Fast Algorithm for the Minimum Chebyshev Distance in RNA Secondary Structure

Tiejun Ke, Changwu Wang(✉), Wenyuan Liu, and Jiaomin Liu

Yanshan University, Hebei 066004, China
wcw@ysu.edu.cn

Abstract. Minimum Chebyshev distance computation between base-pair and structures cost most time while comparing RNA secondary structures. We present a fast algorithm for speeding up the minimum Chebyshev distance computation. Based on the properties of RNA dot plots and Chebyshev distance, this algorithm uses binary search to reduce the size of base pairs and compute Chebyshev distances rapidly. Compared with O(n) time complexity of the original algorithm, the new one takes nearly [O($\log_2 n$), O(1)] time.

Keywords: RNA secondary structure · Minimum Chebyshev distance · RNA secondary structure comparison

1 Introduction

RNA is a multifunctional molecule that can be used as messenger, enzyme and structural component, which plays an important role in the whole life activity of organisms. Research on RNA not only can improve the understanding of basic biological processes but also facilitates the development of drugs, biotechnology, and genetic engineering. The function of the biological macromolecule mainly depends on its structure [1, 2]. Only when the corresponding structure of the molecule is obtained can the function and biochemical process of the molecule be explored in depth. At present, the most accurate measurement methods of RNA molecular structure, require a large amount of investment in time and economy, such as NMR, x-ray crystallography and cryogenic electron microscopy. It also requires considerable technical expertise. In that way we can see, it has very important realistic significance to use computational algorithms to predict RNA structures.

RNA structure prediction can be divided into secondary structure prediction and tertiary structure prediction. RNA secondary structure is one of the fundamental bases of its tertiary structure prediction, and can effectively improve the efficiency of tertiary structure prediction [3–5].

RNA secondary structure prediction methods includes single sequence prediction based on dynamic programming [6–9] or context-free grammar [10–12], multi-sequences comparison based on similar functions with similar structure, soft computing algorithms [13] such as genetic algorithms [14–17], simulated annealing [18, 19], artificial neural networks [20–23], and fuzzy logic [24]. These methods still have space to promote accuracy and performance [25].

Q. Li et al. (Eds.): BROADNETS 2019, LNICST 303, pp. 247–260, 2019.
https://doi.org/10.1007/978-3-030-36442-7_16

The above prediction methods often involve judging the similarity between different structures. For example, [26] generates many candidate structures and then chooses the representative according to the similarity between these structures by using Boltzmann sampling methods such as minimum free energy folding and sub-optimal free energy folding [27, 28], as well as calculating the probability of base pairing.

The similarity measurement method of the secondary structure is related to its representation. For example, tree edit distance under tree representation, base pair (BP) distance [28], relaxed base pair (RBP) scoring method [29] and Hausdorff distance [30–33] under base pair set representation. These methods are helpful to select one or a few representative structures from large quantities of candidate sets.

At present, the performance of the secondary structure prediction algorithm still needs to be improved. In this paper, we present a fast algorithm for computing the minimum Chebyshev distance between RNA secondary structures. According to the properties of Chebyshev distance calculation and the characteristics of RNA secondary structure, we design an auxiliary computing structure, which reduces the amount of calculation and effectively improves the computing efficiency of the related RNA secondary structure prediction method.

2 Minimum Chebyshev Distance Between RNA Secondary Structures

2.1 RNA Secondary Structure

Definition 1. RNA sequence R is composed of four types of bases A, U, G, and C. Its length is n. Its form is defined as follows:

$$R = r_1 r_2 r_3 \ldots r_n, \text{ where } r_i \in \{\mathrm{A}, \mathrm{U}, \mathrm{G}, \mathrm{C}\}, i = 1, 2, \ldots n$$

The left end of R represents the starting point of the base sequence, usually represented by 5'; the right end is the endpoint, usually by 3'. This string represents a single directed biological chain consisting of four bases that starts at 5' and ends at 3'. RNA sequences fold themselves through intramolecular base pairs and stabilize their structures by hydrogen bonds that form base pairs.

There are three cases of base pairing in RNA secondary structure: strong Watson Crick pairing for C-G (G-C) and A-U (U-A), and unstable wobble pairing for G-U (U-G).

