Математическая биология и биоинформатика
2020. Т. 15. № 2. С. 313–337. doi: 10.17537/2020.15.313

========== **ОБЗОРЫ** ==========

Сложность ДНК-последовательностей. Различные подходы и определения

Гусев В.Д.*, Мирошниченко Л.А.**

Институт математики им. С.Л. Соболева СО РАН, Новосибирск, Россия

Аннотация. Важной количественной характеристикой символьных последовательностей (текстов, строк) является сложность, отражающая на интуитивном уровне степень их «неслучайности». Достаточно общий подход к оцениванию сложности сформулировал А.Н. Колмогоров. Он предложил измерять её длиной кратчайшего описания, по которому последовательность восстанавливается однозначно. Поскольку программы, гарантированно осуществляющей поиск кратчайшего описания, не существует, на практике для этой цели используют различные алгоритмические приближения, рассматриваемые в данной работе. Наряду с определениями сложности, предполагающими возможность восстановления последовательности по её описанию, рассмотрен и ряд мер, не обладающих указанным свойством. Основное внимание уделено не столько количественной оценке сложности, сколько выявлению и классификации структурных закономерностей, обусловивших конкретное её значение. Все они в той или иной форме сводятся к проявлениям повторности в самом широком смысле.

Рассматриваемые меры сложности можно условно разделить на статистические, учитывающие частоту встречаемости символов или коротких слов в тексте, «словарные», оценивающие число всевозможных подслов в анализируемой последовательности (тексте) и «структурные», основанные на выделении длинных повторяющихся фрагментов текста и установлении взаимосвязей между ними.

Большинство методов ориентировано на последовательности произвольной языковой природы. Особое внимание, уделяемое ДНК-последовательностям, отраженное в названии статьи, обусловлено значимостью объекта, проявлениями повторности разного типа и многочисленными примерами использования понятия сложности при решении задач классификации и эволюции различных биологических объектов. Значительный интерес представляют локальные структурные особенности, выявляемые в режиме скользящего окна в ДНК-последовательностях, поскольку зачастую *зоны пониженной сложности* в геномах различных организмов имеют отношение к регуляции основных генетических процессов.

Ключевые слова: *ДНК-последовательности, сложность, алгоритмы, сжатие данных, энтропия, статистические меры, лингвистическая сложность, структурные меры сложности.*

ВВЕДЕНИЕ

Интуитивные представления о сложности последовательностей обычно связаны со степенью их неслучайности (регулярности). Последовательности, насыщенные длинными повторами (либо большим числом коротких) представляются более простыми. Наличие периодической компоненты в последовательности воспринимается

*gusev@math.nsc.ru
**luba@math.nsc.ru

как фактор, снижающий сложность, причем тем сильнее, чем меньше период этой компоненты. «Случайные» последовательности считаются наиболее сложными [1]. Так, на интуитивном уровне кажется, что последовательности aaaaaaaaaaaa, abababababab, aaaaaabaaaaa, abababbbabab, abbabaababbb перечислены в порядке возрастания сложности. В [2] информация о времени запоминания двоичных последовательностей длины 8 сопоставляется с оценками избыточности [3,4] этих последовательностей. Сходная информация представлена и в [5], где объектом исследования являлись муравьи. Последовательности характеризуют путь в двоичном дереве-лабиринте от корня до узла, содержащего кормушку. Муравей-разведчик, добравшийся до кормушки, передает информацию о ее местонахождении другим муравьям. Время передачи ассоциируется со сложностью пути (последовательности поворотов «левый-правый») по дереву и подтверждает зависимость измеряемых характеристик от наличия периодической компоненты в последовательности и от длины периода.

Колмогоров [6] был одним из первых, кто предложил измерять сложность объекта числом и указал способ такого измерения: сложность трактуется им как длина наиболее короткого описания объекта, по которому этот объект может быть однозначно восстановлен. Аналогичные идеи высказаны и в работах [7–10]. Однако колмогоровское определение сложности неконструктивно в том смысле, что программы, гарантированно осуществляющей поиск кратчайшего (из всех возможных) описаний, позволяющих восстановить последовательность, не существует. Поэтому колмогоровская сложность обычно рассматривается как гипотетическая нижняя граница длины описания объекта, а на практике используются алгоритмические приближения к вычислению этой длины, что приводит к большому разнообразию определений сложности.

Терминология в области оценивания сложности конечных последовательностей не является устоявшейся. Разные подходы могут иметь одно и то же наименование. Например, термин «композиционная сложность» в разных смыслах используется в [11] и [12]. И, наоборот, сходные подходы могут отличаться наименованиями (оценка числа разных подслов заданного слова лежит в основе лингвистической меры сложности [13] и метода, изложенного в [14], апеллирующего к понятию «топологическая энтропия»). По ходу развития аппарата могут модифицироваться сами меры сложности при сохранении наименования. Так, лингвистическая мера сложности фигурирует в мультипликативном [13, 15] и аддитивном [16] вариантах, а также с полным [13, 16] и усеченным [15] множеством подслов.

Отсутствием единого определения сложности последовательности объясняется и многообразие единиц ее измерения. Сложность можно оценивать длиной кодового слова, коэффициентом сжатия, энтропией или числом шагов некоторого процесса представления последовательности. Более того, возможность восстановления последовательности по ее описанию, не является обязательной прерогативой существующих подходов к оцениванию сложности. Так словарные меры для оценки сложности используют число всевозможных подслов в анализируемом тексте.

Понятие сложности объекта (последовательности в данном случае) как длины описания, по которому этот объект может быть однозначно восстановлен, коррелирует со степенью сжатия последовательности. Сложность последовательности тем выше, чем меньше достигаемая степень компрессии. Некомпрессируемая ни одним из известных методов последовательность с большой вероятностью может рассматриваться как случайная. Именно поэтому во многих определениях мер сложности явно или неявно присутствует длина кодового слова. И, обратно, длину кодового слова, полученную каким-либо алгоритмом сжатия текста (последовательности), можно считать оценкой сложности этой последовательности, даже если в описании алгоритма сжатия отсутствует термин «сложность».

