Pseudoknots in RNA folding

CSCI 5481

Sreenandhreddy Kasireddy
Sebastian Moreno Ahumada
Galaan Omar

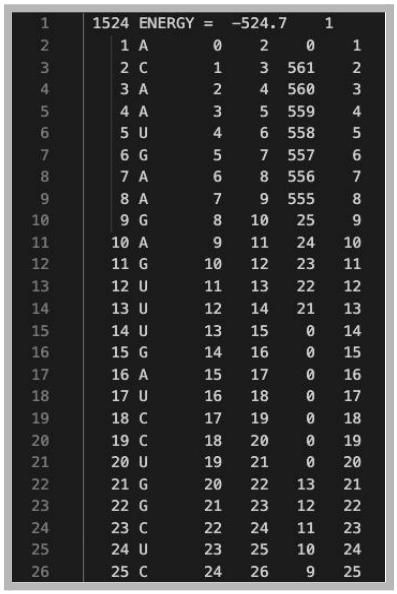
Abstract

RNA folding is the process of RNA molecules finding secondary structure as a result of intrastrand base pairings. These structures relate very closely to the biological functionality and purpose of the molecule. Services such as RNAFold offer predictive structures where the optimal secondary structure is chosen by prioritizing minimum free energy (MFE) of the structure, described as *free energy* $F = -kT \ln Q$. Moreover, applying rules such as i<j<i'<j' and i<i'<j'<j, where i/j and i'/j' are intrastrand base pairings in a consecutive sequence, allows for common structures such as helixes, hairpins, and bulges but omits pseudoknots since i<i'<j<j' is not allowed. However, pseudoknots are practical structures that are commonly found in RNA secondary structure(s) and should be considered. Here we show possible pseudoknots that can form as a result of pairing consecutive unpaired intrastrand bases, given a proposed structure from RNAFold. Using 16S rRNA genes from fourteen different species of the Bacteroides family of bacteria, we were able to find pseudoknots, visualize them, and evaluate similar fold regions amongst Bacteroides. Most notably we found that the 16S genes for Bacteroides Cellulosilyticus and Bacteroides Salyersiae have the most overlapping pseudoknot ranges, indicating a potential commonality in their RNA folding pattern. Our results and visualizations offer an extra layer in the process of RNA folding with regards to possible structures that can be created by pseudoknots and also insight into similar pseudoknot structures amongst 16S Bacteroides genes and other species.

Previous Findings

An RNA molecule folding method that finds the minimum free energy using published values of tracking and destabilizing energies has been discussed in a paper written by M Zuker and P Stiegler. The folding method is based on a dynamic programming algorithm that comes from applied mathematics, which makes the method fast and highly efficient. It allows for larger molecules to be folded than older methods have allowed in the past. They believe that the general solutions when it comes to RNA folding were typically suboptimal, especially when it came to longer chains. To combat this, the algorithm goes a step further than older folding methods by adding in additional information when using their algorithm. Some examples of this include phylogenetic data on secondary structure conservation and evidence of specific long range interactions from the examination of RNA digests. They went on to put this data directly into their algorithm and find an optimal structure that was compatible with the data. They concluded that adding in this additional information was essential due to the finding that the more information they built into the algorithm, the better the predicted folding became. (Zuker & Stiegler, 1981)

Figure 1: Example ct file from RNAfold



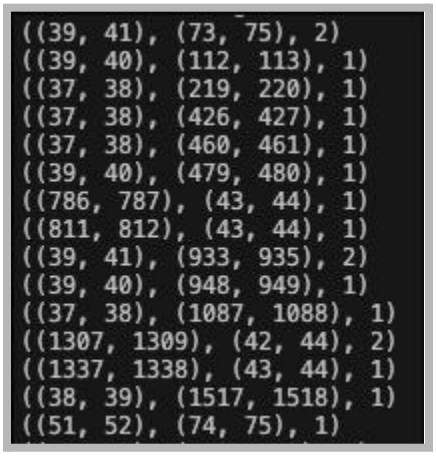
1	1524	ENERGY =	-524.7		1	
2	1	A	0	2	0	1
3	2	C	1	3	561	2
4	3	A	2	4	560	3
5	4	A	3	5	559	4
6	5	U	4	6	558	5
7	6	G	5	7	557	6
8	7	A	6	8	556	7
9	8	A	7	9	555	8
10	9	G	8	10	25	9
11	10	A	9	11	24	10
12	11	G	10	12	23	11
13	12	U	11	13	22	12
14	13	U	12	14	21	13
15	14	U	13	15	0	14
16	15	G	14	16	0	15
17	16	A	15	17	0	16
18	17	U	16	18	0	17
19	18	C	17	19	0	18
20	19	C	18	20	0	19
21	20	U	19	21	0	20
22	21	G	20	22	13	21
23	22	G	21	23	12	22
24	23	C	22	24	11	23
25	24	U	23	25	10	24
26	25	C	24	26	9	25

