Specific Aims

Predicting the secondary structure of an RNA sequence with fast speed and high accuracy has been a long-standing challenge in computational biology [93, 41]. It is an important problem because knowing structures reveals crucial information about the RNA's function, which is useful in many applications ranging from noncoding RNA detection [36, 112, 33] to the design of oligonucleotides for knockdown of message [64, 101]. Being able to rapidly determine the structure is extremely useful given the overwhelming pace of increase in genomic data (about 10^{21} base-pairs per year) [97] and given the small percentage of sequences that have experimentally determined structure. While experimental assays still constitute the most reliable way to determine structures, they are prohibitively costly, slow, and difficult, and therefore computational prediction provides an attractive alternative. In this direction, various studies have greatly improved the accuracy of prediction, but there is not enough attention on the speed of prediction. Existing prediction algorithms [124, 29, 84, 58] scale poorly with longer sequences, running in $O(n^3)$ time to predict nesting structures or $O(n^4) \sim O(n^6)$ time for pseudoknots (n being sequence length). The pseudoknot prediction problem is particularly intractable with current methods. These motifs are sets of non-nested base pairs: given two base pairs with indices $i-j$ and $i'-j'$, a pseudoknot would have $i < i' < j < j'$. A small minority of base pairs are pseudoknotted in natural structures, but most functional RNA structures have at least one [56]. For example, tmRNA (tRNA-like and mRNA-like RNA, which rescues stalled ribosomes) has four pseudoknots [46]. Current methods for predicting structures including pseudoknots tend to largely predict false positive pseudoknots [8, 93]. Therefore, there is a *critical need* to develop faster prediction methods without sacrifice in accuracy.

Recently, PI Huang published, in collaboration with Co-Investigator Mathews, **LinearFold** [45], the first **linear-time** (approximate) algorithm for RNA folding, inspired by linear-time incremental parsing algorithms in computational linguistics developed by PI Huang himself [44]. Unlike classical algorithms which process the RNA sequence in a bottom-up fashion, our LinearFold scans it in a left-to-right (5'-to-3') order, which makes it possible to apply beam search [43], a popular pruning technique, to achieve linear runtime with cost of exact search. Our algorithm is orders of magnitude faster than existing methods, and more surprisingly, results in even higher overall accuracy, esp. on the longest families in our database (16S and 23S rRNAs), and improved accuracies over long-range base pairs (500+ nucleotides apart). Our web server http://linearfold.org is by far the fastest such server today, with a sequence limit of 100,000 *nt* (at least 10x that of other servers), and has been used by researchers from 30+ countries, and our code is released at https://github.com/LinearFold/LinearFold. More recently, we have also made substantial progress in adapting LinearFold to **LinearPartition**, approximating the *partition function* and *base pair probability matrix* in linear time. We propose the following Specific Aims:

- **Specific Aim #1**: **Design linear-time algorithms to predict RNA structures with pseudoknots**. We propose two approaches based on our foundational work LinearFold and LinearPartition: (a) We will linearize existing heuristic pseudoknot prediction algorithms such as ProbKnot and IpKnot by replacing their bottleneck $O(n^3)$ computation of the partition function with our LinearPartition; (b) We will develop LinearKnot, a multi-stack extension of the single-stack LinearFold with the additional stacks for pairs that cross with those on the main stack. This algorithm will be capable of predicting most pseudoknots and still run in linear time.

- **Specific Aim #2**: **Linear-Time Machine Learning for Better Folding Parameters using LinearFold.** The scalability of LinearFold opens up new possiblities to scale up the learning of folding parameters to much longer sequences. We propose to learn more accurate folding parameters by using (a) structured prediction algorithms (such as structured SVMs) with LinearFold, and (b) combined with deep learning techniques such as recurrent neural networks (RNNs) and contextualized embeddings. **cite Bepler 2019**

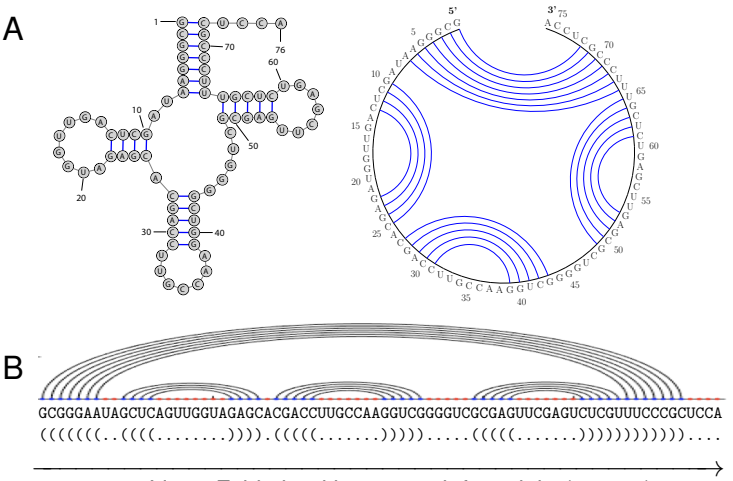
- **Specific Aim #3**: **Application to RNA Secondary Structure Prediction Software.** RNA secondary structure prediction is in widespread use. One popular package is RNAstructure, developed and maintained by co-I Mathews' group [7, 84]. Over 30,000 unique users have registered to download the package and the web servers performed over 125,000 calculations in 2018. We will apply LinearFold in the RNAstructure software package for three important applications: (a) predicting conserved structure in multiple sequence homologs, (b) predicting the base pair interactions between two or more sequences, and (c) predicting target accessibility to DNA or RNA oligonucleotide binding.

A Significance

RNA is Critical to Cellular Function. Over the last three decades, we have become increasingly aware that RNA is actively involved in Biology. Many RNA sequences have intrinsic functions, without being translated to proteins, and we call these RNA sequences noncoding RNA (ncRNA) [31]. ncRNA sequences catalyze reactions [30, 90], regulate gene expression [98, 116, 94], provide site recognition for proteins [5, 109] and serve in trafficking of proteins [110]. It is known that many regions in genomes are transcribed, suggesting there are yet a large number of unknown ncRNA sequences, and new ncRNAs are being reported regularly [11]. A new frontier is the discovery and characterization of long noncoding RNAs [47]. Furthermore, the dual nature of RNA as both a genetic material and functional molecule led to the RNA World hypothesis, that RNA was the first molecule of life [35], and this dual nature has also been utilized to develop in vitro methods to evolve functional sequences [48]. Finally, RNA is an important drug target and agent [99, 89, 19, 14, 34, 12, 80].

Overview of RNA Structure. RNA structure is hierarchical [105]. The primary structure is the covalent structure encoded in the nucleotide sequence. The secondary structure is the set of canonical base pairs (A–U, G–C, and G–U; Fig. 1A–B). The tertiary structure is the 3D structure and the full set of molecular contacts. Secondary structure generally forms faster [119, 115] and is more stable [22, 6, 78] than tertiary structure. Therefore, secondary structure can be accurately predicted independently of tertiary structure.

Prediction of RNA Secondary Structure. The most popular approach to RNA secondary structure prediction is free energy minimization with a dynamic programming algorithm [73, 112]. On average, 73% of known base pairs are correctly predicted in benchmarks of accuracy for sequences shorter than 800 nucleotides [71, 62, 8]. This accuracy is high enough to be useful in developing hypotheses that can be tested experimentally. But these methods have time complexity ranging from $O(n^3)$ to $O(n^6)$, where n is the sequence length, which is too slow for long ncRNAs ($n > 1,000$). This proposal addresses this bottleneck in analysis by providing accurate structure prediction with **linear-time complexity**, borrowing from PI Huang's work in computational linguistics [44] (Fig. 1C).



C

type	*pair distance*	*time*	*space*	*systems/examples*
global	unbounded	$O(n^3)$	$O(n^2)$	RNAstructure, RNAfold, ...
local	bounded	$O(n)$	$O(n)$	Rfold, LocalFold, ...
global	unbounded	$O(n)$	$O(n)$	our **LinearFold** [45]

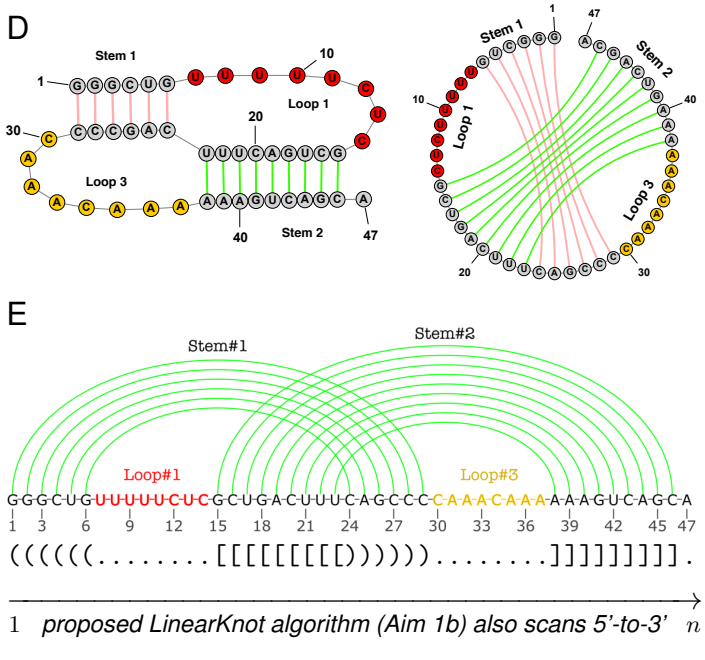


Figure 1: *Overview of RNA secondary structure, our foundational work* **LinearFold** *(Sec. C-0), and one of our proposed algorithms,* **LinearKnot** *(Aim 1b).* **A***: secondary structure of E. coli tRNA^{Gly} and its circle plot;* **B***: the corresponding arc diagram showing nested pairs, the dot-bracket format, and an illustration of our LinearFold algorithm, which scans the sequence left-to-right as opposed to bottom-up as in classical $O(n^3)$ algorithms.* **C***: compared to existing algorithms, LinearFold, though approximate, is the first linear-time algorithm that can predict any pseudoknot-free structure.* **D***: example of pseudoknot structure from human telomerase RNA (and its circle plot);* **E***: the arc diagram showing crossing pairs, the dot-backet format using [and] for stem #2, and the illustration of our proposed LinearKnot algorithm.*

Pseudoknot structure prediction. The biggest barrier to improving RNA secondary structure prediction accuracy is the problem of pseudoknot prediction (Fig. 1D–E). Important algorithms have been developed to predict lowest free energy structures [26, 86, 82], but these scale $O(n^4)$ at best and can only predict a subset of possible topologies. Because of these limitations, heuristics are currently employed [8, 83]. These heuristics suffer from relatively poor accuracy as compared to prediction of non-pseudoknotted pairs. Aim 1 directly addresses this barrier to progress using two different approaches.