Для сжатия ДНК-последовательностей плохо подходят универсальные инструменты сжатия данных. С одной стороны, попытки оценивания информационного содержания ДНК-последовательностей по выборкам значительного объема показали, что это достаточно сложные последовательности, близкие к случайным. С другой стороны, алфавит этих последовательностей состоит из 4 символов (нуклеотидов), что позволяет использовать равномерные двухбитные коды со степенью сжатия 75 %. Поэтому интерес представляют методы со степенью сжатия, превышающей 75 %. Первые инструменты для сжатия ДНК-последовательностей разработаны в 1993–1994 годах [17, 8] и продолжают появляться в наши дни (см., например, [19]). Наряду с алгоритмами сжатия индивидуальных ДНК-последовательностей разрабатываются и «вертикальные» алгоритмы, ориентированные на кодирование с использованием эталонных (референсных) последовательностей и фиксирующие только различия в целевом и эталонном текстах. Идея «вертикальных» алгоритмов высказывается уже в [18], но наибольшее распространение получила в связи с быстрым увеличением числа полностью секвенированных геномов разных людей [20–22]. В [23–25] представлены обзоры по методам сжатия ДНК-последовательностей.

Несмотря на тесную взаимосвязь между алгоритмами сжатия текстов и способами оценивания их сложности, для большинства приложений, связанных с оцениванием сложности, важным элементом является не столько достигаемая степень компрессии, сколько выявление структурных закономерностей, обусловивших сжатие текста (в частности, ДНК-последовательности), и их интерпретируемость. Значительный интерес представляют локальные структурные особенности, выявляемые в режиме скользящего окна. Зоны пониженной сложности характеризуются высоким содержанием различных повторов: как тандемных, так и разнесенных; как прямых, так и комплементарных инвертированных. Именно повторы обеспечивают регуляцию разнообразных генетических процессов в ходе жизненного цикла и являются основой наследственной изменчивости. Тандемно повторяющиеся фрагменты, в частности микросателлиты, нередко используются в качестве биомаркеров для дифференциации близкородственных объектов. С другой стороны, те же микросателлиты затрудняют поиск гомологов по базам данных. При решении этой задачи их приходится локализовывать и отфильтровывать.

Целью данного обзора является освещение определений сложности и методов, направленных на выявление фрагментов ДНК-последовательностей, обладающих пониженной сложностью.

Не претендуя на полноту, рассмотрим следующие классы сложностных мер:

– статистические меры, учитывающие частоту встречаемости символов или коротких слов в тексте. Аномально частое появление отдельных символов (слов) приводит к уменьшению сложности текста;

– словарные меры, основанные на оценивании числа всевозможных подслов в анализируемой последовательности (тексте). Чем больше разнообразие подслов, тем сложнее текст;

– структурные меры, основанные на выделении повторяющихся фрагментов текста и установлении взаимосвязей между ними. Чем больше повторов в тексте и чем они длиннее, тем текст проще.

СТАТИСТИЧЕСКИЕ МЕРЫ СЛОЖНОСТИ

Будем придерживаться следующей системы обозначений: Σ – конечный алфавит, для ДНК-последовательностей $\Sigma = \{a,c,g,t\}$; $\sigma = |\Sigma|$ – число элементов алфавита; S – конечная последовательность, составленная из элементов Σ (текст); $N = |S|$ – длина текста S; $S[i] = S_i$ – элемент S, стоящий в i-й позиции ($1 \le i \le N$); $S\,[i{:}j]$ – фрагмент S, включающий элементы с i-го по j-й ($1 \le i < j \le N$). Термины «слово», «подслово», «цепочка символов»

Математическая биология и биоинформатика. 2020. Т. 15. № 2. doi: 10.17537/2020.15.313

будем использовать как синонимы для обозначения различных фрагментов исходного текста; *l*-грамма – связная цепочка из *l* подряд следующих символов текста (фрагмент $S[i : i + l - 1]$). $S = UV$ – конкатенация (сцепление) последовательностей U и V; $S = Q^k$ – k-кратное повторение слова Q. Специфические обозначения, возникающие при определении разных мер, будут разъясняться по ходу изложения.

Энтропийный подход

Понятие энтропии вероятностного распределения $\vec{p}$ было введено К. Шенноном в [3,4]: $H = -\sum_i p_i \log_2 p_i$. Здесь вектор $\vec{p}$ представляет собой вероятности $\{p_i\}$, $i = 1 \div \sigma$, появления различных элементов алфавита Σ. Энтропия используется обычно для оценки степени отклонения распределения частот элементов алфавита от равномерного (чем меньше значение H, тем более предсказуемым и, соответственно, менее сложным становится текст). Энтропия является нижней границей средней длины кодового слова и может служить оценкой сложности последовательности. Так, для последовательности acaaaaattttttaaaaaga $H = 1.301$.

Обратным образом по отношению к H_1 ведет себя характеристика, называемая избыточностью: $R_1 = 1 - H_1 / \log_2 \sigma$, где σ – размер алфавита. Эта характеристика является нормированной ($0 \le R_1 \le 1$), что облегчает сопоставление разнородных текстов.

Если в качестве объекта рассматривать *l*-граммы, т. е. всевозможные блоки из l символов, порождаемых источником сообщений, то вектор $\vec{p}$ будет содержать вероятности $\{p_i^l\}$ появления различных *l*-грамм. Соответствующую этому распределению энтропию иногда называют блочной: $H_l = -\sum_i p_i^l \log_2 p_i^l$. Блочная энтропия определяет среднее количество информации, содержащееся в блоке длины l. Разность $\Delta_l = H_{l+1} - H_l$ называют дифференциальной энтропией. Она характеризует среднюю неопределенность (условную энтропию) появления $(l + 1)$-й буквы, когда l предыдущих уже известны. При возрастании l учитываются все более далекие статистические связи. Значение l, при котором дифференциальная энтропия $\Delta_l = H_{l+1} - H_l$ достигает максимального значения, т.е. пара (l^*, Δ_{l^*}), где $l^* = \arg\max_{1 \le l \le N} \Delta_l$ рассматривается авторами [26] как мера символьного разнообразия слов и используется для различения растений из семейств пасленовых и крестоцветных на основе анализа их геномов.

На практике используются лишь характеристики H_l невысокого порядка. Это обусловлено тем, что с увеличением l число возможных слов растёт как σ^l и оценка вероятностей их появления по последовательностям ограниченной длины связана со значительными погрешностями. Эта проблема и возможность её «обхода» рассматривалась как в работах самого Шеннона [27], так и других авторов [28–33]. Попытки оценивания информационного содержания ДНК и аминокислотных последовательностей по выборкам значительного объема показали, что это достаточно сложные последовательности, близкие к случайным. Авторы [31], в частности, оценивают редукцию энтропии в аминокислотных последовательностях, обусловленную учетом близких корреляций между символами, всего в 1 %. Компрессионные алгоритмы также оценивают избыточность этих последовательностей примерно в 1 %. Это дает основание авторам [31] подтвердить ранее высказывавшийся тезис о том, что протеиновые последовательности можно рассматривать как слегка отредактированные случайные последовательности. Фиксируемая однопроцентная избыточность объясняется двумя факторами: вкраплениями зон пониженной сложности, возникающих из-за кластеризации отдельных аминокислот, и слабой корреляцией, обусловленной наличием элементов вторичной структуры.

