Technical Report: An n-free-passes CYK algorithm for error-correction and the prediction of non-canonical base-pairs in RNA secondary structure

James W. J. Anderson, Zsuzsanna Sükösd, Christian N. S. Pedersen, and Jotun Hein

April 5, 2013

Abstract

Background: The prediction of non-canonical base-pairs in RNA secondary structure prediction has become increasingly important with the advent of next-generation sequencing technologies, where sequencing errors can introduce artificial non-canonical base-pairs in RNA secondary structure. These base-pairs are not appropriately accounted for by the currently existing models.

Results: Here we focus on SCFG-based RNA secondary structure prediction, and introduce an n-free-passes CYK algorithm, which allows a fixed maximum number of base-pairs to be predicted with a probability from a different distribution than the original grammatical model. Our results show that the "n-free-passes" algorithm improves the prediction of artificial non-canonical base-pairs in RNA sequence, even though it does not improve the prediction of naturally occurring non-canonical base-pairs.

Conclusions: The n-free-passes CYK algorithm is a novel approach to address the problem of predicting non-canonical base-pairs that occur artificially due to sequencing errors. The implementation in PPfold and its source code are available from the authors on request.

Keywords: RNA secondary structure, SCFG, non-canonical base-pairs

1 Background

Accurate RNA secondary structure remains a challenging problem in computational biology. The most commonly used programs use either a thermodynamic model (Markham et al. 2008, Hofacker et al. 1994) or stochastic context-free grammars (SCFGs) (Knudsen & Hein 2003), and implement a dynamic programming algorithm. A review of RNA secondary structure prediction can be found in (Gardner & Giegerich 2004).

With the increasing availability of next-generation sequencing technologies, it is desirable to look for increasingly robust algorithms in bioinformatics. Sequencing errors occur at a relatively high rate: Roche 454 pyrosequencing has, for example, been reported to have error rates of up to 10%. Even though error sequencing correction methods do exist, they do not typically consider the structural conservation of RNA and random sequencing errors can "mutate" canonical base-pairs into non-canonical ones in an unpredictable fashion. Similarly, RNA secondary structure prediction methods usually do not appropriately address sequencing errors or non-canonical base-pairs. Thermodynamic models do not predict non-canonical base-pairs at all because of lack of thermodynamic data, and stochastic context-free grammar based models only predict non-canonical base-pairs with very low probabilities. Because of global dependencies in RNA secondary structure, a single error can create large-scale changes in the underlying model probability distributions.

A recent study (Reinharz et al. 2013) has shown a remarkable ability to predict sequencing errors if the secondary structure is known. By considering a modified Boltzmann distribution (McCaskill 1990) and summing over all possible sequence mutations, the sites of most likely sequencing errors could be identified. Unfortunately, homologous structures are not always available (Engelen & Tahi 2007), and neither the position nor the number of sequencing errors is typically known in advance. It is therefore interesting to investigate the behaviour of the method when we allow a fixed *maximum* number of "free" base-pairs, without knowing their exact location, to possibly correct for errors.

In this work, we investigate the effect of introducing an n-free-passes algorithm in a stochastic context-free grammar-based method. The method is implemented for single sequence prediction in PPfold (Sukosd et al. 2012). A free-pass allows the production of single base-pair with a higher probability than would otherwise be indicated from the training of the SCFG for the actual nucleotides occuring in the sequence. This allows the model to freely choose a low-probability base-pair, and replace it with a high-probability one, in order to maximize the probability of the overall structure prediction. If the correct non-canonical base-pair is chosen, we can expect that the accuracy of the overall prediction will increase. The n-free-passes algorithm can be interpreted in the light of sequencing error correction. Rather than changing the probabilities used in the SCFG, the algorithm allows n coupled changes in the sequence, such that two bases that could otherwise form a low-probability base-pair are replaced with "neutral" bases that can form a base-pair with a higher probability.

2 Methods

Stochastic Context-Free Grammars

Context-free grammars are part of the Chomsky hierarchy (Chomsky 1956) and can be defined as a four-tuple (V, Σ, P, S), containing V, a set of non-terminal variables, Σ, a set of terminal variables (usually the alphabet to generate sequences over, e.g. $\{A, C, G, U\}$), P, a set of production rules,and $S \in V$, a unique start symbol. Starting from the start symbol, one applies the production rules until left only with a string of terminal variables. This sequence of production rules is known as a derivation of a grammar. The stochastic context-free grammar can be obtained from a context-free grammar by associating probabilities with each production rule. In this case, the probability of a derivation is simply the product of the probabilities of the rules used in it.