B Innovation

Our approach is uniquely different from all previous techniques in RNA structure prediction in the following aspects. First of all, our algorithms run in $O(n)$ time compared to previous methods in $O(n^3)$ or even higher time, making our work scale much better for long noncoding RNAs. Secondly, our work is inspired by state-of-the-art linear-time parsing algorithms in computational linguistics [44], whereas classical RNA folding algorithms such as Zuker and McCaskill are instances of natural language parsing algorithm from the 1960s [50]. This proposal thus revives the fruitful, but somewhat forgotten, line of exchanges between computational linguistics and computational biology [92, 91, 107]. Thirdly, our linear-time framework can potentially incorporate more sophisticated energy functions that would go beyond $O(n^3)$ time in traditional algorithms, leading to more accurate predictions without a sacrifice in speed [111]. Lastly, our multi-stack algorithm (Aim 1b) can predict pseudoknot structures in $O(n)$ time in a single 5'-to-3' scan compared to previous pipelined, multi-pass, heuristic approaches [8, 88]. Aim 3 will additionally put the algorithms to work to solve problems in predicting a common structure in multiple homologs, predicting bimolecular structure, and predicting target site accessibility. Our interdisciplinary team with complementary expertise in computational linguistics and machine learning (PI Huang) and RNA structures (co-Investigator Mathews and PI Huang), is well positioned to carry out this project.

C Approach

typo: progress

We start by presenting our foundational work, **LinearFold** [45], the first linear-time algorithm that can predict any *pseudoknot-free* RNA secondary structure, followed by our recent substantial progres on **LinearPartition**, the adaptation of LinearFold to approximate the partition function and base pair probability matrix. We will then extend them in three ways to predict secondary structures *with pseudoknots* in linear-time (Aim 1), to learn better folding parameters in linear-time (Aim 2), and to apply it to RNA secondary structure prediction software (Aim 3).

C-0 Foundational Work: LinearFold: Linear-Time Prediction of Pseudoknot-Free Structures

Background. Most existing tools for RNA secondary structure prediction use classical algorithms developed in the 1980s–1990s [77, 125, 75], which are $O(n^3)$-time dynamic programs searching for the lowest free-energy (or highest-scoring) structure, resembling natural language parsing algorithms from the 1960s. At a very high level, these algorithms compute, in a bottom-up order, the best substructure for each span $[i, j]$ of the input sequence by splitting $[i, j]$ into smaller spans $[i, k]$ and $[k, j]$, thus requiring $O(n^2)$ space and $O(n^3)$ time.

Due to the prohibitive $O(n^3)$ runtime of standard algorithms, there have been faster alternatives that predict a restricted subset of structures, such as Rfold [51], Vienna RNAplfold [10], and LocalFold [54], which run in linear time but only predict base pairs up to L nucleotides apart ($L \ll n$). In addition, it has been a common practice to divide long RNA sequences into short segments (e.g., $\leq$ 700*nt*) and predict structures within each segment only. All these local methods omit long-range base pairs, which theoretical and experimental studies have demonstrated to be common in natural RNAs, especially between the 5' and 3' ends [93, 53, 55].

LinearFold Algorithm. Unlike classical $O(n^3)$-time algorithms that are hard to linearize due to the bottom-up traversal, our LinearFold scans the RNA sequence left-to-right (i.e., 5'-to-3'), incrementally tagging each nucleotide in the dot-bracket format. While this naive version runs in exponential time $O(3^n)$, we borrow an efficient packing idea that PI Huang co-developed in computational linguistics [106, 44] that reduces the running time back to $O(n^3)$. Furthermore, on top of this left-to-right dynamic programming algorithm, we apply beam search,

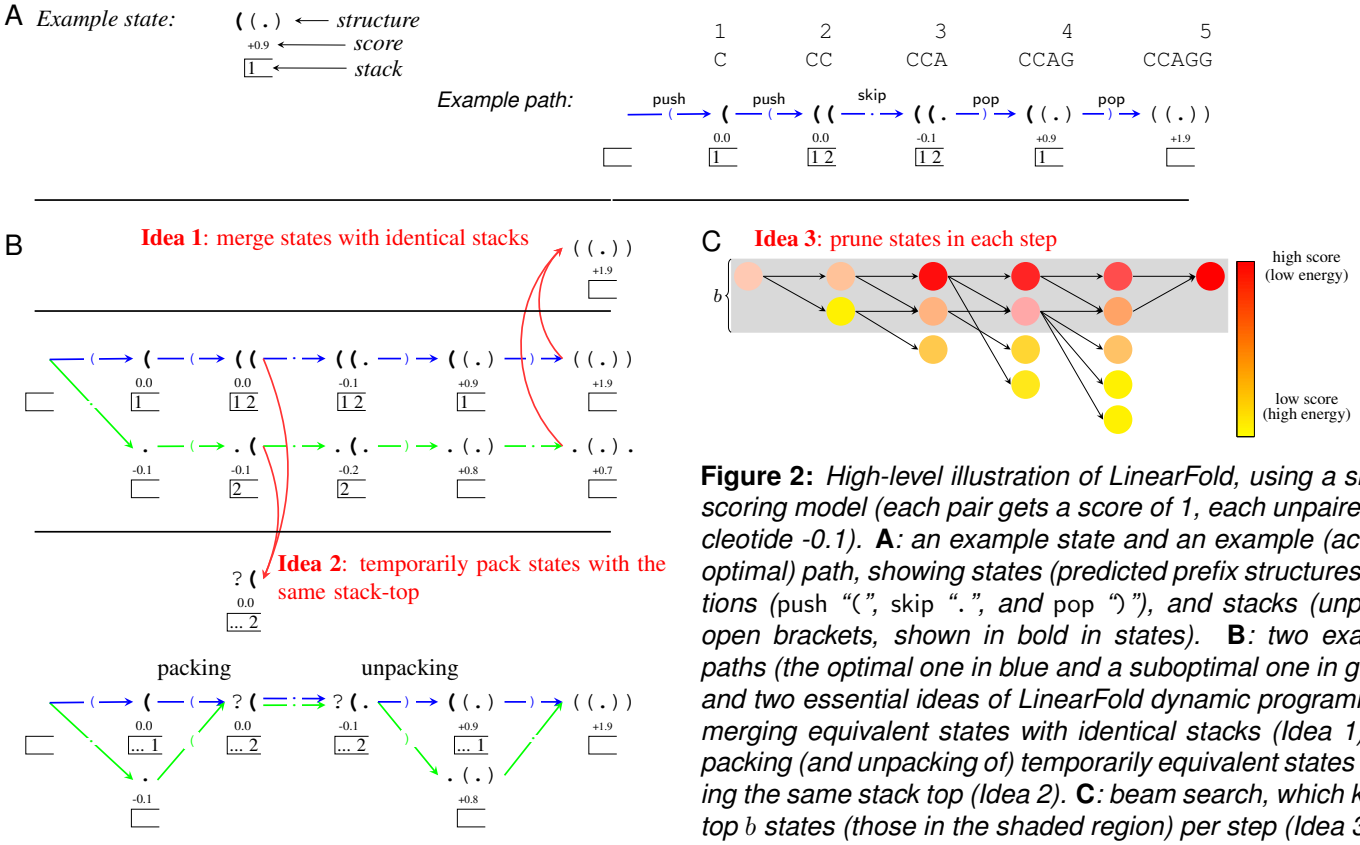


Figure 2: *High-level illustration of LinearFold, using a simple scoring model (each pair gets a score of 1, each unpaired nucleotide -0.1).* **A***: an example state and an example (actually optimal) path, showing states (predicted prefix structures), actions (*push *"(", skip ".", and* pop *")"), and stacks (unpaired open brackets, shown in bold in states).* **B***: two example paths (the optimal one in blue and a suboptimal one in green) and two essential ideas of LinearFold dynamic programming: merging equivalent states with identical stacks (Idea 1) and packing (and unpacking of) temporarily equivalent states sharing the same stack top (Idea 2).* **C***: beam search, which keeps top b states (those in the shaded region) per step (Idea 3).*

a popular heuristic to prune the search space [44], which keeps only the top b highest-scoring (or lowest energy) states for each prefix of the input sequence, resulting in an $O(nb \log b)$ time approximate search algorithm, where b is the beam size chosen by the user. Without loss of generality, we sketch the LinearFold algorithm in Fig. 2 using a simple Nussinov-like scoring function: each pair gets 1 point, and each unpaired nucleotide gets -0.1 points. Below we present our algorithm in four versions, each improving the previous one in efficiency.

0. **Idea 0: Brute-Force: $O(3^n)$ time.** The initial idea mentioned above is to scan the RNA 5'-to-3', maintaining a stack along the way, and performing one of the 3 actions at each step j on nucleotide x_j $(j = 1..n)$:

   (a) *skip*: label x_j as dot ".", i.e., it is unpaired and *skipped*;

   (b) *push*: label x_j as open "(", to be paired with a downstream nucleotide, and *push* j on to the stack; or

   (c) *pop*: label x_j as close ")", paired with the upstream nucleotide x_i where i is the top of the stack, and *pop* i.

   We also require an empty stack at the end, which ensures the output is a well-balanced dot-bracket sequence corresponding to a pseudoknot-free secondary structure. See Fig. 2A for an example path for input sequence CCAGG. A *state* at each step j is a (sub)structure for the prefix input $x_1...x_j$; e.g., ". (" and "((" are two of the possible states for the prefix CC, with [2] and [1,2] being their corresponding stacks. Note that for each state, its stack is simply the positions of unpaired open brackets (shown in bold).

   The above procedure describes a naive exhaustive search without dynamic programming which has exponential runtime $O(3^n)$, as there are up to three actions per step. Next, Fig. 2B sketches the two key dynamic programming ideas that speed up this algorithm to $O(n^3)$ by merging and packing states.

1. **Idea 1: dynamic programming by merging states with identical stacks: $O(2^n)$ time.** We first observe that different states can have the same stack; for example, in Figure 2B, the two states in the last step, ".(.)." and "((.))", both have empty stacks. These states can be merged, because even though they have different histories, going forward they are exactly equivalent. After merging we save the state with the