В [34] избыточность генетических текстов оценивается с помощью характеристики, учитывающей наряду с частотами *l*-грамм их неточные копии.

Префиксные коды переменной длины

Один из первых алгоритмов сжатия сформулировали независимо друг от друга американские учёные Клод Шеннон [3] и Роберт Фано [35]. Алгоритм использует коды переменной длины: часто встречающимся символам соответствуют коды меньшей длины, редко встречающимся – коды большей длины. Коды Шеннона – Фано – префиксные, то есть никакое кодовое слово не является префиксом любого другого. Это свойство позволяет однозначно декодировать любую последовательность кодовых слов. Идея алгоритма проста. Символы алфавита Σ упорядочиваются по убыванию частоты их встречаемости в последовательности. Полученное упорядочение делится на две части, суммарные частоты символов которых максимально близки друг другу. Символам первой части присваивается код «0», второй части – «1». Процесс продолжается рекурсивно для каждой из частей.

Так, для последовательности acaaaaattttttaaaaaga таблица, отражающая процесс построения кодов, выглядит следующим образом:

<table>
<tr><th>Σ</th><th>частота</th><th></th><th></th><th></th><th>коды</th></tr>
<tr><td>a</td><td>12/20</td><td>0</td><td></td><td></td><td>0</td></tr>
<tr><td>t</td><td>6/20</td><td rowspan="3">1</td><td>0</td><td></td><td>10</td></tr>
<tr><td>c</td><td>1/20</td><td rowspan="2">1</td><td>0</td><td>110</td></tr>
<tr><td>g</td><td>1/20</td><td>1</td><td>111</td></tr>
</table>

Закодированная последовательность: 01100000010101010100000001110. Её длина составляет 30 бит или 30/8 = 3.75 байт. Средняя длина кодового слова равна 30/20 = 1.5 и незначительно превышает энтропию $H = 1.301$. Аналогичный результат дает и алгоритм Хаффмана [36]. Для однозначного декодирования кроме закодированной последовательности требуется передавать кодовые слова. При использовании равномерных кодов с длиной кодового слова, равной 2, для последовательности той же длины в 4-х символьном алфавите потребовалось бы 40 бит (5 байт).

Адаптивные методы сжатия информации используются для источников с неизвестной или меняющейся статистикой. В них распределение вероятностей символов заранее не известно и для его оценки используется статистика уже закодированной части сообщения. Адаптивные коды позволяют настраиваться на статистику конкретного источника и сжимать данные, порождаемые им, за один просмотр. Адаптивный код Хаффмана [37], частотный код [38], разработанный на основе алфавитного кода Гилберта – Мура [39], алгоритм «стопка книг» [40], интервальный код [41] и модификации арифметического кодирования [42] – вот далеко не полный перечень адаптивных методов.

Стохастическая сложность [43, 44] позволяет учесть зависимость вероятности появления очередных символов источника от предыдущих даже в случае, когда глубина этой зависимости заранее не известна и, в общем случае, переменна. В основе построения статистической модели для вычисления стохастической сложности лежит принцип кратчайшего описания (Minimum Description Length Principle). Для получения оценки распределения вероятностей символов источника по нескольким предыдущим

символам при кодировании и декодировании строится контекстное дерево, включающее наиболее часто встречающиеся фрагменты последовательности.

В качестве меры сложности текста в случае применения этих методов естественно использовать длину кодового слова, а в процессе кодирования можно выявлять зоны пониженной сложности.

Композиционная сложность

Композиционная сложность тесно связана с энтропийной характеристикой первого порядка. Термин «композиция» чаще всего (но не всегда!) используется для обозначения частотного состава последовательности. Применительно к ДНК-последовательности длины L компонентами вектора $\bar{\nu} = (\nu_a, \nu_c, \nu_g, \nu_t)$ являются числа вхождений соответствующих нуклеотидов ($\nu_a + \nu_c + \nu_g + \nu_t = L$). Последовательность S длины $N >> L$ характеризуется набором из $(N - L + 1)$ таких векторов. Если упорядочить компоненты вектора $\bar{\nu}$ по убыванию и игнорировать информацию о том, какому типу нуклеотидов соответствует каждый компонент, получим вектор повторений $\bar{\nu}' = (\nu_1, \nu_2, \nu_3, \nu_4)$ (следуя терминологии из [11, 45]) или вектор состояний [46]. Одному и тому же вектору повторений может соответствовать множество цепочек длины L, отличающихся друг от друга как переупорядочениями элементов (при фиксированных значениях ν_a, ν_c, ν_g и ν_t), так и их переименованиями (сохраняющими лишь $\bar{\nu}'$). Например, цепочки S_1 = acaacggt и S_2 = aatcgacg при одинаковом составе нуклеотидов (т.е. при $\bar{\nu}_1 = \bar{\nu}_2$) отличаются порядком их следования, тогда как цепочки S_1 и S_3 = tcttcgga отличаются переименованием элементов: a $\leftrightarrow$ t ($\bar{\nu}_1 \neq \bar{\nu}_3$, но $\bar{\nu}_1' = \bar{\nu}_3' = (3, 2, 2, 1)$), а S_2 и S_3 – как переименованием, так и переупорядочением.

Для конкретной цепочки S длины L с фиксированным составом нуклеотидов число возможных переупорядочений $W(\bar{\nu}') = \dfrac{L!}{\prod_k \nu_k!}$. Например, цепочка aac, которой соответствует вектор повторений $\bar{\nu}' = (2, 1, 0, 0)$, имеет ровно 3!/2! = 3 варианта переупорядочений (aac, aca и caa). По аналогии с понятием энтропии по Хартли [47] (частный случай шенноновской энтропии при равновероятных исходах) композиционная сложность цепочки S определяется как логарифмическая функция от W:

$$C_{\text{compos}}(S) = (1/L) \log W.$$

Авторы [11, 46] предлагают применительно к ДНК-последовательностям использовать при логарифмировании основание 4. Нетрудно видеть, что минимальной сложностью (нулевой в соответствии с этим определением) обладают серии мононуклеотидов типа x^L ($x \in$ {a, c, g, t}), а максимальной – цепочки с равномерным распределением оснований.

