アルゴリズム 62－9
（1998．5．20）

RNA 二次構造予測のための近似アルゴリズム

阿久津 達也
東京大学医科学研究所ヒトゲノム解析センター

RNA 二次構造予測は計算生物学の中でも良く研究されている重要な問題の一つである。RNA 二次構造予測は、最も基本的な定式化においては、RNA 配列 (a,u,g,c からなる文字列) が与えられた時、(交差しないように) マッチする相補対 (a,u 対もしくは g,c 対) の個数を最大化する問題として定義される。この問題に対しては、従来、動的計画法に基づく $O(n^3)$ 時間のアルゴリズムしか知られていなかった。本稿では、この問題に対する $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ 時間アルゴリズムを示すとともに、任意の定数 $\epsilon > 0$ に対し $1-\epsilon$ 以内の近似解を求める $O(n^{2.776})$ 時間アルゴリズムを示す。さらに、より現実的な定義のもとでの RNA 二次構造予測問題に対する近似アルゴリズムも示す。

Approximation and Exact Algorithms for Global Free-Energy Minimization of RNA Secondary Structures

Tatsuya Akutsu
Human Genome Center, Institute of Medical Science, University of Tokyo
4-6-1 Shirkanedai, Minato-ku, Tokyo 108-8639, Japan
takutsu@ims.u-tokyo.ac.jp

RNA secondary structure prediction is one of important problems in computational biology. In a simplest form, it is formalized as a problem of, given an RNA sequence of length n, maximizing the number of complementary base pairs. It is well known that this problem can be solved in $O(n^3)$ time by a dynamic programming procedure. Although not a few computational studies have been done on RNA secondary structure prediction, the time complexity had not been improved for this version of the problem. This paper shows, for this version of the problem, a slightly improved $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time exact algorithm, and an $O(n^{2.776})$ time algorithm with guaranteed approximation ratio $1-\epsilon$ for any $\epsilon > 0$. Extensions of the approximation algorithm are also shown for more practical versions of the problem.

1 Introduction

A lot of computational studies have been done for *RNA secondary structure prediction*. This is a problem of, given an RNA sequence, finding its correct secondary structure (a planar graph like structure, see Fig. 1). Usually, RNA secondary structure prediction is modeled as a *free-energy minimization* problem [12, 15, 20]. For this problem, Watermann and Smith proposed simple DP (*dynamic programming*) algorithms [18]. Zuker and Stiegler also proposed a similar algorithm based on DP [21]. The time complexities of those DP algorithms were $O(n^3)$ if we ignore the *destabilizing energy* due to *loop regions*, otherwise it was at least $O(n^4)$.

Although no further improvement had been done on global free-energy minization, several important improvements have been done for finding *locally stabilizing substructures* in an RNA secondary structure [8, 20] Watermann and Smith developed an $O(n^3)$ time algorithm for an arbitrary destabilizing energy function [19]. Kanehisa and Goad developed an $O(n^2)$ time algorithm for a *linear* destabilizing energy function [9]. Eppstein, Galil and Giancarlo developed an $O(n^2\log^2 n)$ time algorithm for a concave or convex destabilizing energy function [5]. Slight improvements have been done for the same case [6, 10].

