected accuracy $\mathbb{E}_{\mathbb{A}}[a^{\gamma}(\boldsymbol{S},\mathbb{A})]$
Structure $\boldsymbol{S}$
Maximum expected accuracy principle
d
Unrealistic
×
×
Pairwise structural alignments are decomposed into their independent components
Conventional
High
Low
Score is high
Pairwise structural alignments are sparsified
((((   )))(((   )))(((   )))
AUUGACGGCGAUCGCUCCGGGUGACGCUAUGC
((((   )))(((   )))((   )))
AUUGACGGCGAUCGCUCCGGGUCUAAUUG
Considered on structure prediction via posterior probabilities
(((. . . .)))
U U G A C G G C G A
U U G A C U U C G A
G G C A – – – G C C
Proposed

Figure 1. (a) The proposed workflow of ConsHomfold and ConsAlifold. ConsHomfold predicts single structures from its input unaligned homologs. ConsAlifold predicts a consensus structure from its input aligned homologs. **The methods do not produce structural alignments.** The methods consist of two steps to (i) estimate probabilities and (ii) predict structures after this estimation. **For ConsAlifold, the estimation step ignores the gaps in the alignment.** (1) First, pair alignment probabilities on sparse pairwise structural alignment for each pair of input homologs are computed. (2) Then, the probabilities are marginalized to be pairing and unpairing probabilities of each homolog. (3) As the final process of the estimation step, average pairing and unpairing probabilities between each homolog and the remaining homologs are gained. These probabilities are utilized to incorporate more than one support homolog into the subsequent predictions of structures. (4) Finally, ConsHomfold predicts the single structure of each input homolog and ConsAlifold predicts the consensus structure of the input alignment. **(b) Examples of a structure $\boldsymbol{S_R}$, a sequence alignment $\boldsymbol{A}_{\mathbb{R}}$, and a structural alignment** $\mathbb{A}_{\mathbb{R}}$**.** A structure $\boldsymbol{S_R}$ and an alignment $\mathbb{A}_{\mathbb{R}}$ are color-coded based on types of loop. **(c) The underlying problems on conventional mono-foldings and ali-foldings.** The maximum likelihood principle, which is equivalent to the free energy minimization on structure predictions (e.g. RNAfold and RNAalifold), exerts less prediction accuracy in general compared to the maximum expected accuracy principle, which is based on Bayes' theorem (e.g. CONTRAfold, CentroidFold, CentroidHomfold, TurboFold, PETfold, CentroidAlifold, ConsHomfold, and ConsAlifold), because the most probable references have their probabilities, which decay exponentially to sequence length. CONTRAfold, CentroidFold (conventional mono-foldings), PETfold, and CentroidAlifold (conventional ali-foldings) consider not pairwise structural alignments but single structures. **(d) Considering all possible pairwise structural alignments costs $O(N^3M^3)$ long hours and $O(N^2M^2)$ huge memory where N and M are the lengths of homologs.** The conventional hom-foldings, CentroidHomfold and TurboFold, decompose all pairwise structural alignments into their independent pairwise sequence alignments and single structures. **ConsHomfold and ConsAlifold take consistently sparse pairwise structural alignments into account.**

$(m-i)+(j-n)+2 \leq 30$ to reduce the time complexities of prediction algorithms. Energy e^{multi}_{ij} is decomposed into the terms of the closing and accessible pairings: $e^{\mathrm{multi}}_{ij} = e^{\mathrm{CBP}} + \sum_{mn|I^{\mathrm{pair}}_{mn}(\boldsymbol{S})=1, m,n \in L_{ij}} e^{\mathrm{ABP}}$ where e^{CBP} and e^{ABP} are free energy per closing and accessible pairing, respectively. Energy e^{outer}_{ij} does not influence the entire free energy $e_{\boldsymbol{S}}$: $e^{\mathrm{outer}}_{ij} = 0$.

As the components $e_{\boldsymbol{S_R}}$ and $e_{\boldsymbol{S_{R'}}}$, many conventional methods scoring structural alignments such as PMcomp, LocARNA, RAF, SPARSE, and LocARNA-P employ the *posterior model*. It scores the components $e_{\boldsymbol{S_R}}$ and $e_{\boldsymbol{S_{R'}}}$ with posterior pairing probability matrices on structure (estimated by *inside-outside algorithms* such as McCaskill's algorithm (42) and its variant algorithms (43, 44, 45)) to simplify computations although the suitability of these matrices has not been discussed. (21, 26, 27, 29, 37) Hence, Turner's model is adopted in this study to prevent a reduction in prediction accuracy due to a matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$.

The component $s_{\boldsymbol{A}_{\mathbb{R}}}$ is computed by the learned parameters, including affine gap scores used in CONTRAlign, which predicts pairwise sequence alignments (46). The components s^{PA} and s^{LA} can be computed to sum RIBOSUM scores (1) across all pair-aligned and loop-aligned positions, respectively: $s^{\mathrm{PA}}=\sum_{ijkl|I^{\mathrm{PA}}_{ijkl}(\mathbb{A})=1}s^{\mathrm{PA}}_{ijkl}$ and $s^{\mathrm{LA}}=\sum_{uv|I^{\mathrm{LA}}_{uv}(\mathbb{A})=1}s^{\mathrm{LA}}_{uv}$ where s^{PA}_{ijkl} and s^{LA}_{uv} are the RIBOSUM pair and loop alignment scores of the pairs (i,j) and (k,l) and the positions u and v, respectively and $I^{\mathrm{LA}}_{uv}(\mathbb{A})$ is 1 if the positions u and v are loop-aligned in the alignment $\mathbb{A}$ and 0 otherwise.

Inside-outside algorithm that computes pair alignment probability matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$

In this section, an efficient (however nevertheless impractical) method that computes a matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$ in the framework of inside-outside algorithm is proposed. A probability p^{PA}_{ijkl} can be written with "inside" partition functions $\alpha^{\mathrm{outer,for}}_{NM}$ and $\alpha^{\mathrm{PA}}_{ijkl}$ and an "outside" partition function $\beta^{\mathrm{PA}}_{ijkl}$: $p^{\mathrm{PA}}_{ijkl}=\frac{\alpha^{\mathrm{PA}}_{ijkl}\beta^{\mathrm{PA}}_{ijkl}}{\alpha^{\mathrm{outer,for}}_{NM}}$ where N and M are the lengths of the sequences $\boldsymbol{R}$ and $\boldsymbol{R}'$, respectively. Inside and outside partition functions can be computed with those of shorter and longer substrings, respectively, and are stored in dynamic programming memory for the remaining computation. A matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$ can be computed by Algorithm 1 with the $O(N^4M^4)$ time and the $O(N^3M^3)$ memory. (Figure 3a)

Algorithm 1 **An inside-outside algorithm that computes a pair alignment probability matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$.**

```
function InOutAlgo($\mathbb{R}$)
    // $x$ and $y$ are partition function labels to represent which state is considered on positions.
    Compute inside partition functions $\alpha^{x}_{ijkl}$ and $\alpha^{y}_{uv}$ on the dynamic programming described in Supplementary section 1.1 // Inside step.
    Compute outside partition functions $\beta^{x}_{ijkl}$ on the dynamic programming described in Supplementary section 1.2 // Outside step.
    Compute pair alignment probabilities $p^{\mathrm{PA}}_{ijkl}$ by $p^{\mathrm{PA}}_{ijkl}=\frac{\alpha^{\mathrm{PA}}_{ijkl}\beta^{\mathrm{PA}}_{ijkl}}{\alpha^{\mathrm{outer,for}}_{NM}}$ // Final step.
    return $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$
```

PROOF. Partition functions $\alpha^{\mathrm{PA}}_{ijkl}$ and $\beta^{\mathrm{PA}}_{ijkl}$ and probabilities p^{PA}_{ijkl} demand the $O(N^2M^2)$ memory. Partition functions α^{y}_{uv} are stored for all the combinations of pair-aligned pairs (i,j) and (k,l) that close the positions u and v, respectively and thus demand the $O(N^3M^3)$ memory.

Partition functions α^{y}_{uv} include the case where the positions u and v are pair-aligned and therefore demand the $O(N^4M^4)$ time. Partition functions α^{x}_{ijkl} are computed from the partition functions $\alpha^{y}_{j-1,l-1}$ in only the $O(1)$ time and thus demand the $O(N^2M^2)$ time as a whole. Partition functions β^{x}_{ijkl} consider the pair-aligned pairs of positions that close the position pairs (i,j) and (k,l) and therefore demand the $O(N^4M^4)$ time.

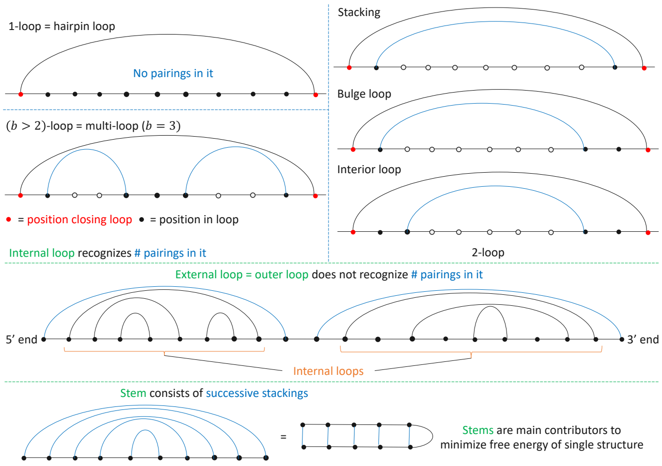

1-loop = hairpin loop
No pairings in it
Stacking
Bulge loop
(b > 2)-loop = multi-loop (b = 3)
Interior loop
● = position closing loop ● = position in loop
2-loop
Internal loop recognizes # pairings in it
External loop = outer loop does not recognize # pairings in it
5' end
3' end
Internal loops
Stem consists of successive stackings
=
Stems are main contributors to minimize free energy of single structure

Figure 2. Different types of loop constitute a single structure. A stacking does not contain unpairings accessible in it. A bulge loop contains unpairings accessible in it on either of the 5' and 3' sides. An interior loop contains unpairings accessible in it on both of these sides.

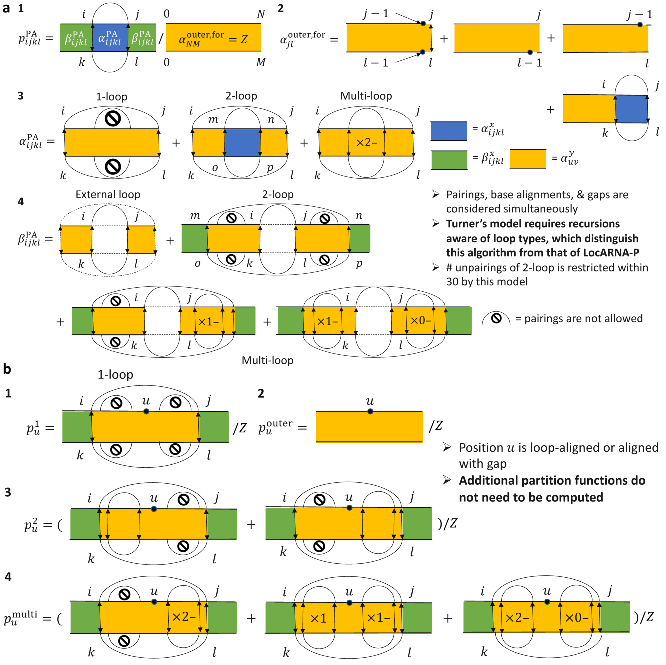
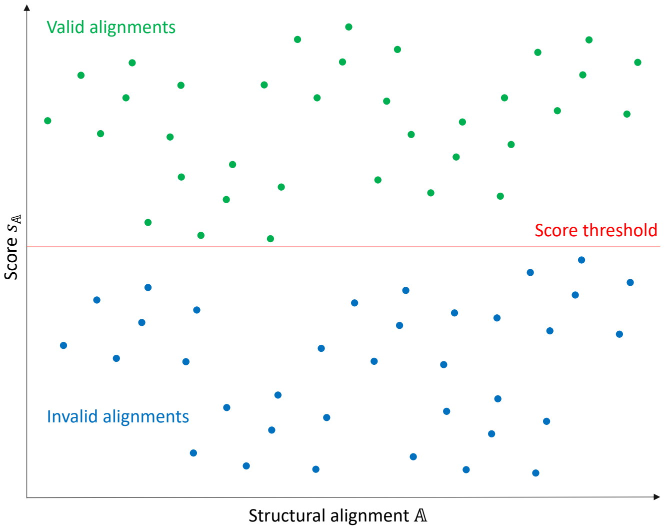