To work with algorithms designed for context-free grammars, it is useful to restrict the statement of the grammar to a normal form. Often Chomsky Normal Form is used, but for RNA secondary structure prediction, it is convenient to use double-emission normal form (Anderson et al. 2012).

This normal form allows production rules of the type $T \to UV$, $T \to .$, and $T \to (U)$, where T, U, V are non-terminal symbols and $(,),. \in \{A, C, G, U\}$ representing paired and unpaired nucleotides. The advantage of double-emission normal form in RNA secondary structure modelling is that it allows for the formation of unpaired and paired bases, whereas Chomsky Normal Form can only output a single terminal variable. It is worth noting that all SCFGs producing valid RNA secondary structures can be written in double-emission normal form. Consequently, algorithms here will be presented for SCFGs in double-emission normal form, however, they can be easily adapted for SCFGs not in this form.

The CYK algorithm

Given an RNA sequence, one approach for identifying a single "best" secondary structure is the CYK algorithm (Younger 1967), which predicts the structure with maximum probability. The CYK algorithm is also a dynamic programming algorithm, which is based on the following recursion relations. Given non-terminal U, and sequence indices i and j, we compute

$$C(U,i,j) = \begin{cases} \mathbb{P}\left[U \to .\right]\ \mathbb{P}_u[s[i]] & \text{if i = j} \\ \max\left\{ \max\limits_{U \to VW} \max\limits_{i \le k < j} \mathbb{P}[U \to VW]\ C(V,i,k)\ C(W,k+1,j), \right. & \\ \left. \max\limits_{U \to (V)} \mathbb{P}[U \to (V)]\ C(V,i+1,j-1)\ \mathbb{P}_d[s[i],s[j]] \right\} & \text{if i < j} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where, for a sequence s with entries $s[1], \ldots, s[k]$, $\mathbb{P}_u[s[i]]$ is the probability of $s[i]$ being unpaired, and $\mathbb{P}_d[s[i], s[j]]$ the probability of $s[i]$ and $s[j]$ being paired. $C(S, 1, k)$ then gives the highest probability of a derivation, and by backtracking through the CYK table the derivation itself, and hence the structure prediction, can be found.

The CYK algorithm is less suitable for use when the grammar is semantically ambiguous (Reeder et al. 2005) (i.e. if more than one derivation is associated with a given secondary structure), because the derivation with highest probability might not correspond to the secondary structure with highest probability, which might be derived in more than one way.

The n-free-passes CYK Algorithm

The n-free-passes algorithm is a modified version of the CYK algorithm, which allows non-canonical base-pairs to be predicted with a higher probability than otherwise indicated by grammar training.

To illustrate the general idea, consider the sequence GGGAAAAACAC, with secondary structure '(((.....)))'. Note the non-canonical base-pair G-A in positions 2 and 10. With no free-passes, the CYK algorithm with a typical RNA SCFG would choose the most likely structure, '.....', because the probability of the base-pair GA is very low. With one free-pass, the CYK algorithm is allowed instead to use its free-pass to draw from an alternative probability distribution, where the free-pass probability $\mathbb{P}'_d$[GA] is higher than the standard probability $\mathbb{P}_d$[GA], and the correct structure '(((.....)))' can become the most likely structure.

Two important issues are also illustrated by this example. The first is the choice of distribution for $\mathbb{P}'_d$. If we choose $\mathbb{P}'_d$ to be greater than the probability of canonical base-pairs, as the algorithm will always choose to use a free-pass. Equally, if $\mathbb{P}'_d$ is lower than the standard probability of non-canonical base-pairs, then it is likely that no alternative structures will be predicted at all. A natural choice for $\mathbb{P}'_d$ might be

$$\max_{\substack{\text{non-canonical} \\ \text{base-pairs}}} \{\mathbb{P}_d[\text{base-pair}]\} < \mathbb{P}'_d[\text{any base pair}] < \min_{\substack{\text{canonical} \\ \text{base-pairs}}} \{\mathbb{P}_d[\text{base-pair}]\} \tag{2}$$

Secondly, by allowing free-passes as proposed, we are not dealing with probabilities anymore, but pseudo-probabilities. When allowing free-passes, the "probability" of a given secondary structure derivation will be greater than or equal to the regular CYK case, and so the probability distribution will not be normalised. Furthermore, the normalising constant will be different for different sequences. However, since the derivation probabilities can be normalised for every sequence, for brevity we will continue to refer to the pseudo-probabilities as probabilities in this paper.