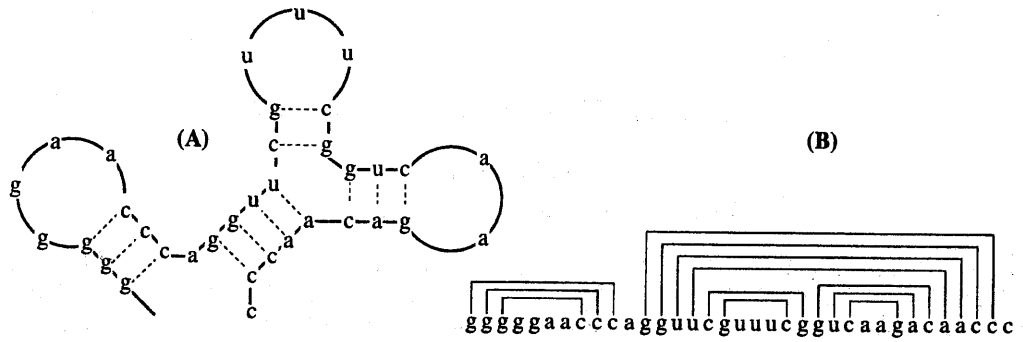

Figure 1: Two representations of RNA secondary structure: (A) 'Clover leaf' representation similar to real structure; (B) Sequence is represented on the horizontal axis. In a basic version, RNA secondary structure prediction is defined as a problem of maximizing the number of base pairs (i.e., (a,u) and (g,c) pairs) which do not intersect each other.

In this paper, we first consider the problem of *maximizing the number of complementary base pairs* (see Fig. 1) [12, 20], which is a simplified and basic version of global free-energy minimization of RNA secondary structure, and denoted by $\mathcal{RNA}_0$ in this paper. A pair of residue (x, y) is called a (*complementary*) *base pair* if $\{x, y\} = \{\mathtt{a}, \mathtt{u}\}$ or $\{x, y\} = \{\mathtt{g}, \mathtt{c}\}$. Although we do not treat $\{\mathtt{g}, \mathtt{u}\}$ as a base pair, all the results can be modified for such a case in a straight-forward way. A set of pairs of indices

$$M = \{(i, j) |\ 1 \leq i < j \leq n,\ (a_i, a_j) \text{ is a base pair } \}$$

is called an *RNA secondary structure* if no distinct pairs $(a_i, a_j), (a_h, a_k)$ in M satisfy $i \leq h \leq j \leq k$ (see Fig. 1). The score of M is defined as the number of base pairs in M (i.e., cardinality of M), and denoted by $score(M)$. Then, $\mathcal{RNA}_0$ is defined as follows: given a string $A = a_1 a_2 \ldots a_n$ over an alphabet $\Sigma = \{\mathtt{a}, \mathtt{c}, \mathtt{g}, \mathtt{u}\}$, to find an RNA secondary structure M with the maximum score. In $\mathcal{RNA}_0$, such an RNA secondary structure is also called an *optimal* RNA secondary structure, and denoted by $OPT_0(A)$.

It is well known that the score of $OPT_0(A)$ can be computed in $O(n^3)$ time using the following simple DP procedure (denoted by $\mathcal{DP}_0$):

$$S(i, j) = \max \begin{cases} S(i-1, j+1) + \mu(a_i, a_j), \\ \max_{i<k\leq j} S(i, k-1) + S(k, j), \end{cases}$$

where we let $S(i, j) = 0$ for all $i \geq j$, and $\mu(x, y) = 1$ if (x, y) is a base pair, otherwise $\mu(x, y) = 0$. Note that the score of $OPT_0(A)$ is given by $S(1, n)$. $OPT_0(A)$ can also be obtained in $O(n^3)$ time using the *traceback* technique [20]. Similarly, we only describe the procedures for computing scores or free-energies in this paper, all of which can be modified for computing secondary structures without increasing the orders of the time complexities using the traceback technique.

In this paper, we first show an $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time algorithm for $\mathcal{RNA}_0$, which is based on Valiant's algorithm for *context-free recognition* [17] and a fast algorithm for *funny matrix multiplication* [7, 13]. Note that funny matrix multiplication is, given $p \times q$ real matrix $X = (x_{ij})$ and $q \times r$ real matrix $Y = (y_{ij})$, to compute $p \times r$ matrix $Z = (z_{ij})$ such that $z_{ij} = \max_{1\leq k\leq q}(x_{ik} + y_{kj})$. For several problems such as the all-pairs shortest path problem [3, 13] and the maximum subarray problem [14], the fastest algorithms were obtained using fast funny matrix multiplication.