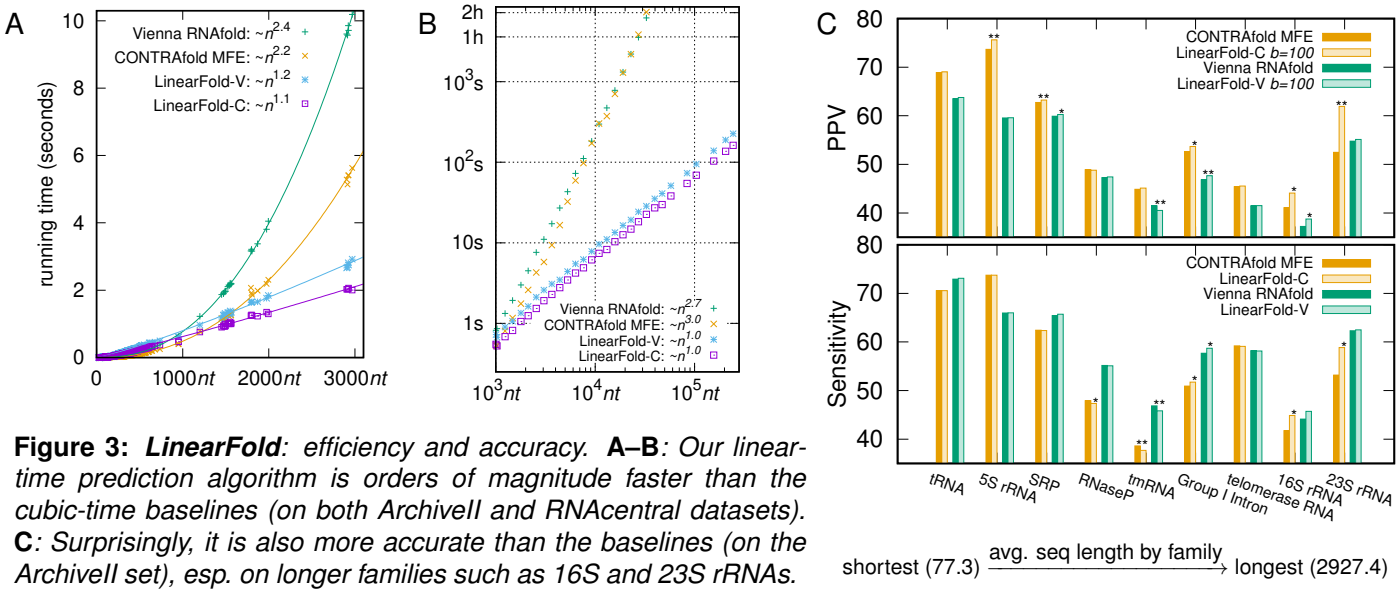


Figure 3: ***LinearFold****: efficiency and accuracy.* **A–B***: Our linear-time prediction algorithm is orders of magnitude faster than the cubic-time baselines (on both ArchiveII and RNAcentral datasets).* **C***: Surprisingly, it is also more accurate than the baselines (on the ArchiveII set), esp. on longer families such as 16S and 23S rRNAs.*

highest score and discard all others which have no potential to lead to optimal structures. This algorithm is faster but still has an exponential time of $O(2^n)$.

2. **Idea 2: dynamic programming by packing temporarily equivalent states: $O(n^3)$ time.** We further observe that even though some states have different stacks, they might share the same stack top; for example, in Figure 2B, in step 2, "`.(`" and "`((`" have [2] and [1,2] as their stacks, with the same stack top 2. Our key insight is that two states with the same stack-top are "temporarily equivalent" and can be "packed" as they would behave equivalently until the stack-top open bracket is closed (i.e., matched), after which they "unpack" and diverge (see Fig. 2B, Idea 2). For example, both "`.(`" and "`((`" are looking for a "G" to match with the stack top x_2 "C", and can be packed as "`?(`" with stack [...2] where `?` and "..." represent histories that are not important for now. After skipping the next nucleotide x_3 "A", they become "`?(.`" and upon matching the next nucleotide x_4 "G" with the stack-top x_2 "C", they unpack, resulting in "`.(.)`" and "`((.)`". These new stacks such as [...2] are called "graph-structured stack" [106]. This method runs in $O(n^3)$ time and is exact. Although this $O(n^3)$ runtime is the same as classical ones, its unique left-to-right (i.e., 5'-to-3', as opposed to bottom-up) nature makes it amenable to linear-time beam search.

3. **Idea 3: Beam Pruning: $O(n)$ time** Finally, we employ beam pruning [43], a popular heuristic widely used in computational linguistics, to reduce the complexity from $O(n^3)$ to $O(n)$, but with the cost of exact search. Basically, at each step j, we only keep the b top-scoring (lowest-energy) states and prune the other, less promising, ones (because they are less likely to be part of the optimal final structure). This results in an approximate search algorithm in $O(nb \log b)$ time, which is linear in sequence length n. See Figure 2C. Although this linear-time search is approximate, it nevertheless achieves higher overall prediction accuracy than the classical cubic-time exact search methods.

add "."

Results of LinearFold. We published two versions of LinearFold: **LinearFold-C** using the machine learned parameters from CONTRAfold [29], and **LinearFold-V** using the thermodynamic parameters developed by Co-I Mathews and used in Vienna RNAfold. Figure 3A confirms that LinearFold's runtime scales linearly with sequence length, while the two baselines scale (near) cubically. For a sequence of ∼10,000*nt* (e.g., the HIV genome), LinearFold takes only 7 seconds while the baselines take 4 minutes. This clearly shows the advantage of LinearFold on very long ncRNAs. We next compare LinearFold with the two baselines in accuracy, reporting both Positive Predictive Value (PPV, the fraction of predicted pairs in the known structure) and sensitivity (the fraction of known pairs predicted) [96] on each RNA family in the ArchiveII dataset from Co-I Mathews

(http://rna.urmc.rochester.edu/pub/archiveII.tar.gz). This is a comprehensive set of well-determined structures first curated in the 1990s [72] and updated later with additional structures [96]. Figure 3B shows that LinearFold is more accurate than the baselines, and interestingly, this advantage is more pronounced on longer sequences, esp. the two longest families in the database, 16S and 23S Ribosomal RNAs.

C-1 Latest Results: LinearPartition: Linear-Time Approximation of the Partition Function

delete

Recently, we have also made substantial progress in adapting LinearFold to to approximate the partition function (and thus the base pair probability matrix) in linear time. This algorithm will be used in Aim 1a and Aim 3a.

Free energy minimization is an important paradigm for predicting structure when a single structure is expected. Many RNA sequences, for example mRNAs, exist in a thermodynamic ensemble of structures [53]. To model these sequences, we need to calculate the *partition function*, which provides a normalization factor from which we can estimate base pairing probabilities [75, 66] or statistically sample structures from the ensemble [25, 70]. The base pairing probabilities also provide confidence estimates for predicted pairs [66, 123]. As a by-product, the pair probabilities also enable maximum expected accuracy (MEA) structure prediction [29, 62]. Given the success of LinearFold, we hypothesize that with reasonable beam size ($b = 100$), its adaptation, the LinearPartition algorithm, will cover much of the probability mass.

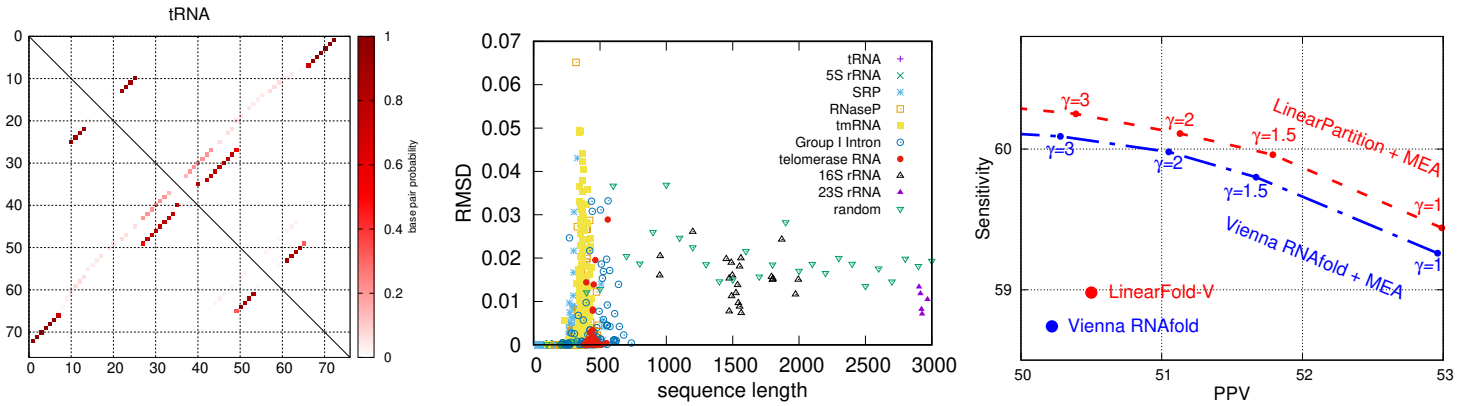
Preliminary Results of LinearPartition. Fig. 4A–B show that our LinearPartition algorithm can indeed approximate the partition function reasonably well. Here we measure root-mean-square deviation (RMSD) between the two probability matrices p and p' (from Vienna RNAfold and LinearPartition, resp.) over the set of all possible pairs pairs(x) on a sequence x (i.e., pairs$(x) = \{1 \le i < j \le |x| \mid x_i x_j \in \{\text{CG, GC, AU, UA, GU, UG}\}, j - i > 3\}$):

$$RMSD(p, p') = \sqrt{\frac{1}{|pairs(x)|} \sum_{(i,j) \in pairs(x)} (p_{i,j} - p'_{i,j})^2}$$

Fig. 4A shows the probability matrix for the example tRNA sequence from Fig. 1A, and both RNAfold and LinearPartition yield identical matrices for all tRNA sequences (i.e., RMSD=0). Fig. 4B shows that RMSD is relatively small across all RNA families in the ArchiveII dataset. The highest deviation is 0.04 for one RNaseP sequence, which means on average, each pair's probability deviation in that worst-case sequence is about 0.0016 between the exact algorithm and our linear-time one. With sequence length increasing, RMSD gradually decreases, since the number of possible pairs grows in $O(n^2)$ but the number of highly probable pairs grows in $O(n)$; on the longest

0.067

0.067



(a) base pair probability matrix (b) RMSD (LinearPartition vs. RNAfold) (c) max. expected accuracy (MEA)

Figure 4: *Substantial results of LinearPartition. (a) LinearPartition (upper triangle) and Vienna RNAfold (lower triangle) result in identical base pair probability matrix for the tRNA sequence in Fig. 1A. (b) root-mean-square deviation (RMSD) is relatively small between LinearPartition and Vienna RNAfold. (c) LinearPartition's base pair probability matrix leads to an improved MEA prediction over the exact partition function from Vienna RNAfold computed in $O(n^3)$ time.*

0.015
23S rRNA family, RMSD is about 0.006 which means each pair's probability deviation is on average only about
0.015 0.000036. We also included 30 random RNA sequences with length 100–3,000 and they behave similarly to natural sequences in terms of RMSD. Fig. 4C shows that LinearPartition even leads to a small improvement in the downstream MEA predictoin using the probability matrix computed in linear time.

C-2 Aim 1: Linear-Time Prediction of RNA Structures with Pseudoknots

Pseudoknots (see Fig. 1D–E) are much harder to model and predict [56]. These **crossing structures** are beyond the expressive capacity of context-free grammars which model pseudoknot-free structures, and generally require super-cubic time; in fact, predicting arbitrary pseudoknot is NP-hard [65, 1]. This is the biggest challenge in RNA secondary structure prediction, and we propose two different linear-time algorithms based on LinearFold.

Background. Existing algorithms for pseudoknot structure prediction fall into two broad categories.

1. **Supercubic-time dynamic programming methods (prohibitively slow)**. Efforts in the first group aim to solve particular subclasses of pseudoknots with exact dynamic programming, at the cost of very high time complexity, ranging from $O(n^4)$ to $O(n^6)$, with increasingly larger subclasses of predictable structures. Examples include Rivas & Eddy (R&E) [86, 87], Akutsu & Uemura (A&U) [107, 1], Dirks & Pierece (D&P) [26], and Reeder & Giegerich (R&G) [82]. Figure 5 shows the relationships among the classes of structures they can predict. By modeling pseudoknots directly, they are considered principled, but are rarely used in practice due to prohibitive computational cost.