a
1
$p^{\mathrm{PA}}_{ijkl} =$
$\beta^{\mathrm{PA}}_{ijkl}$
$\alpha^{\mathrm{PA}}_{ijkl}$
$\beta^{\mathrm{PA}}_{ijkl}$
/
$\alpha^{\mathrm{outer,for}}_{NM} = Z$
i
j
k
l
0
N
0
M
2
$\alpha^{\mathrm{outer,for}}_{jl} =$
j − 1
l − 1
+
3
$\alpha^{\mathrm{PA}}_{ijkl} =$
1-loop
2-loop
Multi-loop
m
n
o
p
×2−
$= \alpha^{x}_{ijkl}$
$= \beta^{x}_{ijkl}$
$= \alpha^{y}_{uv}$
4
$\beta^{\mathrm{PA}}_{ijkl} =$
External loop
2-loop
×1−
×0−
Multi-loop
= pairings are not allowed
Pairings, base alignments, & gaps are considered simultaneously
Turner's model requires recursions aware of loop types, which distinguish this algorithm from that of LocARNA-P
# unpairings of 2-loop is restricted within 30 by this model
b
1
1-loop
$p^{1}_{u} =$
u
/Z
2
$p^{\mathrm{outer}}_{u} =$
/Z
Position u is loop-aligned or aligned with gap
Additional partition functions do not need to be computed
3
$p^{2}_{u} = ($
)/Z
4
$p^{\mathrm{multi}}_{u} = ($
×2−
×1
×1−
×0−
)/Z

Figure 3. (a) An overview of recursions to compute (1) probabilities p^{PA}_{ijkl} and partition functions (2) α^{y}_{uv}, (3) α^{x}_{ijkl}, and (4) β^{x}_{ijkl}. (b) An overview of recursions to compute accessibilities (1) p^{1}_{u}, (2) p^{outer}_{u}, (3) p^{2}_{u}, and (4) p^{multi}_{u}.

Probabilities p^{PA}_{ijkl} are computed from the partition functions α^{x}_{ijkl} and β^{x}_{ijkl} in only the $O(1)$ time and therefore demand the $O(N^2M^2)$ time as a whole. Finally, a matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$ demands the $O(N^4M^4)$ time and the $O(N^3M^3)$ memory.



Invalid alignments $\mathbb{A}$, which are filtered out by score thresold, are given their scores $s_{\mathbb{A}} \to -\infty$ & thus do not affect posterior probabilities because contributions of alignments $\mathbb{A}$ to these probabilities become $\lim_{s_{\mathbb{A}}\to-\infty} \exp s_{\mathbb{A}} = 0$

Figure 4. Sparsification lets Algorithm 1 ignore alignments $\mathcal{A}_{\mathbb{R}}$ whose contributions to the partition function Z are low. Choosing sparsification conditions whose dynamic programming reduces effectively their computational complexities is of extreme importance.

Algorithm 1 is the "simultaneous" solution of Durbin's (*forward-backward*) algorithm, which estimates posterior alignment probability matrices on sequence alignment (47), and McCaskill's algorithm, as expected. Algorithm 1 is also an inside-outside algorithm version of Sankoff's algorithm, which predicts pairwise structural alignments whose score is maximum (18), as expected. However, desirable time and memory complexities of Algorithm 1 are quadratic to deal with long ncRNAs.

Sparsifying pair alignment probability matrix $\boldsymbol{P}_{\mathbb{R}}^{\mathrm{PA}}$

In this section, a solution to make Algorithm 1 lightweight, sparsifying all possible structural alignments $\mathcal{A}_{\mathbb{R}}$, is introduced. From all the possible alignments $\mathcal{A}_{\mathbb{R}}$, *sparsification* can pick out those favorable (e.g. with adequately high scores). It can allow Algorithm 1 to compute only the partition functions α^{x}_{ijkl}, α^{y}_{uv}, and β^{x}_{ijkl} that satisfy sparsification conditions. (Figure 4) Let $p^{\mathrm{pair}}_{ij}(\boldsymbol{R})$ be the pairing probability of the positions i and j given the sequence $\boldsymbol{R}$. In this study, the following sparsification conditions are introduced:

- $|u-v| \leq \delta^{\mathrm{gap}}$ and $|(M-u)-(N-v)| \leq \delta^{\mathrm{gap}}$ for any positions u and v
- $|(j-i)-(l-k)| \leq \delta^{\mathrm{gap}}$ for any pair-aligned pairs (i,j) and (k,l)
- $p^{\mathrm{pair}}_{ij}(\boldsymbol{R}) \geq \epsilon$ for any pairing positions i and j and any sequence $\boldsymbol{R}$

where δ^{gap} and ϵ are sparsification parameters. The first two *banding conditions* let Algorithm 1 not consider the alignments $\mathcal{A}_{\mathbb{R}}$ with too many gaps. (21, 22, 23, 24, 48) The last *pairing condition* makes Algorithm 1 not consider the alignments $\mathcal{A}_{\mathbb{R}}$ with pairings difficult to predict (e.g. distant). (24, 27, 28, 29, 37, 49, 50) If Turner's model is replaced with the posterior model and the banding conditions are removed, Algorithm 1 becomes identical to LocARNA-P. (37) Algorithm 1 with the above conditions is ideal with the $O(L^2)$ time and the $O(L^2)$ memory where $L = \max(N, M)$ if the parameters δ^{gap} and ϵ take sufficiently small and large values, respectively.

PROOF. The numbers of all the possible pairings of positions u and v become $O(N\delta^{\mathrm{bp}})$ and $O(M\delta^{\mathrm{bp}})$ from $O(N^2)$ and $O(M^2)$, respectively where $\delta^{\mathrm{bp}} = \lfloor \frac{1}{\epsilon} \rfloor$ and $\lfloor r \rfloor$ returns the greatest integer less than or equal to the real number r. The number of all the possible combinations of positions u and v becomes $O(L\delta^{\mathrm{gap}})$ from $O(NM)$. The number of all the possible pair-aligned pairs (i,j) and (k,l) becomes $O(\#^{\mathrm{PA}})$ from $O(N^2M^2)$ where $\#^{\mathrm{PA}} = L\delta^{\mathrm{bp}}\delta^{\mathrm{gap}}\delta^{\mathrm{max}}$ and $\delta^{\mathrm{max}} = \max(\delta^{\mathrm{bp}}, \delta^{\mathrm{gap}})$.

Partition functions $\alpha^{\mathrm{PA}}_{ijkl}$ and $\beta^{\mathrm{PA}}_{ijkl}$ and probabilities p^{PA}_{ijkl} demand the $O(\#^{\mathrm{PA}})$ memory. Partition functions α^{y}_{uv} are stored for all the combinations of pair-aligned pairs (i,j) and (k,l) that close the positions u and v, respectively and thus demand the $O(\#^{\mathrm{PA}}L\delta^{\mathrm{gap}})$ memory.

Partition functions α^{y}_{uv} include the case where the positions u and v are pair-aligned and therefore demand the $O((\#^{\mathrm{PA}})^2)$ time. Partition functions α^{x}_{ijkl} are computed from the partition functions $\alpha^{y}_{j-1,l-1}$ in only the $O(1)$ time and thus demand the $O(\#^{\mathrm{PA}})$ time as a whole. Partition functions β^{x}_{ijkl} consider the pair-aligned pairs of positions that close the position pairs (i,j) and (k,l) and therefore demand the $O((\#^{\mathrm{PA}})^2)$ time.

Probabilities p^{PA}_{ijkl} are computed from the partition functions α^{x}_{ijkl} and β^{x}_{ijkl} in only the $O(1)$ time and therefore demand the $O(\#^{\mathrm{PA}})$ time as a whole. Finally, a matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$ demands the $O((\#^{\mathrm{PA}})^2)$ time and the $O(\#^{\mathrm{PA}}L\delta^{\mathrm{gap}})$ memory. If the parameters δ^{gap} and δ^{bp} are sufficiently small, a matrix $\boldsymbol{P}^{\mathrm{PA}}_{\mathbb{R}}$ demands the $O(L^2)$ time and the $O(L^2)$ memory.

Probabilistic consistency transformation

Probabilistic consistency transformation is the technique that converts a probability between a target homolog and each of its support homolog into a metric that summarizes the phylogeny among all the homologs. (24, 30, 31, 51) This transformation is required, because the computational complexities involved in computing posterior probabilities among all the homologs are NP-complete, as with multiple rigorous alignment. In this study, methods that transform probabilities p^{PA}_{ijkl} are proposed. To average probabilities $p^{\mathrm{PA}}_{ijkl}((\boldsymbol{R}^{\mathrm{tar}}, \boldsymbol{R}^{\mathrm{sup}}))$ between the target homolog $\boldsymbol{R}^{\mathrm{tar}}$ and each support homolog $\boldsymbol{R}^{\mathrm{sup}} \in R^{\mathrm{sup}}$, the average pairing probability $p^{\mathrm{pair}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$ is gained:

$$p^{\mathrm{pair}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) = \frac{\sum_{\boldsymbol{R}^{\mathrm{sup}}kl} p^{\mathrm{PA}}_{ijkl}((\boldsymbol{R}^{\mathrm{tar}}, \boldsymbol{R}^{\mathrm{sup}}))}{|R^{\mathrm{sup}}|}$$

where R^{sup} is a set of support homologs of the homolog $\boldsymbol{R}^{\mathrm{tar}}$. To sum probabilities $p^{\mathrm{pair}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$, an average unpairing probability $p^{\mathrm{unp}}_{i}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$ is obtained:

$$p^{\mathrm{unp}}_{i}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) = 1 - \sum_{j:j<i} p^{\mathrm{pair}}_{ji}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) - \sum_{j:i<j} p^{\mathrm{pair}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}).$$

Obviously, the difference between proposed probabilities $p_{ij}^{\text{pair}}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and existing probabilities $p_{ij}^{\text{pair}}(\boldsymbol{R}^{\text{tar}})$ is whether support homologs R^{sup} are considered or not.

Hom-folding that maximizes average expected accuracy

The accuracy of a predicted structure $\boldsymbol{S}$ against a reference alignment $\mathbb{A}$ is measured based on terms of positive and negative predictions:

$$a(\boldsymbol{S},\mathbb{A}) = \alpha_1 TP + \alpha_2 TN - \alpha_3 FP - \alpha_4 FN$$

where TP, TN, FP, and FN are the numbers of true positive, true negative, false positive, and false negative predictions, respectively, and α_h are their scale parameters. In this study, the counts TP, TN, FP, and FN are configured as

$$\begin{cases} TP = \sum_{ij} I_{ij}^{\text{pair}}(\boldsymbol{S}) I_{ij}^{\text{pair}}(\mathbb{A}), TN = \sum_i I_i^{\text{unp}}(\boldsymbol{S}) I_i^{\text{unp}}(\mathbb{A}) \\ FP = \sum_i I_i^{\text{pair}}(\boldsymbol{S}) I_i^{\text{unp}}(\mathbb{A}), FN = \sum_i I_i^{\text{unp}}(\boldsymbol{S}) I_i^{\text{pair}}(\mathbb{A}) \end{cases}.$$

Here, $I_{ij}^{\text{pair}}(\boldsymbol{S})$ is 1 if the positions i and j are pairing in the structure $\boldsymbol{S}$ and 0 otherwise, $I_{ij}^{\text{pair}}(\mathbb{A})$ is 1 if the positions i and j are pairing in the alignment $\mathbb{A}$ and 0 otherwise, $I_i^{\text{unp}}(\boldsymbol{S})$ is 1 if the position i is unpaired in the structure $\boldsymbol{S}$ and 0 otherwise, $I_i^{\text{unp}}(\mathbb{A})$ is 1 if the position i is unpaired in the alignment $\mathbb{A}$ and 0 otherwise, $I_i^{\text{pair}}(\boldsymbol{S}) = 1 - I_i^{\text{unp}}(\boldsymbol{S})$, and $I_i^{\text{pair}}(\mathbb{A}) = 1 - I_i^{\text{unp}}(\mathbb{A})$.