Next, we show the main result of this paper: an $O(n^{2.776})$ time algorithm for $\mathcal{RNA}_0$ which always outputs, for any fixed constant $\epsilon > 0$, an RNA secondary structure with the score at least $1 - \epsilon$ of the maximum. This algorithm is a combination of an approximation algorithm $\mathcal{A}_{approx}$ obtained by modifying $\mathcal{DP}_0$, and an exact algorithm $\mathcal{A}_{exact}$ based on Valiant's algorithm and fast funny matrix multiplication. Although Tamaki and Tokuyama developed an approximation algorithm for the maximum subarray problem [14], their technique could not be applied to $\mathcal{RNA}_0$ and thus a new technique was introduced for $\mathcal{A}_{approx}$. Moreover, although $\mathcal{A}_{approx}$ is a slight modification of $\mathcal{DP}_0$, the modification and the analysis are non-trivial.

Then, we extend the technique used in $\mathcal{A}_{approx}$ for more realistic versions of RNA secondary structure prediction. Since $\mathcal{A}_{approx}$ and their variants are very simple, they may be practical.

2 Exact Algorithm for a Basic Case

Several papers pointed out a relationship between the RNA secondary structure prediction problem and the *recognition problem of (stochastic) context-free grammars* [11, 16]. Based on these observations, we can associate the following context-free grammar (with score) with $\mathcal{RNA}_0$:

	$score(X)$
$X \longrightarrow \mathtt{a}$	0
$X \longrightarrow \mathtt{u}$	0
$X \longrightarrow \mathtt{g}$	0
$X \longrightarrow \mathtt{c}$	0
$X \longrightarrow YZ$	$score(Y) + score(Z)$
$X \longrightarrow \mathtt{a}Y\mathtt{u}$	$score(Y) + 1$
$X \longrightarrow \mathtt{u}Y\mathtt{a}$	$score(Y) + 1$
$X \longrightarrow \mathtt{g}Y\mathtt{c}$	$score(Y) + 1$
$X \longrightarrow \mathtt{c}Y\mathtt{g}$	$score(Y) + 1$

in which an *optimal parsing tree* (i.e., a parsing tree whose root has the highest score among all parsing trees) corresponds to an optimal secondary structure.

Valiant developed an $O(n^\omega)$ time algorithm for context-free recognition [17] using a fast boolean matrix multiplication algorithm, where $O(N^\omega)$ deonotes the time complexity of the current best algorithms (currently, $\omega = 2.376$ [4]) for both the boolean matrix multiplication and the usual matrix multiplication for $N \times N$ matrices.

Note that, in context-free recognition, we only need to decide the existence of a parsing tree for a given sentence, whereas, in $\mathcal{RNA}_0$, we need to compute a parsing tree with the highest score. Based on the above observation, we can obtain an algorithm for computing the score of an optimal parsing tree by replacing the boolean matrix multiplication with the $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time funny matrix multiplication algorithm [13]. Modification and analysis of the algorithm are straight-forward and omitted in this paper.

Theorem 1. $\mathcal{RNA}_0$ (i.e, finding an RNA secondary structure with the maximum number of base pairs) can be solved in $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time.

3 Approximation Algorithms for a Basic Case

3.1 Approximation Algorithm with a Constant Approximation Ratio

Here, we give a simple algorithm for $\mathcal{RNA}_0$ which always outputs an RNA secondary structure with score at least 1/2 of the maximum. The algorithm is based on the following simple observation.

Proposition 1. Suppose that an RNA sequence A consists of residues of types a and u, and let #a and #u be the numbers of letters of a and u appearing in A respectively. Then, the score of $OPT_0(A)$ is equal to $\min\{\#\mathtt{a}, \#\mathtt{u}\}$. Moreover, $OPT_0(A)$ can be computed in linear time.
(Proof) Using the following procedure, we can compute $OPT_0(A)$ in linear time, which consists of $\min\{\#\mathtt{a}, \#\mathtt{u}\}$ base pairs.

```
Let S be an empty stack;
for i = 1 to n do
  if S is empty then push(a_i, S)
  else if (a_i, top(S)) is a base pair then
    begin Output (a_i, top(S)) as a base pair; pop(S) end
  else push(a_i, S);
```