ProbKnot DotKnot

2. **Heuristic methods based on base pair probabilities (faster)**. Other researchers developed faster, heuristic methods on top of the $O(n^3)$ pseudoknot-free infrastructure, so that overall runtime does not
these exceed $O(n^3)$, including These methods, such as Probknot [8], IPknot [88], and Dotknot [], all work in two steps: first they compute the base-pair probability matrix (i.e., the partition function) using the $O(n^3)$ McCaskill algorithm [75] which resembles the classical $O(n^3)$ algorithms in structure, and then each method uses its own different heuristic to choose pairs from that matrix. These heuristic algorithms run in $O(n^3)$ time (dominated by the partition function) and are being used in practice, but with very limited success because (a) they are not principled, as the partition function itself is computed under pseudoknot-free assumptions, and (b) cubic-time is still too slow for longer RNAs where pseudoknots are more common.

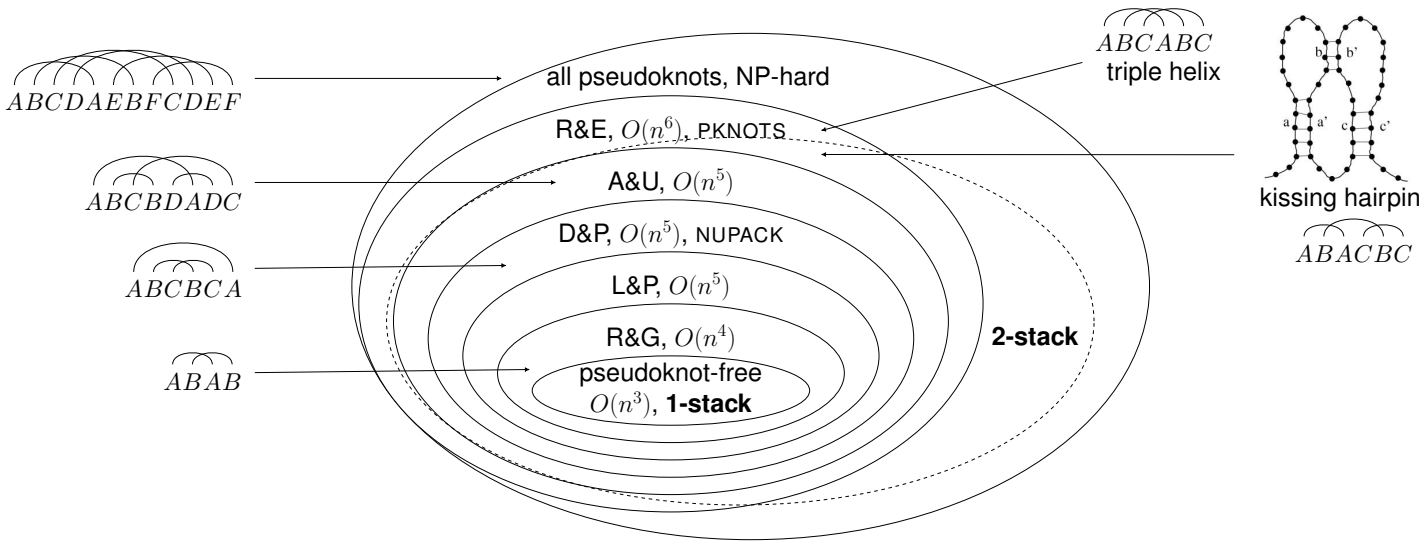


Figure 5: *Classification of pseudoknot structures that can be predicted by existing algorithms, summarizing preivous theoretical results [18, 42]. The existing algorithms are R&E [86, 87], A&U [107, 1], D&P [26], L&P [65], and R&G [82].*

We propose to "linearize" the heuristic methods using LinearPartition, and **LinearKnot**, a left-to-right, multistack extension of LinearFold capturing a large subclass of pseudoknot structures that can run in linear time using beam search (Aim 1b).

Aim 1a: LinearPartition and Linear-Time Heuristic Pseudoknot Prediction

IPknot **DotKnot**

We first propose a very easy heuristic approach, "linearizing" the partition function-based heuristic methods such as ProbKnot (a component of the RNAstructure software package by Co-I Mathews), IpKnot, and Dotknot by replacing their bottleneck $O(n^3)$-time calculation of the partition function with our LinearPartition. After this first step, the second step of all these heuristic methods uses rather simple heuristic criteria to choose pairs from the base pair probability matrix. For example, the second step of probknot selects base pairs (i, j) where the i–j pairing probability is the largest for both bases i and j. This might appear as $O(n^2)$ in the worst case, but since the linear-time beam search used in LinearFold and LinearPartition only returns $O(nb)$ pairs where b is the constant beam size, this second step is still $O(n)$, giving an overall linear-time method, LinearProbKnot. We can similary get LinearIpKnot, LinearDotKnot, etc.

With these promising substantial results of LinearPartition, we believe LinearProbknot (and LinearIpknot, LinearDotKnot, etc) should be as accurate as, if not more accurate than, their original $O(n^3)$ versions.

LinearIPknot

Aim 1b: LinearKnot: Linear-Time Multi-Stack Pseudoknot Prediction

While the above approach is a low-hanging fruit, it is not a principled method as the partition function itself is computed with pseudoknot-free assumptions. We now present a more principled approach to directly model pseudoknots, by generalizing LinearFold (single-stack) to LinearKnot (multiple stacks), with additional stacks modeling pairs that cross with those on the main stack, (see also Fig. 1E). For a given RNA sequence, we first assume (a priori) that there are at most m "pages" [15] where each page contains a subset pseudoknot-free pairs modeled by one stack, with m stacks in total. LinearFold is simply the special case of LinearKnot with $m = 1$. As in LinearFold, our LinearKnot also scans the RNA sequence left-to-right, incrementally taggin each nucleotide, but this time with extra symbols (`[ ]` for page 2; { } for page 3), e.g.:

```
CCUUACCUCUUCCUGGGUAGAGGCAGG
(((..[[[[..{{.)))..]]]]..}}
```

5' LinearKnot scans left-to-right 3'

In other words, LinearKnot predicts all base pairs and simultaneously assign the page index to each base pair.

Since most pseudoknot structures in practice can be covered by 2 pages (in the ArchiveII dataset, only 5 sequences are in 3 pages, out of 1,036 pseudoknotted sequences), in the following we only present a concrete version of the two-stack algorithm. We let the first stack (main stack) to be the same "graph-structured stack" as in LinearFold, but the second stack (side stack) is very different: it records the exact list of unmatched nucleotides (i.e., those `[`'s). Therefore we add 2 actions, push-2 and pop-2, on top of the 3 from LinearFold (skip, push, pop). This exact algorithm runs in time $O(4^w n^3)$ where w is the maximum size of the side stack, and since there are four types of nucleotides, we incur a $O(4^w)$ overhead on top of LinearFold's $O(n^3)$ without beam search. But again,

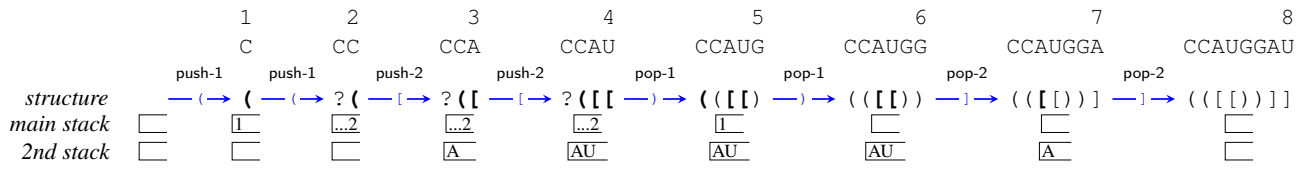


Figure 6: *Example of two-stack LinearKnot on sequence* `CCAUGGAU` *with structure* `(([])]]`.

we can apply beam search to make it run in linear-time in practice. Our preliminary results in Table 1 shows that small values of w can cover a vast majority of known pseudoknot structures in both ArchiveII and PseudoBase.

previous work	% covered		our work (w/o beam search)	% covered	
	ArchiveII	*PB*		ArchiveII	*PB*
classical, 1-page: $O(n^3)$	73.2	*0*	1-stack (LinearFold), $O(n^3)$	73.2	*0*
R&G PKnotsRG: $O(n^4)$	73.5	*57.5*	2-stack, $w = 8, \sim O(n^4)$	95.2	*88.7*
D&P NUPACK: $O(n^5)$	90.6	*89.2*	2-stack, $w = 11, \sim O(n^{4.9})$	97.5	*97.0*
R&E PKnots: $O(n^6)$	100	*100*	2-stack, $w = 16, \sim O(n^{5.8})$	99.9	*97.7*

Table 1: *Preliminary results of Aim 1a: Coverage statistics of existing and our algorithms on ArchiveII and PseudoBaes (PB). Here we convert $O(4^w n^3)$ to $O(n^x)$ using n=3,000 (∼23s rRNAs).*

% pseudoknot structure covered

C-3 Aim 2: Linear-Time Machine Learning using Linear-Time Folding

Our linear-time prediction algorithms will not only improve the speed, but likely **also improve the prediction accuracy** by training on larger datasets than was previously possible with $O(n^3)$ prediction algorithms. In Aim 1's preliminary results, we used energy parameters from CONTRAfold which are learned from the 151-sequence Rfam dataset, using conditional random fields (CRF) [52] as the learning algorithm and $O(n^3)$-time Zuker algorithm as the search algorithm. This CRF learning algorithm is a notable instance of *discriminative learning*, which directly optimizes the conditional distribution $p(y \mid x)$ instead of the joint distribution $p(x, y)$ as in their generative counterparts such as hidden markov models (here input x is an RNA sequence, and output y is a desired structure). As a result, discriminative learning generally outperforms generative models in terms of accuracy, as evidenced in [29], but requires dramatically longer training time since the learner needs to execute search repeatedly on each sequence. We argue that this combination of CRF plus exact search has these limitations and is **not scalable**:

1. the resulting model would inevitably favor $O(n^3)$ predictors rather than $O(n)$ ones;
2. the $O(n^3)$-time exact search algorithm is too slow to be used *repeatedly* during training on longer sequences, which is why CONTRAfold is only trained on 151 short sequences while there are many more data available.

CONTRAfold v2.02 trained on S-Processed with sequence length limit 700

3. the CRF learner itself (independent of search) converges too slowly (requiring 100+ passes over the whole training data), which is well-known in machine learning and computational linguistics [16].

To alleviate these problems and scale to larger datasets, we propose to

CONTRAfold default training epoch is 100

Algorithm 1 Structured Perceptron [16].