Results

Primarily, our study demonstrates visualization of possible pseudoknots within RNA secondary structures. Initially, we convert an RNA sequence into a ct file (*Figure 1*) using the RNAfold web server, with the ct file reflecting the optimal MFE secondary structure without pseudoknots. The MFE value can be seen in the first row. Upon obtaining the ct file, we get the indexes of the complementary base from the fifth column for each nucleotide pairing. We extracted the index ranges in which consecutive non-paired bases are located, denoted by 0's in the fifth column. We also extracted the sequence, found in the second column, into a string for later reference in the visualizer.

Next, we calculated pairs of positional index ranges for possible pseudoknots within the structure. We first started off with a nested for-loop to compare all unpaired ranges against each other. Then, we compared two sequences by using frameshifts (to account for differing size ranges) and also applying the inverse of one sequence to find the pairing complement of the other. Compliments that were found (i.e pseudoknots) had their respective pair of index ranges stored along with their length, in a tuple. In this initial approach, we encountered an issue in which a multitude of pseudoknots were of length one or two bases (*figure 2*). To correct this flaw in our logic, we implemented a ***minimum threshold*** for both the length of consecutive non-paired bases extracted (in paragraph 1) and the length of pseudoknots ranges. This allowed us to filter out smaller and less practical proposed pseudoknots. We found that the best value(s) for these thresholds were of size ***five*** since it allowed for distinct and significant knots to be visualized and analyzed.

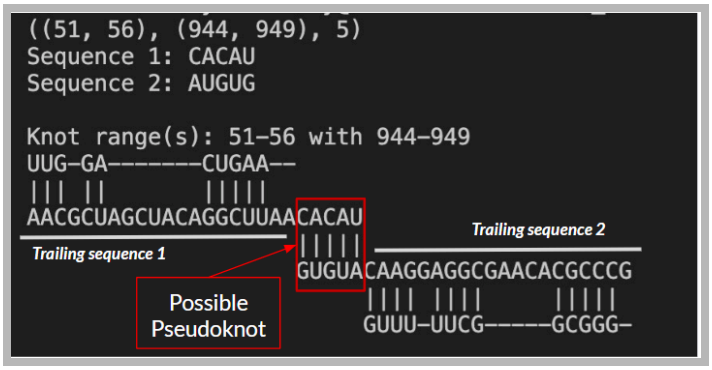
Figure 2:
Proposed pseudoknots without threshold parameters



```
((39, 41), (73, 75), 2)
((39, 40), (112, 113), 1)
((37, 38), (219, 220), 1)
((37, 38), (426, 427), 1)
((37, 38), (460, 461), 1)
((39, 40), (479, 480), 1)
((786, 787), (43, 44), 1)
((811, 812), (43, 44), 1)
((39, 41), (933, 935), 2)
((39, 40), (948, 949), 1)
((37, 38), (1087, 1088), 1)
((1307, 1309), (42, 44), 2)
((1337, 1338), (43, 44), 1)
((38, 39), (1517, 1518), 1)
((51, 52), (74, 75), 1)
```

Lastly, the visualizer for our program enhances analysis by displaying the RNA secondary structure with the pseudoknots and base pairings. As seen in *figure 3*, the visualization includes different parts. Firstly, the indexes found for a given pseudoknot are shown above the visualization. Next, the sequences respective to the indexes, along with their trailing bases, are shown with their respective pairings to highlight the nature of our found pseudoknots. Besides the pseudoknot pairing(s), we also see what the trailing sequence is paired to. This is denoted by '-' for no pairing and ' | ' for other secondary structure base pairings. This approach offers us a quick way to gain insight into the RNA secondary structure's complexity.