□

For an RNA sequence $A = a_1 \ldots a_n$, let $A(\mathtt{a},\mathtt{u})$ (resp. $A(\mathtt{c},\mathtt{g})$) be the subsequence of A consisting of letters of a and u (resp. c and g). Then, $score(OPT_0(A))$ is at most the sum of $score(OPT_0(A(\mathtt{a},\mathtt{u})))$ and $score(OPT_0(A(\mathtt{c},\mathtt{g})))$. Choosing the better one between $OPT_0(A(\mathtt{a},\mathtt{u}))$ and $OPT_0(A(\mathtt{c},\mathtt{g}))$, we have:

Theorem 2 For $\mathcal{RNA}_0$, an RNA secondary structure with the score at least 1/2 of the maximum can be computed in linear time.

3.2 $1-\epsilon$ Approximation Algorithm

The $1-\epsilon$ approximation algorithm is a combination of an exact algorithm $\mathcal{A}_{exact}$ and an approximation algorithm $\mathcal{A}_{approx}$: $\mathcal{A}_{exact}$ is used when $score(OPT_0(A))$ is small (presizely, $score(OPT_0(A)) = O(n^\gamma)$ where γ is a constant to be determined later), otherwise $\mathcal{A}_{approx}$ is used. Note that the linear time algorithm in Section 3.1 can be used for estimating $score(OPT_0(A))$.

First, we describe $\mathcal{A}_{exact}$. As observed in [14], funny matrix multiplication for $N \times N$ integer matrices whose maximum absolute value of the entries is bounded by Q can be done in $O(Q(\log Q)N^\omega)$ time by slightly modifying the algorithm in [3]. Using this in the modified Valiant's algorithm described in Section 2, we can obtain $\mathcal{A}_{exact}$.

Lemma 1. $\mathcal{A}_{exact}$ computes $OPT_0(A)$ in $O(Q(\log Q)n^\omega)$ time if $score(OPT_0(A)) \leq Q$.
(Proof) The maximum absolute value of elements in matrices appearing in the execution of $\mathcal{A}$ is bounded by $score(OPT_0(A))$. Therefore, each funny matrix multiplication for $N \times N$ matrices in the modified Valiant's algorithm can be done in $O(Q(\log Q)N^\omega)$ time. It is straight-forward to see that the total time complexity is $O(Q(\log Q)n^\omega)$. □

Next, $\mathcal{A}_{approx}$ is obtained by modifying the original $O(n^3)$ time DP procedure ($\mathcal{DP}_0$) for $\mathcal{RNA}_0$. Note that $S(i,j)$ (in $\mathcal{DP}_0$) is equal to of $score(OPT_0(A_{i,j}))$ where $A_{i,j} = a_i a_{i+1} \ldots a_j$.

Lemma 2. $|S(i,j) - S(i+h, j+k)| \leq |h| + |k|$.
(Proof) From the definition of $\mathcal{RNA}_0$, both $|S(i,j) - S(i,j+1)| \leq 1$ and $|S(i,j) - S(i+1,j)| \leq 1$ hold. □

In $\mathcal{A}_{approx}$, we do not compute $\max_{i<k\leq j} S(i,k-1) + S(k,j)$ exactly. Instead, we compute the maximum of $S(i,k-1) + S(k,j)$ for $O(n^\alpha + n^{1-\beta})$ values of k's (see Fig. 2), where α and β ($0 < \alpha, \beta < 1$) are appropriate constants to be determined later.

