离散数学与理论计算机科学.style id='3'>第XXX卷：XXX期 #XXX（2023年）

*RNA*二级结构组合集合的排名与反排名算法

Yuriy Shablya DmitryKruchinin

俄罗斯托木斯克托木斯克控制系统与无线电电子学国立大学

收稿日期：*2023*年*1*月*27*日；录用日期：*XXX*。

本文研究长度为 n 、含有 m 个碱基对的RNA二级结构组合集合。为便于紧凑表示，我们用相应的莫茨金词对RNA二级结构进行编码。对于该组合集合，我们构造了与/或树结构，找到了该组合集合与与/或树变体集合之间的双射，并开发了对与/或树变体进行排名和反排名的算法。所开发的排名与反排名算法的时间复杂度为 $O(m^2(n-m))$（适用于 $m < n-2m$），以及 $O(m(n-m)^2)$（适用于 $m > n-2m$）。与现有算法相比，新算法不需要预处理步骤，且时间复杂度更优。

关键词 ds: 组合生成、与/或树、RNA二级结构、莫茨金词、排序、反排名 ng

arXiv:2301.11890v1 [cs.DS] 27 Jan 2023

1 引言

组合生成是一个科学分支，处于计算机科学与组合学的交叉领域，致力于开发用于组合集合排序与反排名的方法和算法。组合生成的基础知识可见于Kreher和Stinson（1999）、Ruskey（2003）和Knuth（2011）。许多信息对象都可以表示为组合集合，例如排列、划分、树、路径等。在这种情况下，可以使用组合生成方法来处理这些集合，并对其进行编码、生成和存储。在处理大型组合集合时（例如研究各种化学和生物结构），这一点尤为重要。

本文中，我们拟探讨 RNA（核糖核酸）的结构。RNA 是任何生物细胞结构中的主要且最重要的组成元素之一。RNA分子是一条由核苷酸组成的链，包含四种含氮碱基：A（腺嘌呤）、C（胞嘧啶）、G（鸟嘌呤）和 U（尿嘧啶）。RNA链中的核苷酸序列构成 RNA一级结构。从数学角度来看，RNA一级结构是从集合 $\{A, C, G, U\}$ 中取出的元素序列，即 $a = (a_1,\ a_2,\ .....)$，其中 $a_i \in \{A, C, G, U\}$。

国际标准连续出版物号投稿至离散数学与理论计算机科学 ©2023年 作者所有，依据知识共享署名 4.0 国际许可协议授权发布

此外，RNA链中的两个碱基之间可以形成氢键（例如，A–U 和 C–G 之间的 Watson-Crick 碱基对）。RNA分子中氢键的形成会产生不同的环，从而使 RNA分子结构更加复杂。RNA链形成的、带有氢键的形状构成 RNA二级结构。从数学角度来看，RNA二级结构可以表示为一个图，其中图的节点是核苷酸，边表示碱基之间的连接。有关 RNA结构特征的更多详情，请参阅 Saenger（1984）、Reidys（2011）和 Waterman（2018）。

RNA分子在细胞核酸加工和基因表达中发挥着重要作用。根据 Picardi（2015）的观点，通过对RNA结构进行详细研究，可以更深入地理解这些过程。例如，其中一个研究方向是RNA二级结构的预测：Zuker 和 Sankoff（1984）；Oluoch 等（2019）；Sato 等（2021）；Sato 和 Kato（2022）。此外，通过开发计算算法来改进此类研究工作，也是现代生物信息学的主要方向之一。

我们考虑RNA一级结构的组合集合，记为 A_n。RNA一级结构的集合由一个参数 n （RNA链中的核苷酸数量）描述。该组合集合可表示为具有重复元素的 n-排列集合，其中 $m = 4$。也就是说，从由 $m = 4$ 个元素组成的集合中选择 n 个元素 $\{A, C, G, U\}$，且这些元素可以重复。因此，此类排列集合的排名与反排名算法也可用于RNA一级结构的组合集合。

接下来，我们考虑 RNA 二级结构的组合集合，将其记为$A_{n,\ m}$。该组合集合由两个参数描述： n 表示 RNA 链中的核苷酸数，m 表示碱基对数。另有一些研究致力于开发 RNA 二级结构集合的组合生成算法：Nebel 等人（2011）；Seyedi-Tabari 等人（2010）；Mohammadi 和 Nowzari-Dalini（2018）。

Nebel 等人（2011）提出了用于生成长度为 n的 m 随机 RNA 二级结构集合的算法（未指定碱基对数）。这些组合生成算法采用 Flajolet 等人（1994）的方法开发。与 Viennot 和 Vauchaussade de Chaumont（1985）的做法一样，RNA 二级结构以正确括号序列编码，并允许添加空格（或莫茨金词）。所开发算法的时间复杂度为 $O(mn\log\ n)$，并且需要执行时间复杂度为 $O(n^2)$的预处理步骤。这种方法的主要缺点是无法控制所生成 RNA 二级结构中的碱基对数，同时也缺少排名和反排名算法。