Definition 2. RNA secondary structure is a base pair set S on its sequence R. Its elements (i, j) represent the pairing of r_i and r_j in the sequence, where $1 < i < j < n$. At the same time, the following constraints must be satisfied:

1. $(i,j) \in S \Rightarrow j - i \geq 4$;
2. $(i,j) \in S \Rightarrow \begin{cases} (i,j') \notin S, (j' \neq j) \\ or \\ (i',j) \notin S (i \neq i') \end{cases}$;

$$3.\ \left.\begin{array}{l}(i,j)\in S\\(i',j')\in S\\i\le i'\end{array}\right\}\Rightarrow\left\{\begin{array}{c}i=i',j=j'\\or\\i<j<i'<j'\\or\\i<i'<j'<j\end{array}\right.$$

Constraint 1 indicates that the ring formed by base pairing will not be too sharp (the loop area in Fig. 1). Constraint 2 means that the same base participates in at most one base pairing. Constraint 3 denotes that RNA secondary structure does not contain pseudoknots.

Definition 3. RNA Sequence R's secondary structure plane graph [34] is a labeled graph with n vertices $G = (V, E)$, where $V = \{r_1, r_2, \ldots, r_n\}$, and its adjacency matrix $A = \{a_{i,j}\}$ satisfies the following conditions:

1. $a_{i\cdot i+1} = 1\ 0 < i < n$;
2. $(i,j) \in S \Leftrightarrow \begin{cases} a_{i\cdot j} = 1 \\ j - i \ge 4 \end{cases}$.

Take the RNA molecule numbered PDB_00194 in RNA STRAND (the RNA secondary STRucture and statistical ANalysis Database) as an example. According to *Definition* 1, its sequence is:

$$R = \text{UGCUCCUAGUACGAGAGGACCGGAGUG}$$

According to Definition 2, its secondary structure can be expressed as:

$$S = \{(2,26),(3,25),(4,24),(5,23),(6,22),(7,21),(10,19),(11,18),(12,17),(13,16)\}$$

According to Definition 3, the secondary structure planar graph is shown in Fig. 1.

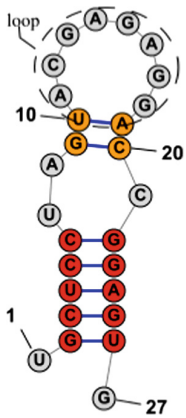

Fig. 1. PDB_00194 secondary structure planar graph. Note: This figure is generated by RNAStructure [35]

2.2 Chebyshev Distance

Definition 4. If two vectors or two points p and q have coordinates of p_i and q_i respectively, then the Chebyshev distance between them is defined as follows:

$$d_{Cheb}(p,q) = \max_i(|p_i - q_i|)$$

In the 2-dimensional plane, the Chebyshev distance between two points $p(i_1, j_1)$ and $q(i_2, j_2)$ is:

$$d_{Cheb}(p,q) = max\{|i_1 - i_2|, |j_1 - j_2|\}$$

Property 1: In a two-dimensional plane, a set of points with equal Chebyshev distance to a certain point constitutes a square [36].

If the fixed point and moving point are satisfied $d_{Cheb}(p,m) = r(r > 0)$ then $\max\{|i - i'|, |j - j'|\} = r$. This formula is equivalent to $\begin{cases} |i - i'| = r \\ |j - j'| \le r \end{cases}$ or $\begin{cases} |j - j'| = r \\ |i - i'| \le r \end{cases}$.

Then the trajectory of point m is a square with point p as the center and side length $2r$, as shown in Fig. 2.

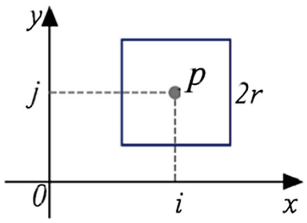

Fig. 2. Trajectories of points with the same Chebyshev distance from a certain point p.

2.3 Minimum Chebyshev Distance of RNA Secondary Structure

Assume that there are two RNA secondary structures S_1 and S_2 respectively, where $(i,j) \in S_1$, $(i',j') \in S_2$. To calculate the minimum Chebyshev distance between S_2 and base pairs (i, j) in S_1, we can operate as the following steps:

Step 1: Calculate the distance between two base pairs:

$$d_{Cheb}((i,j),(i',j')) = \max(|i - i'|, |j - j'|)$$

Step 2: Calculate the distance between base pairs (i, j) and S_2:

$$d_a((i,j),S_2) = \min_{(i',j') \in S_2}\{d_{Cheb}((i,j),(i',j'))\}$$

The specific calculation process is shown in Algorithm 1.