Because the accuracy $a(\boldsymbol{S},\mathbb{A})$ and the γ-dependent accuracy $a^{\gamma}(\boldsymbol{S},\mathbb{A}) = \gamma TP + TN$ are equivalent, the expected accuracy to be maximized is gained:

$$\mathbb{E}_{\mathbb{A}_{\mathbb{R}}}[a^{\gamma}(\boldsymbol{S}_{\boldsymbol{R}},\mathbb{A}_{\mathbb{R}})] = \gamma \sum_{ij \mid I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{R}})=1} p_{ij}^{\text{pair}} + \sum_{i \mid I_i^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{R}})=1} p_i^{\text{unp}} \quad (1)$$

where $\gamma = \frac{\alpha_1 + 2\alpha_4}{\alpha_2 + \alpha_3}$, $p_{ij}^{\text{pair}} = \sum_{kl} p_{ijkl}^{\text{PA}}$, and $p_i^{\text{unp}} = 1 - \sum_{j:j<i} p_{ji}^{\text{pair}} - \sum_{j:i<j} p_{ij}^{\text{pair}}$.

PROOF.

$$\begin{aligned} a(\boldsymbol{S},\mathbb{A}) &= \alpha_1 TP + \alpha_2 TN - \alpha_3 FP - \alpha_4 FN \\ &= \alpha_1 TP + \alpha_2 TN - \alpha_3 (N^* - TN) - \alpha_4 (P^* - 2TP) \\ &= (\alpha_1 + 2\alpha_4) TP + (\alpha_2 + \alpha_3) TN - \alpha_3 N^* - \alpha_4 P^* \\ &= (\alpha_1 + 2\alpha_4) TP + (\alpha_2 + \alpha_3) TN + \text{const.} \end{aligned} \quad (2)$$

where $P^* = 2TP + FN = \sum_i I_i^{\text{pair}}(\mathbb{A})$ and $N^* = TN + FP = \sum_i I_i^{\text{unp}}(\mathbb{A})$. To divide the both sides of Equation **2** by the scaler $\alpha_2 + \alpha_3$, the equivalence $\frac{a(\boldsymbol{S},\mathbb{A})}{\alpha_2+\alpha_3} = \gamma TP + TN + \text{const.}$ is obtained.

$$\begin{aligned} \text{LHS} &= \mathbb{E}_{\mathbb{A}_{\mathbb{R}}}[a^{\gamma}(\boldsymbol{S}_{\boldsymbol{R}},\mathbb{A}_{\mathbb{R}})] \\ &= \sum_{\mathbb{A}_{\mathbb{R}}} p_{\mathbb{A}_{\mathbb{R}}} (\gamma TP + TN) \\ &= \sum_{\mathbb{A}_{\mathbb{R}}} p_{\mathbb{A}_{\mathbb{R}}} (\gamma \sum_{ij} I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{R}}) I_{ij}^{\text{pair}}(\mathbb{A}_{\mathbb{R}}) + \sum_i I_i^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{R}}) I_i^{\text{unp}}(\mathbb{A}_{\mathbb{R}})) \\ &= \gamma \sum_{ij \mid I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{R}})=1} \sum_{\mathbb{A}_{\mathbb{R}} \mid I_{ij}^{\text{pair}}(\mathbb{A}_{\mathbb{R}})=1} p_{\mathbb{A}_{\mathbb{R}}} \\ &\quad + \sum_{i \mid I_i^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{R}})=1} \sum_{\mathbb{A}_{\mathbb{R}} \mid I_i^{\text{unp}}(\mathbb{A}_{\mathbb{R}})=1} p_{\mathbb{A}_{\mathbb{R}}} \\ &= \gamma \sum_{ij \mid I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{R}})=1} p_{ij}^{\text{pair}} + \sum_{i \mid I_i^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{R}})=1} p_i^{\text{unp}} \\ &= \text{RHS.} \end{aligned}$$

Deriving the expected accuracy that makes posterior probabilities explicit like Equation **1**, which enables dynamic programming to maximize this accuracy, is called *posterior decoding*. (52) The predicted structure that maximizes expected accuracy $\mathbb{E}_{\mathbb{A}_{\mathbb{R}}}[a^{\gamma}(\boldsymbol{S}_{\boldsymbol{R}},\mathbb{A}_{\mathbb{R}})]$ can be computed to conduct Nussinov-type dynamic programming (53) based on the following recursion:

$$a_{ij} = \max \begin{cases} p_i^{\text{unp}} + a_{i+1,j} & (i \text{ unpairs}) \\ a_{i,j-1} + p_j^{\text{unp}} & (j \text{ unpairs}) \\ \gamma p_{ij}^{\text{pair}} + a_{i+1,j-1} & (i,j \text{ pair}) \\ \max_{k:i \le k < j} (a_{ik} + a_{k+1,j}) & (\text{Bifurcates}) \end{cases}. \quad (3)$$

In Equation **3**, only the one support homolog $\boldsymbol{R}'$ of the target homolog $\boldsymbol{R}$ is considered to predict the single structure of the homolog $\boldsymbol{R}$. In order to consider more than one support homolog, it is sufficient that probabilities p_{ij}^{pair} and p_i^{unp} are replaced with the probabilities $p_{ij}^{\text{pair}}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $p_i^{\text{unp}}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ in Equation **3**, respectively. If the parameter γ is large, Equation **3** emphasizes positives and thus predicts more pairings. If the parameter γ is small, Equation **3** emphasizes negatives and thus predicts more unpairings.

Ali-folding that maximizes mixed expected accuracy

Let $\boldsymbol{A}_{R^{\text{hom}}}$ be the sequence alignment among the set of homologs R^{hom}. Single structure prediction is extended to consensus structure prediction of sequence alignment to view positions i on a sequence $\boldsymbol{R}$ as columns i on an alignment $\boldsymbol{A}_{R^{\text{hom}}}$ in Equation **3**. It is known that pairing probabilities of columns i and j given an alignment $\boldsymbol{A}_{R^{\text{hom}}}$ (= $p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$), which can be computed by RNAalifold (32), improve the prediction accuracy of consensus structure. (32, 34) Thus, the mixture of probabilities $p_{ij}^{\text{pair}}(\boldsymbol{R}, R^{\text{sup}})$ and $p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$ is used on Equation **3** to predict the consensus structure of an alignment $\boldsymbol{A}_{R^{\text{hom}}}$ (= $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$):

$$p_{ij}^{\text{mix}}(\boldsymbol{A}_{R^{\text{hom}}}) = \tau \frac{\sum_{\boldsymbol{R}} p_{i^*j^*}^{\text{pair}}(\boldsymbol{R}, R^{\text{hom}} \setminus \boldsymbol{R})}{|R^{\text{hom}*}|} + (1-\tau) p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$$

where $0 \leq \tau \leq 1$, $\boldsymbol{R} \in R^{\text{hom}}$, $R^{\text{hom}*}$ is the subset of the sequences R^{hom} that is not mapped to the gaps on the columns i and j in the alignment $\boldsymbol{A}_{R^{\text{hom}}}$, and i^* and j^* are the positions on the sequence $\boldsymbol{R}$ mapped to the columns i and j in the alignment $\boldsymbol{A}_{R^{\text{hom}}}$ respectively. The parameter τ is a mixing coefficient. Likewise, the mixture of probabilities $p_i^{\text{unp}}(\boldsymbol{R}, R^{\text{sup}})$ and $p_i^{\text{unp}}(\boldsymbol{A}_{R^{\text{hom}}})$ is used on Equation **3**:

$$p_i^{\text{mix}}(\boldsymbol{A}_{R^{\text{hom}}}) = \tau \frac{\sum_{\boldsymbol{R}} p_{i^*}^{\text{unp}}(\boldsymbol{R}, R^{\text{hom}} \setminus \boldsymbol{R})}{|R^{\text{hom}*}|} + (1-\tau) p_i^{\text{unp}}(\boldsymbol{A}_{R^{\text{hom}}})$$

where $p_i^{\text{unp}}(\boldsymbol{A}_{R^{\text{hom}}}) = 1 - \sum_{j:j<i} p_{ji}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}}) - \sum_{j:i<j} p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$. Finally, the following recursion, which predicts a structure $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$, is obtained:

$$a_{ij} = \max \begin{cases} p_i^{\text{mix}}(\boldsymbol{A}_{R^{\text{hom}}}) + a_{i+1,j} & (i \text{ unpairs}) \\ a_{i,j-1} + p_j^{\text{mix}}(\boldsymbol{A}_{R^{\text{hom}}}) & (j \text{ unpairs}) \\ \gamma p_{ij}^{\text{mix}}(\boldsymbol{A}_{R^{\text{hom}}}) + a_{i+1,j-1} & (i,j \text{ pair}) \\ \max_{k:i \leq k < j}(a_{ik} + a_{k+1,j}) & (\text{Bifurcates}) \end{cases}.$$

Modifying Algorithm 1 to also compute average loop accessibilities on sparse pairwise structural alignment

The posterior probability that a position u is accessible from λ-loops is called the loop accessibility of the position u. Let loop accessibility matrices given a pair $\mathbb{R}$ be $P_{\mathbb{R}}^{\text{pair},\lambda^*} = (p_{ij}^{\lambda^*})$ and $P_{\mathbb{R}}^{\text{unp},\lambda} = (p_u^{\lambda})$ where $\lambda^* \in \{2, \text{multi}, \text{outer}\}$, $p_{ij}^{\lambda^*} = p_{ij}^{\lambda^*}(\mathbb{R}) = \sum_{\mathbb{A} | I_{ijkl}^{\text{PA}}(\mathbb{A}) I_i^{\lambda^*}(\mathbb{A}) I_j^{\lambda^*}(\mathbb{A}) = 1} p_{\mathbb{A}}$, and $p_u^{\lambda} = p_u^{\lambda}(\mathbb{R}) = \sum_{\mathbb{A} | I_u^{\text{unp}}(\mathbb{A}) I_u^{\lambda}(\mathbb{A}) = 1} p_{\mathbb{A}}$. Here, $I_u^{\lambda}(\mathbb{A})$ returns 1 if the position u is accessible from a λ-loop in the alignment $\mathbb{A}$. Accessibilities $p_{ij}^{\lambda^*}$ can be computed while computing probabilities p_{ijkl}^{PA} because $\sum_{kl} p_{ijkl}^{\text{PA}} = \sum_{\lambda^*} p_{ij}^{\lambda^*}$. (Supplementary section 1.3) Accessibilities p_u^{λ} ask Algorithm 1 for additional computations. (Figure 3b) Matrices $P_{\mathbb{R}}^{\text{PA}}$, $P_{\mathbb{R}}^{\text{pair},\lambda^*}$, and $P_{\mathbb{R}}^{\text{unp},\lambda}$ are computed by Algorithm 2 with the $O(N^4 M^4)$ time and the $O(N^3 M^3)$ memory. The sparsifications applied to Algorithm 1 are also applied to Algorithm 2. Therefore, the time and memory complexities of Algorithm 2 become $O(L^2)$ if the parameters δ^{gap} and ϵ take sufficiently small and large values, respectively as Algorithm 1.