Seyedi-Tabari 等人（2010年）提出了用于列举、排名和反排名长度为 n 、含有 m 个碱基对的 RNA二级结构组合集合的算法。随后，Mohammadi 和 Nowzari-Dalini（2018）提出了列举算法的并行版本。这些组合生成算法采用了一种基于计算序列前缀的方法，其思路与 Ryabko（1998年）的方法相似。在这种情况下，RNA二级结构被编码为一棵具有n 个内部节点和 m 个外部节点的树，该树也被编码为一个 E序列。排名和反排名算法的时间复杂度为多项式 $O\left(nm - m^2\right)$ 和 $O\left(n^2 + m^2\right)$，但这两种算法都需要一个预处理步骤，其时间复杂度为 $O\left((n-m)\left|A_{n,\ m}\right|\right)$，其中$\left|A_{n,\ m}\right|$ 是集合$A_{n,\ m}$的基数。由于$\left|A_{n,\ m}\right|$ 可以通过二项式系数计算（见公式（2））， 因此预处理步骤具有指数时间复杂度。因此，该方法的主要缺点在于预处理步骤具有指数时间复杂度，且需要大量内存空间来存储预处理计算的结果。

本文研究的主要目的是开发用于对长度为 n 、含有 m 个碱基对的RNA二级结构组合集合进行排名和反排名的新型高效算法（具有多项式时间复杂度且无需预处理步骤）。

2 主要结果

2.1 研究方法

开发组合生成算法有多种方法。本文提出采用 Shablya 等人（2020）所述的方法开发组合生成算法，该方法基于与/或树。要将此方法用于开发排序与反排序算法，必须将给定的组合集合表示为与/或树结构，并确保其变体总数等于该组合集合的基数函数值。

与/或树是一种树结构，包含两种类型的节点：与节点和或节点（图 2 中，与节点指向子节点的边由额外的曲线连接）。与/或树的一个变体是通过为每个或节点仅保留一条边并移除其余所有边而得到的树结构。如果我们知道组合集合 A 的基数函数 f，并且 f 属于代数 $\{\mathbb{N}, +, \times, R\}$（仅使用自然数、加法和乘法运算以及原始递归算子），那么就可以构造相应的与/或树结构。因此，我们需要知道所考虑的 RNA二级结构组合集合所需的基数函数形式。

2.2 组合集合

为了实现紧凑表示，我们用相应的莫茨金词对 RNA二级结构进行编码。图 1 展示了一个长度为 $n = 27$、含有 $m = 9$个碱基对的 RNA二级结构示例。

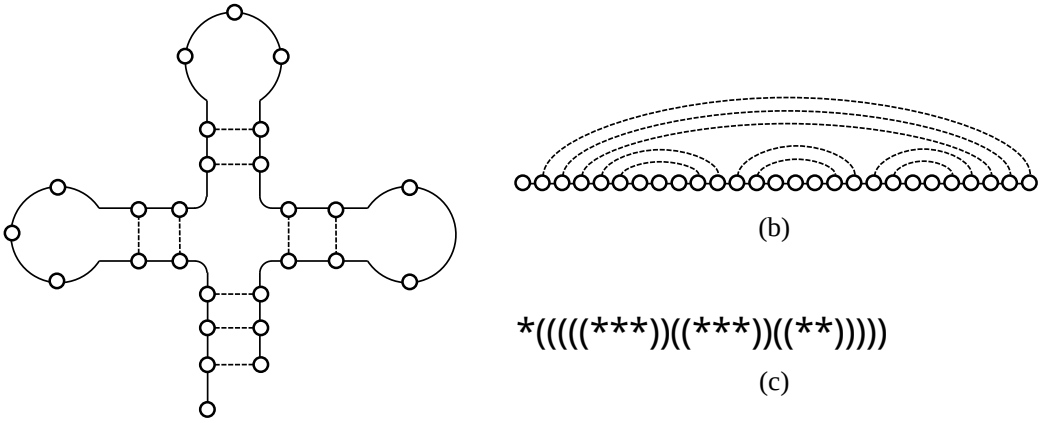

图 1： 一个长度为 $n = 27$ 、含有 $m = 9$ 个碱基对的 RNA二级结构表示示例。

图 1(a) 展示了 RNA二级结构由核苷酸链构成的“真实”形状（节点由实线连接），以及碱基对（节点由虚线连接）。图 1(b) 将 RNA二级结构表示为 RNA链的直线。图 1(c) 则将该 RNA二级结构表示为相应的莫茨金词，即正确括号序列，并可在其中添加空格（以“*”表示）。因此，我们可以用序列 $a = (a_1, \ldots\ldots, a_n)$ 对 RNA二级结构进行编码，其中 $a_i \in \{$ “(”、“)”、“*” $\}$。在这种情况下，“(”和“)”表示一个碱基对，“*”表示一个不含氢键的核苷酸。