Pseudocode for the n-free-passes CYK algorithm for SCFGs in double-emission normal form is given in Algorithm 1. For a sequence s, $C(n, S, 1, |s|)$ then gives the maximum probability of deriving the sequence s with n free-passes. Backtracking can be done to find the secondary structure associated with this probability in the same way as with the original CYK algorithm.

Algorithm 1 CYK Algorithm with at most n free-passes

for $m = 0$ **to** n **do**
  for $i = 0$ **to** $|s|$ **do**
    for $j = i$ **to** $|s|$ **do**
      $C(m, U, i, j) = 0$
      if $i = j$ **then**
        $C(m, U, i, i) = \mathbb{P}[U \to .]\ \mathbb{P}_u[s[i]]$
      if $i < j$ **then**

$$C(m,U,i,j) = \max \begin{cases} \max\limits_{0\le l\le m} \max\limits_{i\le k<j} \max\limits_{U\to VW} [\mathbb{P}[U \to VW]\ C(l,V,i,k)\ C(m-l,W,k+1,j)] \\ \max\limits_{U\to(V)} [\mathbb{P}[U \to (V)]\ \mathbb{P}_d[s[i]s[j]]\ C(m,V,i+1,j-1)] \\ \max\limits_{U\to(V)} [\mathbb{P}[U \to (V)]\ \mathbb{P}'_d[s[i]s[j]]\ C(m-1,V,i+1,j-1)\ \mathbb{1}_{m>0}] \end{cases}$$

The complexity of the original CYK algorithm is $\mathcal{O}(k^3)$ in time and $\mathcal{O}(k^2)$ in space, where k is the length of the RNA sequence. With n free-passes, the CYK table must be completed for n additional "layers", and so the complexity becomes $\mathcal{O}(n \times k^3)$ in time and $\mathcal{O}(n \times k^2)$ in space. If only a small number of errors or non-canonical base-pairs is expected, this is only a constant time multiplier, but if the error rate were proportional to the length of the sequence, this algorithm would become $\mathcal{O}(k^4)$.

3 Implementation and Testing

PPfold and the CYK algorithm

PPfold is a recent multithreaded reimplementation of the Pfold algorithm (Knudsen & Hein 1999, Knudsen & Hein 2003), and the two programs use the same lightweight stochastic context-free grammar, which we will refer to as KH99. The KH99 grammar is:

$$\begin{array}{lcl|l} S & \to & L & LS \\ L & \to & . & (F) \\ F & \to & (F) & LS \end{array} \tag{3}$$

where . indicates a single unpaired nucleotide, and (and) indicate two nucleotides forming a base-pair. KH99 is not presented in double-emission normal form but the adaption of the algorithm is trivial (Anderson et al. 2012).

Typically, a SCFG is parameterised with a separate nucleotide distribution. The probability of a SCFG generating a particular unpaired nucleotide, say an unpaired A, is the product of the SCFG probability of producing any unpaired nucleotide multiplied with the likelihood that the unpaired nucleotide is an A, that is: $\mathbb{P}[L \to A] = \mathbb{P}[L \to s]\mathbb{P}_u[A]$. In this way, the probability of the derivation corresponding to an RNA secondary structure string can also be written as a product of the SCFG rule probabilities and the nucleotide probabilities. The nucleotide probabilities (likelihoods) are typically obtained using frequency counts from RNA sequences with known secondary structures.

In Pfold and PPfold, the dynamic programming algorithm known as the inside-outside algorithm (Lari & Young 1990) is implemented, to compute posterior basepairing and single-stranded probabilities. Using these probabilities, the predicted secondary structure is determined by maximizing the expected number of correctly predicted secondary structure elements. For the Pfold base-pair probabilities, the maximum probability non-canonical base-pair is UU with probability 0.0027, and the minimum probability canonical base-pair is GU with probability 0.0490. Throughout, then, we will use a free-pass probability of 0.025.

Testing data

To test the algorithm, we took a data set of 443 RNA sequences with known secondary structures from RNASTRAND (Andronescu et al. 2008). The data set was filtered to remove too similar sequences (greater than 80% base pair similarity) or sequences with ambiguous base-pairs. The dataset contained no pseudoknotted structures, as these cannot be predicted by standard SCFG methods.