Algorithm 1. original minimum Chebyshev distance between base pair and secondary Structure

Input: Secondary structure $S = \{(i_1, j_1)(i_2, j_2), \ldots, (i_m, j_m)\}$; base pair $p = (i', j')$
Output: minimum Chebyshev distance *mindist* between p and S
processing:

```
mindist = chebyshev((i', j'),(i_1, j_1))
for t = 2, 3, ..., m do
    dist = chebyshev((i', j'),(i_t, j_t))
    if mindist > dist then
        mindist = dist
    end if
end for
```

Note: chebyshev((*i*, *j*), (*i*', *j*') is used to calculate the Chebyshev distance between point (i, j) and point (i', j').

3 Fast Algorithm for Computing Minimum Chebyshev Distance of RNA Secondary Structure

3.1 Details of the Algorithm Idea

Definition 5. RNA secondary structure dot plot [37] is shown in Fig. 3. In the $n*n$ square grid, the intersection points of row i and column j are used to represent base pairs (i, j), which represent the entire secondary structure S, where n represents the length of the RNA sequence.

The grid in Fig. 3 and its black square point are shown as a dot plot representation of PDB_00194, where the numbers in the x-axis and y-axis directions represent the position of the base in the RNA respectively, and the base pairs are represented by black squares. All black square points in the ellipse represent the secondary structure of PDB_00194. The dot plot can be viewed as part of a plane rectangular coordinate system.

The minimum chebyshev distance from one base pair in the RNA secondary structure to another secondary structure is the minimum chebyshev distance calculated from a point to a point set. Let a base pair in the RNA secondary structure be $p(i', j')$ and another secondary structure S where and the minimum chebyshev distance between p and S is r_{min}. If the base pair p is the center and r_{min} is the radius of square G, then G and S will obtain the following properties:

Property 2: G does not contain any base pairs in S
Property 3: At least one of the base pairs in S is on the edge of G.

Based on the above two properties, the algorithm uses the binary search method to quickly screen out the square G that meets the condition, and the radius is the minimum Chebyshev distance.

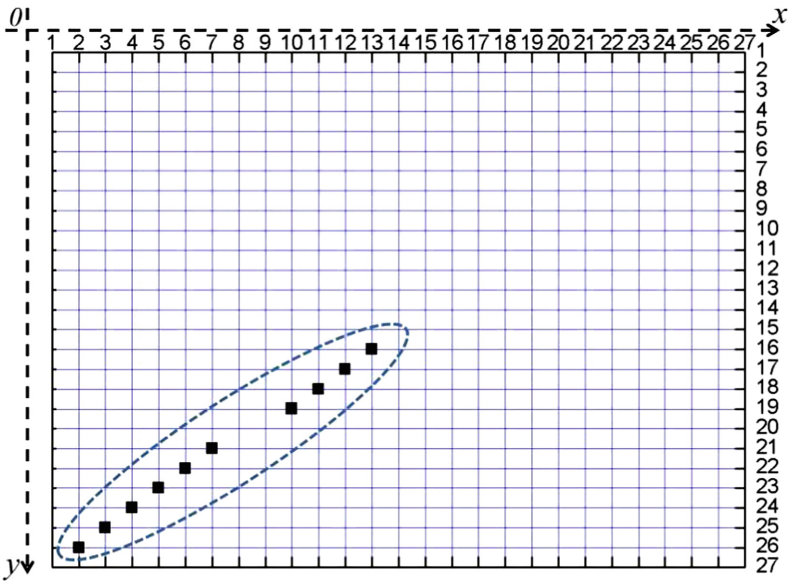

Fig. 3. PDB_00194 dot plot

To facilitate the description of the algorithm, i in the base pair (i, j) is referred to as the *left base*, and j is referred to as the *right base*. Set the secondary structure $S = \{(i_1, j_1), (i_2, j_2), \ldots, (i_m, j_m)\}$, and the other two sets of structures S_x and S_y are

$$p_x = (i, j, leftNN, rightNN) \in S_x$$

$$p_y = (i', j', downNN, upNN) \in S_y,$$

p_x, p_y has the following characteristics:

1. The elements in S_x and S_y are arranged in ascending order according to left base i and right base j respectively;
2. $1 \leq i \leq i_m, 1 \leq j' \leq \max\{j_1, j_2, \ldots, j_m\}$;
3. (i, j) and (i', j') represent the pairing of bases. If the i-number bases are not paired, then j = 0; if the j′-number bases are not paired, then i′ = 0;
4. *LeftNN* and *rightNN* in p_x represent the nearest left and right nearest neighbor points in the x-axis direction of the secondary structure dot plot that are closest to the straight line $x = i$. Similarly, when $i' = 0$, *downNN* and *upNN* in p_y refer to the nearest lower nearest neighbor and upper nearest neighbor in y-axis direction from $y = j'$.