Probabilistic consistency transformation for accessibilities $p_{ij}^{\lambda^*}$ and p_u^{λ} are also proposed. To average an accessibility $p_{ij}^{\lambda^*}((\boldsymbol{R}^{\text{tar}}, \boldsymbol{R}^{\text{sup}}))$ between the target homolog $\boldsymbol{R}^{\text{tar}}$ and each support homolog $\boldsymbol{R}^{\text{sup}}$, the average accessibility $p_{ij}^{\lambda^*}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ is obtained:

Algorithm 2 **A variant algorithm of Algorithm 1 that computes a pair alignment probability matrix $\boldsymbol{P}_{\mathbb{R}}^{\text{PA}}$ and loop accessibility matrices $P_{\mathbb{R}}^{\text{pair},\lambda^*}$ and $P_{\mathbb{R}}^{\text{unp},\lambda}$.**

```
function InOutAlgo*($\mathbb{R}$)
    Compute inside partition functions $\alpha_{ijkl}^{x}$ and $\alpha_{uv}^{y}$ on the dynamic programming described in Supplementary section 1.1
    Compute outside partition functions $\beta_{ijkl}^{x}$ on the dynamic programming described in Supplementary section 1.2
    Compute pair alignment probabilities $p_{ijkl}^{\text{PA}}$ by $p_{ijkl}^{\text{PA}} = \frac{\alpha_{ijkl}^{\text{PA}} \beta_{ijkl}^{\text{PA}}}{\alpha_{NM}^{\text{outer,for}}}$
    Compute loop accessibilities $p_{ij}^{\lambda^*}$ and $p_u^{\lambda}$ on the dynamic programming described in Supplementary section 1.3
    return $\boldsymbol{P}_{\mathbb{R}}^{\text{PA}}$, $P_{\mathbb{R}}^{\text{pair},\lambda^*}$, $P_{\mathbb{R}}^{\text{unp},\lambda}$
```

$$p_{ij}^{\lambda^*}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}}) = \frac{\sum_{\boldsymbol{R}^{\text{sup}}} p_{ij}^{\lambda^*}((\boldsymbol{R}^{\text{tar}}, \boldsymbol{R}^{\text{sup}}))}{|R^{\text{sup}}|}.$$

Likewise, the average accessibility $p_u^{\lambda}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ is gained:

$$p_u^{\lambda}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}}) = \frac{\sum_{\boldsymbol{R}^{\text{sup}}} p_u^{\lambda}((\boldsymbol{R}^{\text{tar}}, \boldsymbol{R}^{\text{sup}}))}{|R^{\text{sup}}|}.$$

Data collection for benchmark

From Rfam, which collects thousands RNA families (54), 1473 RNA families whose reference seed structural alignments had at most 200 columns and that contained at most ten sequences were collected as dataset "origin". Reference single structures were obtained to map the reference seed consensus structure of each RNA family to each sequence on dataset "origin". The obtained set which contains the sequences and their single structures of each RNA family is called test set "unaligned". Reference consensus structures were obtained to leave only the reference seed consensus structure of each RNA family on dataset "origin". The obtained set which contains the sequences and their consensus structure of each RNA family is called test set "aligned".

Competitors for benchmark

TurboFold v6.2, CentroidHomfold v0.0.16, CONTRAfold v2.02, CentroidFold v0.0.16, and RNAfold v2.4.14 (Table 1) were compared to ConsHomfold using their default parameters.

RNAfold RNAfold is the most standard mono-folding and predicts a structure $\boldsymbol{S}_{\boldsymbol{R}}$ to minimize its energy: $\arg\min_{\boldsymbol{S}_{\boldsymbol{R}}} e_{\boldsymbol{S}_{\boldsymbol{R}}}$. (4) This minimization is equivalent to the maximum likelihood prediction $\arg\max_{\boldsymbol{S}_{\boldsymbol{R}}} p_{\boldsymbol{S}_{\boldsymbol{R}}} = \frac{\exp(-e_{\boldsymbol{S}_{\boldsymbol{R}}})}{\sum_{\boldsymbol{S}\in\mathcal{S}_{\boldsymbol{R}}}\exp(-e_{\boldsymbol{S}})}$ where $\mathcal{S}_{\boldsymbol{R}}$ is a set of all possible structures $\boldsymbol{S}_{\boldsymbol{R}}$. (36)

CONTRAfold CONTRAfold is a mono-folding that predicts a structure $\boldsymbol{S}_{\boldsymbol{R}}$ maximizing its expected accuracy

$$2\gamma \sum_{ij|I^{\mathrm{pair}}_{ij}(\boldsymbol{S}_{\boldsymbol{R}})=1} p^{\mathrm{pair}}_{ij}(\boldsymbol{R}) + \sum_{i|I^{\mathrm{unp}}_{i}(\boldsymbol{S}_{\boldsymbol{R}})=1} p^{\mathrm{unp}}_{i}(\boldsymbol{R})$$

where $p^{\mathrm{unp}}_{i}(\boldsymbol{R}) = 1 - \sum_{j:j<i} p^{\mathrm{pair}}_{ji}(\boldsymbol{R}) - \sum_{j:i<j} p^{\mathrm{pair}}_{ij}(\boldsymbol{R})$. (8, 9) This accuracy is based on the counts

$$\begin{cases} TP = \sum_i I^{\mathrm{pair}}_i(\boldsymbol{S}) I^{\mathrm{pair}}_i(\boldsymbol{S}'), TN = \sum_i I^{\mathrm{unp}}_i(\boldsymbol{S}) I^{\mathrm{unp}}_i(\boldsymbol{S}') \\ FP = \sum_i I^{\mathrm{pair}}_i(\boldsymbol{S}) I^{\mathrm{unp}}_i(\boldsymbol{S}'), FN = \sum_i I^{\mathrm{unp}}_i(\boldsymbol{S}) I^{\mathrm{pair}}_i(\boldsymbol{S}') \end{cases}$$

where $\boldsymbol{S}' \in \mathcal{S}_{\boldsymbol{R}}$. The count TP does not mind the pairing partner j of a position i if only the position i is pairing.

CentroidFold CentroidFold is a mono-folding that predicts a structure $\boldsymbol{S}_{\boldsymbol{R}}$ maximizing its expected accuracy $\sum_{ij|I^{\mathrm{pair}}_{ij}(\boldsymbol{S}_{\boldsymbol{R}})=1}[(\gamma+1)p^{\mathrm{pair}}_{ij}(\boldsymbol{R})-1]$. (9) This accuracy is based on the counts

$$\begin{cases} TP = \sum_{ij} I^{\mathrm{pair}}_{ij}(\boldsymbol{S}) I^{\mathrm{pair}}_{ij}(\boldsymbol{S}'), TN = \sum_{ij} I^{\mathrm{unp}}_{ij}(\boldsymbol{S}) I^{\mathrm{unp}}_{ij}(\boldsymbol{S}') \\ FP = \sum_{ij} I^{\mathrm{pair}}_{ij}(\boldsymbol{S}) I^{\mathrm{unp}}_{ij}(\boldsymbol{S}'), FN = \sum_{ij} I^{\mathrm{unp}}_{ij}(\boldsymbol{S}) I^{\mathrm{pair}}_{ij}(\boldsymbol{S}') \end{cases}$$

where $I^{\mathrm{unp}}_{ij}(\boldsymbol{S})$ is 1 if the positions i and j are unpairing in the structure $\boldsymbol{S}$ and 0 otherwise. The counts TP, TN, FP, and FN are biased to negatives since a position i can be pairing with at most one position j and thus most pairs (i,j) are unpairing. This bias becomes remarkable when sequences are long.

TurboFold TurboFold is a hom-folding that iteratively, alternately estimates posterior probabilities on single structure and those on pairwise sequence alignment of homologs. (31) During this estimation (called the *turbo decoding* (55, 56)), the probabilities estimated currently (e.g. on pairwise sequence alignment) are incorporated into those estimated immediately afterwards (e.g. on single structure). After η iterations of the alternate estimation, TurboFold predicts the single structures of the homologs to maximize the expected accuracy based on the posterior probabilities estimated finally by the turbo decoding. From the viewpoint of structural alignment, TurboFold is said to decompose a multiple structural alignment of homologs into its single structures and pairwise sequence alignments on probability. TurboFold predicts a structure $\boldsymbol{S}_{\boldsymbol{R}^{\mathrm{tar}}}$ maximizing its expected accuracy

$$2\gamma \sum_{ij|I^{\mathrm{pair}}_{ij}(\boldsymbol{S}_{\boldsymbol{R}^{\mathrm{tar}}})=1} p^{\mathrm{pair,iter}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) + \sum_{i|I^{\mathrm{unp}}_{i}(\boldsymbol{S}_{\boldsymbol{R}^{\mathrm{tar}}})=1} p^{\mathrm{unp,iter}}_{i}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$$

where $p^{\mathrm{pair,iter}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$ is the average pairing probability of the positions i and j given the sequence $\boldsymbol{R}^{\mathrm{tar}}$ and the sequences R^{sup} estimated by the turbo decoding and $p^{\mathrm{unp,iter}}_{i}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) = 1 - \sum_{j:j<i} p^{\mathrm{pair,iter}}_{ji}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) - \sum_{j:i<j} p^{\mathrm{pair,iter}}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$. In the benchmark of this study, TurboFold was retried with the parameter $\eta = 1$ when this method failed with its default parameters (including the parameter $\eta = 3$).

CentroidHomfold CentroidHomfold is a hom-folding that extends CentroidFold to incorporate homologs and *factorize*s a probability p^{PA}_{ijkl} into independent posterior probabilities:

$$p^{\mathrm{PA}}_{ijkl} \approx p^{\mathrm{PA},\times}_{ijkl} = p^{\mathrm{pair}}_{ij}(\boldsymbol{R}) p^{\mathrm{ali}}_{ik}(\mathbb{R}) p^{\mathrm{ali}}_{jl}(\mathbb{R}) p^{\mathrm{pair}}_{kl}(\boldsymbol{R}')$$

where $p^{\mathrm{ali}}_{ik}(\mathbb{R})$ is the alignment probability of the positions i and k given the pair $\mathbb{R}$ on pairwise sequence alignment. This factorization lets CentroidHomfold avoid the computation of probabilities p^{PA}_{ijkl}. (30) CentroidHomfold connects probabilities $p^{\mathrm{PA},\times}_{ijkl}$ into a single metric via probabilistic consistency transformation as ConsHomfold and then predict the single structures of the homologs using the metrics obtained by this transformation. CentroidHomfold predicts a structure $\boldsymbol{S}_{\boldsymbol{R}^{\mathrm{tar}}}$ maximizing its expected accuracy

$$\sum_{ij|I^{\mathrm{pair}}_{ij}(\boldsymbol{S}_{\boldsymbol{R}^{\mathrm{tar}}})=1} [(\gamma+1) p^{\mathrm{pair},\times}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}}) - 1]$$

where $p^{\mathrm{pair},\times}_{ij}(\boldsymbol{R}^{\mathrm{tar}}, R^{\mathrm{sup}})$ is the average pairing probability of the positions i and j given the sequence $\boldsymbol{R}^{\mathrm{tar}}$ and the sequences R^{sup} obtained by this transformation.

CentroidAlifold v0.0.16, RNAalifold v2.4.14, and PETfold v2.1 (Table 1) were compared to ConsAlifold using their default parameters.