From the set of 443 structures, three data sets were created. Firstly, the 443 structures were partitioned into two sets, those structures which contained natural non-canonical base-pairs, and those structures which did not. There were 206 structures containing non-canonical base-pairs, which will be hereafter called the *NC dataset*. The NC dataset contained an average of 3.74 non-canonical base-pairs per secondary structure, with 43 of the secondary structures only containing a single non-canonical base-pair. The 237 structures which did not contain non-canonical base-pairs will be denoted the *Non-NC dataset*.

Secondly, we created a simulated dataset by taking all the structures in the Non-NC dataset, choosing a base-pair at random, and changing it to a random non-canonical base-pair. Errors simulated in base-pairs might simulate a sequencing or processing error, and allows for examining the performance of the n-free-passes CYK to evade the effects of this error. We note that errors were not simulated in unpaired regions. For a given secondary structure, the probability of two sequences which only differ by an unpaired base will differ only slightly– by the ratio of the unpaired nucleotide probabilities– and so the n-free-passes CYK algorithm would simply choose the nucleotide with higher probability. Consequently, this error simulation would not allow for useful evaluation of the n-free-passes CYK algorithm. This dataset will be known throughout as the *Sim-NC dataset*.

Measuring accuracy

To evaluate prediction accuracies, we used the *F-score*, which is the harmonic mean of the sensitivity and the positive predictive value (PPV) of the base-pair (bp.) predictions, compared to the reference structure. These quantities are defined as:

$$\text{sensitivity} = \frac{TP}{\text{number of bp. in reference}} \tag{4}$$

$$\text{PPV} = \frac{TP}{\text{number of bp. in prediction}} \tag{5}$$

$$\text{F-score} = 2 \times \frac{\text{sensitivity} \times \text{PPV}}{\text{sensitivity} + \text{PPV}} \tag{6}$$

where TP is the number of correctly predicted base-pairs.

4 Results and discussion

Probability of a free-pass

To investigate the behaviour of the method as a function of the free-pass probability, we plotted the accuracy against the fractional free-pass probability for all 3 datasets, shown in Figure 1. We expected that the best performance would be obtained if the probability of the free-pass was between

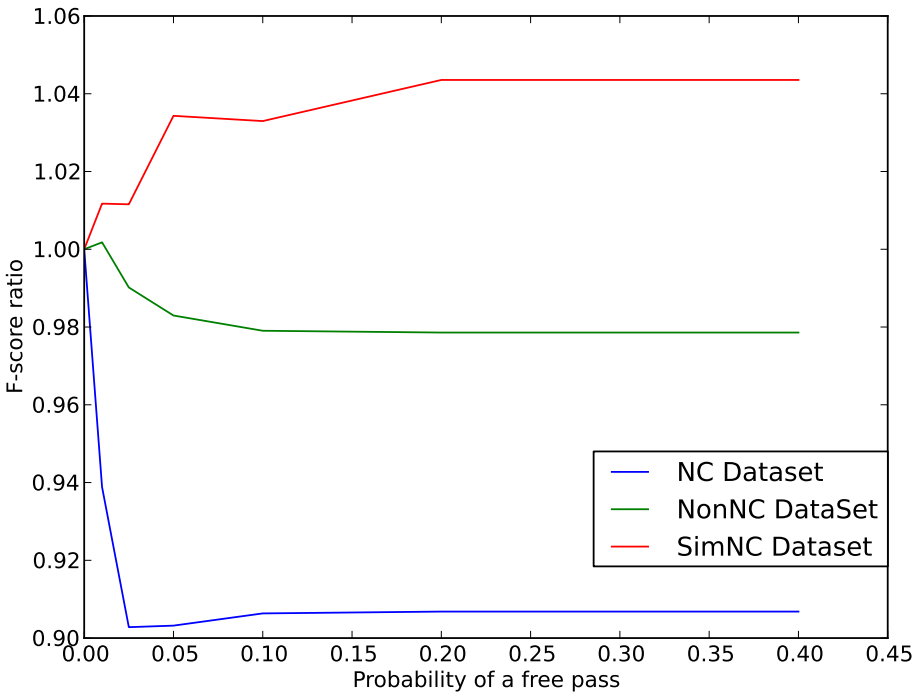


Figure 1: The relative F-score, $\frac{\text{F-score with 1 free pass}}{\text{F-score without free passes}}$ is plotted as a function of free pass probability, for all 3 datasets. A free pass probability of 0.049 corresponds to the probability of a G-U wobble base pair. Probabilities below this value are smaller than the probabilities of all canonical pairs. Probabilities larger than 0.049 are greater than the probabilities of all non-canonical pairs.