S_x and S_y are used to quickly locate the nearest point to a fixed point in the direction of the x-axis and y-axis respectively. In this paper, they are called NNI (Nearest Neighbor Indexes), represented by arrays in the algorithm.

The calculation process mainly consists of the following two steps:

Step 1: Generate the NNI

Take PDB_00194 as an example, the base pair set which represents its secondary structure is sorted by i by default, while the set sorted by j is {(13,16), (12,17), (11,18), (10,19), (7,21), (6,22), (5,23), (4,24), (3,25), (2,26)}, and the generated NNI is shown in Fig. 4.

Step 2: Calculate the minimum Chebyshev distance between the base pair p(i', j') and S

Quickly locate the closest point in the x-axis direction to i' based on the generated NNI:

Case 1: If there is a base pair in S that coincides with $p(i', j')$, we can determine that the minimum Chebyshev distance is 0, and end the calculation;
Case 2: If i or j has a paired base in S, return the base pair;
Case 3: If neither i nor j in S has no paired bases, return two base pairs in the x-axis direction from $x = i$ and the y-axis direction closest to $y = j$ respectively.

Calculate the minimum Chebyshev distance r_0 according to the base pair returned in case 2 or case 3. Let the square area with p(I′, j′) as the center and r_0 as the initial radius be G. According to property 2, first determine whether there is a base pair in G and if so, reduce the radius of G to r/2; if it does not exist, and there are no base pairs on the four sides of G, then the radius is enlarged to 3r/4. Thus, using the binary search to

No	*i*	*j*	*LeftNN*	*RightNN*
1	1	0	0	2
2	2	26	0	3
3	3	25	2	4
4	4	24	3	5
5	5	23	4	6
6	6	22	5	7
7	7	21	6	10
8	8	0	7	10
9	9	0	7	10
10	10	19	7	11
11	11	18	10	12
12	12	17	11	13
13	13	16	12	0

a)NNI sorted by *i*

No	*i*	*j*	*DownNN*	*UpNN*
1	0	1	0	16
2	0	2	0	16
3	0	3	0	16
4	0	4	0	16
5	0	5	0	16
6	0	6	0	16
7	0	7	0	16
8	0	8	0	16
9	0	9	0	16
10	0	10	0	16
11	0	11	0	16
12	0	12	0	16
13	0	13	0	16
14	0	14	0	16
15	0	15	0	16
16	13	16	0	17
17	12	17	16	18
18	11	18	17	19
19	10	19	18	21
20	0	20	19	21
21	7	21	19	22
22	6	22	21	23
23	5	23	22	24
24	4	24	23	25
25	3	25	24	26
26	2	26	25	0

b)NNI sorted by *j*

Fig. 4. NNI of PDB 00194

scale the radius of G until the property 3 is satisfied, that is, G is empty, and at least one base pair is on the four sides, the radius of the current G is the minimum Chebyshev distance sought. The specific process is shown in Algorithm 2.

Figure 5 shows the minimum Chebyshev distance calculation process between the points p = (12, 25) and the PDB_00194 structure.

Algorithm 2. fast calculation for the minimum Chebyshev distance between base pair and secondary structure

Input: point $p(i', j')$, secondary structure $S = \{(i_1, j_1), (i_2, j_2), ..., (i_m, j_m)\}$

Output: minimum Chebyshev distance r

Process:

According to S to generate S_x, S_y, and

$$S_x = \{(g_k, h_k, leftNN_k, rightNN_k)\}_{k=1}^{i_m},$$

$$S_y = \{(u_k, v_k, downNN_k, upNN_k)\}_{k=1}^{\max(j_1, j_2, ..., j_m)}$$

///To get the initial fixed-length Chebyshev distance square radius

if $i > i_m$ **then**

$Xchebydist$ = chebyshev($p, S_x.leftNN_{i_m}$)

else

$Xchebydist$=min(chebyshev(p,$S_x.leftNN_{i'}$),
chebyshev(p,$S_x.rightNN_{i'}$))

end if

if $Xchebydist$==0 **then**

return 0

end if

Similarly, the minimum Chebyshev distance in the y-axis direction from the nearest two points to the point p can be obtained as $Ychebydist$