Input: data $D = \{(x^{(t)}, y^{(t)})\}_{t=1}^{n}$ and feature map $\mathbf{\Phi}$
Output: weight vector $\mathbf{w}$
Let: $\textsc{Exact}(x, \mathbf{w}) \triangleq \mathbf{argmax}_{s \in \mathcal{Y}(x)} \mathbf{w} \cdot \mathbf{\Phi}(x, s)$
Let: $\Delta\mathbf{\Phi}(x, y, z) \triangleq \mathbf{\Phi}(x, y) - \mathbf{\Phi}(x, z)$

repeat
  for each example (x, y) **in** D **do**
    $z \leftarrow \textsc{Exact}(x, \mathbf{w})$
    if $z \neq y$ **then**
      $\mathbf{w} \leftarrow \mathbf{w} + \Delta\mathbf{\Phi}(x, y, z)$
until converged

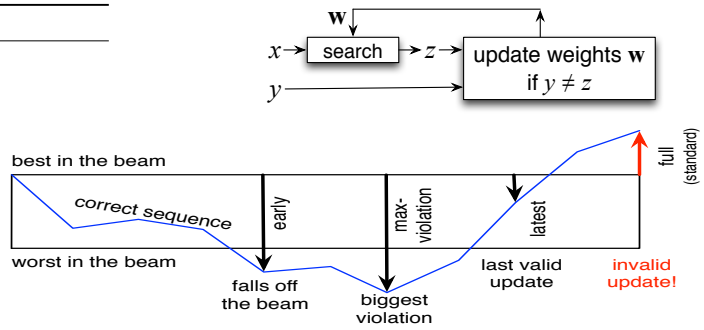


Figure 7: *(left) vanilla structured perceptron. (top right) standard update; (bottom right) specialised updates [17, 43] tailored to inexact search (beam search).*

1. use our $O(n)$ (approximate) beam search prediction that is fast enough to be used *repeatedly* during training on longer sequences; as a by-product, the resulting model can perfectly match $O(n)$ predictors at test time;
2. use the **structured perceptron** (Fig. 7) [16], another popular discriminative training algorithm, as the alternative to CRF since the former converges much faster (making only 5–10 passes over the training data);
3. use specialized variants of structured perceptron developed by PI Huang and others [17, 43] that are tailored to approximate search since vanilla structured perceptron requires exact search (it is well known both empirically and theoretically that vanilla perceptron performs poorly with inexact search [43]). These variants, in particular early update and max-violation update (Fig. 7), have been widely used in computational linguistics as standard recipes for learning with inexact search.

Aim 2a: learning the first prediction model trained from long sequences and large datasets

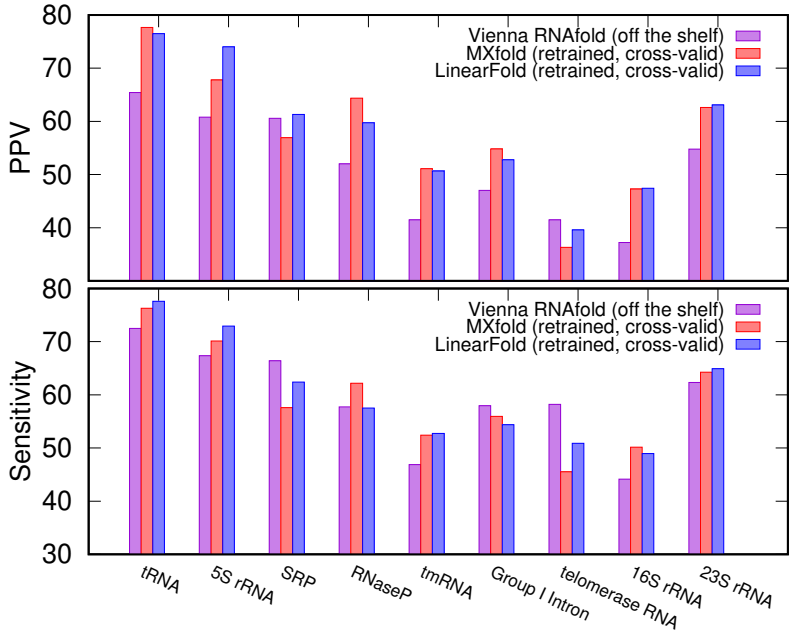


Figure 8: *Preliminary results of Learning to Fold with LinearFold (Aim 2a). Cross-validation training on the ArchiveII dataset (train on all other 9 families, test on 1 family) shows that training with LinearFold results in more accurate predictions than previous machine learning work (MXfold) and thermodynamic parameters in Vienna RNAfold.*

We will first use the max-violation perceptron (Fig. 7, right) developed by PI Huang [43] to learn a better CONTRAfold-style model with $O(n)$ beam search prediction, on both CompaRNA (1,987 sequences) and co-PI Mathews's dataset (3,857 sequences). We hypothesize, based on extensive experience from similar scenarios in computational linguistics [120, 122, 121], that the resulting model will likely lead to significantly more accurate predictions than what we achieved in preliminary results using CONTRAfold model learned from small data. This will be the first RNA prediction model **discriminatively trained from large datasets with longer sequences** (1,000+ sequence-structure pairs, maximum length over 4,000nt); previous work either use less powerful generative models or train on shorter sequences, for instance [2] generatively trained on sequences less than 700 nt. We will then incorporate more sophisticated features corresponding to the more refined energy functions in Aim 1c such as a non-linear function of multiloop length. These more sophisticated features can also be learned without difficulty via perceptron.

Aim 2b: learning better pseudoknot prediction model from large datasets and long sequences

We will then extend this learning framework to learn a pseudoknot model from data, using either the linear-time beam search multi-stack parsing algorithm, or the linearized Dirks & Pierce or Rivas & Eddy algorithms. The physics-based free energy parameters for pseudoknot structures are much less studied than those of pseudoknot-free structures, which can explain the relatively low accuracy of pseudoknot prediction software so far [56, 57].

The existing machine-learned pseudoknot prediction models, on the other hand, are trained on short sequences (e.g., avg. length 126nt in [3]), while most pseudoknotted sequences tend to be much longer (avg. lengths 729nt and 395nt in CompaRNA and the co-PI Mathews's dataset, resp.). We thus envision our learned pseudoknot model from these large datasets will result in significantly more accurate predictions than current physics-based and machine-learned ones. Co-PI Mathews developed a thermodynamic model based on polymer theory, and we would start by using those functional forms [37].

Aim 2c: deep learning for automatic feature engineering without physics

Finally, we aim a more ambitious goal of automatic feature engineering using recent advances in deep learning. So far the CONTRAfold model, though claimed to be "without physics", is still inspired by physics-based free energy functions, and has a close correspondence between feature templates (such as helix stacking and terminal mismatch) and free energy functions. The only difference is in the specific parameter *values* (i.e., feature *weights*), which CONTRAfold learned from data. We instead take a different path, aiming at automating the feature template engineering without **any** hints from physics. Fortunately, recent advances in deep learning provide powerful tools for learning *representations*, and PI Huang's recent work on neural network-based parsing [20, 21] can be a good start of using bidirectional recurrent neural networks (RNNs) to model the sequence. These bidirectional RNNs are particularly good at summarizing the whole sequence, thus providing long-distance lookahead information such as whether there is matching k-mer downstream from the current position that can form a helix with the current k-mer.

C-4 Aim 3: Applications of LinearFold to RNA Secondary Structure Predction Software

RNA secondary structure prediction is in widespread use. One popular package is RNAstructure, developed and maintained by co-PI Mathews' group [7, 84]. Over 30,000 unique users have registered to download the package and the web servers performed over 100,000 calculations in 2017. For this aim, we will apply the linear algorithms from the preliminary results and from aims 1 and 2 in the RNAstructure software package. We will deploy it for three important applications: (a) *TurboFold* (for predicting a conserved structure in multiple sequence homologs) [39, 102], (b) bimolecular structure prediction (to predict the base pair interactions between two sequences) [23, 81], and (c) *OligoWalk* (for predictions of target accessibility to DNA or RNA oligonucleotide binding) [68, 60, 61, 64]. These software tools from RNAstructure are well-used and there will be immediate impact from the faster throughput enabled by linear-scaling in the dynamic programming algorithms on which they are based.

Aim 3a: Accelerate and Improve *TurboFold* using LinearPartition from Aim 1b

For this aim, we will improve the prediction of the conserved structure using multiple sequence homologs [93, 40, 4]. The gold standard for RNA secondary structure determination is comparative sequence analysis, by which the conserved structure is determined across a set of sequence homologs [79]. The secondary structure is well conserved, but the sequence is not as well conserved. Because of this, we can observe compensating base pair changes, by which a base pair is conserved, but the sequences show two changes to conserve the pair. For example, a G-C base pair in one sequence replaced during evolution with an A-U base pair in another sequence [93]. It is these compensating base pair changes that provide evidence for the secondary structure [85]. A multiple sequence alignment is produced to indicate the conserved structures and compensating changes.

Because sequences for ncRNA are not conserved across evolution, it can be difficult to align the sequences. The structure, however, provides additional information needed to align the nucleotides to demonstrate homology. This leads to the chicken-and-egg problem that characterizes the problem of comparative sequence analysis; the sequences cannot be aligned without the structure and the structure cannot be determined until the sequences are aligned. To solve this, the Mathews lab developed methods that simultaneously align and fold multiple sequences. We implemented the first used dynamic programming algorithm that could predict the complete secondary structure and alignment of two sequences [69]. The dynamic programming algorithms are

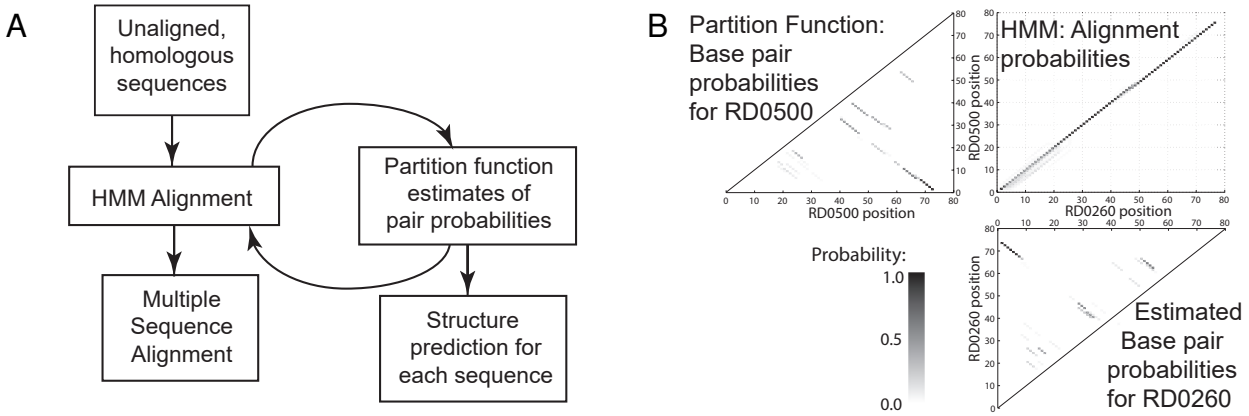


Figure 9: *TurboFold. Panel A shows the information flow in TurboFold. A complete block diagram with more detail can be found as Fig. 1 in [102]. The iterative loop at the center requires three passes to converge. Then the alignment and predicted conserved structures are produced. Panel B illustrates the TurboFold extrinsic information calculation. The upper left corner shows a probability dot plot estimated for tRNA sequence RD0500 using thermodynamics. Black are highly probable base pairs (close to 1) and gray are less probable pairs. There is some uncertainty in the pairing located at nucleotides 15-30. The upper right corner is the HMM alignment of RD0500 and RD0260. Again, black are highly probable nucleotide alignments (close to 1) and gray are less probable. There is some uncertainty in the alignment at nucleotides 10-20. These probabilities are then used to estimate the probability of base pairs in RD0260 as shown in the lower right panel. These are derived using:* $\widetilde{\pi}(i,j) = \sum_{1 \le k < l \le N} \pi(k,l)\alpha(i,k)\alpha(j,l)$. *We call* $\widetilde{\pi}(i,j)$ *the proclivity, i.e. a propensity for base pairing between nucleotides* i *and* j *in the second sequence.* $\pi(k,l)$ *is the base pairing probabilities estimated for the first sequence using the partition function and* $\alpha(i,k)$ *is the alignment probability between nucleotides* i *in sequence 1 and* k *in sequence 2. The TurboFold procedure integrates this across all pairwise proclivities, to determine a total extrinsic information for each sequence (eq. 2 in [39]). The extrinsic information is then used in the partition function calculation as a pseudo free energy change to improve the estimates of base pairing probability. Using the extrinsic information as a free energy change is the statistical mechanics equivalent to a log odds value in statistics.*

computationally costly in spite of methods we used to restrict the search space in alignments and structure [69, 67, 108, 38, 118, 32].