Число возможных переименований элементов 4-буквенного алфавита, не меняющих значения вектора $\bar{\nu}'$, задается выражением $F(\bar{\nu}') = \dfrac{4!}{\prod_i h_i!}$, где h_i – встречаемость числа i ($0 \le i \le L$) в векторе повторений $\bar{\nu}'$, т.е. число ν_k ($1 \le k \le 4$), таких что $\nu_k = i$. Например, для цепочки a^8 ($\bar{\nu}' = (8,0,0,0)$) имеем $h_0 = 3$, $h_8 = 1$, остальные $h_i = 0$, отсюда $F(\bar{\nu}') = 4!/3! = 4$. Если в векторе $\bar{\nu}'$ нет повторяющихся компонентов, то $F(\bar{\nu}') = 4!$. С учетом того, что для каждого варианта переименования (т.е. для фиксированного набора частот нуклеотидов) существует $W(\bar{\nu}')$ возможностей переупорядочения, можно получить априорную оценку правдоподобия вектора $\bar{\nu}'$ в виде $Prob(\bar{\nu}') = F(\bar{\nu}') \times W(\bar{\nu}') / 4^L$.

Априорные вероятности векторов повторений (ожидаемые вероятности) предлагается сравнивать с наблюдаемыми, реализуемыми при скольжении окна размера

L вдоль последовательности. Существенные отклонения значений $Q(\bar{v}') = \log(P_{obs}(\bar{v}') / P_{exp}(\bar{v}'))$ от 0 сигнализируют о скрытых закономерностях, характеризующих специфику различных генетических процессов.

Авторы [11] провели детальное изучение локальной композиционной сложности в подборках функционально эквивалентных последовательностей (преимущественно экзонов или интронов) разных организмов. Исследовалась зависимость (в среднем) величины Q от сложности C_{compos} коротких ($L \le 8$) олигонуклеотидов. Показано, что во всех случаях эта зависимость обнаруживает явный линейный тренд с отрицательным значением тангенса угла наклона (чем больше сложность, тем меньше отличаются P_{obs} от P_{exp}). Наилучшее согласование наблюдаемых и ожидаемых значений отмечается при значении $C_{compos} \approx 0.6$, где Q близко к нулю. При малых значениях сложности ($0 \le C_{compos} \le 0.2$) P_{obs} существенно (в разы) превосходит P_{exp}, т. е. количество низкосложностных олигонуклеотидов в реальных последовательностях значимо превосходит ожидаемое их количество, полученное в предположении равновероятности появления всех олигонуклеотидов. Авторы [46] отмечают, что по углу наклона аппроксимирующей кривой можно достаточно уверенно различать отдельные таксономические группы, а в пределах одной группы – кодирующие области от некодирующих (последние содержат больше низкосложных олигонуклеотидов).

Другие примеры использования композиционной сложности приведены в [48, 49]. Главная цель – локализация и фильтрация областей с низкой сложностью, затрудняющих поиск гомологов по базе данных. Дальнейшее развитие этой меры в направлении учета негомогенности реальных последовательностей и возможности сравнения по сложности последовательностей разной длины представлено в [46].

Термин «композиционная сложность», как уже отмечалось выше, встречается и в ряде других работ [12, 50–52], но уже в другом смысле – как мера неравномерности распределения нуклеотидного состава по длине последовательности. Основная идея сводится к тому, что достаточно длинные последовательности (преимущественно, эукариотические), как правило, характеризуются сложной композиционной гетерогенностью. Они могут быть сегментированы на относительно гомогенные по нуклеотидному составу области (домены), которые, в свою очередь, могут быть разделены на подобласти с более однородным составом и т. д. Для выяснения потенциально возможных точек сегментации используется мера дивергенции Йенсена – Шеннона [53]. Если потенциально возможная точка сегментации делит последовательность S длины N на две области S_1 и S_2 с длинами N_1 и N_2 ($S = S_1S_2$, $N = N_1 + N_2$), то значение меры $JS_2(S_1, S_2) = H(S) - \left(\frac{N_1}{N} H(S_1) + \frac{N_2}{N} H(S_2) \right) \ge 0$, где $H(P) = -\sum_i p_i \log_2 p_i$ – шенноновская энтропия вероятностных распределений, характеризующих нуклеотидный состав последовательностей S, S_1 и S_2 соответственно. Чем выше значение JS_2, тем больше различие по составу нуклеотидов между S_1 и S_2. Значимость получаемого на реальной последовательности значения JS_2 оценивается путем сопоставления его с аналогичными данными для случайной последовательности той же длины и с тем же составом нуклеотидов. Чем меньше вероятность наблюдать при сегментации случайной последовательности сопоставимые или более высокие значения дивергенции, тем более значимо наблюдаемое значение JS_2.

Алгоритм сегментации итеративный [52]. На первой итерации среди всех потенциально возможных значимых (т. е. превышающих по уровню значимости задаваемый порог) точек сегментации отбирается одна с максимальным значением JS_2. Она делит исходную последовательность S на две: S_1 и S_2. Для каждой из них процесс итеративно повторяется. Каждая новая точка сегментации в отдельно взятой подобласти должна: а) фиксировать максимальное значении дивергенции между левым и правым

сегментами разбиения; б) гарантировать значимость наблюдаемого различия; в) проходить тест на сохранение значимых различий между новыми элементами разбиения (левый и правый сегменты подобласти) и уже существующими, смежными с ними. По окончании этого процесса последовательность S оказывается разбитой на $m \geq 2$ сегментов, значимо отличающихся друг от друга по составу нуклеотидов ($S = S_1S_2\ldots S_m$). В соответствии с этим композиционная сложность *SCC* (Sequence Compositional Complexity) определяется как

$$SCC(S) = JS_m(S) = H(S) - \sum_{i=1}^{m} \frac{N_i}{N} H(S_i) = \sum_{i=1}^{m} \frac{N_i}{N} \left(H(S) - H(S_i) \right).$$

Она учитывает, как количество областей в разбиении, так и различия в составах нуклеотидов выделенных областей.

Заметим, что так определенная сложность последовательности (как мера композиционной гетерогенности) по своим свойствам может существенно отличаться от рассмотренных выше мер, учитывающих различные проявления повторности безотносительно к позиционной привязке. В частности, приведенный авторами [52] иллюстрирующий пример по сопоставлению двух достаточно длинных фрагментов ДНК *E. coli* и человека показал, что второй сложнее первого по мере *SCC*, хотя общая тенденция, отмечаемая многими авторами по разным мерам сложности, противоположна: сложность в среднем снижается при переходе от прокариотов к эукариотам.