$SquareRadius$=min($Xchebydist$, $Ychebydist$)

//Find the minimum Chebyshev distance using the //binary search method

$radiusStart$ = 0

$radiusEnd$ = $SquareRadius$

while $pointnumber$>0 **do**

r=($radiusStart$+$radiusEnd$)/2

$pointnumber$=pointinsquare(p,r,S_x,S_y)

if $pointnumber$ > 0 **then**

$radiusEnd$ = r - 1

else if sidehaspoint(p,r,S_x,S_y) == False **then**

$radiusStart$ = r +1

$pointnumber$ = 1// Let the loop continue

end if

end while

Note: pointinsquare(p, r, S_x, S_y) is used to calculate the number of base pairs of the secondary structure S in the square with the radius r and the center point p; sidehaspoint(p, r, S_x, S_y) is used to determine whether the secondary structure S has a base pair just on or in the square with the radius r and the center point p.

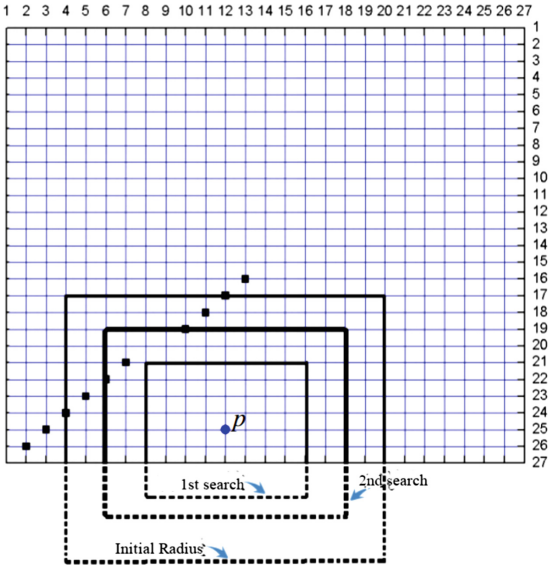

Fig. 5. Schematic diagram of the minimum Chebyshev distance calculation process

It can be seen intuitively from Fig. 5 that after the algorithm zooms through the search radius of 3 times, the minimum Chebyshev distance from the point p is 5.

3.2 Algorithm Complexity Analysis

Assuming that the length of the RNA sequence R is n, and the set of secondary structures generated by folding is $\{S_1, S_2, \ldots, S_m\}$. We want to calculate the minimum Chebyshev distance from a base pair of S_1 to S_2. According to Definition 3, it can be seen that the number of base pairs in S_1 and S_2 does not exceed $n/2$.

When solving the problem with Algorithm 1, a base pair in S_1 needs to traverse all base pairs in S_2 to calculate Chebyshev distance to find the minimum value. The number of traversals is at most $n/2$, and the time complexity is $O(n)$.

NNI generation and traversal calculation analysis are needed when solving with Algorithm 2. According to the characteristics of RNA secondary structure in Definition 2, the NNI could be generated in $O(n)$ time complexity by the counting sorting algorithm. Since the NNI only needs to be initialized before comparison, if each secondary structure in the set calculates the minimum Chebyshev distance from other secondary structures, the average time complexity is $O(n/m^2)$. According to Table 1 in the following experiment, this time is negligible.

When traversing the computational analysis, the best case traversal is 1, so the time complexity is O(1); in the worst case, the time complexity $O(\log_2 n)$ of the binary search. Therefore, the time complexity of Algorithm 2 is nearly $[O(\log_2 n), O(1)]$.

Compared to the original algorithm, the improved algorithm requires two arrays sorted by the left base and right base respectively, and the length of the array is determined by the maximum number of base numbers participating in the pairing. In

the worst case, the lengths of the two arrays are all Sequence length, so its spatial complexity is O(n).

4 Experiment and Results Analysis

4.1 Experimental Data

The experimental data (see Table 1 for details) are derived from mRNA sequences of different lengths from the NCBI (National Center for Biotechnology). The data details are shown in Table 1 where TSC means the times for structure comparison and then the default parameters are set by mfold (v2.3). Calculate the fold, and finally compare the CT secondary data of the RNA secondary structure generated by the same sequence folding.