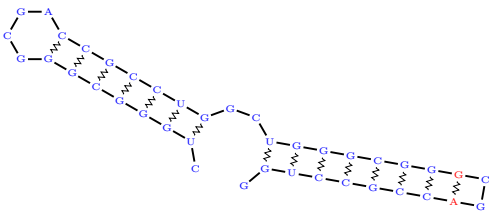
the highest probability for non-canonical base-pairs (0.0027) and the lowest probability for canonical base-pairs (0.0490). Against our expectations, any nonzero probability decreased the average accuracy in the case of the NC dataset, had a very small negative effect in the case of the non-NC dataset, and only gave rise to positive changes in the case of the simulated dataset. These results indicate that the n-free-passes approach does not generally increase the accuracy of RNA secondary structure prediction when the sequences do not include sequencing errors. However, it appears that the algorithm does improve prediction quality in the case where "non-canonical" base-pairs are not of natural origin but rather are introduced artificially due to errors in the sequence.

We suggest the observed pattern can be explained as follows. In the case where the sequences include naturally occurring non-canonical base-pairs or no non-canonical base-pairs at all, the original KH99 model is already as well trained as possible to predict the correct structure. The n-free-passes approach does not capture any additional non-canonical pairing signals. If it does add non-canonical base-pairs, these are chosen more-or-less randomly among the large number of available options, and will therefore in a large number of cases be incorrect. In the case where the sequences include sequencing errors, however, the original grammatical model is no longer appropriate. The appearing "non-canonical" base-pairs include the additional signal that an alternative, closely related sequence would be significantly more adequately explained by the model. In these cases, therefore, it is possible to improve prediction quality.

Case studies

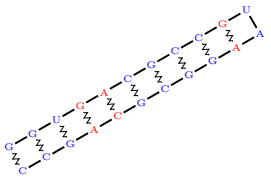
Next, we wanted to investigate the patterns observed in Figure 1 in more detail. On a closer inspection of the resulting structure predictions, we identified that sequences generally fell into

A



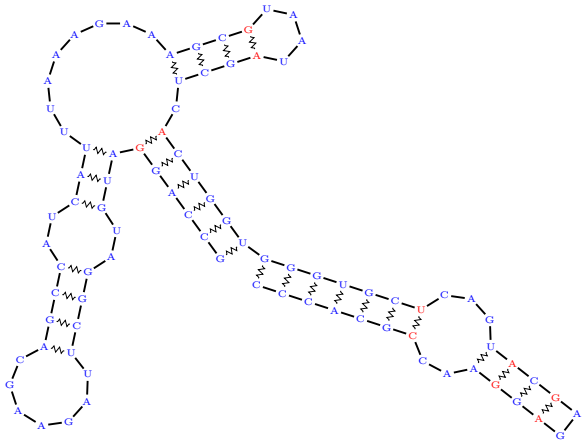
```
Sequence:          CUGGGCGGGCGACCGCCUGGCUGGGCGGGCGACCGCCUGG
True Structure:    .(((((((....)))))))..(((((((..)))))))).
Zero Free-Passes:  (((((((....)))))))((((((((....))))))))
One Free-Pass:     (((((((....)))))))((((((((....))))))))
Two Free-Passes:   ((((((((((((((((((..))))))))))))))))))
```

B



```
Sequence:          GGUGAACGCCGUAAGGCGAAGCC
True Structure:    ((((((((((..))))))))))
Zero Free-Passes:  .....((((....))))).....
One Free-Pass:     (((((((((....)))))))))
Two Free-Passes:   (((((((((....)))))))))
```

C



```
Sequence:          GCCAGGAUGUAGGCUUAGAAGCAGCCAUCAUUUAAAGAAAGCGUAAUAGCUCACUGGUGGGUGCUCAGUACGAGAGGAACGGCACCC
True Structure:    ((((((((((..((((.......))))..)))).........(((.....))))..)))))).((((((...((((..))))..))))))
Zero Free-Passes:  (((((.....(((((.......))))..............(((.....)))..)))))).((((((..................))))))
One Free-Pass:     (((((.....(((((.......))))..............(((.....)))..)))))).((((((..................))))))
Two Free-Passes:   ((((((((((((((.......))))))))))).........(((.....)))..)))))).((((((..................))))))
Three Free-Passes: ((((((((((((((.......))))))))))).........(((.....)))..)))))).((((((..................))))))
Four Free-Passes:  ((((((((((((((..))))))))))).........(((.....)))..)))))).((((((..................))))))
```