Алгоритмы серии SIMPLE

Алгоритмы серии SIMPLE предназначены для обнаружения в тексте фрагментов с низкой сложностью (простых последовательностей), характеризующихся кластеризацией коротких (от 1 до 4 символов) прямых повторов. Механизм возникновения таких повторов в ДНК-последовательностях и первая версия алгоритма описаны в [54], дальнейшее развитие применительно к ДНК-последовательностям представлено в [55], а к аминокислотным – в [56]. Хорошо известные простые области в геномах это микросателлиты, теломерные области, CpG-островки, горячие точки рекомбинации, протяженные повторы триплетов. Функции простых областей в протеиновых последовательностях изучены недостаточно, известны лишь функции некоторых (в частности, полиглутаминовых трактов и областей богатых аргинином).

Измерение простоты последовательности ведется в скользящем окне фиксированного размера (например, $10 + l$ – для протеиновых последовательностей и $64 + l$ – для последовательностей ДНК). Фиксируется l-грамма ($l = 1, 2, 3, 4$), расположенная в центре окна, и вычисляется ее частота в окне *SS* (Simplicity Score). Усреднение этой характеристики по всем окнам даёт характеристику всей последовательности *SF* (Simplicity Factor). Чем выше значение *SF*, тем менее сложна последовательность.

Относительная мера кластеризации мотивов *RSF* (Relative Simplicity Factor) вычисляется делением *SF* на среднее значение соответствующих *SF* для 100 случайных последовательностей с той же длиной и тем же частотным составом. Этот показатель служит оценкой значимости *SF*. В общем случае можно получить меру для повторов разной длины, суммируя *SS* по всем l от 1 до 4. Алгоритм позволяет также фиксировать значимые мотивы.

Для случайных последовательностей значения *RSF* близки к 1. Значения *RSF* для случайной подборки прокариотических последовательностей длиной не менее 1000 символов каждая (см. описание эксперимента в [54]) лежат в диапазоне от 1 до 1.4, а для подборки эукариотических последовательностей – от 1 до 2 и выше.

Алгоритмы серии SIMPLE используются для предварительной фильтрации низкосложностных участков в программах поиска гомологов по базам данных. Для этих же целей могут быть использованы программы SEG [48], CAST[57] и др. Сюда же могут быть отнесены многочисленные алгоритмы поиска совершенных и несовершенных тандемных повторов, не имеющих ограничений на длину периода [58, 59].

СЛОВАРНЫЕ МЕРЫ СЛОЖНОСТИ

Количество различных подслов (факторов) фиксированной длины l в бесконечных последовательностях над конечным алфавитом (symbolic complexity) давно является предметом исследования специалистов в области «комбинаторики на словах» (см. для обзора [60, 61]). Этот показатель рассматривается как индикатор случайности последовательности. Например, для двоичных последовательностей он может меняться в диапазоне от константы для периодических последовательностей до экспоненты 2^l для последовательности Чемперноуна, полученной конкатенацией двоичных представлений целых чисел 0, 1, 2, …, l, … (011011100101110…).

Для конечных последовательностей (в частности, геномных) число различных подслов, встречающихся в окне W (фрагменте последовательности S, сдвигающемся вдоль последовательности с определенным шагом) фиксированного размера ($|W|$), авторы [14] называют топологической энтропией. Авторы выбрали для своих исследований размер окна в 500 символов с шагом 250 символов. Эксперименты проводились и с другими размерами окон, результаты оказались согласованными.

Оценка числа различных подслов заданного слова (последовательности S) лежит и в основе лингвистической меры сложности. Для устранения зависимости сложности от длины последовательности авторами [13, 16] введена нормировка. А именно, лингвистическая сложность последовательности S определяется как отношение размера словаря l-грамм ($l = \overline{1, N}$), присутствующих в этой последовательности, к их максимально возможному числу. Пусть M_l – число различных слов длины l в исходной последовательности длины N (размер словаря l-грамм). Максимально возможное значение M_l для последовательностей длины N над алфавитом размера σ равно $\min(\sigma^l, N - l + 1)$.

Мультипликативный вариант лингвистической сложности предложен E.N. Трифоновым [13] в 1990г.: $LC_1(S) = \prod_{l=1}^{N} \frac{M_l}{(\sigma^l, N-l+1)}$

Аддитивный вариант [16] этой меры выглядит следующим образом:

$$LC_2(S) = \frac{\sum_{l=1}^{N} M_l}{\sum_{l=1}^{N} \min(\sigma^l, N-l+1)}$$

Например, для S = aaaaaaaaaaaaaaaacgta (N = 20, Σ = {a,c,g,t})

$$LC_2(S) = \frac{4+5+5+\ldots+5+4+3+2+1}{4+16+18+17+16+\ldots+2+1} = \frac{89}{191} \cong 0.466$$

Краткий обзор применений лингвистической сложности дан в [62]. Одно из основных применений – поиск зон пониженной сложности в геномах различных организмов [14, 16]. Чаще всего они обусловлены тандемной повторностью и имеют отношение к регуляции основных генетических процессов, в частности, влияют на экспрессию генов. О чувствительности метода свидетельствует тот факт, что в отдельных геномах (например, в *Haemophilus influenzae*) с его помощью обнаружены все известные SSRs

(simple sequence repeats) [16]. Взаимосвязь участков низкой сложности с кривизной ДНК (DNA curvature) исследовалась в [15].

Принципиальные различия обнаружены в сложностных профилях gc-богатых прокариотических геномов по сравнению с другими (at-богатыми геномами) на участках, фланкирующих кодирующие области. Характерным признаком at-геномов является наличие зон пониженной сложности, расположенных примерно на расстоянии 50 bp до старта трансляции и непосредственно после ее завершения [16]. Таким образом профиль лингвистической сложности может служить удобным вспомогательным признаком при обнаружении неизвестных генов.

В [63] сложность трактуется как отражение степени наложения разных функциональных сигналов. Поэтому, если какая-либо подборка слабых сигналов (например, нуклеосомного позиционирования) демонстрирует значительный разброс по значениям сложности в отдельных реализациях, ее целесообразно разделить на две группы – высоко- и низкосложностных последовательностей и изучать интересующий нас сигнал по группе последовательностей низкой сложности, где он будет менее зашумлен другими сигналами.