Table 1. Experimental data details

Locus	Gene	Length	Number of generated structures	TSC	Base pair repetition rate
NM_130776	XAGE3	493	26	650	48.29%
NM_002988	CCL18	793	24	552	37.52%
NM_002475	MYL6B	869	27	702	36.09%
NM_033142	CGB7	880	22	462	41.05%
NM_016815	GYpC	1019	41	1640	45.48%
NM_004019	DMD	1634	33	1056	45.42%
NM_002231	CD82	1715	49	2352	42.81%
NM_013958	NRG1	1740	38	1406	48.60%
NM_000781	CYp11A1	1821	65	4160	39.85%
NM_021785	RAI2	2338	35	1190	40.00%
NM_004381	ATF6B	2622	35	1190	44.71%
NM_033172	B3GALT5	2711	43	1806	45.70%
NM_005656	TMpRSS2	3226	42	1722	60.89%

4.2 Experimental Environment

The experimental environment is as follows:

Operating system: ubuntu 17.10 64-bit version; development language: python 2.7.14; CPU: Core i7 3630QM; memory: 16 GB.

4.3 Experimental Steps

Experiments were carried out by taking the common Hausdorff distance and RBP scoring methods as examples to calculate the distance between RNA secondary

structures. Each method uses the minimum Chebyshev distance original algorithm and the improved algorithm proposed in this paper to compare the running time overhead.

Because there are more identical base pairs in different secondary structures generated by the same sequence, the base pair repetition in the structure comparison is considered in the experiment, and the Algorithm 1 is slightly improved, as shown in the Algorithm 3.

Algorithm 3. Improved original algorithm

Input: secondary structure $S = \{(i_1, j_1),(i_2, j_2), \ldots ,(i_m, j_m)\}$; base pair $p = (i', j')$
Output: Minimum Chebyshev distance *mindist* between p and S
Process:

```
mindist = chebyshev((i', j'),(i_1, j_1))
t = 2
while (mindist > 0 and t < m + 1) do
   dist = chebyshev((i', j'),(i_t, j_t))
   if mindist > dist then
      mindist = dist
   end if
   t = t + 1
end while
```

4.4 Experimental Results

The experiment uses the running time of the average structure comparison in the same sequence to measure the pros and cons of the algorithm. For example, the sequence NM_130776 with length 493 is folded to generate 26 candidate structures. The comparison times of the two structures are 26 * (26 − 1) = 650 times. The total time of the improved hausdorff distance calculation algorithm is 7.291402102 s compared with the original 130.0833249 s, and the average time of the improved algorithm is 11.2175 417 ms compared with the original 200.1281922 ms.

The experimental results are as follows:

4.5 Analysis of Results

It can be seen from Table 1 that the secondary sequence generated by the same sequence has a higher rate of coincidence of base pairs, and further exerts the advantage of rapid calculation of the improved algorithm. The experimental results in Table 2 show that the improved algorithm significantly improves the calculation speed.

Table 2. Experimental results

Locus	TSC	Average running time per structure (unit: milliseconds)			
		Improved Hausdorff	Original Hausdorff	Improved RBP	Original RBP
NM_130776	650	11.2175417	200.1281922	11.49805692	187.2433091
NM_002988	552	29.89899076	504.4981216	29.81576668	471.2811341
NM_002475	702	50.83871499	985.6605442	53.64886037	925.020497
NM_033142	462	32.92536788	670.0163032	34.59124028	622.7872052
NM_016815	1640	53.65096767	1292.021342	56.01231529	1204.217655
NM_004019	1056	82.25197819	1951.261859	85.54889781	1800.510011
NM_002231	2352	112.8718082	2922.761558	120.2368014	2744.817355
NM_013958	1406	92.43744808	2507.75789	97.35820982	2313.373188
NM_000781	4160	154.4170339	3445.17768	162.1970053	3162.275498
NM_021785	1190	174.5501597	4165.3774	187.9682958	3904.11921
NM_004381	1190	270.955447	6404.746261	293.4771656	5991.996467
NM_033172	1806	244.5065211	7262.653743	261.8198942	6713.290925
NM_005656	1722	201.0514477	9254.406022	213.9117422	8557.3886

5 Conclusion

The distance calculation optimization method proposed in this paper is a great improvement in the computational efficiency of the original RNA secondary structure minimum Chebyshev distance algorithm. This method locates the target calculation points quickly by using the characteristics and regularity of RNA secondary structure. It also reduces the calculation range. Although only different structures generated by the same sequence are tested in this paper. This method can be generalized to compare the structures generated by different sequences.

The minimum Chebyshev distance also has applications in image processing, such as digital image forensics [38], cell tracking [39], hyperspectral imaging [40], and neural network research [41, 42]. The algorithm can also be used to calculate the minimum Chebyshev distance between other points and point sets.