Figure 2: Non-canonical base-pairs are highlighted in red in both the sequence and graphical representation of the secondary structure. Subfigure A is an example of a structure prediction which gets considerably worse when free-passes are allowed, Subfigure B an example of a structure prediction which gets considerably better when free-passes are allowed, and Subfigure C an example of a structure prediction which does not change much when free-passes are allowed. See text for further analysis.

three categories:

(A) Prediction quality is significantly worsened, Δ F-score $\leq$ -0.05

(B) Prediction quality is significantly improved, Δ F-score ≥ 0.05

(C) No significant change in the F-score. $|\Delta$ F-score$| < 0.05$

The majority of the structures fell in category (C). In Figure 2, we have chosen 3 structures which illustrate the above categories. For varying number of free-passes, all with probability 0.025, secondary structure predictions were produced to demonstrate the algorithm.

In the case of (A), we have a structure with a single non-canonical base-pair, a GA on the inside of a stem. With zero free-passes (that is, the CYK prediction) the prediction quality is reasonably good, with additional CG base-pairs produced on the outsides of each of the stems. The one free-pass prediction is identical to the zero free-passes prediction. When KH99 produces unpaired bases in helices, it must use a specific sequence of production rules to change from producing paired nucleotides to producing paired nucleotides, which does not depend on how many unpaired bases KH99 goes on to produce. In this way, the probability of producing a GA base-pair on the inside of the stem is not higher than leaving the two bases unpaired. When two free-passes are used, though, the predicted structure changes entirely to a single stem. The sequence is almost a base-pairing palindrome– the first base pairing with the last, and so on– but there are two exceptions to this. With two (or more) free-passes, a structure with one stem occurs with higher probability than a structure with two stems, and so the structure prediction is significantly worse.

Structure (B) is an RNA hairpin containing three non-canonical base-pairs. The prediction with zero free-passes avoids all of these non-canonical base-pairs, predicting only a small stem in the middle of the sequence. With a single free-pass, however, two non-canonical base-pairs are predicted for a much better structure prediction. Once a single non-canonical base-pair is allowed at a slightly higher probability, predicting a second non-canonical base-pair, despite the low probability, allows the SCFG to produce a single stem structure. Two (or more) free-passes produce the same structure, but by allowing a second free-pass, the structure occurs at a higher probability. As with structure (A), even with a free-pass the algorithm does not produce the non-canonical base-pair on the inside of a stem.

Finally, structure (C) is a more complex structure, with five non-canonical base-pairs. However, many of the non-canonical base-pairs are on the inside of stems, which we have seen will not be predicted with free-passes. With zero free-passes, the structure prediction is reasonably accurate. With a single free-pass added, the prediction is identical. Two or three free-passes, though, allows KH99 to significantly extend a stem, creating a higher probability structure. In particular, the true non-canonical base-pairs are not predicted, the model instead choosing to extend a stem– a choice which allows more efficient use of production rules by KH99, and hence a higher probability structure. Until larger numbers of free-passes are allowed, though, the structure stays approximately the same.

Statistics

Results of the implementation of the n-free-passes CYK for all three datasets can be found in Table 1. For for clarity of analysis, we show only the results for a single free-pass. We see that, in all three data sets, the majority of secondary structure predictions have little or no change in F-score. This is particularly pronounced in the Non-NC dataset, where there are no non-canonical base-pairs, so we might expect fewer changes.

Structures which did produce positive or negative changes in F-score tended, on average, to be shorter. This is partitially due to the dependence of F-score on sequence length, as is also the case for many other RNA secondary structure metric (Freyhult et al. 2005). Predicting five more correct base-pairs, as in (B) of Figure 2, will create a much higher change in F-score than predicting five more correct base-pairs for a significantly longer sequence. However, the opposite is true in the case of the Non-NC dataset, where the average length of the positively improved predictions is notably

	#structures	Avg. length	% increase in C bps
NC			
Δ F-score $\leq$ -0.05	43	128	3
$\lvert\Delta$F-score $\rvert < 0.05$	139	191	2
Δ F-score ≥ 0.05	24	119	16
Sim-NC			
Δ F-score $\leq$ -0.05	28	133	5
$\lvert\Delta$F-score $\rvert < 0.05$	178	167	2
ΔF-score ≥ 0.05	31	109	11
Non-NC			
Δ F-score $\leq$ -0.05	24	130	11
$\lvert\Delta$F-score $\rvert < 0.05$	197	157	2
Δ F-score ≥ 0.05	16	172	17

Table 1: Probability = 0.025, 1 free pass compared to 0 free passes.

higher. This is due to several outliers with length over 300 in a small data set, skewing the sample average.