К достоинствам лингвистической меры сложности можно отнести простоту и наглядность определения, эффективность реализации (линейные в зависимости от длины текста алгоритмы, основанные на использовании суффиксных деревьев [64] и графа слов [65]), а также возможность сравнения двух текстов по словарям содержащихся в них *l*-грамм [14,62].

СТРУКТУРНЫЕ МЕРЫ СЛОЖНОСТИ

Все представленные в данном разделе меры сложности близки по духу к колмогоровскому определению этого понятия [6]. Все они используются (или могут быть использованы) для сжатия и последующего восстановления последовательностей без потери информации. Однако, при анализе последовательностей, как уже говорилось выше, на первый план выходит не столько достигаемая степень компрессии, сколько обусловившие её структурные особенности текста, их наглядность и интерпретируемость.

Модификации меры сложности Лемпеля – Зива

В 1976 г. Лемпель и Зив предложили меру сложности конечных символьных последовательностей [66], которая лежит в основе многих алгоритмов сжатия текстовой информации. Основная идея мер типа LZ состоит в том, что второе и последующие вхождения некоторого фрагмента последовательности заменяются ссылками на его первое вхождение. Алгоритмы сжатия LZ-77 [67] и LZ-78 [68] были предложены самими авторами, на их базе разработаны многочисленные модификации: LZSS, LZW, LZWL,LZO, LZMA, LZX, LZRW, LZJB, LZT, LZ4, Brotli, Zstandart и др.

LZ-77 в комбинации с другими методами (в основном, статистическими) лежит и в основе многочисленных алгоритмов сжатия ДНК-последовательностей (biocompress-2 [19], DNAcompress[69], DNASC [70] и др.).

Сложность последовательности по Лемпелю – Зиву равна числу шагов порождающего ее процесса. Допустимыми операциями при этом являются: копирование «готового» фрагмента из уже синтезированной части текста и (или) генерация нового символа.

Схема порождения последовательности S, называемая в дальнейшем сложностным разложением, может быть представлена в виде конкатенации фрагментов $H(S) = S[1 : i_1]\, S[i_1 + 1 : i_2] \ldots S[i_{k-1} + 1 : i_k] \ldots S[i_{m-1} + 1 : N]$, где $S[i_{k-1} + 1 : i_k]$ – фрагмент, синтезируемый на k-м шаге, $m = m_H(S)$ – число шагов процесса. Из всевозможных схем выбирается минимальная по числу шагов, которая и определяет сложность текста:

$C_{LZ}(S) = \min_H \{ m_H(S)\}$. Минимальность числа шагов обеспечивается выбором для копирования на каждом шаге максимально длинного прототипа из словаря или предыстории.

В словарном варианте меры сложности компоненты для копирования выбираются из словаря, достраиваемого по ходу построения разложения. Так, после k-го шага процесса к словарю добавляется k-й компонент, т. е. слово $S[i_{k-1} + 1 : i_k - 1]\, S[i_k]$, где $S[i_{k-1} + 1 : i_k - 1]$ совпадает со словом максимальной длины, имеющемся в словаре перед k-м шагом, а элемент $S[i_k]$ приписывается к k-му компоненту с помощью операции «генерация символа». В «текстовом» варианте компонент $S[i_{k-1} + 1 : i_k]$ формируется путем копирования фрагмента текста $S[1 : i_k - 1]$, совпадающего с $S[i_{k-1} + 1 : i_k]$. При множественности вариантов выбирается фрагмент, обеспечивающий максимально возможное значение i_k. В последнем случае операция «генерация символа» может быть использована только при необходимости, когда нечего копировать. Поскольку нас больше интересуют закономерности, присутствующие в последовательностях, далее будем иметь в виду текстовый вариант меры сложности. Каждый компонент сложностного разложения кодируется парой (позиция, с которой начинается копирование; длина компонента). При генерации символа указывается пара (0; генерируемый символ).

Например, пошаговый процесс построения S = ataatataatatttttaatatt выглядит следующим образом: $H(S)$ = a * t * a * ata * taatat* ttttt * aatatt. Элементы алфавита (в данном случае это два первых компонента «a» и «t») получены с использованием операции генерации символа, остальные компоненты копируются из сформированного префикса. На рисунке 1 стрелками поясняется схема копирования. Начало каждой стрелки указывает на стартовую позицию, с которой осуществляется копирование очередного компонента разложения. Конец стрелки указывает на первый элемент этого компонента. Процесс копирования, начавшийся с какого-либо элемента синтезированного префикса S, может быть продолжен с использованием элементов, только что синтезированных на данном шаге. Примером может служить построение компонента ttttt. Это свойство лежит в основе алгоритма выявления тандемных повторов (периодичностей) [71]. Число компонентов в разложении определяет сложность последовательности $c(S)$. В нашем примере $c(S) = 7$. Кодировка выглядит следующим образом: (0; a)(0; t) (1; 1)(1; 3)(2; 6)(12; 5)(8; 6).

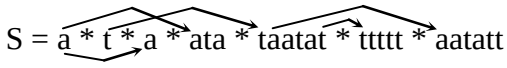


Рис. 1. Схема сложностного разложения последовательности S = ataatataatatttttaatatt.

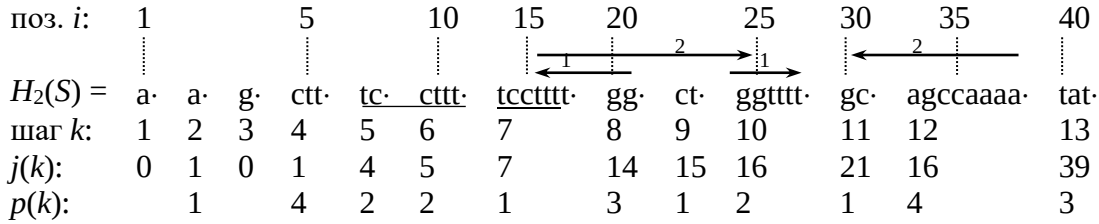
ДНК-ориентированный вариант меры сложности (мера C_2)

Вариант меры сложности, предназначенный для анализа ДНК-последовательностей (мера C_2), основан на использовании нескольких типов операций копирования в режиме конкурирования. На каждом шаге синтеза последовательности выбирается та операция, которой соответствует максимальный по длине прототип в уже синтезированном префиксе последовательности. Компонент сложностного разложения с номером k характеризуется своей длиной l_k, указателем копирования $j(k)$ и типом используемой операции (подстановки) $p(k)$. В ДНК-ориентированной мере сложности, предложенной в [72] и исследованной в [73–77], фигурируют 4 типа операций копирования: прямое (p_1, соответствует обычным повторам); копирование фрагмента с изменением направления считывания: (p_2, соответствует симметричным повторам); прямое комплементарное копирование (p_3, фиксирует прямые повторы с точностью до комплементарной подстановки a ↔ t, g ↔ c); симметричное комплементарное копирование (p_4, фиксирует