References

1. National Natural Science Foundation of China, Chinese Academy of Sciences. Major Scientific Issues in the Study of RNA in China's Discipline Development Strategy. Science Press, China (2017)
2. Krieger, E., Sander, B., Vriend, G.: Homology modeling. In: Structural Bioinformatics, vol. 44 (2003)
3. Thiel, B.C., Flamm, C., Hofacker, I.L.: RNA structure prediction: from 2D to 3D. Emerg. Top. Life Sci. **1**(3), 275–285 (2017)

4. Galvanek, R., Hoksza, D.: Template-based prediction of RNA tertiary structure using its predicted secondary structure. In: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 2238–2240. IEEE (2017)
5. Zhao, Y., Huang, Y., Gong, Z., Wang, Y., Man, J., Xiao, Y.: Automated and fast building of three-dimensional RNA structures. Sci. Rep. **2**, 1–6 (2012)
6. Waterman, M.S., Smith, T.F.: RNA secondary structure: a complete mathematical analysis. Math. Biosci. **42**(3–4), 257–266 (1978)
7. Nussinov, R., Jacobson, A.B.: Fast algorithm for predicting the secondary structure of single-stranded RNA. Proc. Natl. Acad. Sci. U.S.A. **77**(11), 6309–6313 (1980)
8. Waterman, M.S., Smith, T.F.: Rapid dynamic programming algorithms for RNA secondary structure. Adv. Appl. Math. **7**(4), 455–464 (1986)
9. Akutsu, T.: Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots. Discret. Appl. Math. **104**(1–3), 45–62 (2000)
10. García, R.: Prediction of RNA pseudoknotted secondary structure using stochastic context free grammars (SCFG). CLEI Electron. J. **9**(2) (2006)
11. Sakakibara, Y., Brown, M., Hughey, R., Mian, I.S., Sjölander, K., Underwood, R.C., Haussler, D.: Stochastic context-free grammars for tRNA modeling. Nucleic Acids Res. **22** (23), 5112–5120 (1994)
12. Anderson, J.W.J., Tataru, P., Staines, J., Hein, J., Lyngsø, R.: Evolving stochastic context-free grammars for RNA secondary structure prediction. BMC Bioinform. **13**(1), 78 (2012)
13. Pal, S.K., Ray, S.S., Ganivada, A.: RNA secondary structure prediction: soft computing perspective. Granular Neural Networks, Pattern Recognition and Bioinformatics. SCI, vol. 712, pp. 195–222. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57115-7_7
14. Batenburg, F.H.V., Gultyaev, A.P., Pleij, C.W.: An APL-programmed genetic algorithm for the prediction of RNA secondary structure. J. Theor. Biol. **174**(3), 269–280 (1995)
15. Shapiro, B.A., Bengali, D., Kasprzak, W., Wu, J.C.: RNA folding pathway functional intermediates: their prediction and analysis. J. Mol. Biol. **312**, 27–44 (2001)
16. Wiese, K.C., Deschenes, A., Glen, E.: Permutation based RNA secondary structure prediction via a genetic algorithm. In: Proceedings of the 2003 Congress on Evolutionary Computation, pp. 335–342 (2003)
17. Wiese, K.C., Deschênes, A.A., Hendriks, A.G.: RnaPredict - an evolutionary algorithm for RNA secondary structure prediction. IEEE/ACM Trans. Comput. Biol. Bioinf. **5**(1), 25–41 (2008)
18. Schmitz, M., Steger, G.: Description of RNA folding by simulated annealing. J. Mol. Biol. **255**(1), 254–266 (1996)
19. Tsang, H.H., Wiese, K.C.: SARNA-predict: accuracy improvement of RNA secondary structure prediction using permutation based simulated annealing. IEEE/ACM Trans. Comput. Biol. Bioinf. **7**(4), 727–740 (2010)
20. Liu, Q., Ye, X., Zhang, Y.: A Hopfield neural network based algorithm for RNA secondary structure prediction. In: Proceedings of the 1st International Conference on Multi-symposiums on Computer and Computational Sciences, pp. 1–7 (2006)
21. Haynes, T., Knisley, D., Knisley, J.: Using a neural network to identify secondary RNA structures quantified by graphical invariants. MATCH Commun. Math. Comput. Chem. **60**, 277–290 (2008)
22. Zou, Q., Zhao, T., Liu, Y., Guo, M.: Predicting RNA secondary structure based on the class information and Hopfield network. Comput. Biol. Med. **39**(3), 206–214 (2009)
23. Koessler, D.R., Knisley, D.J., Knisley, J., Haynes, T.: A predictive model for secondary RNA structure using graph theory and a neural network. BMC Bioinform. **11**, S6–S21 (2010)