Also notable is the percent increase in the number of canonical base-pairs for each catagory. As might be expected, when the F-score changes very little, the number of canonical base-pairs is almost the same as it was previously. To make a significant change in F-score, for better or worse, the prediction needs to predict more than just the non-canonical base-pairs. As in (B) of Figure 2, where prediction of one non-canonical base-pair allows for prediction of other, canonical, base-pairs, we would expect larger F-score change to require larger change in canonical base-pairs.

Discussion

As the n-free-passes approach changes the KH99 model for RNA secondary structure in fundamental ways, parameterisation is of concern. All grammar probabilities were taken from (Knudsen & Hein 1999), so that testing was independent of parameterisation. In that work, the SCFG probabilities were obtained through counting in derivations of known structures, and similarly the nucleotide probabilities were determined from frequency counts. These parameters maximise the probability of ordinary SCFG prediction, not n-free-passes SCFG prediction. Since KH99 is semantically unambiguous, the production rules used to generate the known secondary structures would be identical for the n-free-passes algorithm, so the probabilities for the production rules will remain constant. The only probabilities that might change in our model are therefore the nucleotide probabilities. If we allowed a free-pass every time there was a non-canonical base-pair in the training structure, we would have zero probabilities for all non-canonical base-pairs, as a free-pass would be preferable each time. Allowing free-passes, in addition to the usual small non-canonical base-pair probabilities, will permit occasional prediction of non-canonical base-pairs not predicted by free-passes, as we see in (B) of Figure 2.

Additionally, one might consider an alternative approach of allowing non-canonical base pairs only in the backtracking part of the CYK algorithm. However, this does not alter the underlying probability distribution created by the SCFG, and we would not predict more than a single additional base-pair. Part of the design of the n-free-passes algorithm was to allow a complete change of structure if it is preferred by the algorithm, and allowing non-canonical base-pairs in the backtracking would not achieve this.

Another possibility would have been to implement the n-free-passes algorithm for the inside-outside algorithm. The advantage of this would be that we could sum over the entire probability distribution generated by the SCFG, rather than simply taking the maximum. Unfortunately, this is not possible to do in practice. For a single free-pass, it would be necessary to sum over all possible subsequences and substructures that can be generated; unlike in the CYK algorithm, the inside-

outside algorithm would require summing probability contributions from multiple non-canonical base-pairs. If only a single nucleotide or base-pair were of interest, the inside-outside algorithm could be used to sum over all possible predictions that can happen outside of that nucleotide or base-pair. But with an entire structure prediction, the sum would not be over the correct distribution.

The algorithm can be used to answer different questions with just a few small changes. For example, changing the initialization condition to only include a non-zero probability on the top "layer" ($m = n$, $i = j$), as well as prohibiting the "copying" of values from a lower layer

$$\left(\max_{U \to (V)} \left[\mathbb{P}[U \to (V)] \, \mathbb{P}'_d[s[i]s[j]] \, C(m-1, V, i+1, j-1) \, \mathbb{1}_{m>0} \right] \right)$$

, would cause the algorithm to return a structure with *exactly* n-free-passes, in contrast to the current implementation, which returns *maximum* n-free-passes.

However, the extension of the n-free-passes algorithm to alignments is less simple. In the case of alignments, nucleotide likelihoods are replaced with alignment likelihoods, which are derived from an evolutionary model, and will vary considerably depending on the sequence content and evolutionary tree. Similarly, one would not expect a whole column to be erroneous, but only a single entry in that column. To implement this, the choice of distribution $\mathbb{P}'$ is therefore not trivial, and the Felsenstein pruning algorithm (Felsenstein 1981) would also have to be modified in addition to the CYK algorithm to allow a fixed number of free-passes in this framework.