Математическая биология и биоинформатика. 2020. Т. 15. № 2. doi: 10.17537/2020.15.313

комплементарные инвертированные повторы). Алгоритм вычисления меры C_2 легко модифицировать на случай, когда используется только одна из перечисленных операций копирования или две (как правило, прямое и комплементарное инвертированное)

Например, схема синтеза *S* (фрагмента некодирующей последовательности из генома дрожжей *S. cerevisiae*) с пошаговой оптимизацией выбора операции копирования (от 1 до 4) имеет вид:



Сложность последовательности *S* по мере C_2 равна числу компонентов в $H_2(S)$, т. е. $C_2(S) = k_{max} = 13$. В приведенном разложении первый и третий компоненты генерируются ($j(1) = j(3) = 0$), остальные копируются. Например, компонент разложения с номером $k = 10$ (ggtttt) копируется симметричным образом с фрагмента $S[16 : 21]$ = ttttgg ($j(10) = 16$). При наличии различных (по позиции) вариантов копирования используется прототип, ближайший к порождаемому компоненту. Позиционная близость компонента и его прототипа может свидетельствовать об их совместном функционировании. Стрелками и подчеркиванием выделены наиболее характерные структуры: тандемный повтор $(tccttt)^2$ с началом в 7 позиции, симметричный повтор (1) и симметричный комплементарный повтор (2), образующий структуру шпилечного типа с симметричной петлей gttttg. Составные структуры подобного рода достаточно типичны, равно как и комбинации типа «комплементарный палиндром», фланкированный симметричными или прямыми повторами [78], локальные фракталы [71] и т. п.

Последняя структура связана с повторением в ограниченном участке текста палиндромов или комплементарных палиндромов и интересна тем, что приводит к усилению конструкции, т. е. к образованию новых палиндромов (обычных или комплементарных) большей длины. Например, при тандемном повторении симметричной цепочки catac образуется симметричная цепочка вдвое большей длины cataccatac. Аналогично, при повторении комплементарного палиндрома, скажем tgca, возникает новый комплементарный палиндром вдвое большей длины tgca tgca. Алгоритм выявления всех фракталоподобных структур такого рода, разработанный авторами статьи, описан в [71].

Основное достоинство ДНК-ориентированной меры C_2 состоит в том, что соответствующие ей сложностные разложения участков с аномально низкой сложностью легко интерпретируются в терминах структур, понятных биологу. Мера эффективно работает как в режиме скользящего окна (сложностные профили [79]), выявляя локальные структурные закономерности, так и при анализе генома в целом [80], выявляя его крупноблочную структуру (аномально длинные тандемные и разнесенные повторы, повторяющиеся гены, встроенные мобильные элементы и т. п.).

В [81] идея сложностного разложения перенесена на пары и группы текстов. В случае двух текстов один текст представляется в виде покрытия фрагментами из другого. Каждый фрагмент – это общий для двух текстов повтор одного из 4 типов (p_1–p_4). Чем длиннее компоненты разложения, тем меньше элементов в покрытии, и, соответственно, ближе тексты. В случае нескольких текстов каждый из них можно представить в виде конкатенации фрагментов из других текстов (на условиях конкуренции). Указатели копирования $j(k)$ при этом фиксируют связи данного текста с остальными, длины копируемых компонентов $l(k)$ характеризуют силу связи, а значения $p(k)$ – тип связи.

Математическая биология и биоинформатика. 2020. Т. 15. № 2. doi: 10.17537/2020.15.313

Меры близости текстов, построенные на таких разложениях, в некоторой степени коррелированы с эволюционным (редакционным) расстоянием, но обладают рядом преимуществ по сравнению с традиционными алгоритмами выравнивания: они адекватно реагируют на крупноблочные перестройки, эффективнее в вычислительном отношении, у них расширен спектр допустимых операций.

В общем случае при пошаговой оптимизации выбора способа копирования необязательно ограничивать себя четырьмя способами, фигурирующими в определении меры C_2. Можно допускать любую из $|\Sigma|!$ возможных подстановок на элементах алфавита в сочетании с прямым и симметричным копированием (всего $2|\Sigma|!$ вариантов выбора). Соответствующая мера сложности (C_3) и алгоритм ее вычисления, позволяющий обойти факториальный перебор, рассмотрены в [76]. С помощью этой меры можно выделять значимые подстановки, соответствующие аномально длинным компонентам сложностного разложения, при любом (конечном) размере алфавита. Мера C_3 пока что не нашла широкого применения при анализе ДНК- и аминокислотных последовательностей. Она в значительной степени носит дешифровочный характер, поэтому ее можно рассматривать как инструмент для получения «отрицательных результатов». Применительно к ДНК-последовательностям это означает, что не выявлено значимых подстановок кроме тождественной (при прямом копировании) и комплементарной (при симметричном копировании). Применительно к аминокислотным последовательностям, опять же, единственно значимой подстановкой является тождественная (прямое копирование). Вопрос о наличии значимых подстановок при различных агрегированиях алфавита аминокислот пока не исследован. Отметим, что ввиду своей универсальности, мера C_3 применима для разных языковых систем. В частности, она может быть использована для поиска плагиатов в текстах программ или сходных (с точностью до звуковысотного переноса) мелодических фрагментов в музыкальных текстах.

Мера C_2 использовалась для анализа блочных перестроек в промоторах генов гормона роста у позвоночных (от человека до рыб) [82]. Попытки сравнения этих последовательностей путем попарного и множественного выравнивания имели лишь частичный успех. Удалось получить хорошее выравнивание для последовательностей млекопитающих (приматы, копытные, грызуны), однако попытки выравнивания промоторных последовательностей курицы, амфибии (bullfrog) и пяти видов рыб оказались неудачными из-за большого числа блочных перестроек (протяженных делеций и вставок, перестановок блоков, их тиражирования) у эволюционно более далеких организмов. Проведенный анализ промоторов с использованием меры C_2 позволил подтвердить и расширить концепцию «блочного перемешивания промоторов» [83]. Блочная структура промоторов гена гормона роста у рыб оказалась наиболее разнообразной и вариативной. Она демонстрирует значимое сходство со структурой аналогичного промотора у амфибии. В то же время блочная структура этого промотора у курицы оказалась более похожей на структуру промоторов млекопитающих. Гипотеза о блочной структуре промоторов и перемешивании блоков в ходе эволюции подтвердилась и при анализе промоторов гена β-глобина [84]. Подборки функционально эквивалентных последовательностей, формируемые из баз данных в автоматическом режиме, часто содержат дубликаты, вложения, пересечения, высокогомологичные экземпляры, вкрапления мобильных элементов и т.п. Возможность выявления такого рода взаимосвязей между текстами подборки с помощью совместных сложностных разложений иллюстрируется в [80, 81]. Режим получения усредненного сложностного профиля для подборок сфазированных функциональных сайтов описан в [85].