24. Song, D., Deng, Z.: A fuzzy dynamic programming approach to predict RNA secondary structure. In: Bücher, P., Moret, B.M.E. (eds.) WABI 2006. LNCS, vol. 4175, pp. 242–251. Springer, Heidelberg (2006). https://doi.org/10.1007/11851561_23
25. Oluoch, I.K., Akalin, A., Vural, Y., Canbay, Y.: A review on RNA secondary structure prediction algorithms. In: 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), Ankara, Turkey, pp. 18–23. IEEE (2018)
26. Zuker, M.: Mfold web server for nucleic acid folding and hybridization prediction. Nucleic Acids Res. **31**, 3406–3415 (2003)
27. Markham, N.R., Zuker, M.: UNAFold: software for nucleic acid folding and hybridization. In: Keith, J.M. (ed.) Bioinformatics: Structure, Functions and Applications, vol. 453, pp. 3–31. Humana Press, Totowa (2008)
28. Ding, Y., Chan, C.Y., Lawrence, C.E.: Clustering of RNA secondary structures with application to messenger RNAs. J. Mol. Biol. **359**, 554–571 (2006)
29. Agius, P., Bennett, K.P., Zuker, M.: Comparing RNA secondary structures using a relaxed base-pair score. RNA **16**(5), 865–878 (2010)
30. Schirmer, S., Ponty, Y., Giegerich, R.: Introduction to RNA secondary structure comparison. Methods Mol. Biol. **1097**(1097), 247–273 (2014)
31. Lopez, M.A., Reisner, S.: Hausdorff approximation of convex polygons. Comput. Geom. **2** (32), 139–158 (2005)
32. Moulton, V., Zuker, M., Steel, M., Pointon, R., Penny, D.: Metrics on RNA secondary structures. J. Comput. Biol.: J. Comput. Mol. Cell Biol. **7**(1–2), 277–292 (2000)
33. Chen, Q., Chen, B., Zhang, C.: Interval based similarity for function classification of RNA pseudoknots. In: Chen, Q., Chen, B., Zhang, C. (eds.) Intelligent Strategies for Pathway Mining. LNCS (LNAI), vol. 8335, pp. 175–192. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04172-8_8
34. Fu, W., Huang, J., Xu, L.: RNA secondary structure representation and conversion algorithms. Comput. Eng. Appl. (14), 43–45, 85 (2004)
35. Reuter, J.S., Mathews, D.H.: RNAstructure: software for RNA secondary structure prediction and analysis. BMC Bioinform. **11**(1), 129 (2010)
36. Zhang, Y.: Track exploration under Chebyshev distance. Math. Teach. **8**, 19–22 (2015)
37. Tsang, H.H., Jacob, C.: RNADPCompare: an algorithm for comparing RNA secondary structures based on image processing techniques, pp. 1288–1295. IEEE (2011)
38. Kang, X., Wei, S.: Identifying tampered regions using singular value decomposition in digital image forensics. In: 2008 International Conference on Computer Science and Software Engineering, Wuhan, Hubei, pp. 926–930 (2008)
39. Chowdhury, A.S., Chatterjee, R., Ghosh, M., Ray, N.: Cell tracking in video microscopy using bipartite graph matching. In: 2010 20th International Conference on pattern Recognition, Istanbul, pp. 2456–2459 (2010)
40. Demirci, S., Erer, I., Ersoy, O.: Weighted Chebyshev distance classification method for hyperspectral imaging. In: Proceedings of SPIE 9482, Next-Generation Spectroscopic Technologies VIII, p. 948218 (2015)
41. Ritter, G.X., Urcid-Serrano, G., Schmalz, M.S.: Lattice associative memories that are robust in the presence of noise. In: Proceedings of SPIE 5916, Mathematical Methods in Pattern and Image Analysis, p. 59160Q (2005)
42. Ritter, G.X., Urcid, G.: Learning in lattice neural networks that employ dendritic computing. In: Kaburlasos, V.G., Ritter, G.X. (eds.) Computational Intelligence Based on Lattice Theory, vol. 67, pp. 25–44. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72687-6_2