Figure 3: Visualizer of Pseudoknot in Terminal



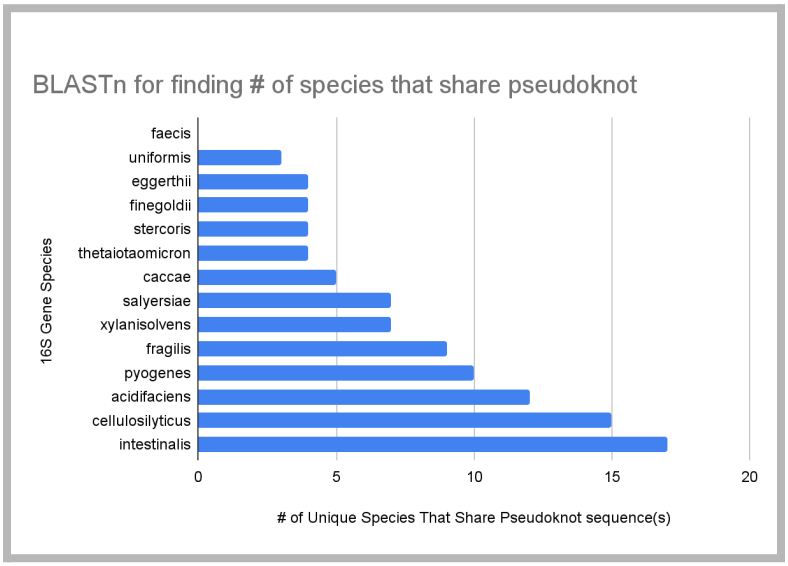
Applying our tool, we were able to analyze similar secondary pseudoknot structures amongst fourteen different 16S rRNA genes from the bacteria family, Bacteroides. This can be seen in *figure 4*. Notably, Bacteroides Cellulosilyticus and Bacteroides Salyeriae have three pseudoknot ranges in common, indicating a potential commonality in their RNA folding pattern. A full table containing all proposed pseudoknots ranges for each 16S gene can be found in ***methods & supplementary material*** below.

Figure 4: Commonality of two species with ranges

Species 1	Species 2	Pseudoknot range matches
Acidifaciens	Xylanisolvens	2
Caccae	Fragilis	1
Fragilis	Finegoldii	1
Intestinalis	Cellulosilyticus	1
Cellulosilyticus	Salyersiae	3
Pyogenes	Eggerthii	1

Figure 5:

Lastly, taking the given sequences from the visualizer (pseudoknot sequence and its trailing sequence) we were able to find other species that may have similar RNA secondary structure with BLASTn alignment. Referencing *figure 5*, we show the tally of unique species that have the same pseudoknot ranges to a respective 16S rRNA gene. We found that Bacteroides Intestinalis had the most unique species that share the proposed pseudoknot sequence(s).



Conclusion

In our study, we were able to find and visualize pseudoknots amongst 16S rRNA Bacteroides and compare similar fold regions. Applying our tool, we were able to find that Bacteroides Cellulosilyticus and Bacteroides Salyeriae have three distinct pseudoknot ranges in common with each other, thus signifying that they have some similarities in their RNA folding patterns. Although these two had the most matches, there were other pairs that shared either one or two pseudoknot range matches. Future work could include comparing results with different RNA web services, such as GeneBee that includes pseudoknot in their computed secondary structure. We can also alter our original ct file(s) by using different parameters in the RNAFold web server. Lastly, we can benchmark our performance, which may help us analyze computational speeds with larger data sets more efficiently, and this will help us optimize our current code for better performance.

Methods & Supplementary Material

Firstly, for analysis, we extracted fourteen 16S rRNA genes from different Bacteroides species from the NTB bacteria genomic listings. The ones we selected for analyzing (as seen in the last three paragraphs of the results portion) were Acidifaciens, Caccae, Fragilis, Xylanisolvens, Uniformis, Thetaiotaomicron, Faecis, Intestinalis, Stercoris, Pyogenes, Finegoldii, Eggerthii, Cellulosilyticus, and Salyersiae.