RNAalifold RNAalifold is the most standard ali-folding and predicts a structure $\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}$ minimizing its average energy $\arg\min_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}} e_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}}$ where $e_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}}$ is the average energy of the structure $\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}$. (32) This minimization is equivalent to the maximum likelihood prediction $\arg\max_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}} p_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}} = \frac{\exp(-e_{\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}})}{\sum_{\boldsymbol{S}^*\in\mathcal{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}}\exp(-e_{\boldsymbol{S}^*})}$ where $\mathcal{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}$ is a set of all possible structures $\boldsymbol{S}_{\boldsymbol{A}_{\mathbb{R}\mathrm{hom}}}$. (36)

CentroidAlifold CentroidAlifold is an ali-folding that predicts a structure $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$ maximizing its expected accuracy $\sum_{ij|I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}})=1}[(\gamma+1)p_{ij}^{\text{mix}*}(\boldsymbol{A}_{R^{\text{hom}}})-1]$ where $I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}})$ returns 1 if the columns i and j are pairing in the structure $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$ and 0 otherwise and $p_{ij}^{\text{mix}*}(\boldsymbol{A}_{R^{\text{hom}}})=\tau\frac{\sum_{\boldsymbol{R}}p_{i^*j^*}^{\text{pair}}(\boldsymbol{R})}{|R^{\text{hom}}|}+(1-\tau)p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$. (34)

PETfold PETfold is an ali-folding that predicts a structure $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$ to maximize its expected accuracy

$$\sum_{ij|I_{ij}^{\text{pair}}(\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}})=1} p_{ij}^{\text{mix}**}(\boldsymbol{A}_{R^{\text{hom}}})$$
$$+\gamma^{-1}\sum_{i|I_{i}^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}})=1} p_{i}^{\text{mix}**}(\boldsymbol{A}_{R^{\text{hom}}})$$

where $p_{ij}^{\text{mix}**}(\boldsymbol{A}_{R^{\text{hom}}})=\kappa\frac{\sum_{\boldsymbol{R}}p_{i^*j^*}^{\text{pair}}(\boldsymbol{R})}{|R^{\text{hom}*}|}+p_{ij}^{\text{pair}}(\boldsymbol{A}_{R^{\text{hom}}})$, $0<\kappa$, $I_{i}^{\text{unp}}(\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}})$ returns 1 if the column i is unpairing in the structure $\boldsymbol{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$ and 0 otherwise, and $p_{i}^{\text{mix}**}(\boldsymbol{A}_{R^{\text{hom}}})=\kappa\frac{\sum_{\boldsymbol{R}}p_{i^*}^{\text{unp}}(\boldsymbol{R})}{|R^{\text{hom}*}|}+p_{i}^{\text{unp}}(\boldsymbol{A}_{R^{\text{hom}}})$. (33) CentroidAlifold and PETFold become equivalent to McCaskill-MEA (57) when $\tau=1$ and $\kappa\rightarrow\infty$ except their count configuration, respectively.

CapR

CapR v1.1.1 was compared to ConsHomfold and ConsAlifold. CapR computes loop accessibilities

$$p_u^{\lambda^{**}}(\boldsymbol{R})=\frac{\sum_{\boldsymbol{S}|I_u^{\lambda^{**}}(\boldsymbol{S})=1}\exp(-e_{\boldsymbol{S}})}{\sum_{\boldsymbol{S}}\exp(-e_{\boldsymbol{S}})}$$

where $\lambda^{**}\in\{1,\text{stem},\text{bulge},\text{interior},\text{multi},\text{outer}\}$ and $I_u^{\lambda^{**}}(\boldsymbol{S})$ returns 1 if the position u is accessible from a λ^{**}-loop in the structure $\boldsymbol{S}$. (39) Here, the position u is said to be in a stem loop if the position u closes a stacking or is accessible from it. Also, CapR changes the definition of a ($\lambda^{**}\neq$stem)-loop to exclude pairing positions in the ($\lambda^{**}\neq$stem)-loop from this definition. To enable genome-wide analysis, this method considers all possible local structures $\boldsymbol{S}$ imposing $|j-i|\leq W$ on any pairing pairs (i,j) where W is the *maximum span* of pairings, which regulates the structures $\boldsymbol{S}$ (43). In this study, the span $W=200$ was used though the length of the sequence applied CapR to was less than 200, i.e. this method took all possible structures $\boldsymbol{S}$ into account. CapR does not incorporate support homologs of a target homolog.

Metrics for prediction accuracy

Positive predictive value (= PPV), sensitivity, false positive rate (= FPR), the F1 score, and the Matthews correlation coefficient (= MCC) are calculated from the numbers of true and false positives and negatives:

$$\begin{cases} PPV=\frac{TP}{TP+FP},SENS=\frac{TP}{TP+FN},FPR=\frac{FP}{TN+FP} \\ F1=2\frac{PPV\times SENS}{PPV+SENS} \\ MCC=\frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \end{cases}.$$

For single structures, the counts TP, TN, FP, and FN are configured as

$$\begin{cases} TP=\sum_{ij}I_{ij}^{\text{pair}}(\boldsymbol{S})I_{ij}^{\text{pair}}(\boldsymbol{S}'),TN=\sum_{i}I_{i}^{\text{unp}}(\boldsymbol{S})I_{i}^{\text{unp}}(\boldsymbol{S}') \\ FP=\sum_{i}I_{i}^{\text{pair}}(\boldsymbol{S})I_{i}^{\text{unp}}(\boldsymbol{S}'),FN=\sum_{i}I_{i}^{\text{unp}}(\boldsymbol{S})I_{i}^{\text{pair}}(\boldsymbol{S}') \end{cases}.$$

For consensus structures, the counts TP, TN, FP, and FN are configured in two ways as

$$\begin{cases} TP=\sum_{ij}I_{ij}^{\text{pair}}(\boldsymbol{S}^*)I_{ij}^{\text{pair}}(\boldsymbol{S}^{*\prime}) \\ TN=\sum_{i}I_{i}^{\text{unp}}(\boldsymbol{S}^*)I_{i}^{\text{unp}}(\boldsymbol{S}^{*\prime}) \\ FP=\sum_{i}I_{i}^{\text{pair}}(\boldsymbol{S}^*)I_{i}^{\text{unp}}(\boldsymbol{S}^{*\prime}) \\ FN=\sum_{i}I_{i}^{\text{unp}}(\boldsymbol{S}^*)I_{i}^{\text{pair}}(\boldsymbol{S}^{*\prime}) \end{cases}$$

(called *columnwise counts*) where $\boldsymbol{S}^{*\prime}\in\mathcal{S}_{\boldsymbol{A}_{R^{\text{hom}}}}$ and

$$\begin{cases} TP=\sum_{ijc}I_{i^*j^*}^{\text{pair}}(\boldsymbol{S}_c^*)I_{i^*j^*}^{\text{pair}}(\boldsymbol{S}_c^{*\prime}) \\ TN=\sum_{ic}I_{i^*}^{\text{unp}}(\boldsymbol{S}_c^*)I_{i^*}^{\text{unp}}(\boldsymbol{S}_c^{*\prime}) \\ FP=\sum_{ic}I_{i^*}^{\text{pair}}(\boldsymbol{S}_c^*)I_{i^*}^{\text{unp}}(\boldsymbol{S}_c^{*\prime}) \\ FN=\sum_{ic}I_{i^*}^{\text{unp}}(\boldsymbol{S}_c^*)I_{i^*}^{\text{pair}}(\boldsymbol{S}_c^{*\prime}) \end{cases}$$

(called *mapwise counts*) where $\boldsymbol{S}_c^*$ and $\boldsymbol{S}_c^{*\prime}$ are the single structures obtained to map the structures $\boldsymbol{S}^*$ and $\boldsymbol{S}^{*\prime}$ to the c-th sequence, respectively.

Implementations and benchmark environments

ConsHomfold and ConsAlifold implemented in Rust employ multi-threading to give their users more efficient computing. Probabilities and partition functions are computed under the log scale using the logsumexp trick $\log\sum_a\exp x_a=\log\sum_a\exp(x_a-\max_a(x_a))+\max_a(x_a)$, which mitigates the undesirable effect of extremely large and small values (e.g. the overflow and underflow of floating point values), in these methods where x_a is a real number. Sparse data structures were implemented by `FxHashMap` (the fastest, memory-efficient hash table in Rust to our best knowledge) provided by https://github.com/Amanieu/hashbrown. The implementation choice of these structures is critical because the efficiency of these structures dominates the entire running time and memory usage of the methods. In this study, the methods used the parameters

$$\delta^{\text{gap}}=\begin{cases} |N-M|+1 & \text{(External loop)} \\ \max(\min(|N-M|+1,20),2) & \text{(Internal loop)} \end{cases}$$

, $\epsilon=0.005$, and $\tau=0.5$ (used as the default value of the parameter τ by CentroidAlifold (34)). For the

Table 1. The profile and benchmark running time of methods that predict structures.

Method	Folding type	Posterior probability type	Running time	Time complexity
ConsHomfold	Hom-folding	Average on sparse pairwise structural alignment	2.95 m (Turner) 2.01 m (Posterior)	$O(R^2L^2+RL^3)$
TurboFold	Hom-folding	Average on iterative, alternate pairwise structural alignment	4.48 m	$O(R^2L^2+RL^3)$
CentroidHomfold	Hom-folding	Average on factorized pairwise structural alignment	32.4 s	$O(R^2L^3)$
CONTRAfold	Mono-folding	Single structure	8.24 s	$O(RL^3)$
CentroidFold	Mono-folding	Single structure	5.76 s	$O(RL^3)$
RNAfold	Mono-folding	N/A (Free energy minimization)	2.02 s	$O(RL^3)$
ConsAlifold	Ali-folding	Average on sparse pairwise structural alignment + consensus structure	5.23 m	$O(R^2L^2+RL^3)$
CentroidAlifold	Ali-folding	Average on single structure + consensus structure	7.19 s	$O(RL^3)$
PETfold	Ali-folding	Average on single structure + consensus structure	6.29 s	$O(RL^3)$
RNAalifold	Ali-folding	N/A (Average free energy minimization)	1.26 s	$O(RL^3)$

The third column shows the type of posterior probabilities used to predict structures. The forth column represents the benchmark running time of the methods that predict single and consensus structures on test sets "unaligned" and "aligned", respectively. For the γ-dependent methods (other than RNAfold and RNAalifold), running time was measured at the parameter $\gamma = 1$. For the ali-foldings, running time was measured using ProbCons v1.12 (51) to generate input sequence alignments of the ali-foldings. For ConsHomfold, running time in the case where Turner's model is replaced with the posterior model was also measured. All the methods were executed utilizing `multiprocessing.Pool` (https://docs.python.org/3/library/multiprocessing.html), a popular Python package that provides running functions under multi-processing, to relieve the large volume of test sets "unaligned" and "aligned". Each of ConsHomfold and ConsAlifold was run to combine their multi-threading and this multi-processing configuring eight processes each to be assigned to eight threads. On the other hand, each of the methods, other than these methods, was performed utilizing 64 processes though TurboFold supports multi-threading (with the low saturation of thread utilization on test set "unaligned" resulting in longer running time). The rightmost column shows rough time complexities of these methods. R and L are the number and the maximum sequence length of homologs, respectively.

comparison with Turner's model, the posterior model was also implemented in ConsHomfold to use the scoring $e_{\boldsymbol{S}} = -\sum_{ij|I^{\text{pair}}_{ij}(\boldsymbol{S})=1} \ln p^{\text{pair}}_{ij}(\boldsymbol{R})$. For the benchmark of running time, programs were run on a computer composed of an "Intel Xeon CPU" CPU with 64 threads and a clock rate of 2.30 GHz and 240 GB of RAM. Otherwise, programs were run on a computer composed of an "AMD EPYC 7501" CPU with 64 threads and a clock rate of 2 GHz and 128 GB of RAM.

RESULTS

Benchmark of ConsHomfold and ConsAlifold with their competitors

ConsHomfold and ConsAlifold perform the best trade-off of the metrics PPV, $SENS$, and FPR (Figure 5a) among the state-of-the-art methods that predict structures while requiring comparable running time (Table 1). Also, ConsAlifold demonstrates better transitions of the metrics $F1$ and MCC than CentroidAlifold and PETfold. (Figure 5b3–4) However, PETfold does not drop the metrics $F1$ and MCC within the range $-7 \leq \log_2 \gamma \leq -4$ compared to ConsAlifold and CentroidAlifold. (Figure 5b3–4) The above accuracy performances on consensus structure are also confirmed when columnwise counts were used instead of those mapwise. (Figure 6) RNAalifold shows competitive accuracy on all the metrics PPV, $SENS$, FPR, $F1$, and MCC, except the metric MCC when ProbCons was used. (Figure 5ab, Figure 6) The columnwise counts suffer from the quality of input sequence alignments compared to those mapwise. (Figure 5ab, Figure 6)

TurboFold displays a superior transition of the metric $F1$ to the other methods that predict single structures whereas competing with ConsHomfold on the metric MCC. (Figure 5b1–2) As expected, ConsHomfold with the posterior model exerts significantly less predictive power than that with Turner's model across all the metrics PPV, $SENS$, FPR, $F1$, and MCC (Figure 5ab), though the former records the faster running time than the latter (Table 1).