所考虑的 RNA二级结构组合集合的基数函数由 Schmitt and Waterman（1994）定义：$A_{n,\ m}$

$$f(n,m) = |A_{n,m}| = S_n^m = S_{n-1}^m + \sum_{i=0}^{m-1} \sum_{j=2i}^{n-2(m-i)-1} S_{n-2-j}^{m-1-i} S_j^i, \tag{1}$$

其中，$S_n^0 = 1$ 当 $n \geq 0$ 时，$S_n^m = 0$ 当 $m \geq \frac{n}{2}$ 时。

S_n^m 的取值构成整数序列 $A089732$，见 OEIS基金会（2022），并可使用 Schmitt and Waterman（1994）给出的以下显式公式计算：

$$S_n^m = \frac{1}{n-m}\binom{n-m}{m}\binom{n-m}{m+1} \tag{2}$$

当 $n > m \geq 0$ 且 $S_0^0 = 1$时。表 1 列出了 S_n^m的前几个值。

表 1： S_n^m的前几个值。

	m				
n	0	1	2	3	4
0	1				
1	1				
2	1				
3	1	1			
4	1	3			
5	1	6	1		
6	1	10	6		
7	1	15	20	1	
8	1	21	50	10	
9	1	28	105	50	1
10	1	36	196	175	15

生成函数在整数序列研究中发挥着重要作用。以下关于数列 S_n^m的二元生成函数见于 OEIS基金会（2022）：

$$S(x,y) = \sum_{n\geq 0}\sum_{m\geq 0} S_n^m x^n y^m = \frac{1 - x + x^2 y - \sqrt{(1-x-x^2y)^2 - 4x^3y}}{2x^2y}.$$

数列 S_n^m 与纳拉亚纳数 N_n^m 之间存在关联。还值得注意的是，纳拉亚纳数 N_n^m 与著名的卡特兰数 C_n 相关（更多信息请参见 Petersen（2015））。在这种情况下，纳拉亚纳数 N_n^m 将由卡特兰数 C_n 定义的组合集合划分为多个子集，划分依据是某个额外参数 m。例如，卡特兰数的一种组合解释是 Dyck n-路径的数量；那么纳拉亚纳数表示具有 m 个峰的 Dyck n-路径的数量（参见 Kruchinin等人（2022））。

纳拉亚纳数的以下二元生成函数 N_n^m 由 OEIS 基金会公司（2022）给出：

$$N(x,y)=\sum_{n>0}\sum_{m>0}N_n^m x^n y^m=\frac{1-x-xy-\sqrt{(1-x-xy)^2-4x^2y}}{2x},$$

并且对于 $n\geq m>0$，我们有

$$N_n^m=\frac{1}{n}\binom{n}{m-1}\binom{n}{m}.$$

因此，我们可以通过它们生成函数之间的关系，说明数列 S_n^m 与纳拉亚纳数 N_n^m 之间的关系：

$$S(x,y)=1+\frac{N(x,xy)}{xy}$$

或者通过它们的显式公式：

$$S_n^m=N_{n-m}^{m+1},$$

其中 $n>m\geq 0$ 且 $S_0^0=1$。

2.3 与/或树

由于公式 (1) 属于代数 $\{\mathbb{N},+,\times,R\}$，我们可以为 S_n^m 构建与/或树结构，如图 2 所示。在该与/或树中：

- 公式 (1) 中的每个加法都表示为一个或节点，其中所有项都表示为其子节点；
- 公式 (1) 中的每个乘法都表示为一个与节点，其中所有因子都表示为其子节点；
- 公式 (1) 中的每个递归运算都表示为与/或树中的递归（用带三角形的节点表示）。

对于该与/或树结构，有以下初始条件：

- 如果我们得到一个标记为 S_n^0 的节点，那么它就是与/或树中的一个叶节点，其中 $n\geq 0$
- 如果我们得到一个标记为 $S_n^m$where $m\geq \frac{n}{2}$ 的节点，那么就需要移除该节点，并移除包含该节点的子树，直到或节点最近的子节点为止。