We define a sequence of indices $f_i^+(h)$ and $f_j^-(h)$ for $h = 0, 1, 2, \cdots$ by

$$f_i^+(0) = i + \lceil n^\alpha \rceil \qquad f_j^-(0) = j - \lceil n^\alpha \rceil$$
$$f_i^+(h+1) = f_i^+(h) + \lceil (f_i^+(h) - i)^\beta \rceil \qquad f_j^-(h+1) = f_j^-(h) - \lceil (j - f_j^-(h))^\beta \rceil$$

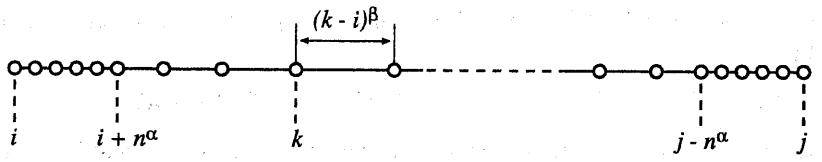

Figure 2: In $\mathcal{A}_{apporx}$, $\max_k S(i,k-1)+S(k,j)$ is computed not for all k, but for $O(n^\alpha + n^{1-\beta})$ values of k's, where such k's are represented by white circles in this figure.

Next, we define $\mathcal{I}(i,j)$ by

$$\begin{aligned}\mathcal{I}(i,j) &= \{\, k \mid i < k \le n^\alpha \text{ or } j - n^\alpha \le k \le j \,\} \bigcup \\ &\quad \{\, f_i^+(h) \mid f_i^+(h) \le (i+j)/2 \,\} \bigcup \{\, f_j^-(h) \mid f_j^-(h) \ge (i+j)/2 \,\}.\end{aligned}$$

Then, $\mathcal{A}_{approx}$ is expressed by the following DP procedure:

$$S'(i,j) = \max \begin{cases} S'(i-1,j+1) + \mu(a_i, a_j), \\ \max_{k \in \mathcal{I}(i,j)} S'(i,k-1) + S'(k,j), \end{cases}$$

where we let $S'(i,j) = 0$ for $i \ge j$.

Lemma 3. $\mathcal{A}_{approx}$ works in $O(n^{2+\alpha} + n^{3-\beta})$ time.
(Proof) Since $j - i \le n$, the size of $\mathcal{I}(i,j)$ is bounded by

$$\begin{aligned}|\mathcal{I}(i,j)| &\le 2n^\alpha + 4\left(\frac{\frac{n}{2}}{(\frac{n}{2})^\beta} + \frac{\frac{n}{4}}{(\frac{n}{4})^\beta} + \frac{\frac{n}{8}}{(\frac{n}{8})^\beta} + \cdots\right) \\ &\le 2n^\alpha + 4n^{1-\beta} \cdot \frac{1}{1-(\frac{1}{2})^{1-\beta}} \le O(n^\alpha + n^{1-\beta}).\end{aligned}$$

Since $\max_{k \in \mathcal{I}(i,j)} S'(i,k-1) + S'(k,j)$ is computed for $O(n^2)$ pairs (i,j), $\mathcal{A}_{approx}$ takes $O(n^{2+\alpha} + n^{3-\beta})$ time. □

Here, we define the *error* of an secondary structure M to $OPT_0(A)$ to be $score(OPT_0(A)) - score(M)$ (note that this value must be non-negative).

Lemma 4. The error of an secondary structure M computed by $\mathcal{A}_{approx}$ is $O(n^{1+\alpha\beta-\alpha})$.
(Proof) Note that, for each (i,j), we define the *error* of $S'(i,j)$ (in $\mathcal{A}_{approx}$) to be $S(i,j) - S'(i,j)$. Here, we show that, for all i,j, the following inequality holds:

$$S(i,j) - S'(i,j) \le \max\{\, Cm \cdot n^{\alpha\beta-\alpha} - C \cdot m^\beta,\ 0\}$$

for some constant C, where we let $m = j - i$. We prove this inequality by means of the induction on m.

Case (i) $m \le n^\alpha$

In this case, the error is always 0 and thus the above inequality holds.

Case (ii) $m > n^\alpha$

In this case, we assume that the inequality holds for all m' such that $m' < m$, and we consider the following recurrence in $\mathcal{A}_{approx}$: $S'(i,j) = \max_{k \in \mathcal{I}(i,j)}\{S'(i,k) + S'(k+1,j)\}$.