The general approach presented in this paper could be extended in a number of ways. By considering SCFG rules which simulate base-pair stacking as in (Dowell & Eddy 2004), it might be possible to gain a more realistic biological signal from sites of possible sequencing errors. Similarly, base-pair stacking will encourage the extension of helices in predictions. The n-free-passes approach could also be considered in the case of the thermodynamic method, and algorithms that can predict pseudoknots.

5 Conclusions

In this paper, we have developed an n-free-passes CYK algorithm for the prediction of non-canonical base-pairs in RNA secondary structure, which allows the prediction of maximum n base-pairs with probabilities that are different from the original SCFG model. If this probability is set appropriately, n low-probability non-canonical base-pairs can be replaced with higher-probability ones. We implemented the algorithm for the PPfold SCFG, and tested it on 443 sequences. Our results show that the n-free-passes approach does not generally increase the accuracy of an RNA secondary structure prediction in the case where sequences do not include sequencing errors, but does increase accuracy when "artificial" non-canonical base-pairs are introduced due to sequencing errors.

References

Anderson, J. W. J., Tataru, P., Staines, J., Hein, J. & Lyngso, R. (2012), 'Evolving stochastic context-free grammars for rna secondary structure prediction', *BMC Bioinformatics* **13**(1), 78.

Andronescu, M., Bereg, V., Hoos, H. & Condon, A. (2008), 'Rna strand: The rna secondary structure and statistical analysis database', *BMC Bioinformatics* **9**(1), 340. 18700982.

Chomsky, N. (1956), 'Three models for the description of language', *Information Theory, IRE Transactions on* **2**(3), 113–124.

Dowell, R. & Eddy, S. (2004), 'Evaluation of several lightweight stochastic context-free grammars for rna secondary structure prediction', *BMC Bioinformatics* **5**(1), 71.

Engelen, S. & Tahi, F. (2007), 'Predicting rna secondary structure by the comparative approach: how to select the homologous sequences', *BMC Bioinformatics* **8**(1), 464.

Felsenstein, J. (1981), 'Evolutionary trees from dna sequences: A maximum likelihood approach', *Journal of Molecular Evolution* **17**(6), 368–376.

Freyhult, E., Gardner, P. & Moulton, V. (2005), 'A comparison of rna folding measures', *BMC Bioinformatics* **6**(1), 241.

Gardner, P. & Giegerich, R. (2004), 'A comprehensive comparison of comparative rna structure prediction approaches', *BMC Bioinformatics* **5**(1), 140.

Hofacker, I., Fontana, W., Stadler, P., Bonhoeffer, L., Tacker, M. & Schuster, P. (1994), 'Fast folding and comparison of rna secondary structures.', *Chemical Monthly* **125**(2), 167–188. SP: 167.

Knudsen, B. & Hein, J. (1999), 'Rna secondary structure prediction using stochastic context-free grammars and evolutionary history.', *Bioinformatics* **15**(6), 446–454.

Knudsen, B. & Hein, J. (2003), 'Pfold: Rna secondary structure prediction using stochastic context-free grammars', *Nucleic acids research* **31**(13), 3423–3428.

Lari, K. & Young, S. J. (1990), 'The estimation of stochastic context-free grammars using the inside-outside algorithm', *Computer Speech and Language* **4**(1), 35–56.

Markham, N. R., Zuker, M., Keith, J. M. & Walker, J. M. (2008), *UNAFold*, Bioinformatics, Humana Press, pp. 3–31. Methods in Molecular Biology; SP: 3.

McCaskill, J. S. (1990), 'The equilibrium partition function and base pair binding probabilities for rna secondary structure', *Biopolymers* **29**(6-7), 1105–1119.

Reeder, J., Steffen, P. & Giegerich, R. (2005), 'Effective ambiguity checking in biosequence analysis', *BMC Bioinformatics* **6**(1), 153.

Reinharz, V., Ponty, Y. & Waldispühl, J. (2013), A linear inside-outside algorithm for correcting sequencing errors in structured RNA sequences, *in* 'RECOMB - 17th Annual International Conference on Research in Computational Molecular Biology - 2013', Beijing, Chine.

Sukosd, Z., Knudsen, B., Kjems, J. & Pedersen, C. N. S. (2012), 'Ppfold 3.0: fast rna secondary structure prediction using phylogeny and auxiliary data', *Bioinformatics* **28**(20), 2691–2692.

Younger, D. (1967), 'Recognition and parsing of context-free languages in time n3', *Information and Control* **10**(2), 189–208.