图3展示了一个与/或树结构的示例，其中 S_n^m， $n=6$ 和 $m=2$。其变体总数等于 $S_6^2=6$。

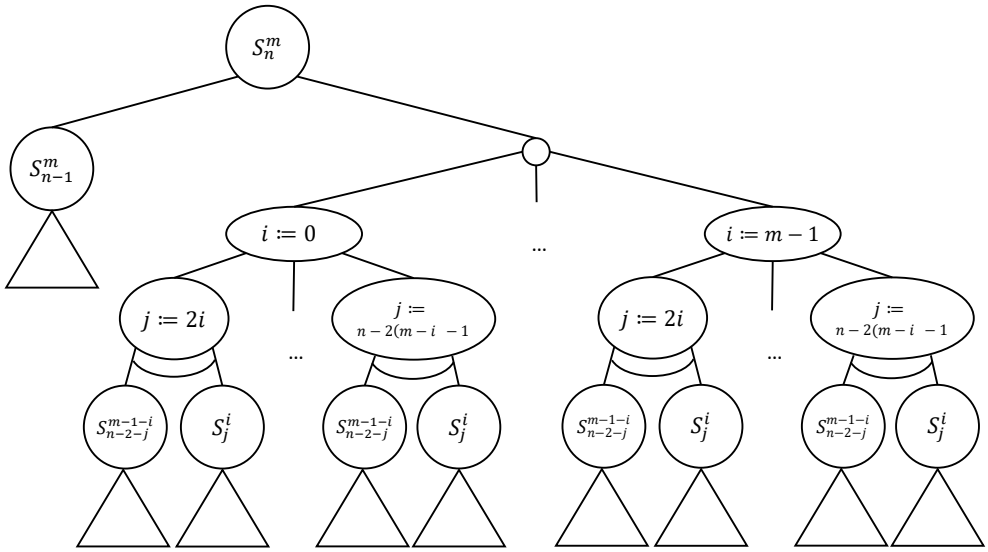

图 2： S_n^m的与/或树。

定理 1 长度为 n 、含有 m 个碱基对的 *RNA*二级结构集合，与 S_n^m的与/或树变体集合之间存在双射。

证明： 将长度为 n 且含有 m 个碱基对的 RNA二级结构表示为长度为 n 、含有 m 对括号且可添加空格的正确括号序列。这类 RNA二级结构的总数等于 S_n^m。图2中所示的 S_n^m对应的与/或树的变体总数也等于 S_n^m。因此，长度为n 且含有 m 个碱基对的 RNA二级结构集合与 S_n^m对应的与/或树的变体集合之间的双射由以下规则定义：

- 或节点 S_n^m中每个被选中的左子节点，都会使序列$a = (a_1, \ldots, a_n)$增加一个空格：序列 a 表示为 $a = \text{concat}(\text{('*')}, b)$，也就是说，序列的第一个元素是添加的空格（$a_1 =$ '*'），而序列的其余部分 $b = (a_2, \ldots, a_n)$是一个长度为 $n-1$、含有 m 对括号的正确括号序列（它对应于标记为 S_{n-1}^m的节点的子树）；

- 或节点 S_n^m中每个被选中的右子节点，都会使序列 $a = (a_1, \ldots, a_n)$增加一对括号：序列 a 表示为 $a = \text{concat}(\text{('('}), b, \text{(')'}), c)$，其中$b = (a_2, \ldots, a_{n-j-1})$是一个长度为 $n-2-j$、含有 $m-1-i$ 对括号的正确括号序列（它对应于标记为 S_{n-2-j}^{m-1-i} 的节点的子树），而 $c = (a_{n-j+1}, \ldots, a_n)$是一个长度为 j 、含有 i 对括号的正确括号序列（它对应于标记为 S_j^i的节点的子树）； i 和 j 的值由与/或树中选定的子节点决定。

- each 叶节点标记为 S_n^0，决定向序列 $a = (a_1$、.....、 a_n 中添加 n 个空格).