Let k' be the integer maximizing $S'(i,k') + S'(k'+1,j)$ under the condition that $i \le k' < j$, and let $k'' \in \mathcal{I}(i,j)$ be the integer maximizing $S'(i,k'') + S'(k''+1,j)$ under the condition that $k'' \in \mathcal{I}(i,j)$. From Lemma 2 and the definition of $\mathcal{I}(i,j)$, it is seen that $S'(i,k') + S'(k'+1,j) -$

$S'(i,k'') - S'(k''+1,j)$ is $O(h^\beta)$, where $h = \min(k''-i, j-k'')$. Then, the error of $S'(i,j)$ is bounded by

$$Ch \cdot n^{\alpha\beta-\alpha} - C \cdot h^\beta + C(m-h) \cdot n^{\alpha\beta-\alpha} - C \cdot (m-h)^\beta + D \cdot h^\beta$$

where D is an appropriate constant, and we assume without loss of generality that $h > n^\alpha$. It is not difficult to verify that this value is at most $Cm \cdot n^{\alpha\beta-\alpha} - C \cdot m^\beta$ for $C \gg D$. □

Theorem 3. For $\mathcal{RNA}_0$, an RNA secondary structure with the score at least $1-\epsilon$ of the maximum can be computed in $O(n^{2.776})$ time, where ϵ is any fixed positive number.

(Proof) First, we estimate $score(OPT_0(A))$ using the linear time algorithm described in Section 3.1. If the estimated value is at most n^γ, an optimal structure is computed using $\mathcal{A}_{exact}$, otherwise an approximate structure is computed using $\mathcal{A}_{approx}$. Then, from Lemma 3, the time complexity is $O(n^{\gamma+\omega}\log n + n^{2+\alpha} + n^{3-\beta})$.

From Lemma 4, the ratio of the score of an approximate solution computed by $\mathcal{A}_{approx}$ to the optimal score is $\dfrac{score(OPT_0(A)) - O(n^{1+\alpha\beta-\alpha})}{score(OPT_0(A))}$, which is less than $1-\epsilon$ for any fixed $\epsilon > 0$ if $1+\alpha\beta-\alpha < \gamma$ and n is sufficiently large.

Here, we let $\alpha = 0.776$, $\beta = 0.224$, $\gamma = 0.398$ and $\omega = 2.376$. Then, $1+\alpha\beta-\alpha < \gamma$ is satisfied and the theorem follows. □

4 Approximation Algorithms for More Practical Cases

Although the above algorithms are not practical, the technique developed for $\mathcal{A}_{approx}$ can be applied to more practical versions of RNA secondary structure prediction. Since the quality of predicted RNA structures heavily depends on the energy functions which are derived empirically, a lot of practical versions have been proposed based on various energy functions [12, 15, 20]. In this section, we show that the developed technique can be applied to many of them.

4.1 Energy Function for Adjacent Base Pairs

In $\mathcal{RNA}_0$, energy function is defined for each base pair. On the other hand, energy functions defined for adjacent base pairs are widely used [15, 16]. In this case, energy function μ is defined for adjacent base pairs $(a_ia_{i+1}, a_{j+1}a_j)$. Formally, an energy function is a function from $\Sigma \times \Sigma \times \Sigma \times \Sigma$ to the set of *negative* reals. Note that, in this case, the global free-energy (i.e., the total score) is always negative and the problem is defined as a minimization problem.

Under this kind of energy functions, an optimal RNA secondary structure can be computed in $O(n^3)$ time using a DP procedure similar to $\mathcal{DP}_0$ [15]. Moreover, a context-free grammar (with score) can also be associated as in Section 2, and thus we can derive an $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time exact algorithm.

Since energy function $\mu(a_ia_{i+1}, a_{j+1}a_j)$ takes values between 0 and E where E is a negative constant, the property similar to Lemma 2 still holds in this case, and thus we can obtain an approximation algorithm as in Section 3.

Theorem 4. Under energy functions defined for adjacent base pairs, an optimal RNA secondary structure can be computed in $O(n^3(\log\log n)^{1/2}/(\log n)^{1/2})$ time, and an RNA secondary structure with the free-energy at most $1-\epsilon$ of the minimum can be computed in $O(n^{2.776})$ time.