Once extracted, each gene's respective ct files were generated through RNAFold (default parameters) and then ran through our program with the parameters described above in ***results***. The functionality of our program also allows for appending proposed pseudoknots to a CSV file for a respective gene. A table with all pseudoknots for each respective 16S Bacteroides gene can be seen below in *figure 6*. *Figure 4* is a simplified and analyzed version of *figure 6*. The analysis for this was manually inspecting rows for overlapping pseudoknot ranges amongst other Bacteroides species and highlighting them to match their corresponding counterpart. Any commonalities found were then condensed into *figure 4*.

Lastly, the analytical process for finding other species with similar pseudoknot sequences with BLASTn is as follows. Firstly, proposed pseudoknot sequences and their respective trailing bases from our visualizer were run through BLASTn (default parameters). Given a list of species with 100% query matches, for each range in the pair sequences for a given pseudoknot, we only recorded the intersect between the two lists. This tailored results to only consider species that had both required sequences of the proposed pseudoknot. This was done for each proposed knot for each of the fourteen 16S Bacteroides genes. A table with our work can be seen here. Finally, we tallied the number of unique species, excluding duplicates, for a given gene's pseudoknots and produced a graph respectively, as seen in *figure 5* above.

Figure 6:

Range1 Start	Range1 End	Range2 Start	Range2 End	Knot Size	ct file/species
					./fold_files/bacteroides_acidifaciens.ct
811	816	1118	1123	5	
993	998	1308	1313	5	
1118	1124	1338	1344	6	
					./fold_files/bacteroides_caccae.ct
302	307	1288	1293	5	
					./fold_files/bacteroides_fragilis.ct
300	305	1288	1293	5	
616	621	650	655	5	
1147	1153	1287	1293	6	
					./fold_files/bacteroides_xylanisolvens.ct
811	816	1119	1124	5	
1119	1125	1339	1345	6	
					./fold_files/bacteroides_uniformis.ct
51	56	944	949	5	
					./fold_files/bacteroides_thetaiotaomicron.ct
183	188	1319	1324	5	
219	224	990	995	5	
477	482	1243	1248	5	
1082	1087	1447	1452	5	
					./fold_files/bacteroides_faecis.ct
					./fold_files/bacteroides_intestinalis.ct
440	445	833	838	5	
471	476	1273	1278	5	
					./fold_files/bacteroides_stercoris.ct
217	222	988	993	5	
214	219	1020	1025	5	
219	224	1277	1282	5	
					./fold_files/bacteroides_pyogenes.ct
165	171	1022	1028	6	
265	270	1119	1124	5	
1120	1125	1340	1345	5	
					./fold_files/bacteroides_finegoldii.ct
1147	1153	1287	1293	6	
					./fold_files/bacteroides_eggerthii
1117	1123	1337	1343	6	
					./fold_files/bacteroides_cellulosilyticus.ct
213	218	984	989	5	
440	445	833	838	5	
562	567	807	812	5	
987	992	1303	1308	5	
1112	1117	1478	1483	5	
					./fold_files/bacteroides_salyersiae.ct
15	20	778	783	5	
14	19	1292	1297	5	
213	218	984	989	5	
562	567	807	812	5	
987	992	1303	1308	5	

Files and code packet (zip)

- Ex. command
    - python3 rna_folding.py ./fold_files/bacteroides_cellulosilyticus.ct 5
        - '5' parameter stands for the minimum consecutive gap length of unpaired sequences
        - *pseudoknots of length 5 are only considered, you can edit this value on the call to find_knots() in main

Acknowledgement:

Sebastian: Code, Writeup, and Slides
Sreenandhreddy: Writeup, and Slides
Galaan: Writeup, and Slides

Previous findings citations:

(1) Michael Zuker, Patrick Stiegler, Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information, *Nucleic Acids Research*, Volume 9, Issue 1, 10 January 1981, Pages 133–148, https://doi.org/10.1093/nar/9.1.133