□

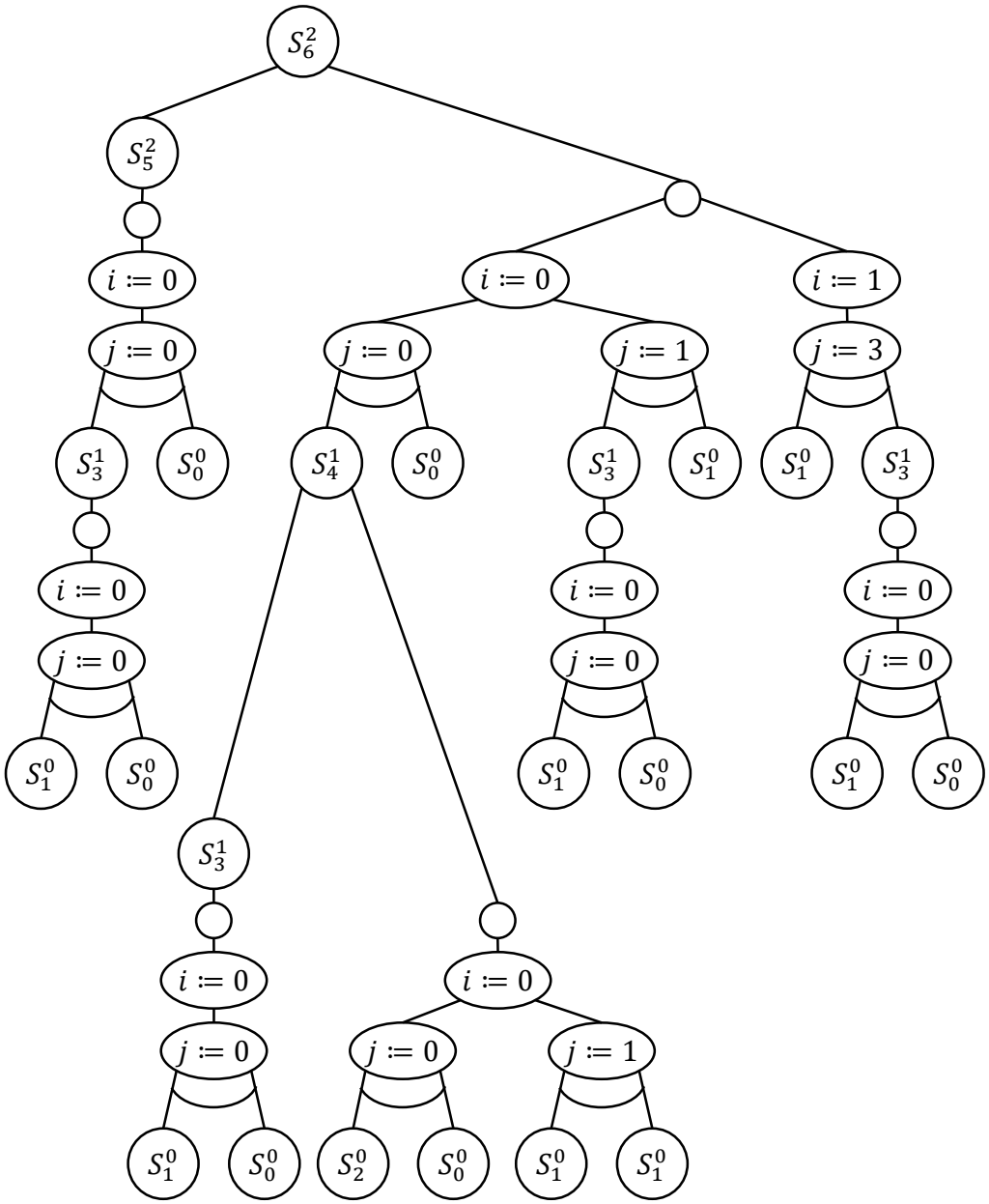
S_6^2
S_5^2
$i \coloneqq 0$
$i \coloneqq 0$
$i \coloneqq 1$
$j \coloneqq 0$
$j \coloneqq 0$
$j \coloneqq 1$
$j \coloneqq 3$
S_3^1 S_0^0
S_4^1 S_0^0
S_3^1 S_1^0
S_1^0 S_3^1
$i \coloneqq 0$
$i \coloneqq 0$
$i \coloneqq 0$
$j \coloneqq 0$
$j \coloneqq 0$
$j \coloneqq 0$
S_1^0 S_0^0
S_1^0 S_0^0
S_1^0 S_0^0
S_3^1
$i \coloneqq 0$
$i \coloneqq 0$
$j \coloneqq 0$
$j \coloneqq 0$
$j \coloneqq 1$
S_1^0 S_0^0
S_2^0 S_0^0
S_1^0 S_1^0

图 3： S_6^2的与/或树。

在得到的双射规则中，concat 函数表示合并序列，即对于给定的两个序列 $a = (a_1, \ldots, a_n)$ 和 $b = (b_1, \ldots, b_m)$，我们得到以下序列：

$$c = \texttt{concat}(a, b) = (a_1, \ldots, a_n, b_1, \ldots, b_m) = (c_1, \ldots, c_{n+m}).$$

为了紧凑表示，S_n^m的与/或树变体将编码为一个序列$v = (v_1, v_2)$，其中：

- 空序列 $v = ()$ 对应于选择标记为 S_n^0 的叶节点；
- v_1 对应于选择标记为 S_n^m的或节点的左子节点（$v_1 = 0$）或右子节点（$v_1 = 1$）；
- 如果 $v_1 = 0$，那么 v_2 对应于标记为 S_{n-1}^m的节点的子树变体；
- 如果 $v_1 = 1$，那么 v_2 由序列 $v_2 = (I, J, vl, vr)$ 编码，其中：
  - I 对应于与/或树中 i 的选定值；
  - J 对应于与/或树中 j 的选定值；
  - vl 对应于标记为 S_{n-2-j}^{m-1-i} 的节点的子树变体；
  - vr 对应于标记为 S_j^i的节点的子树变体。

算法 1 和算法 2 给出了所构建的规则，用于在 RNA二级结构集合与相应与/或树的变体集合之间建立双射。在这些算法中，()表示空序列。

让我们更详细地考察这些算法的计算复杂度：

- 在算法 1 中，有 $n - 2m$ 次递归调用，其中 $a_1 =$ '*'（每次这样的递归调用都需要创建一个长度为 $n - 1$ 的新序列）；此外，还有 m 次递归调用，其中 $a_1 \neq$ '*'（每次这样的递归调用都需要在 for 循环中最多进行 n 次计算，并创建长度最多为 $n-2$ 的新序列）。因此，假设对 $O(1)$ 中的数进行代数运算，算法 1 的多项式时间复杂度为 $O(n(n-m) + nm)$；
- 在算法 2 中，最多有 $n - 2m$ 次递归调用，其中 $v_1 = 0$（每次这样的递归调用都需要创建一个长度为 n 的新序列）；此外，还有 m 次递归调用，其中 $v_1 = 1$（每次这样的递归调用都需要创建一个长度为 n 的新序列）。因此，假设对 $O(1)$ 中的数进行代数运算，算法 2 的多项式时间复杂度为 $O(n(n-m)+nm)$。