Comparison between conventional and proposed posterior probabilities of example ncRNA: tRNA

The conventional probabilities $p^{\text{pair}}_{ij}(\boldsymbol{R})$ and accessibilities $p^{\lambda **}_{u}(\boldsymbol{R})$ of a tRNA are contrasted to its proposed probabilities $p^{\text{pair}}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and accessibilities $p^{\lambda *}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $p^{\lambda}_{u}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ in terms of homology consideration. (Figure 5c) Conserved and nonconserved pairings are clarified through the gaps between the probabilities $p^{\text{pair}}_{ij}(\boldsymbol{R})$ and $p^{\text{pair}}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ of this RNA. (Figure 5c1)

Proposed loop accessibilities of example ncRNAs: tRNA and microRNA

The single (cloverleaf) structure of a tRNA predicted by ConsHomfold is decorated with the proposed accessibilities $p^{\lambda *}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $p^{\lambda}_{u}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$. (Figure 5d1) The reliability of each pairing and unpairing in this structure can be assessed through the accessibilities $p^{\lambda *}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $p^{\lambda}_{u}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$.

The single structure of pri-miR-16-2, which is one of primary microRNAs (= pri-miRNAs), modeled to investigate the mechanism of human Drosha and DGCR8 (58) that cleaves metazoan pri-miRNAs in order to generate their mature miRNAs through Dicer (59) using Cryo-EM (60) is compared to that predicted by ConsHomfold. (Figure 5d2–3) The outlines of these structures agree though stems not found in the model structure were mispredicted by

ConsHomfold. (Figure 5d2–3) The reliability of these stems is as high as the other parts of the predicted structure. (Figure 5d3) Accessibilities $p_{ij}^{\lambda^*}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $p_u^{\lambda}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ can help biologists validate model single structures as in the presented model structure. The entire reliability of the consensus structure of pri-miR-16-2 and its homologs predicted by MAFFT and ConsAlifold is lower than that of the structure predicted by ConsHomfold though the contours of these structures coincide. (Figure 5d3–4)

DISCUSSION

Probabilistic, consistent consideration is possibly simple answer for further improvement of prediction accuracy

Considering consistently pairwise structural alignments on structure prediction is the prospect to improve further this prediction, though the effectiveness of this consideration has not been focused on. ConsHomfold and ConsAlifold demonstrate that this consideration improves the prediction accuracy on this prediction. This improvement is possibly successful in resolving other prediction problems, such as sequence alignment predictions and certainly structural alignment predictions. It is likely that CentroidAlign (49) and MAFFT (61), which can predict sequence alignments considering structural alignments via their decomposition, will become universally accepted for the adoption of the consistent consideration, instead of this decomposed consideration. DAFS, which can predict structural alignments with this decomposed consideration (28), will be also enhanced by this adoption.

Turner's model is also effective in structural alignments

A majority of conventional methods that predict structural alignments use the posterior model to score the single structures in structural alignments being computed, because aligning more than two sequences with Turner's model (expected to display more predictive power than the posterior model) is computationally complicated. However, Turner's model can be used avoiding this complication to predict structural alignments of the maximum expected accuracy principle as DAFS (28) where probabilistic consistency transformation, which decomposes (NP-complete) multiple structural alignments to be considered into pairwise structural alignments, is available. It is valuable that rebuilding popular methods that predict structural alignments, such as LocARNA and SPARSE, in this principle on Turner's model, because ConsHomfold proves that this model is superior to the posterior model in terms of prediction accuracy.

Extending ConsHomfold and ConsAlifold to more sophisticated prediction of structures

At present, ConsHomfold and ConsAlifold cannot predict pseudoknotted structures and those enhanced by single structure probing data. Augmenting these methods to also predict these structures is a straightforward future task that can be explored in the future. An algorithm that computes posterior pairing probabilities on pseudoknotted single structure (62, 63) cannot be simply extended to compute those on pseudoknotted pairwise structural alignment, because this algorithm demands the $O(N^5)$ running time and the $O(N^4)$ memory usage (41) whereas McCaskill's algorithm demands the $O(N^3)$ running time and the $O(N^2)$ memory usage (42). IPknot offers a reasonable way to use McCaskill's algorithm for the prediction of pseudoknotted structures. (41) More specifically, this method decomposes a pseudoknotted structure into its pseudoknot-free structures and then maximize the expected accuracy of these structures based on posterior pairing probabilities on pseudoknot-free single structure computed by McCaskill's algorithm. Importing the methodology of IPknot and replacing McCaskill's algorithm with Algorithm 1 and the proposed probabilistic consistency transformation in this methodology, ConsHomfold and ConsAlifold will predict pseudoknotted structures.

RNAfold and RNAalifold can predict structures limited by SHAPE reactivity data. (64) TurboFold can conduct its hom-folding utilizing this data. (65) Many methods that predict structures incorporating the data including the above methods add a new term called *SHAPE-origin pseudo-free energy* to the free energy of a structure. (64, 65, 66, 67, 68, 69) This pseudo-energy is obtained to convert the data into the pseudo-free energy per unpairing position and then sum this energy across all unpairing positions. ConsHomfold and ConsAlifold will predict structures constrained by the data to introduce SHAPE-origin pseudo-free energy.

Proposed loop accessibilities are worth using in field of RNA-binding protein

CapR revealed that several RNA-binding proteins bind their target RNAs by recognizing the loop types of the binding regions in the RNAs (71, 72, 73) through CLIP-seq data (74). (39) The proposed loop accessibilities in this study can be used to analyze these proteins and the target RNAs including their structures more precisely. RNAcontext (75) and RCK (76) demonstrated that loop accessibilities improve the prediction accuracy of predicting the binding of the proteins to their candidate target RNAs compared to the conventional methods that do not use these accessibilities, such as MEMERIS (77) and MatrixREDUCE (78). The proposed accessibilities have the potential to ameliorate the prediction accuracy of RNAcontext and RCK.

CONCLUSION

In this study, the below approaches have been proposed:

- a hom-folding (ConsHomfold) and an ali-folding (ConsAlifold) that consider sparse pairwise structural alignments on their probability distributions
- a quadratic homolog-aware algorithm to compute different kinds of average posterior probabilities on sparse pairwise structural alignment, some of which are used to conduct these foldings, and the others of which are helpful in analyzing RNAs and their structures.

ConsHomfold and ConsAlifold have represented better trade-off between PPV, sensitivity, and FPR on Rfam-based benchmarks than other state-of-the-art methods that predict structures including those that consider stochastically