Recently, the Mathews lab developed the *TurboFold* approach that iteratively refines the sequence alignment and structure prediction using a belief propagation framework Figure 9 [39, 102, 103]. This is breakthrough because of its accuracy and excellent scaling. Using traditional folding algorithms [66], the scaling is $O(SN^3 + S^2N^2)$ where S in the number of sequence homologs and N is the average sequence length. The first part of the expression reflects the prediction of secondary structure partition functions for each sequence and the second term reflects the pairwise sequence alignments. The time cost is dominated by the first term.

For this proposal, we will accelerate *TurboFold* using the linear partition function in aim 1. This will alleviate the time bottleneck of the calculations, and give us the opportunity to improve *TurboFold*. The first improvement we will explore is how additional sequences improve the accuracy. Our past benchmarks focused on calculations with 5, 10, or 20 sequences. Going forward, we will explore using additional sequences up to 100. There is a need for a method that can automatically determine the comparative analysis structure for large number of homologs. The majority of RNA families in the rfam database, for example, have only partially determined structures [49]. Statistical analysis using R-scape shows that most families are have structures poorly supported with the existing alignments [85]. Our improved *TurboFold* can be used to refine the alignments and structures.

We will also develop ways to incorporate additional information into the partition function calculation. Because of time constraints, our extrinsic information (inferred by belief propagation) is currently limited to whether a nucleotide is paired or unpaired [39]. With a linear scaling algorithm, we will incorporate additional information, including whether a nucleotide is paired upstream or downstream and whether unpaired nucletiotides are in hairpin, internal, or multibranch loops. We will benchmark the accelerated and enhanced *TurboFold* using sets

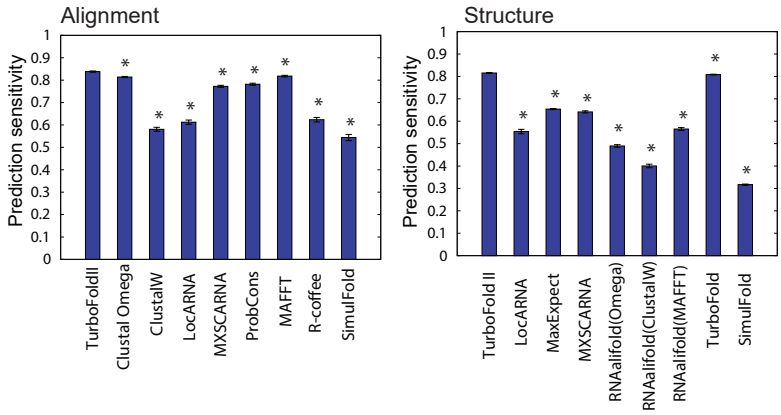


Figure 10: *Benchmarks of RNase P RNA Predictions with TurboFold [102]. The alignment or structure sensitivities (a.k.a. recall; percent of accepted alignments or base pairs that are correctly predicted) are plotted and the stars indicate significantly different performance as compared to TurboFold II using a paired t-test (p¡0.05) [117]. TurboFold II [102] (including alignment refinement) is benchmarked against TurboFold [39] (without alignment refinement); alignment methods Clustal Omega [95], Clustal W [104], ProbCons [27], MAFFT [28], and R-coffee [114]; structure prediction methods MaxExpect [59] and RNAalifold [9]; and simultaneous alignment and structure prediction methods LocARNA [113], MXSCARNA[100], and SimulFold [76]. MaxExpect uses a single sequence to predict structure and performs better than RNAalifold, which predicts a conserved structure but uses a fixed input alignment. This highlights the importance of determining conserved structures and alignments in a single framework.*

of homologous sequences as we have done in previous studies [32, 39, 102, 103].

Our methods for determining conserved structures are already in widespread use. They provide an important initial hypothesis for the starting structure, but the predictions do not yet replace manual sequence comparison in accuracy. By accelerating the fundamental structure prediction, we can computationally afford to improve the model and therefore improve the accuracy of predictions.

Aim 3b: Accelerate and Improve bimolecular and multistrand structure prediction

Many ncRNAs function by interacting with other RNA sequences by base pairing. We developed several software tools for predicting base pairing structures between two sequences (bimolecular) [81, 68, 23, 24]. There are two important barriers to accurate bimolecular structure prediction. First, RNA sequences form self-structure that prevents bimolecular structure prediction. We account for this in our AccessFold algorithm, which is part of the RNAstructure software package and uses a partition function calculation to approximate the accessibility [23] The second barrier is that many simple bimolecular structures that also include unimolecular pairs, such as the kissing hairpin [13], are tantamount to pseudoknots in our algorithms.

The linear algorithms from aims 1 and 2 provide important new ways to improve our existing AccessFold algorithm. First, the linear partition function calculation (Aim 1a) will provide a much faster solution to the accessibility calculation. Second, the linear pseudoknot prediction (aim 2) will provide us with the algorithm needed to elevate AccessFold to include predictions of intramolecular and bimolecular pairs. Currently, AccessFold only provides the pairs between strands, ignoring the pairs within a strand.

We will advance AccessFold to incorporate the new algorithms from aims 1 and 2 and test it against a database of bimolecular structures we developed previously [23]. This will have immediate impact on our ability to predict bimolecular structures by improving speed and also providing additional structure information to our users.

Aim 3c: Accelerate predictions of target binding accessibility

We developed the OligoWalk algorithm to estimate the affinity of complementary oligonucleotides (either DNA or RNA) to a long RNA target, such as an mRNA or long-ncRNA [68, 63] This algorithm has important impact in siRNA and antisense DNA design methods [63, 60, 74]. As part of the calculation, a partition function is used to estimate the target accessibility as the site of the complementarity. This is the bottleneck of these calculations.

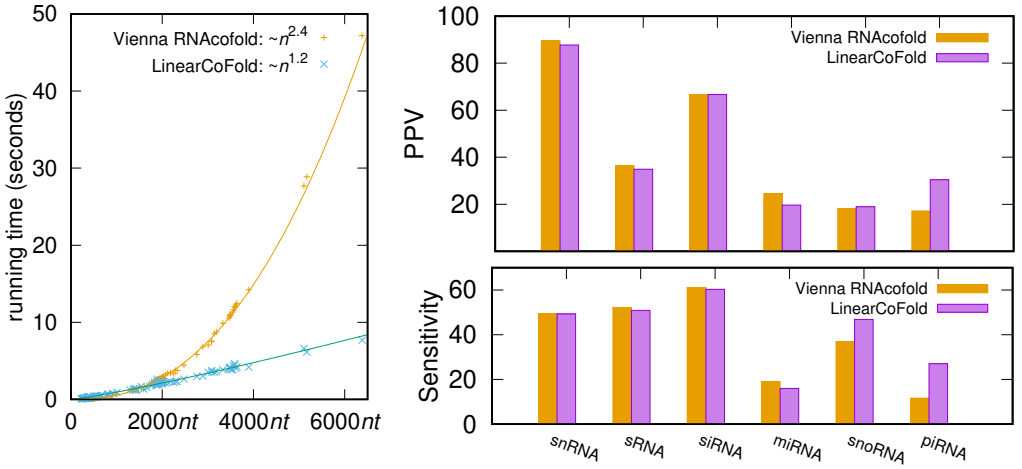


Figure 11: *Preliminary results of Aim 3b, LinearCoFold, linear-time bimolecular folding, predicting both intra- and intermolecular base pairs. Similar to LinearFold, our LinearCoFold is much faster than Vienna RNAcofold, while being more accurate on longer families.*

Using the linear partition function (Aim 1a), we will accelerate the OligoWalk algorithm in RNAstructure. This is well-used on our webservers and therefore a deployment on the RNAstructure webserver will reduce wait times for users and also relieve part of the server workload [61].

Literature Cited

[1] Tatsuya Akutsu. Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots. *Discrete Applied Mathematics*, 104(1):45–62, 2000.

[2] Mirela Andronescu, Anne Condon, Holger H Hoos, David H Mathews, and Kevin P Murphy. Computational approaches for RNA energy parameter estimation. *RNA*, 16(12):2304–2318, 2010.

[3] Mirela S Andronescu, Cristina Pop, and Anne E Condon. Improved free energy parameters for rna pseudoknotted secondary structure prediction. *RNA*, 16(1):26–42, 2010.

[4] K. Asai and M. Hamada. Rna structural alignments, part ii: non-sankoff approaches for structural alignments. *Methods Mol. Biol.*, 1097:291–301, 2014.

[5] Jean-Pierre Bachellerie, Jérôme Cavaillé, and Alexander Hüttenhofer. The expanding snoRNA world. *Biochimie*, 84(8):775–790, 2002.

[6] Aloke Raj Banerjee, John A Jaeger, and Douglas H Turner. Thermal unfolding of a group i ribozyme: the low-temperature transition is primarily disruption of tertiary structure. *Biochemistry*, 32(1):153–163, 1993.

[7] S. Bellaousov, J. S. Reuter, M. G. Seetin, and D. H Mathews. Web servers for rna secondary structure prediction and analysis. *Nucleic Acids Res*, 41:W471–W474, 2013.

[8] Stanislav Bellaousov and David H Mathews. Probknot: fast prediction of RNA secondary structure including pseudoknots. *Rna*, 16(10):1870–1880, 2010.

[9] S. H. Bernhart, I. L. Hofacker, S. Will, A. R. Gruber, and P. F. Stadler. Rnaalifold: improved consensus structure prediction for rna alignments. *BMC Bioinformatics*, 9:474, 2008.

[10] Stephan H Bernhart, Ivo L Hofacker, and Peter F Stadler. Local RNA base pairing probabilities in large sequences. *Bioinformatics*, 22(5):614–615, 2005.