因此，两种算法的时间复杂度均为多项式级别 $O(n(n-m)+nm)$。此外，考虑到约束条件 $m < \frac{n}{2}$，我们得到 $O(n^2)$。

得到的双射使我们能够将上文提到的组合生成方法应用于开发算法，以对 S_n^m 所对应的已构造与/或树结构的变体进行排名与反排名。将这些组合生成算法与得到的双射相结合，我们将得到对所考虑的 RNA二级结构组合集合进行排名与反排名的算法。

RNASSToVariant ($a =$**(**a_1**, .. . ,** a_n**),** n**,** m**) 开始 如果** $m = 0$ **则返回 () 如果** $a_1 =$**'*' 则** v**:=(0, RNASSToVariant ((**a_2**, .. . ,** a_n**),** $n - 1$**,** m**)) 否则** s_1 **:=** $0 s_2$ **:=** 0**对于** $i := 1$ **到** n **循环 如果** $a_i =$**'(' 则** s_1 **:=** $s_1 + 1$**如果** $a_i =$**')' 则** $s_2 := s_2 + 1$**如果** $s_1 = s_2$ **则** $I := m - s_1 J :=$ $n - ib :=$ **(**a_2**, .. . ,** a_{n-J-1}**)**$c :=$ **(**a_{n-J+1}**, .. . ,** a_n**) 跳出循环 结束** $vl :=$ **RNASSToVariant (**b**,** $n - 2 - J$**,** $m - 1 - I$**)**$vr :=$ **RNASSToVariant (**c**,** J**,** I**)**$v :=$ **(1, (**I**,** J**,** vl**,** vr**)) 结束 返回** v**结束**

算法 1: 一种算法，可将正确括号序列（允许添加空格）转换为 S_n^m的与/或树变体。

VariantToRNASS ($v = (v_1, v_2)$, n, m) 开始 如果 $m = 0$ 则 返回 ($a_1, \ldots,$ a_n) := ('*' . . . ,'*') 如果 $v_1 := 0$ 则 $a :=$ concat (('*'), VariantToRNASS (v_2, $n - 1$, m)) 否则 (I, J, vl, vr) := $v_2 b :=$ VariantToRNASS (vl, $n - 2 - J$, $m - 1 - I$)$c :=$ VariantToRNASS (vr, J, I)$a :=$ concat (('('), b, (')'), c) 结束 返回 a结束

算法 2: 一种将 S_n^m 的与/或树变体转换为正确括号序列，并可添加空格的算法。

2.4 排序与反排名算法

我们采用 Shablya 等人（2020）所述的通用方法，开发了用于对构建的与/或树结构的变体进行排名与反排名的算法。算法3列出了对 S_n^m 的与/或树变体进行排名所需的操作，算法4列出了对其变体进行反排名所需的操作。

让我们更详细地考察这些算法的计算复杂度：

- 在算法3中，最多有 $n-2m$ 次递归调用，其中 $v_1 = 0$ （每次这样的递归调用只需进行一次赋值）；此外，还有 m 次递归调用，其中 $v_1 = 1$ （每次这样的递归调用都需要计算 S_j^i，对于 i 最多计算 m 次，对于 j 最多计算 $n-2m$ 次）。根据公式（2）， S_n^m 的计算对于 $m < n-2m$ 的多项式时间复杂度为 $O(m)$，对于 $m > n-2m$ 则为 $O(n-m)$。因此，假设对 $O(1)$ 中的数进行代数运算，算法3对于 $m < n-2m$ 的多项式时间复杂度为 $O(m^2(n-m))$，对于 $m > n-2m$ 则为 $O(m(n-m)^2)$。

- 在算法4中，最多有 $n-2m$ 次递归调用，其中 $r < S_{n-1}^m$ （每次这样的递归调用只需进行一次赋值）；此外，还有 m 次递归调用，其中 $r \geq S_{n-1}^m$ （每次这样的递归调用都需要计算 S_J^I，对于 I 最多计算 m 次，对于 J 最多计算 $n-2m$ 次）。因此，假设对 $O(1)$ 中的数进行代数运算，算法4对于 $m < n-2m$ 的多项式时间复杂度为 $O(m^2(n-m))$，对于 $m > n-2m$ 则为 $O(m(n-m)^2)$。

考虑约束 $m < \frac{n}{2}$ 后，所提出的两种算法的时间复杂度均为多项式时间复杂度 $O(n^3)$。相比之下，Seyedi-Tabari 等人（2010）的排名和反排名算法具有更优的多项式时间复杂度 $O(n^2)$，但需要执行一个时间和空间复杂度均为指数级的预处理步骤。由于所开发的算法不需要预处理步骤，因此其整体时间和空间复杂度更优。此外，还可以为所开发的算法添加一个预处理步骤，用于计算 S_n^m 的所有必要取值。由此，我们得到一个时间复杂度为多项式时间复杂度 $O(n^2)$ 的预处理步骤，以及具有相同多项式时间复杂度 $O(n^2)$的新排名和反排名算法。