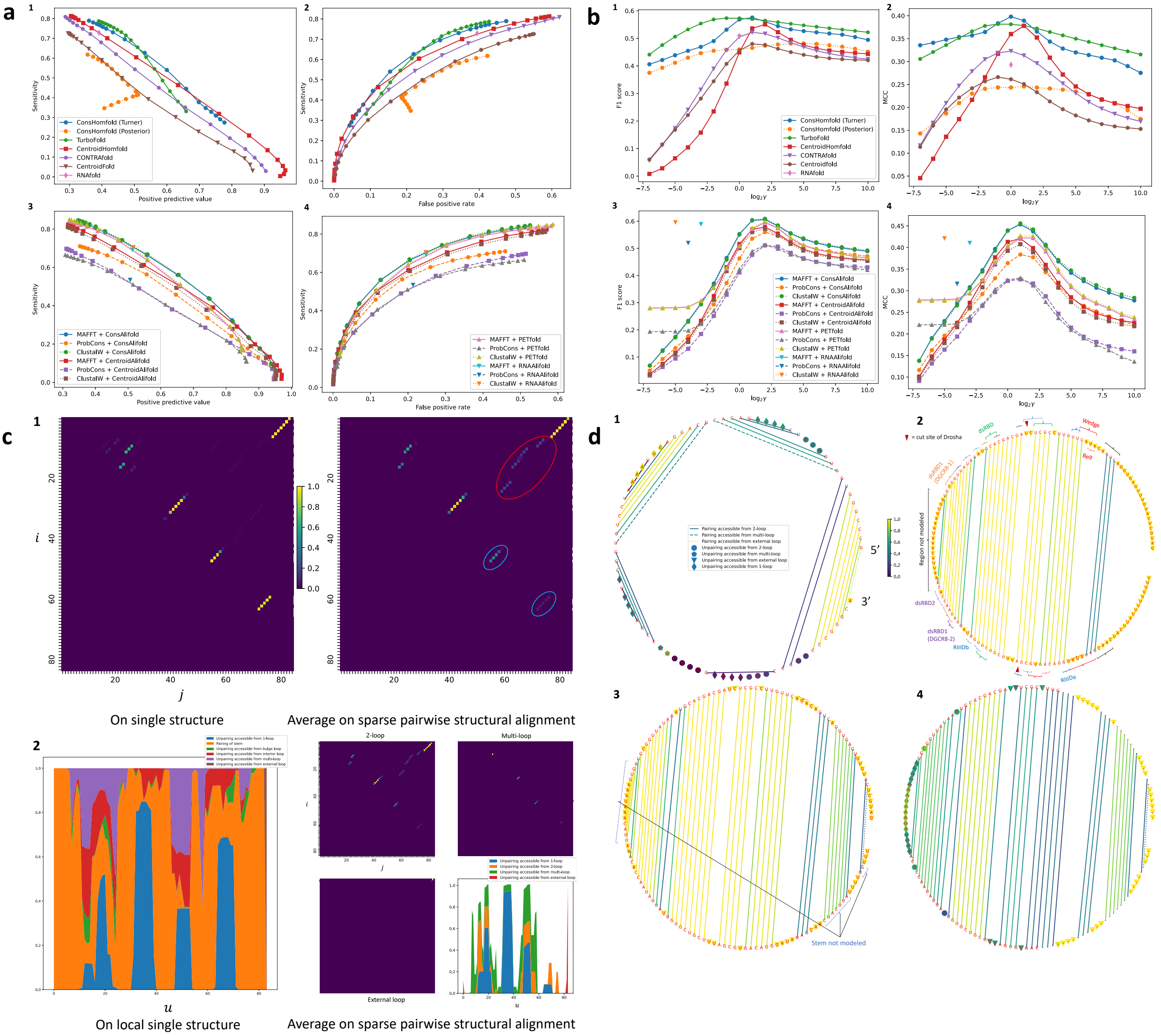


Figure 5. **(a) The trade-off curves composed of (1, 3) pairs $(PPV, SENS)$ and (2, 4) pairs $(FPR, SENS)$ at each parameter $\gamma = 2^g : g \in \{-7, \ldots, 10\}$, respectively.** These curves are unavailable for RNAfold and RNAalifold because they do not depend on the parameter γ. The methods that predict (1, 2) single and (3, 4) consensus structures were measured using test sets "unaligned" and "aligned", respectively. Mapwise counts were used for consensus structures. Turner's and the posterior models were compared on ConsHomfold. The MAFFT v7.470 (61), ProbCons, and ClustalW v2.1 (70) were used with their default parameters to generate input sequence alignments of the ali-foldings. **(b) The transitions of the metrics (1, 3) $F1$ and (2, 4) MCC across parameters $g = \log_2\gamma$, respectively.** The used test sets, the comparison between Turner's and posterior models, counts for consensus structures, and input alignment settings are the same as Figure 5a. For RNAfold, the metrics $F1$ and MCC are plotted at the parameter $g = 0$. For RNAalifold, the metrics $F1$ and MCC computed with MAFFT, ProbCons, and ClustalW are plotted at the parameters $g = -3$, $g = -4$, and $g = -5$, respectively. **(c) The various probability distributions of a tRNA.** (1) The comparison between probabilities (left) $p^{\text{pair}}_{ij}(\boldsymbol{R})$ and (right) $p^{\text{pair}}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$. Probabilities $p^{\text{pair}}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ are supported by other five tRNAs. The red and blue circles display conserved and nonconserved pairings, respectively. (2) The comparison between (left) accessibilities $p^{\lambda^{**}}_{u}(\boldsymbol{R})$ and (right) accessibilities $p^{\lambda^*}_{ij}$ and p^{λ}_{u}. **(d) The structures of a tRNA and pri-miR-16-2 color-coded by accessibilities $p^{\lambda^*}_{ij}$ and p^{λ}_{u}.** (1) The single structure of a tRNA supported by other five tRNAs predicted by ConsHomfold at the parameter $\gamma = 2^{10} = 1024$. The single structures of pri-miR-16-2 (2) modeled in the study that determined the Cryo-EM structure of human Drosha and DGCR8 binding this RNA (60) and (3) predicted by ConsHomfold at the parameter $\gamma = 2^3 = 8$ using the ten homologs of this RNA. Each pairing and unpairing are color-coded by maximum accessibilities $\max_{\lambda^*} p^{\lambda^*}_{ij}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$ and $\max_{\lambda} p^{\lambda}_{u}(\boldsymbol{R}^{\text{tar}}, R^{\text{sup}})$. The sites of this model structure bound by components of Drosha and DGCR8 retrieved from the study are also displayed. Belt, Wedge, dsRBD, RIIIDa, and RIIIDb are components of Drosha. dsRBD1 and dsRBD2 are components of DGCR8. DGCR8-1 and DGCR8-2 are different copies of DGCR8. (4) The consensus structure predicted by ConsAlifold at the parameter $\gamma = 2^3 = 8$. This structure is of the sequence alignment among pri-miR-16-2 and the homologs of this RNA predicted by MAFFT with its default parameters. Each column of this alignment is represented by the most frequent character including a gap in this column. Each pairing and unpairing are color-coded by maximum accessibilities $\frac{\max_{\lambda^*} \sum_{\boldsymbol{R}} p^{\lambda^*}_{i^* j^*}(\boldsymbol{R}, R^{\text{hom}} \setminus \boldsymbol{R})}{|R^{\text{hom}*}|}$ and $\frac{\max_{\lambda} \sum_{\boldsymbol{R}} p^{\lambda}_{u^*}(\boldsymbol{R}, R^{\text{hom}} \setminus \boldsymbol{R})}{|R^{\text{hom}*}|}$ where u^* is the position on the sequence $\boldsymbol{R}$ mapped to the column u in this alignment.

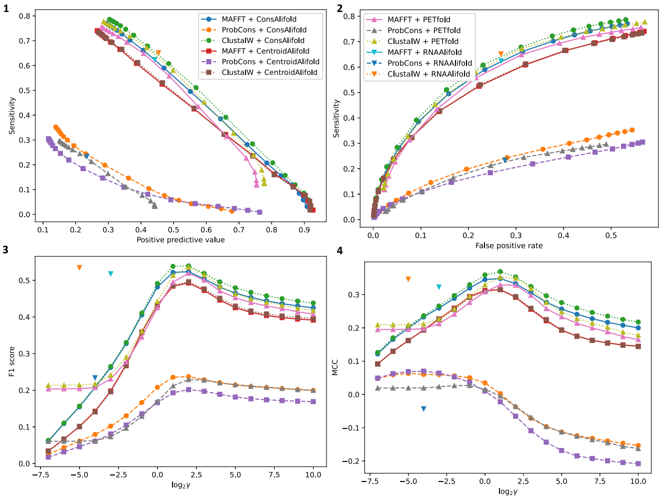


Figure 6. The trade-off curves composed of (1) pairs $(PPV, SENS)$ and (2) pairs $(FPR, SENS)$ at each parameter $\gamma = 2^g$ on columnwise counts, respectively. (b) The transitions of the metrics (3) $F1$ and (4) MCC across parameters $g = \log_2 \gamma$ on columnwise counts, respectively. The ali-foldings were measured using test set "aligned". The input alignment settings are the same as Figure 5a.

pairwise structural alignments to decompose them into their independent components. ConsAlifold has also displayed superior transitions of the F1 score and the MCC to the conventional methods of this method. From these results, It has been concluded that the consistent, probabilistic consideration of sparse pairwise structural alignments improves the prediction accuracy of structures.

ConsHomfold and ConsAlifold demand the reasonable running time supported by the compared time complexities of the benchmarked methods. It has been confirmed that Turner's model, which is the most popular to score single structures, adopted in this study significantly more fits to score structural alignments than the posterior model which is used by many conventional methods that score structural alignments. Turner's model possibly raises the prediction accuracy of methods that predict structural alignments using the posterior model.

Conventional loop accessibilities on single structure succeeded in the analysis and prediction of RNA-binding proteins and the *in silico* RNA aptamer selection on HT-SELEX (79, 80). There is a possibility that the loop accessibilities proposed in this study have a broader range of impactful applications, as with the conventional accessibilities.

DATA AVAILABILITY

ConsHomfold, ConsAlifold, the data used in this study, and Python scripts to generate the figures and tables in this study are freely available at https://github.com/heartsh/conshomfold and https://github.com/heartsh/consalifold managed by M.T.

SUPPLEMENTARY DATA

Supplementary Data is available from NAR Online.

FUNDING

This work was not supported by any funding.

Conflict of interest statement. None declared.

ACKNOWLEDGEMENTS

We would like to thank Dr. Kiyoshi Asai, Dr. Martin Frith, and the members of their laboratories for discussing this study with us for years. Also, we offer our thanks to Dr. Risa Kawaguchi for sharing valuable information about the scaling and logsumexp methods on exact probability computations with us. We are appreciate being advised on work closely related to this study, LocARNA-P and PETfold, by kind anonymous reviewers. We would like to thank Editage (www.editage.com) for English language editing. Most computations were performed on the NIG supercomputer at the ROIS National Institute of Genetics.

AUTHOR CONTRIBUTIONS

M.T. conceived the proposed method, implemented it in the presented programs, experimented with them, and wrote this paper.

REFERENCES

1. Klein,R.J. and Eddy,S.R. (2003) RSEARCH: Finding homologs of single structured RNA sequences. *BMC Bioinf.*, **4**, 44.
2. Washietl,S., Hofacker,I.L. and Stadler,P.F. (2005) Fast and reliable prediction of noncoding RNAs. *Proc. Natl. Acad. Sci. U. S. A.*, **102**, 2454–2459.
3. Havgaard,J.H., Lyngs,R.B., Stormo,G.D. and Gorodkin,J. (2005) Pairwise local structural alignment of RNA sequences with sequence similarity less than 40%. *Bioinformatics*, **21**, 1815–1824.
4. Lorenz,R., Bernhart,S.H., Hner zu Siederdissen,C., Tafer,H., Flamm,C., Stadler,P.F. and Hofacker,I.L. (2011) ViennaRNA Package 2.0. *Algorithms Mol. Biol.*, **6**, 26.
5. Zuker,M. (2003) Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, **31**, 3406–3415.
6. Knudsen,B. and Hein,J. (2003) Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Res.*, **31**, 3423–3428.
7. Ding,Y., Chan,C.Y. and Lawrence,C.E. (2004) Sfold web server for statistical folding and rational design of nucleic acids. *Nucleic Acids Res.*, **32**, W135–W141.
8. Do,C.B., Woods,D.A. and Batzoglou,S. (2006) CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, **22**, e90–e98.
9. Hamada,M., Kiryu,H., Sato,K., Mituyama,T. and Asai,K. (2009) Prediction of RNA secondary structure using generalized centroid estimators. *Bioinformatics*, **25**, 465–473.
10. Armache,J.P., Jarasch,A., Anger,A.M., Villa,E., Becker,T., Bhushan,S., Jossinet,F., Habeck,M., Dindar,G., Franckenberg,S., Marquez,V., Mielke,T., Thomm,M., Berninghausen,O., Beatrix,B., Sding,J., Westhof,E., Wilson,D.N. and Beckmann,R. (2010) Cryo-EM structure and rRNA model of a translating eukaryotic 80S ribosome at 5.5-resolution. *Proc. Natl. Acad. Sci. U. S. A.*, **107**, 19748–19753.
11. Spitale,R.C., Flynn,R.A., Zhang,Q.C., Crisalli,P., Lee,B., Jung,J., Kuchelmeister,H.Y., Batista,P.J., Torre,E.A., Kool,E.T. and Chang,H.Y. (2015) Structural imprints in vivo decode RNA regulatory mechanisms. *Nature*, **519**, 486–490.
12. Siegfried,N.A., Busan,S., Rice,G.M., Nelson,J.A.E. and Weeks,K.M. (2014) RNA motif discovery by SHAPE and mutational profiling (SHAPE-MaP). *Nat. Methods*, **11**, 959–965.
13. Kertesz,M., Wan,Y., Mazor,E., Rinn,J.L., Nutter,R.C., Chang,H.Y. and Segal,E. (2010) Genome-wide measurement of RNA secondary structure in yeast. *Nature*, **467**, 103–107.