[11] Ewan Birney, John A Stamatoyannopoulos, Anindya Dutta, Roderic Guigó, Thomas R Gingeras, Elliott H Margulies, Zhiping Weng, Michael Snyder, Emmanouil T Dermitzakis, Robert E Thurman, et al. Identification and analysis of functional elements in 1% of the human genome by the encode pilot project. *Nature*, 447(7146):799–816, 2007.

[12] Daniela Castanotto and John J Rossi. The promises and pitfalls of RNA-interference-based therapeutics. *Nature*, 457(7228):426–433, 2009.

[13] Kung-Yao Chang and Ignacio Tinoco. Characterization of a" kissing" hairpin complex derived from the human immunodeficiency virus genome. *Proceedings of the National Academy of Sciences*, 91(18):8705–8709, 1994.

[14] Jessica L Childs-Disney, Meilan Wu, Alexei Pushechnikov, Olga Aminova, and Matthew D Disney. A small molecule microarray platform to select RNA internal loop- ligand interactions. *ACS chemical biology*, 2(11):745–754, 2007.

[15] Peter Clote, Stefan Dobrev, Ivan Dotu, Evangelos Kranakis, Danny Krizanc, and Jorge Urrutia. On the page number of RNA secondary structures with pseudoknots. *Journal of mathematical biology*, 65(6-7):1337–1357, 2012.

[16] Michael Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP*, 2002.

[17] Michael Collins and Brian Roark. Incremental parsing with the perceptron algorithm. In *Proceedings of ACL*, 2004.

[18] Anne Condon, Beth Davy, Baharak Rastegari, Shelly Zhao, and Finbarr Tarrant. Classifying RNA pseudoknotted structures. *Theoretical Computer Science*, 320(1):35–50, 2004.

[19] Stanley T Crooke. Antisense strategies. *Current molecular medicine*, 4(5):465–487, 2004.

[20] James Cross and Liang Huang. Incremental parsing with minimal features using bi-directional lstm. In *Proceedings of ACL*, 2016.

[21] James Cross and Liang Huang. Span-based constituency parsing with a structure-label system and provably optimal dynamic oracles. In *Proceedings of EMNLP*, 2016.

[22] DM Crothers, PE Cole, CW Hilbers, and RG Shulman. The molecular mechanism of thermal unfolding of escherichia coli formylmethionine transfer RNA. *Journal of molecular biology*, 87(1):63–88, 1974.

[23] L. DiChiacchio, M. F. Sloma, and D. H. Mathews. Accessfold: predicting rna-rna interactions with consideration for competing self-structure. *Bioinformatics*, 32:1033–1039, 2016.

[24] Laura DiChiacchio and David H Mathews. Predicting rna–rna interactions using rnastructure. In *RNA Structure Determination*, pages 51–62. Springer, 2016.

[25] YE Ding, Chi Yu Chan, and Charles E Lawrence. RNA secondary structure prediction by centroids in a boltzmann weighted ensemble. *Rna*, 11(8):1157–1166, 2005.

[26] Robert M Dirks and Niles A Pierce. A partition function algorithm for nucleic acid secondary structure including pseudoknots. *Journal of computational chemistry*, 24(13):1664–1677, 2003.

[27] C. B. Do, M. S. Mahabhashyam, M. Brudno, and S. Batzoglou. Probcons: Probabilistic consistency-based multiple sequence alignment. *Genome Res*, 15:330–340, 2005.

[28] C. B. Do, M. S. Mahabhashyam, M. Brudno, and S. Batzoglou. Improved accuracy of multiple ncrna alignment by incorporating structural information into a mafft-based framework. *BMC Bioinformatics*, 9:212, 2008.

[29] Chuong B Do, Daniel A Woods, and Serafim Batzoglou. Contrafold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90–e98, 2006.

[30] Jennifer A Doudna and Thomas R Cech. The chemical repertoire of natural ribozymes. *Nature*, 418(6894):222–228, 2002.

[31] Sean R Eddy. Non-coding RNA genes and the modern RNA world. *Nature Reviews Genetics*, 2(12):919–929, 2001.

[32] Y. Fu, G. Sharma, and D. H. Mathews. Dynalign ii: common secondary structure prediction for rna homologs with domain insertions. *Nucleic Acids Res*, 42:13939–13948, 2014.

[33] Yinghan Fu, Zhenjiang Zech Xu, Zhi J Lu, Shan Zhao, and David H Mathews. Discovery of novel ncRNA sequences in multiple genome alignments on the basis of conserved and stable secondary structures. *PloS one*, 10(6):e0130200, 2015.

[34] Peter C Gareiss, Krzysztof Sobczak, Brian R McNaughton, Prakash B Palde, Charles A Thornton, and Benjamin L Miller. Dynamic combinatorial selection of molecules capable of inhibiting the (cug) repeat RNA- mbnl1 interaction in vitro: discovery of lead compounds targeting myotonic dystrophy (dm1). *Journal of the American Chemical Society*, 130(48):16254–16261, 2008.

[35] Walter Gilbert. Origin of life: The RNA world. *Nature*, 319(6055), 1986.

[36] Andreas Gruber, Sven Findeiss, Stephan Washietl, IVOL HOFACKER, and PETER F STADLER. RNAz 2.0: improved noncoding RNA detection. In *Pacific Symposium on Biocomputing*, volume 15, pages 69–79, 2010.

[37] Christine E Hajdin, Stanislav Bellaousov, Wayne Huggins, Christopher W Leonard, David H Mathews, and Kevin M Weeks. Accurate shape-directed rna secondary structure modeling, including pseudoknots. *Proceedings of the National Academy of Sciences*, 110(14):5498–5503, 2013.

[38] A. O. Harmanci, G. Sharma, and D. H. Mathews. Efficient pairwise rna structure prediction using probabilistic alignment constraints in dynalign. *BMC Bioinformatics*, 8:130, 2007.

[39] A. O. Harmanci, G. Sharma, and D. H. Mathews. Turbofold: Iterative probabilistic estimation of secondary structures for multiple rna sequences. *BMC Bioinformatics*, 12:108, 2011.

[40] J. H. Havgaard and J. Gorodkin. Rna structural alignments, part i: Sankoff-based approaches for structural alignments. *Methods Mol. Biol.*, 1097:275–290, 2014.

[41] Ivo L Hofacker and Ronny Lorenz. Predicting RNA structure: advances and limitations. *RNA Folding: Methods and Protocols*, pages 1–19, 2014.

[42] Ivo L Hofacker and Peter F Stadler. RNA secondary structures. *Reviews in Cell Biology and Molecular Medicine*, 2006.

[43] Liang Huang, Suphan Fayong, and Yang Guo. Structured perceptron with inexact search. In *Proceedings of NAACL*, 2012.

[44] Liang Huang and Kenji Sagae. Dynamic programming for linear-time incremental parsing. In *Proceedings of ACL 2010*, Uppsala, Sweden, 2010.

[45] Liang Huang, He Zhang, Dezhong Deng, Kai Zhao, Kaibo Liu, David Hendrix, and David Mathews. Approximate RNA folding by 5'-to-3' dynamic programming and beam search. *Bioinformatics*, 35, 2019. ISMB 2019. An earlier version available on bioRxiv: `https://www.biorxiv.org/content/early/2018/02/14/263509`; Web server: `http://linearfold.org`, Code: `https://github.com/LinearFold/LinearFold`.

[46] Corey M Hudson and Kelly P Williams. The tmRNA website. *Nucleic acids research*, page 1109, 2014.

[47] Per Johnsson, Leonard Lipovich, Dan Grandér, and Kevin V Morris. Evolutionary conservation of long non-coding RNAs; sequence, structure, function. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1840(3):1063–1071, 2014.

[48] Gerald F Joyce. In vitro evolution of nucleic acids. *Current opinion in structural biology*, 4(3):331–336, 1994.

[49] I. Kalvari, J. Argasinska, N. Quinones-Olvera, E. P. Nawrocki, E. Rivas, S. R. Eddy, A. Bateman, R. D. Finn, and A. I. Petrov. Rfam 13.0: shifting to a genome-centric resource for non-coding rna families. *Nucleic Acids Res*, 46:D335–D342, 2018.

[50] T. Kasami. An efficient recognition and syntax analysis algorithm for context-free languages. Technical Report AFCRL-65-758, Air Force Cambridge Research Laboratory, Bedford, MA†, 1965.

[51] Hisanori Kiryu, Taishin Kin, and Kiyoshi Asai. Rfold: an exact algorithm for computing local base pairing probabilities. *Bioinformatics*, 24(3):367–373, 2007.

[52] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, 2001.

[53] Wan-Jung C Lai, Mohammad Kayedkhordeh, Erica V Cornell, Elie Farah, Stanislav Bellaousov, Robert Rietmeijer, Enea Salsi, David H Mathews, and Dmitri N Ermolenko. mrnas and lncrnas intrinsically form secondary structures with short end-to-end distances. *Nature communications*, 9(1):4328, 2018.

[54] Sita J Lange, Daniel Maticzka, Mathias Möhl, Joshua N Gagnon, Chris M Brown, and Rolf Backofen. Global or local? predicting secondary structure and accessibility in mRNAs. *Nucleic acids research*, 40(12):5215–5226, 2012.

[55] Thomas JX Li and Christian M Reidys. The rainbow spectrum of RNA secondary structures. *Bulletin of mathematical biology*, 80(6):1514–1538, 2018.

[56] Biao Liu, David H Mathews, and Douglas H Turner. RNA pseudoknots: folding and finding. *F1000 biology reports*, 2, 2010.

[57] Biao Liu, Neelaabh Shankar, and Douglas H Turner. Fluorescence competition assay measurements of free energy changes for rna pseudoknots. *Biochemistry*, 49(3):623–634, 2009.

[58] Ronny Lorenz, Stephan H Bernhart, Christian Hoener Zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. ViennaRNA package 2.0. *Algorithms for Molecular Biology*, 6(1):1, 2011.

[59] Z. J. Lu, J. W. Gloor, and D. H. Mathews. Improved rna secondary structure prediction by maximizing expected pair accuracy. *RNA*, 15:1805–1813, 2009.

[60] Z. J. Lu and D. H. Mathews. Fundamental differences in the equilibrium considerations for sirna and antisense oligodeoxynucleotide design. *Nucleic Acids Res*, 36:3738–3745, 2008.

[61] Z. J. Lu and D. H. Mathews. Oligowalk: An online sirna design tool utilizing hybridization thermodynamics. *Nucleic Acids Res*, 36:W104–W108, 2008.

[62] Zhi John Lu, Jason W Gloor, and David H Mathews. Improved RNA secondary structure prediction by maximizing expected pair accuracy. *Rna*, 15(10):1805–1813, 2009.

[63] Zhi John Lu and David H Mathews. Efficient sirna selection using hybridization thermodynamics. *Nucleic acids research*, 36(2):640–647, 2007.

[64] Zhi John Lu and David H Mathews. Efficient siRNA selection using hybridization thermodynamics. *Nucleic acids research*, 36(2):640–647, 2008.

[65] Rune B Lyngsø and Christian NS Pedersen. RNA pseudoknot prediction in energy-based models. *Journal of computational biology*, 7(3-4):409–427, 2000.

[66] D. H. Mathews. Using an rna secondary structure partition function to determine confidence in base pairs predicted by free energy minimization. *RNA*, 10:1178–1190, 2004.