RankVariant RNASS ($v = (v_1, v_2)$, n, m) 开始 如果 $m = 0$则 返回 0 如果 $v_1 = 0$ 则 $r :=$ RankVariant RNASS ($v_2, n-1, m$) 否则 $(I, J, vl, vr) := v_2 l_1 :=$ RankVariant RNASS ($vl, n-2-J, m-1-I$)$l_2 :=$RankVariant RNASS (vr, J, I)$r := l_1 + S_{n-2-J}^{m-1-I} l_2 + S_{n-1}^m + \sum_{i=0}^{I-1} \left(\sum_{j=2i}^{m-i-1} S_{n-2-j}^{m-1-i} S_j^i + \sum_{j=2I}^{J-1} S_{n-2-j}^{m-1-I} S_j^I \right)$ 结束 返回 r结束 算法3：对 S_n^m 的与/或树变体进行排序的算法。

UnrankVariant RNASS (r, n, m)
开始
  若 $m = 0$ **则返回** ()
  若 $r < S_{n-1}^{m}$ **则** $v :=$ (0, UnrankVariant RNASS (r, $n-1$, m))
  else
    $r := r - S^m$
    $sum := 0^{n-1}$
    $I := 0$
    $J := 0$
    当 $sum + S_{n-2-J}^{m-1-I} S_J^I \le r$ **时执行**
      $sum := sum + S_{n-2-J}^{m-1-I} S_J^I$
      如果 $J \le n-2\ (m-I)\ -1$，**则** $J := J+1$
      else
        $I := I+1$
        $J := 2I$
      end
    end
    $r := r - sum$
    $l_1 :=$ r 模 S_{n-2-J}^{m-1-I}
    $l_2 := \lfloor r \,/\, S_{n-2-J}^{m-1-I} \rfloor$
    $vl :=$ UnrankVariant RNASS (l_1, $n-2-J$, $m-1-I$)
    $vr :=$UnrankVariant RNASS (l_2, J, I)
    $v :=$(1, $(I,\ J,\ vl,\ vr)$)
  end
  返回 v
end

算法4：用于对 S_n^m的与/或树变体进行反排名的算法。

将算法 1 与算法3结合，可得到对长度为 n 、含有 m 个碱基对的 RNA二级结构组合集合进行排序的算法。将算法4与算法 2结合可执行逆向操作（反排名）。表2和表3给出了对长度为 n 、含有 m 个碱基对的 RNA二级结构组合集合进行排序的示例（这些结构以长度为 n 、含有 m 对括号且允许添加空格的正确括号序列表示）。

表2：对长度为 $n = 6$ 且具有 $m = 2$ 个碱基对的RNA二级结构集合进行排序的示例。

正确括号序列	与/或树变体	Rank
$*((*))$	(0, (1, (0, 0, (1, (0, 0, 0, 0)), 0)))	0
$(*(*))$	(1, (0, 0, (0, (1, (0, 0, 0, 0))), 0))	1
$((**))$	(1, (0, 0, (1, (0, 0, 0, 0)), 0))	2
$((*)*)$	(1, (0, 0, (1, (0, 1, 0, 0)), 0))	3
$((*))*$	(1, (0, 1, (1, (0, 0, 0, 0)), 0))	4
$(*)(*)$	(1,(1, 3,0),(1,(0,0,0,0),0)))	5

Tab. 3: 一个长度为 $n = 8$ 、含有 $m = 3$ 个碱基对的RNA二级结构集合的排序示例。

正确括号序列	与/或树变体	Rank
∗(((∗)))	(0, (1, (0, 0, (1, (0, 0, (1,(0,0,(),())),()))),())))	0
(∗((∗)))	(1, (0, 0, (0, (1, (0, 0, (1, (0, 0, 0, ())), ()))), ())	1
((∗(∗)))	(1、(0、0, (1, (0, 0, (0, (1, (0, 0, 0, ())), ()), ())	2
(((∗∗)))	(1, (0, 0, (1, (0, 0, (1, (0, 0, (), ())), ()), ())	3
(((∗)∗))	(1, (0, 0, (1, (0, 0, (1, (0, 1, (), ())), ()), ())	4
(((∗))∗)	(1, (0, 0, (1, (0, 1, (1, (0, 0, (), ())), ()), ())	5
（（∗）（∗））	(1, (0, 0, (1, (1, 3, (), (1, (0, 0, (), ()))), ())	6
(((∗)))∗	(1,(0,1,(1,(0, 0, (1, (0, 0, (), ())), ()), ())	7
((∗))(∗)	(1, (1, 3, (1, (0, 0, (), ())), (1, (0, 0, 0, ())))	8
(∗)((∗))	(1, (2, 5, (), (1, (0, 0, (1, (0, 0, (), ())), ())))	9

3 结论

本文研究了长度为 n 、含有 m 个碱基对的 RNA二级结构组合集合。为了实现紧凑表示，RNA二级结构被编码为正确的括号序列，并允许添加空格。对于该组合集合，我们构造了一个与/或树结构，找到了组合集合与该与/或树的变体集合之间的双射，并开发了用于对该与/或树的变体进行排名与反排名的算法。