14. Lu,Z., Zhang,Q.C., Lee,B., Flynn,R.A., Smith,M.A., Robinson,J.T., Davidovich,C., Gooding,A.R., Goodrich,K.J., Mattick,J.S., Mesirov,J.P., Cech,T.R. and Chang,H.Y. (2016) RNA Duplex Map in Living Cells Reveals Higher-Order Transcriptome Structure. *Cell*, **165**, 1267–1279.
15. Mathews,D.H., Disney,M.D., Childs,J.L., Schroeder,S.J., Zuker,M. and Turner,D.H. (2004) Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc. Natl. Acad. Sci. U. S. A.*, **101**, 7287–7292.
16. Watts,J.M., Dang,K.K., Gorelick,R.J., Leonard,C.W., Bess Jr,J.W., Swanstrom,R., Burch,C.L., Weeks,K.M. (2009) Architecture and secondary structure of an entire HIV-1 RNA genome. *Nature*, **460**, 711–716.
17. Wu,Y., Shi,B., Ding,X., Liu,T., Hu,X., Yip,K.Y., Yang,Z.R., Mathews,D.H. and Lu,Z.J. (2015) Improved prediction of RNA secondary structure by integrating the free energy model with restraints derived from experimental probing data. *Nucleic Acids Res.*, **43**, 7247–7259.
18. Sankoff,D. (1985) Simultaneous Solution of the RNA Folding, Alignment and Protosequence Problems. *SIAM J. Appl. Math.*, **45**, 810–825.
19. Sato,K. and Sakakibara,Y. (2005) RNA secondary structural alignment with conditional random fields. *Bioinformatics*, **21**, ii237–ii242.
20. Feng,D. and Doolittle,R.F. (1987) Progressive sequence alignment as a prerequisitetto correct phylogenetic trees. *J. Mol. Evol.*, **25**, 351–360.
21. Hofacker,I.L., Bernhart,S.H.F. and Stadler,P.F. (2004) Alignment of RNA base pairing probability matrices. *Bioinformatics*, **20**, 2222–2227.
22. Havgaard,J.H., Torarinsson,E. and Gorodkin,J. (2007) Fast Pairwise Structural RNA Alignments by Pruning of the Dynamical Programming Matrix. *PLoS Comput. Biol.*, **3**, e193.
23. Torarinsson,E., Havgaard,J.H. and Gorodkin,J. (2007) Multiple structural alignment and clustering of RNA sequences. *Bioinformatics*, **23**, 926–932.
24. Kiryu,H., Tabei,Y., Kin,T. and Asai,K. (2007) Murlet: a practical multiple alignment tool for structural RNA sequences. *Bioinformatics*, **23**, 1588–1598.
25. Tabei,Y., Kiryu,H., Kin,T. and Asai,K. (2008) A fast structural multiple alignment method for long RNA sequences. *BMC Bioinf.*, **9**, 33.
26. Smith,C., Heyne,S., Richter,A.S., Will,S. and Backofen,R. (2010) Freiburg RNA Tools: a web server integrating INTARNA, EXPARNA and LOCARNA. *Nucleic Acids Res.*, **38**, W373–W377.
27. Do,C.B., Foo,C. and Batzoglou,S. (2008) A max-margin model for efficient simultaneous alignment and folding of RNA sequences. *Bioinformatics*, **24**, i68–i76.
28. Sato,K., Kato,Y., Akutsu,T., Asai,K. and Sakakibara,Y. (2012) DAFS: simultaneous aligning and folding of RNA sequences via dual decomposition. *Bioinformatics*, **28**, 3218–3224.
29. Will,S., Otto,C., Miladi,M., Mhl,M. and Backofen,R. (2015) SPARSE: quadratic time simultaneous alignment and folding of RNAs without sequence-based heuristics. *Bioinformatics*, **31**, 2489–2496.
30. Hamada,M., Sato,K., Kiryu,H., Mituyama,T. and Asai,K. (2009) Predictions of RNA secondary structure by combining homologous sequence information. *Bioinformatics*, **25**, i330–i338.
31. Tan,Z., Fu,Y., Sharma,G. and Mathews,D.H. TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Res.*, **45**, 11570–11581.
32. Bernhart,S.H., Hofacker,I.L., Will,S., Gruber,A.R. and Stadler,P.F. (2008) RNAalifold: improved consensus structure prediction for RNA alignments. *BMC Bioinf.*, **9**, 474.
33. Seemann,S.E., Gorodkin,J. and Backofen,R. (2008) Unifying evolutionary and thermodynamic information for RNA folding of multiple alignments. *Nucleic Acids Res.*, **36**, 6355–6362.
34. Hamada,M., Sato,K. and Asai,K. (2011) Improving the accuracy of predicting secondary structure for aligned RNA sequences. *Nucleic Acids Res.*, **39**, 393–402.
35. Ding,Y., Chi,Y.C. and Lawrence,C.E. (2005) RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble. *RNA*, **11**, 1157–1166.
36. Hamada,M., Kiryu,H., Iwasaki,W. and Asai,K. (2011) Generalized Centroid Estimators in Bioinformatics. *PLoS One*, **6**, e16450.
37. Will,S., Joshi,T., Hofacker,I.L., Stadler,P.F. and Backofen,R. (2012) LocARNA-P: Accurate boundary prediction and improved detection of structural RNAs. *RNA*, **18**, 900–914.
38. Turner,D.H. and Mathews,D.H. (2010) NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.*, **38**, D280–D282.
39. Fukunaga,T., Ozaki,H., Terai,G., Asai,K., Iwasaki,W. and Kiryu,H. (2014) CapR: revealing structural specificities of RNA-binding protein target recognition using CLIP-seq data. *Genome Biol.*, **15**, R16.
40. Bradley,R.K., Roberts,A., Smoot,M., Juvekar,S., Do,J., Dewey,C., Holmes,I. and Pachter,L. (2009) Fast Statistical Alignment. *PLoS Comput Biol.*, **5**, e1000392.
41. Sato,K., Kato,Y., Hamada,M., Akutsu,T. and Asai,K. (2011) IPknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming. *Bioinformatics*, **27**, i85–i93.
42. McCaskill,J.S. (1990) The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, **29**, 1105–1119.
43. Kiryu,H., Kin,T. and Asai,K. (2008) Rfold: an exact algorithm for computing local base pairing probabilities. *Bioinformatics*, **24**, 367–373.
44. Hofacker,I.L., Priwitzer,B., Stadler,P.F. (2004) Prediction of locally stable RNA secondary structures for genome-wide surveys. *Bioinformatics*, **20**, 186–190.
45. Bernhart,S.H., Hofacker,I.L. and Stadler,P.F. (2006) Local RNA base pairing probabilities in large sequences. *Bioinformatics*, **22**, 614–615.
46. Do,C.B., Gross,S.S. and Batzoglou,S. (2006) *CONTRAlign: Discriminative Training for Protein Sequence Alignment*. In Proceedings of the Tenth Annual International Conference on Computational Molecular Biology, RECOMB
47. Durbin,R., Eddy,S., Krogh,A. and Mitchison,G. (1998) *Biological sequence analysis*. Cambridge University press, Cambridge, England.
48. Dowell,R.D. and Eddy,S.R. (2006) Efficient pairwise RNA structure prediction and alignment using sequence alignment constraints. *BMC Bioinf.*, **7**, 400.
49. Hamada,M., Sato,K., Kiryu,H., Mituyama,T. and Asai,K. (2009) CentroidAlign: fast and accurate aligner for structured RNAs by maximizing expected sum-of-pairs score. *Bioinformatics*, **25**, 3236–3243.
50. Will,S., Reiche,K., Hofacker,I.L., Stadler,P.F. and Backofen,R. (2007) Inferring Noncoding RNA Families and Classes by Means of Genome-Scale Structure-Based Clustering. *PLoS Comput. Biol.*, **3**, e65.
51. Do,C.B., Mahabhashyam,M.S.P., Brudno,M. and Batzoglou,S. (2005) ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Res.*, **15**, 330–340.
52. Kall,L., Krogh,A. and Sonnhammer,E.L.L. (2005) An HMM posterior decoder for sequence feature prediction that includes homology information. *Bioinformatics*, **21**, i251–i257.
53. Nussinov,R., Pieczenik,G., Griggs,J.R. and Kleitman,D.J. (1978) Algorithms for Loop Matchings. *SIAM J. Appl. Math.*, **35**, 68–82.
54. Kalvari,I., Argasinska,J., Quinones-Olvera,N., Nawrocki,E.P., Rivas,E., Eddy,S.R., Bateman,A., Finn,R.D. and Petrov,A.I. (2018) Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic Acids Res.*, **46**, D335–D342.
55. Harmanci,A.O., Sharma,G. and Mathews,D.H. (2007) *Toward Turbo Decoding of RNA Secondary Structure*. In 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP 07, IEEE.
56. Harmanci,A.O., Sharma,G. and Mathews,D.H. (2011) TurboFold: Iterative probabilistic estimation of secondary structures for multiple RNA sequences. *BMC Bioinf.*, **12**, 108.
57. Kiryu,H., Kin,T. and Asai,K. (2007) Robust prediction of consensus secondary structures using averaged base pairing probability matrices. *Bioinformatics*, **23**, 434–441.
58. Landthaler,M., Yalcin,A. and Tuschl,T. (2004) The Human DiGeorge Syndrome Critical Region Gene 8 and Its D. melanogaster Homolog Are Required for miRNA Biogenesis. *Curr. Biol.*, **14**, 2162–2167.
59. Lee,Y., Ahn,C., Han,J., Choi,H., Kim,J., Yim,J., Lee,J., Provost,P., Rdmark,O., Kim,S. and Kim,V.N. (2003) The nuclear RNase III Drosha initiates microRNA processing. *Nature*, **425**, 415–419.
60. Partin,A.C., Zhang,K., Jeong,B.C., Herrell,E., Li,S., Chiu,W. and Nam,Y. (2020) Cryo-EM Structures of Human Drosha and DGCR8 in Complex with Primary MicroRNA. *Mol. Cell*, **78**, 411–422.
61. Katoh,K. and Standley,D.M. (2013) MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability. *Mol. Biol. Evol.*, **30**, 772–780.
62. Dirks,R.M. and Pierce,N.A. (2003) A partition function algorithm for nucleic acid secondary structure including pseudoknots. *J. Comput. Chem.*, **24**, 1664–1677.
63. Dirks,R.M. and Pierce,N.A. (2004) An algorithm for computing nucleic acid base-pairing probabilities including pseudoknots. *J. Comput. Chem.*, **25**, 1295–1304.

64. Lorenz,R., Luntzer,D., Hofacker,I.L., Stadler,P.F. and Wolfinger,M.T. SHAPE directed RNA folding. *Bioinformatics*, **32**, 145–147.
65. Tan,Z., Sharma,G. and Mathews,D.H. (2017) Modeling RNA Secondary Structure with Sequence Comparison and Experimental Mapping Data. *Biophys. J.*, **113**, 330–338.
66. Deigan,K.E., Li,T.W., Mathews,D.H. and Weeks,K.M. (2009) Accurate SHAPE-directed RNA structure determination. *Proc. Natl. Acad. Sci. U. S. A.*, **106**, 97–102.
67. Washietl,S., Hofacker,I.L., Stadler,P.F. and Kellis,M. (2012) RNA folding with soft constraints: reconciliation of probing data and thermodynamic secondary structure prediction. *Nucleic Acids Res.*, **40**, 4261–4272.
68. Zarringhalam,K., Meyer,M.M., Dotu,I., Chuang,J.H. and Clote,P. (2012) Integrating Chemical Footprinting Data into RNA Secondary Structure Prediction. *PLoS One*, **7**, e45160.
69. Sksd,Z., Swenson,M.S., Kjems,J. and Heitsch,C.E. (2013) Evaluating the accuracy of SHAPE-directed RNA secondary structure predictions. *Nucleic Acids Res.*, **41**, 2807–2816.
70. Larkin,M.A., Blackshields,G., Brown,N.P., Chenna,R., McGettigan,P.A., McWilliam,H., Valentin,F., Wallace,I.M., Wilm,A., Lopez,R.,Thompson,J.D., Gibson,T.J. and Higgins,D.G. (2007) Clustal W and Clustal X version 2.0. *Bioinformatics*, **23**, 2947–2948.
71. Ray,D., Kazan,H., Chan,E.T., Pea Castillo,L., Chaudhry,S., Talukder,S., Blencowe,B.J., Morris,Q. and Hughes,T.R. (2009) Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. *Nat. Biotechnol.*, **27**, 667–670.
72. Li,X., Kazan,H., Lipshitz,H.D. and Morris,Q.D. (2014) Finding the target sites of RNA-binding proteins. *Wiley Interdiscip. Rev.: RNA*, **5**, 111–130.
73. Taliaferro,J.M., Lambert,N.J., Sudmant,P.H, Dominguez,D., Merkin,J.J., Alexis,M.S., Bazile,C.A and Burge,C.B. (2016) RNA Sequence Context Effects Measured In Vitro Predict In Vivo Protein Binding and Regulation. *Mol. Cell*, **64**, 294–306.
74. Licatalosi,D.D., Mele,A., Fak,J.J., Ule,J., Kayikci,M., Chi,S.W., Clark,T.A., Schweitzer,A.C., Blume,J.E., Wang,X., Darnell,J.C. and Darnell,R.B. (2008) HITS-CLIP yields genome-wide insights into brain alternative RNA processing. *Nature*, **456**, 464–469.
75. Kazan,H., Ray,D., Chan,E.T., Hughes,T.R. and Morris,Q. (2010) RNAcontext: A New Method for Learning the Sequence and Structure Binding Preferences of RNA-Binding Proteins. *PLoS Comput. Biol.*, **6**, e1000832.
76. Orenstein,Y., Wang,Y. and Berger,B. (2016) RCK: Accurate and efficient inference of sequence- and structure-based protein-RNA binding models from RNAcompete data. *Bioinformatics*, **32**, i351-i359.
77. Hiller,M., Pudimat,R., Busch,A. and Backofen,R. (2006) Using RNA secondary structures to guide sequence motif finding towards single-stranded regions. *Nucleic Acids Res.*, **34**, e117.
78. Foat,B.C., Morozov,A.V. and Bussemaker,H.J. (2006) Statistical mechanical modeling of genome-wide transcription factor occupancy data by MatrixREDUCE. *Bioinformatics*, **22**, e141–e149.
79. Jolma,A., Kivioja,T., Toivonen,J., Cheng,L., Wei,G., Enge,M., Taipale,M., Vaquerizas,J.M., Yan,J., Sillanp,M.J., Bonke,M., Palin,K., Talukder,S., Hughes,T.R., Luscombe,N.M., Ukkonen,E. and Taipale,J. (2010) Multiplexed massively parallel SELEX for characterization of human transcription factor binding specificities. *Genome Res.*, **20**, 861–873.
80. Ishida,R., Adachi,T., Yokota,A., Yoshihara,H., Aoki,K., Nakamura,Y. and Hamada,M. (2020) RaptRanker: in silico RNA aptamer selection from HT-SELEX experiment based on local sequence and structure information. *Nucleic Acids Res.*, **https://doi.org/10.1093/nar/gkaa484**, advanced article: peer-reviewed and published.