[67] D. H. Mathews. Predicting a set of minimal free energy rna secondary structures common to two sequences. *Bioinformatics*, 21:2246–2253, 2005.

[68] D. H. Mathews, M. E. Burkard, S. M. Freier, J. R. Wyatt, and D. H. Turner. Predicting oligonucleotide affinity to nucleic acid targets. *RNA*, 5:1458–1469, 1999.

[69] D. H. Mathews and D. H. Turner. Dynalign: An algorithm for finding the secondary structure common to two rna sequences. *J. Mol. Biol.*, 317:191–203, 2002.

[70] David H Mathews. Revolutions in rna secondary structure prediction. *Journal of molecular biology*, 359(3):526–532, 2006.

[71] David H Mathews, Matthew D Disney, Jessica L Childs, Susan J Schroeder, Michael Zuker, and Douglas H Turner. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proceedings of the National Academy of Sciences of the United States of America*, 101(19):7287–7292, 2004.

[72] David H Mathews, Jeffrey Sabina, Michael Zuker, and Douglas H Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of molecular biology*, 288(5):911–940, 1999.

[73] David H Mathews and Douglas H Turner. Prediction of RNA secondary structure by free energy minimization. *Current opinion in structural biology*, 16(3):270–278, 2006.

[74] Olgam V Matveeva, DH Mathews, AD Tsodikov, SA Shabalina, RF Gesteland, JF Atkins, and SM Freier. Thermodynamic criteria for high hit rate antisense oligonucleotide design. *Nucleic acids research*, 31(17):4989–4994, 2003.

[75] John S McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–1119, 1990.

[76] I. M. Meyer and I. Miklos. Simulfold: simultaneously inferring rna structures including pseudoknots, alignments, and trees using a bayesian mcmc framework. *PLoS Comput. Biol.*, 3:e149, 2007.

[77] Ruth Nussinov and Ann B Jacobson. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proceedings of the National Academy of Sciences*, 77(11):6309–6313, 1980.

[78] Bibiana Onoa, Sophie Dumont, Jan Liphardt, Steven B Smith, Ignacio Tinoco, and Carlos Bustamante. Identifying kinetic barriers to mechanical unfolding of the t. thermophila ribozyme. *Science*, 299(5614):1892–1895, 2003.

[79] N. R. Pace, B. C. Thomas, and C. R. Woese. *Probing RNA structure, function, and history by comparative analysis*. Cold Spring Harbor Laboratory Press, second edition, 1999.

[80] Prakash B Palde, Leslie O Ofori, Peter C Gareiss, Jaclyn Lerea, and Benjamin L Miller. Strategies for recognition of stem- loop RNA structures by synthetic ligands: Application to the hiv-1 frameshift stimulatory sequence. *Journal of medicinal chemistry*, 53(16):6018–6027, 2010.

[81] D. Piekna-Przybylska, L. DiChiacchio, D. H. Mathews, and R. A. Bambara. A sequence similar to trna3lys gene is embedded in hiv-1 u3/r and promotes minus strand transfer. *Nat. Struct. Mol. Biol.*, 17:83–89, 2009.

[82] Jens Reeder and Robert Giegerich. Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics. *BMC bioinformatics*, 5(1):1, 2004.

[83] Jihong Ren, Baharak Rastegari, Anne Condon, and Holger H Hoos. Hotknots: heuristic prediction of rna secondary structures including pseudoknots. *Rna*, 11(10):1494–1504, 2005.

[84] Jessica S Reuter and David H Mathews. RNAstructure: software for RNA secondary structure prediction and analysis. *BMC bioinformatics*, 11(1):1, 2010.

[85] E. Rivas, J. Clements, and S. R. Eddy. A statistical test for conserved rna structure shows lack of evidence for structure in lncrnas. *Nat. Methods*, 14:45–48, 2016.

[86] Elena Rivas and Sean R Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *Journal of molecular biology*, 285(5):2053–2068, 1999.

[87] Elena Rivas and Sean R Eddy. The language of RNA: a formal grammar that includes pseudoknots. *Bioinformatics*, 16(4):334–340, 2000.

[88] Kengo Sato, Yuki Kato, Michiaki Hamada, Tatsuya Akutsu, and Kiyoshi Asai. Ipknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming. *Bioinformatics*, 27(13):i85–i93, 2011.

[89] Peter Sazani, Federica Gemignani, Shin-Hong Kang, Martin A Maier, Muthiah Manoharan, Magnus Persmark, Donna Bortner, and Ryszard Kole. Systemically delivered antisense oligomers upregulate gene expression in mouse tissues. *Nature biotechnology*, 20(12):1228–1233, 2002.

[90] William G Scott. Ribozymes. *Current opinion in structural biology*, 17(3):280–286, 2007.

[91] David B Searls. The language of genes. *Nature*, 420(6912):211–217, 2002.

[92] David B Searls et al. Formal language theory and biological macromolecules. *Series in Discrete Mathematics and Theoretical Computer Science*, 47:117–140, 1999.

[93] Matthew G Seetin and David H Mathews. RNA structure prediction: an overview of methods. *Bacterial Regulatory RNA: Methods and Protocols*, pages 99–122, 2012.

[94] Alexander Serganov and Evgeny Nudler. A decade of riboswitches. *Cell*, 152(1):17–24, 2013.

[95] F. Sievers, A. Wilm, D. Dineen, T. J. Gibson, K. Karplus, W. Li, R. Lopez, H. McWilliam, M. Remmert, J. Soding, J. D. Thompson, and D. G. Higgins. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Molecular systems biology*, 7:539, 2011.

[96] M.F. Sloma and D.H. Mathews. Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA, In Press.*, 2016.

[97] Zachary D Stephens, Skylar Y Lee, Faraz Faghri, Roy H Campbell, Chengxiang Zhai, Miles J Efron, Ravishankar Iyer, Michael C Schatz, Saurabh Sinha, and Gene E Robinson. Big data: astronomical or genomical? *PLoS Biol*, 13(7):e1002195, 2015.

[98] Gisela Storz and Susan Gottesman. 20 versatile roles of small RNA regulators in bacteria. *Cold Spring Harbor Monograph Archive*, 43:567–594, 2006.

[99] Steven J Sucheck and Chi-Huey Wong. RNA as a target for small molecules. *Current opinion in chemical biology*, 4(6):678–686, 2000.

[100] Y. Tabei, H. Kiryu, T. Kin, and K. Asai. A fast structural multiple alignment method for long rna sequences. *BMC Bioinformatics*, 9:33, 2008.

[101] Hakim Tafer, Stefan L Ameres, Gregor Obernosterer, Christoph A Gebeshuber, Renée Schroeder, Javier Martinez, and Ivo L Hofacker. The impact of target site accessibility on the design of effective siRNAs. *Nature biotechnology*, 26(5):578–583, 2008.

[102] Z. Tan, Y. Fu, G. Sharma, and D. H. Mathews. Turbofold ii: Rna structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Res*, 45:11570–11581, 2017.

[103] Z. Tan, G. Sharma, and D. H. Mathews. Modeling rna secondary structure with sequence comparison and experimental mapping data. *Biophys. J.*, 113:330–338, 2017.

[104] J. D. Thompson, D. G. Higgins, and T. J. Gibson. Clustal w: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 22:4673–4680, 1994.

[105] Ignacio Tinoco and Carlos Bustamante. How RNA folds. *Journal of molecular biology*, 293(2):271–281, 1999.

[106] Masaru Tomita. Graph-structured stack and natural language parsing. In *Proceedings of the 26th annual meeting on Association for Computational Linguistics*, pages 249–257, Morristown, NJ, USA, 1988. Association for Computational Linguistics.

[107] Yasuo Uemura, Aki Hasegawa, Satoshi Kobayashi, and Takashi Yokomori. Tree adjoining grammars for RNA structure prediction. *Theoretical computer science*, 210(2):277–303, 1999.

[108] A. V. Uzilov, J. M. Keegan, and D. H. Mathews. Detection of non-coding rnas on the basis of predicted secondary structure formation free energy change. *BMC Bioinformatics*, 7:173, 2006.

[109] Markus C Wahl, Cindy L Will, and Reinhard Lührmann. The spliceosome: design principles of a dynamic rnp machine. *Cell*, 136(4):701–718, 2009.

[110] Peter Walter and Giinter Blobel. Signal recognition particle contains a 7s RNA essential for protein. *Nature*, 299:21, 1982.

[111] Max Ward, Amitava Datta, Michael Wise, and David H Mathews. Advanced multi-loop algorithms for rna secondary structure prediction reveal that the simplest model is best. *Nucleic acids research*, 45(14):8541–8550, 2017.

[112] Stefan Washietl, Sebastian Will, David A Hendrix, Loyal A Goff, John L Rinn, Bonnie Berger, and Manolis Kellis. Computational analysis of noncoding RNAs. *Wiley Interdisciplinary Reviews: RNA*, 3(6):759–778, 2012.

[113] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler, and R. Backofen. Inferring noncoding rna families and classes by means of genome-scale structure-based clustering. *PLoS Comput. Biol.*, 3:e65, 2007.

[114] A. Wilm, D. G. Higgins, and C. Notredame. R-coffee: a method for multiple alignment of non-coding rna. *Nucleic Acids Res*, 36:e52, 2008.

[115] SA Woodson. Recent insights on RNA folding mechanisms from catalytic RNA. *Cellular and Molecular Life Sciences CMLS*, 57(5):796–808, 2000.

[116] Ligang Wu and Joel G Belasco. Let me count the ways: mechanisms of gene regulation by miRNAs and siRNAs. *Molecular cell*, 29(1):1–7, 2008.

[117] Z. Xu, A. Almudevar, and D. H. Mathews. Statistical evaluation of improvement in rna secondary structure prediction. *Nucleic Acids Res*, 40:e26, 2011.

[118] Z. Xu and D. H. Mathews. Multilign: an algorithm to predict secondary structures conserved in multiple rna sequences. *Bioinformatics*, 27:626–632, 2011.

[119] PP Zarrinkar and JR Williamson. Kinetic intermediates in RNA folding. *Science*, 265:918–924, 1994.

[120] Kai Zhao and Liang Huang. Minibatch and parallelization for structured online learning. In *Proceedings of NAACL*, 2013.

[121] Kai Zhao and Liang Huang. Type-driven incremental semantic parsing with polymorphism. In *Proceedings of NAACL 2015*, 2015.

[122] Kai Zhao, Liang Huang, Haitao Mi, and Abe Ittycheriah. Hierarchical mt training using max-violation perceptron. In *Proceedings of ACL*, Baltimore, Maryland, June 2014.

[123] Jeffrey Zuber, B Joseph Cabral, Iain McFadyen, David M Mauger, and David H Mathews. Analysis of rna nearest neighbor parameters reveals interdependencies and quantifies the uncertainty in rna secondary structure prediction. *RNA*, 24(11):1568–1582, 2018.

[124] Michael Zuker. Calculating nucleic acid secondary structure. *Current opinion in structural biology*, 10(3):303–310, 2000.

[125] Michael Zuker and Patrick Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic acids research*, 9(1):133–148, 1981.