4.2 Destabilizing Energy

We did not consider free-energy for unpaired residues so far. However, such residues are also important determinants of RNA stability, and several energy functions are proposed for unpaired

residues [12, 15, 20]. A consecutive part of unpaired residues is called a *loop*, where there are several kinds of loops such as *bulge loop*, *end loop* and *interior loop* [20]. Usually, an energy function for loops takes positive value and is called a *destabilizing* energy function.

Watermann and Smith proposed an $O(n^3)$ time algorithm for computing locally destabilizing RNA secondary structures (i.e., minimum energy RNA secondary structures without *multibranch loops*) [19, 20]. In their algorithm, energy for a *bulge loop* is computed by

$$\min_{k \geq 1}\{\mu(a_i, a_j) + \nu(k) + S(i+k+1, j-1)\}$$

where $\nu(k)$ corresponds to the destabilizing energy for a *bulge loop*. Since this 'min' is computed for $O(n^2)$ pairs of (i, j), the computation time for bulge loops becomes $O(n^3)$. However, we can reduce the computation time if we compute 'min' only for k's such that $k - i \in \mathcal{I}(i, j)$, where we assume that $|\nu(k) - \nu(k+1)| \leq const$ holds for any k. Similarly, we can reduce the computation time for the other loops. Although we omit details, we can prove that an RNA secondary structure whose score is at most $O(n^{1+\alpha\beta-\alpha})$ larger than the minimum can be computed in $O(n^{2+\alpha} + n^{3-\beta})$ time. Thus, letting $\alpha = \beta = \frac{1}{2}$, we have the following:

Theorem 5. Under the energy function including destabilizing energy and the condition that an RNA secondary structure does not have multibranch loops, an RNA secondary structure with the free-energy at most $1 - \epsilon$ of the minimum can be computed in $O(n^{2.5})$ time if the minimum free-energy is $\Omega(n^{0.75+\delta})$, where $\epsilon, \delta > 0$ are arbitrary small constants.

4.3 Pseudoknots

Although *pseudoknots* (special kinds of substructures) are taken into account in a few algorithms, pseudoknots appear in several important RNA's [16]. For a basic version (i.e., maximizing the number of base pairs) of RNA secondary structure prediction with *simple* pseudoknots, an $O(n^4)$ time algorithm was proposed by Uemura *et al.* based on *tree-adjoining* grammar [16], and then a simpler $O(n^4)$ time algorithm was developed without tree-adjoining grammar [2].

In the latter algorithm, computation of $S_{pseudo}(i, j)$ (for all (i, j)'s) takes $O(n^4)$ time and this is a crucial part for improving the time complexity [2]. A similar technique as in $\mathcal{A}_{approx}$ can be applied to this part, where details are omitted.

Theorem 6. For a basic version of RNA secondary structure prediction with simple pseudoknots, an RNA secondary structure with the score at least $1 - \epsilon$ of the maximum can be computed in $O(n^{3.5})$ time if the maximum score is $\Omega(n^{0.75+\delta})$, where $\epsilon, \delta > 0$ are arbitrary small constants.

5 Concluding Remarks

In this paper, we proposed approximation algorithms and exact algorithms for RNA secondary structure prediction. The most important contribution of this paper is that it shows that the well-known $O(n^3)$ DP algorithm and its variants are not necessarily optimal.

Although the exact algorithms are not complicated, the approximation algorithms (excluding $\mathcal{A}_{exact}$) are very simple and may be practical. Of course, secondary structures obtained by the approximation algorithms may be different from optimal secondary structures. However, optimal secondary structures do not necessarily coincide with real secondary structures because energy functions used are empirically derived, and thus optimal secondary structures are also approximations of real secondary structures. Indeed, a lot of heuristic algorithms without guaranteed approximation ratio have been proposed for RNA secondary structure prediction [1]. Therefore, the proposed approximation algorithms may be practical.