所开发的排序和反排名算法的时间复杂度为：对于$m < n - 2m$，为多项式时间复杂度 $O(m^2(n-m))$；对于 $m > n - 2m$，为 $O(m(n-m)^2)$。所进行的计算实验也证实了这些算法的时间复杂度推导结果。

Seyedi-Tabari 等人提出的现有算法，可对所考虑的长度为 n 、含有 m 个碱基对的RNA二级结构组合集合进行排序和反排名，其时间复杂度也为多项式时间复杂度。然而，使用这些排序和反排名算法需要进行预处理步骤，而该步骤具有指数时间和空间复杂度。与 Seyedi-Tabari 等人的算法相比，新算法无需预处理步骤。因此，所开发的排序和反排名算法具有更优的时间复杂度（所得复杂度为多项式）。

参考文献

P. Flajolet、P. Zimmerman 和 B. Cutsem。组合结构随机生成的演算。理论计算机科学，132(1–2):1–35，1994。doi: 10.1016/0304-3975(94)90226-7。D. E. Knuth。计算机程序设计艺术，第*4A*卷：组合算法，第*1*部分。Addison-Wesley Professional，美国，2011。D. L. Kreher 和 D. R. Stinson。组合算法：生成、枚举与搜索。CRC出版社，美国，1999。D. Kruchinin、V. Kruchinin 和 Y. Shablya。广义纳拉亚纳数的一些性质。*QuaestionesMathematicae*，45(12):1949–1963，2022。doi: 10.2989/16073606.2021.1980448。

S. Mohammadi 和 A. Nowzari-Dalini。一种用于生成长度为 n 且含有 k 个碱基对的RNA二级结构的并行算法。《伊朗计算机科学杂志》，1:11–17，2018。doi: 10.1007/s42044-017-0001-2。M. E. Nebel、A. Scheid 和 F. Weinberg。根据天然分布随机生成RNA二级结构。《分子生物学算法》，6(1):24，2011。doi: 10.1186/1748-7188-6-24。OEIS基金会。在线整数序列百科全书，2022。URL https://page. math.tu-berlin.de/~felsner/SemWS17-18/Ruskey-Comb-Gen.pdf。I. K. Oluoch、A. Akalin、Y. Vural 和 Y. Canbay。RNA二级结构预测算法综述。载于*IEEE*大数据、深度学习与打击网络恐怖主义国际大会（*IBIGDELFT 2018*），第18–23页，2019。doi: 10.1109/ibigdelft.2018.8625347。T. K. Petersen。欧拉数。Birkhauser高级教材：Basler Lehrbücher，美国，2015。doi: 10.1007/978-1-4939-3091-3。E. Picardi。*RNA*生物信息学。Humana出版社，美国，2015。doi: 10.1007/978-1-4939-2291-8。C. Reidys。*RNA*组合计算生物学：假结与中性网络。Springer New York，美国，2011。doi: 10.1007/978-0-387-76731-4。F. Ruskey。组合生成。工作版本（1j-csc 425/520），2003。URL https://page. math.tu-berlin.de/~felsner/SemWS17-18/Ruskey-Comb-Gen.pdf。B. Y. Ryabko。组合对象的快速枚举。《离散数学与应用》，8(2): 163–182，1998年。doi: 10.4213/dm426。W. Saenger。《核酸结构原理》。Springer New York，美国，1984。doi: 10.1007/ 978-1-4612-5190-3。K. Sato 和 Y. Kato。针对长序列的含假结RNA二级结构预测。《生物信息学简报》，23(1):bbab395，2022。doi: 10.1093/bib/bbab395。K. Sato、M. Akiyama 和 Y. Sakakibara。结合热力学积分和深度学习的RNA二级结构预测。《自然通讯》，12:941，2021。doi: 10.1038/s41467-021-21194-4。W. R. Schmitt 和 M. S. Waterman。线性树与RNA二级结构。《离散应用数学》，51(3):317–323，1994。doi: 10.1016/0166-218x(92)00038-n。E. Seyedi-Tabari、H. Ahrabian 和 A. Nowzari-Dalini。一种生成不同类型RNA的新算法。《国际计算机数学杂志》，87(6):1197–1207，2010年。doi: 10.1080/ 00207160802140049。Y. Shablya、D. Kruchinin 和 V. Kruchinin。一种基于与/或树开发组合生成算法的方法及其应用。《数学》，8(6):962，2020。doi: 10.3390/math8060962。G. Viennot 和 M. Vauchaussade de Chaumont。按复杂度枚举RNA二级结构。《生物学与医学中的数学》，第360–365页，1985。doi: 10.1007/978-3-642-93287-8\ 50。

M. S. Waterman。《计算生物学导论》。CRC出版社，美国，2018。

M. Zuker 和 D. Sankoff。RNA二级结构及其预测。《数学生物学通报》，46(4):591–621，1984。doi: 10.1007/bf02459506。