Дуговые структуры на символьных последовательностях

Стариковская Татьяна Андреевна
Студент 3 курса
Московский государственный университет им. М.В.Ломоносова,
Москва, Россия
E-mail: tat.starikovskaya@gmail.com.

1 Определения.

Рассмотрим алфавит *Alp = {0, 1, …, s}*. Будем считать, что для этого алфавита задана *матрица разрешенных спариваний* A – матрица размера $s \times s$ со значениями в *{1, 0},* причем $\forall k$ *A[k, k] = 0*. Пусть *w* - слово длины *n* в алфавите *Alp*. *Дугой* в слове *w* будем называть такую пару позиций *(i, j)* в слове *w*, что *A[w[i], w[j]] = 1*. *Дуговая структура* на слове *w* – это набор дуг, в котором любые 2 дуги не пересекаются, то есть для любых двух дуг (*k, l)* и (*p, t)* выполнено либо $k < l < p < t$, либо $k < p < t < l$, либо $p < t < k < l$, либо $p < k < l < t$. *Весом* дуговой структуры называется количество дуг в ней. *Оптимальная структура* для данного слова *w* – дуговая структура с максимальным количеством дуг. На слове может быть несколько оптимальных структур.

Дуговые структуры на словах используются при моделировании пространственного строения рибонуклеиновых кислот (РНК) [1]; при этом слово *w* описывает последовательность мономеров в полимерной молекуле РНК. При моделировании РНК количество букв равно 4; четыре буквы разбиты на две пары (например, {0,2}; {1, 3}), спаривания разрешены только между разными элементами пары (т.е. только между 0 и 2 и между 1 и 3).
Строение молекулы РНК (т.н. вторичная структура) соответствует оптимальной дуговой структуре.

Цель работы – изучение свойств дуговых структур как комбинаторных объектов и (в дальнейшем) разработка эффективных алгоритмов построения оптимальных дуговых структур. Лучший из известных в настоящее время алгоритмов (см. [1]) основан на методе динамического программирования и имеет временную сложность $O(n^3)$.

2 Процесс сокращения.

На специальных видах последовательностей оптимальные дуговые структуры могут быть найдены за линейное время работы алгоритма.
Утверждение 1. Пусть слово $w \in \{0,1\}^*$; разрешены дуги между 0 и 1. Тогда (1) вес оптимальной дуговой структуры равен *min{#0, #1}*; (2) оптимальная структура может быть построена за время $O(n)$.

Алгоритм (не приводится) основан на т.н. «процессе сокращения».
Определение. Пусть дано слово *w*. (Левым) *процессом сокращения* назовем следующую последовательность действий: найдем в слове *w* первое вхождение двух подряд идущих букв *i, j*, таких, что *A[i, j] = 1* и удалим эту пару. Будем продолжать так делать до тех пор, пока в полученном слове будет хотя бы одно такое вхождение. Весом процесса сокращения назовем количество удаленных пар букв.

Следующее утверждение дает нижнюю оценку средней длины слов, которые получаются в итоге процесса сокращения.

Утверждение 2. Рассмотрим алфавит $\{0,1,2,3\}$; разрешены дуги только между 0 и 2 и между 1 и 3. Применим процесс сокращения ко всем словам длины *n* в этом алфавите. Тогда средняя длина получившихся слов будет больше $\frac{n}{2}$.

3 Сильные отрезки.

Определение. Отрезок называется *сильным*, если в любую его оптимальную структуру входит дуга, соединяющая первую и последнюю буквы.
Используя понятие сильного отрезка, можно усовершенствовать алгоритм [1] и получить алгоритм с оценкой времени работы $O(S*n)$, где S – количество сильных фрагментов в исходном слове *w* [2]. В связи с этим представляет интерес оценка количества сильных фрагментов в слове *w* длины *n.* Как показал Анд. А. Мучник (персональное сообщение), в алфавите {0, 1, 2, 3} с указанной выше матрицей спаривания, для всякого *n* существует слово длины *n,* содержащее больше, чем $\left(\frac{n^2}{6}\right)$ сильных отрезков. Приведенное ниже утверждение может быть полезно при оценке среднего количества сильных фрагментов случайного слова длины *n.*

Утверждение 3. Пусть *r(n)* – среднее количество сильных отрезков на всех последовательностях длины n; $\varphi(n)$ – доля сильных слов длины n среди всех слов длины n. Тогда

$$r(n+1) = \frac{1}{4}n + n*\varphi(1) + (n-1)*\varphi(2) + \ldots 1*\varphi(n).$$

4 Список литературы:

[1] Уотермен М. (ред). Математические методы для анализа последовательностей ДНК, М. Мир, 1999.
[2] Wexler Y., Zilberstein C., Ziv-Ukelson M. A Study of Accessible Motifs and RNA Folding Complexity. Proceedings of RECOMB 2006: 473-487.