Finally, we conclude with open problems: (i) Development of an $O(n^{3-\epsilon})$ time exact algorithm for $\mathcal{RNA}_0$ where ϵ is some positive constant; (ii) Removement of "if ... $\Omega(n^{0.75+\delta})$" in Theorem 5 and Theorem 6; (iii) Extension of the proposed approximation algorithms for parsing of stochastic context-free grammars (in this case, Lemma 2 does not hold).

References

[1] J. P. Abrahams, M. Berg, E. Batenburg and C. Pleij, Prediction of RNA secondary structure, including pseudoknotting by computer simulation, *Nucleic Acids Research* **18** (1990) 3035–3044.

[2] T. Akutsu, DP algorithms for RNA secondary structure prediction with pseudoknot, *Genome Informatics 1997*, Universal Academy Press (1997) 173–179.

[3] N. Alon, Z. Galil and O. Margalit, On the exponent of the all pairs shortest path problem, *Proc. 32nd IEEE Symp. Foundations of Computer Science* (1991) 569–575.

[4] D. Coppersmith and S. Winograd, Matrix multiplication via arithmetic progression, *J. Symbolic Computation* **9** (1990) 251–280.

[5] D. Eppstein, Z. Galil and R. Giancarlo, Speeding up dynamic programming, *Proc. 29th IEEE Symp. Foundations of Computer Science* (1988) 488–496.

[6] D. Eppstein, Z. Galil, R. Giancarlo and G. F. Italiano, Sparse dynamic programming II: convex and concave cost functions, *J. ACM* **39** (1992) 546–567.

[7] M. L. Fredman, New bounds on the complexity of the shortest path problem, *SIAM Journal on Computing* **5** (1976) 83–89.

[8] Z. Galil and K. Park, Dynamic programming with convexity, concavity and sparsity, *Theoretical Computer Science* **92** (1992) 49–76.

[9] M. Kanehisa and W. B. Goad, Pattern recognition in nucleic acid sequences II: an efficient method for finding locally stable secondary structures, *Nucleic Acids Research* **10** (1982) 265–277.

[10] L. L. Larmore and B. Schieber, On-line dynamic programming with applications to the prediction of RNA secondary structure, *J. Algorithms* **12** (1991) 490–515.

[11] Y. Sakakibara, M. Brown, E. Hughey, I. S. Mian, K. Sjlölander, R. C. Underwood and D. Haussler, Stochastic context-free grammars for tRNA modeling, *Nucleic Acids Research* **22** (1994) 5112–5120.

[12] J. Setubal and J. Meidanis, *Introduction to Computational Molecular Biology*, PWS Pub. Co., Boston (1997).

[13] T. Takaoka, A new upper bound on the complexity of all pairs shortest path problem, *Information Processing Letters* **43** (1992) 195–199.

[14] H. Tamaki and T. Tokuyama, Algorithms for maximum subarray problem based on matrix multiplication, *Proc. 9th ACM-SIAM Symp. Discrete Algorithms* (1998) 446–452.

[15] D. H. Turner, N. Sugimoto and S. M. Freier, RNA structure prediction, *Ann. Rev. Biophys. Biophys. Chem.* **17** (1988) 167–192.

[16] Y. Uemura, A. Hasegawa, S. Kobayashi and T. Yokomori, Grammatically modeling and predicting RNA secondary structures, *Proc. Genome Informatics Workshop VI* (1995) 67–76.

[17] L. G. Valiant, General context-free recognition in less than cubic time, *Journal of Computer and System Sciences* **10** (1975) 308–315.

[18] M. S. Watermann and T. F. Smith, RNA secondary structure: a complete mathematical analysis, *Math. Biosciences* **41** (1978) 257–266.

[19] M. S. Watermann and T. F. Smith, Rapid dynamic programming algorithms for RNA secondary structure, *Advances in Applied Mathematics* **7** (1986) 455–464.

[20] M. S. Watermann, *Introduction to Computational Biology*, Capman & Hall, London (1995).

[21] M. Zuker and P. Stiegler, Optimal computer folding for large RNA sequences using thermodynamics and auxiliary information, *Nucleic Acids Research* **9** (1981) 133–148.