Редуцированный до двух операций (прямое и симметричное копирование при тождественной подстановке) вариант меры C_2 использовался для сравнения последовательностей дисков политенных хромосом при построении филогенетического

дерева семейства хирономид (комаров-звонцов: Diptera, Chironomidae) [86]. Эти насекомые являются хорошими биоиндикаторами состояния водоемов на планете. Использование операции симметричного копирования здесь является принципиальным, поскольку эволюция последовательностей дисков, отражающих рисунок гигантских политенных хромосом, в основном происходит за счет инверсий (разрыв хромосомы в двух точках и изменение порядка следования дисков между ними на противоположный).

В эволюционирующих во времени языковых системах чаще встречаются несовершенные повторы, чем совершенные. Обобщения мер сложности, использующие копирование с искажениями [87, 88], лучше соответствует реальному положению вещей. Алгоритмическая реализация несовершенного копирования сложнее, чем совершенного, равно как и оценка значимости получаемых результатов. Несмотря на это, повторы с искажениями используются в некоторых алгоритмах сжатия ДНК-последовательностей (см., например, GenCompress [89]). Алгоритм DNACompress [69] также использует для кодирования приближенные повторы, выделяемые с помощью инструмента PatternHunter [90] в сочетании с подходом Лемпеля-Зива [67].

Существуют и определения сложности последовательностей, в идейном плане схожие с определениями типа LZ, но с другим порядком формирования компонентов в процессе синтеза последовательности. Если в схеме Лемпеля – Зива и её модификациях последовательность «генерируется» слева – направо, то при вычислении суффиксной сложности [91] компоненты формируются справа – налево. Схема сборки [92] синтезирует последовательность в произвольном порядке.

Грамматическая сложность

Грамматическая сложность связана с построением грамматики для языка, представленного одной конкретной («индивидуальной») последовательностью. Контекстно-свободной грамматикой называется четверка $G = (\Sigma, Q, P, q_0)$, где Σ – множество терминальных символов (алфавит языка); Q – множество нетерминальных символов $(\Sigma \cap Q) = \varnothing$; P – конечное множество правил вывода (продукций) вида $\alpha \to \beta$, где $\alpha \in Q$, $\beta \in (Q \cup \Sigma)^*$, т. е. β – цепочка символов произвольной длины из объединенного алфавита; q_0 – начальный символ ($q_0 \in Q$).

Сложность K любого правила $\alpha \to \beta$ из P определяется длиной его правой части, т.е. $K(\alpha \to \beta) = |\beta|$. Сложность последовательности S, порождаемой грамматикой G, определяется суммарной длиной правил вывода $K_G(S) = \sum_{P \in G} K(\alpha \to \beta)$. Поскольку текст может быть порожден разными грамматиками, интерес представляет характеристика $K(S) = \min_G K_G(S)$, которая называется грамматической сложностью текста [93]. Для оценки этой величины существуют различные приближенные алгоритмы [93–96]. Неплохие приближения могут быть получены и путем построения конкретной грамматики, адекватно отражающей проявления повторности в тексте. Пример построения такой грамматики описан в [97]. Общая идея состоит в рекурсивной замене повторяющихся слов порождающими их грамматическими правилами. Синтез грамматических правил осуществляется таким образом, чтобы на каждом шаге процесса выполнялись два условия:

P1 (условие «биграммной уникальности»): никакая пара смежных фрагментов не появляется в правых частях правил вывода более одного раза;

P2 (условие «полезности»): каждое правило $\alpha \to \beta$ используется не менее двух раз.

Грамматика строится слева направо. Для первого символа текста она имеет вид $q_0 \to S_1$. После i шагов имеем грамматику для порождения $S[1:i]$, первое правило которой имеет вид $q_0 \to uv$, где $u \in (Q \cup \Sigma)^*$; $v \in (Q \cup S_i)$. На $(i+1)$-м шаге S_{i+1} добавляется к правилу q_0. При этом образуется биграмма vS_{i+1}. Если она ранее не

Математическая биология и биоинформатика. 2020. Т. 15. № 2. doi: 10.17537/2020.15.313

встречалась, переходим к следующему шагу. Если в грамматике есть правило $q_k \to vS_{i+1}$, правило $q_0 \to uvS_{i+1}$ меняется на $q_0 \to uq_k$. Если такого правила нет, а биграмма vS_{i+1} в грамматике где-то есть, вводится новое правило типа $q_j \to vS_{i+1}$ и оба вхождения vS_{i+1} заменяются на q_j. После проведения такой замены правило P1 выполняется, но может нарушиться P2. Если какой-то нетерминал A остался в единственном числе (в правых частях правил вывода), его вхождение меняется на соответствующую правую часть правила A → w, а само правило исключается из грамматики.

Например, для первых 9 символов последовательности S = abcdbcabcd получена грамматика G = (Σ = {a,b,c,d}, Q = {q_0, A, B}, P = {q_0→ BdAB; A→bc; B→aA}. Тогда 10-й шаг процесса построения грамматики выглядит следующим образом:

10	abcdbcabcd	S→ BdABd A→bc B→aA	Bd встречается дважды
		S→CAC A→bc B→aA C→Bd	нарушается P2 B встречается один раз
		S→ CAC A→bc C→aAd	

Сложность этой грамматики равна 8.

Возможности метода иллюстрируются на естественно-языковых [98], музыкальных текстах и ДНК-последовательностях [99], но идеология подхода (построение иерархии повторов [100]) применима к текстам любой языковой природы. Авторы делают упор на выявление иерархической структуры текста, однако отмечают и высокий компрессионный потенциал метода. На основе грамматической сложности построены такие алгоритмы сжатия ДНК(РНК)-последовательностей как DNASequitur [101] и RNACompress [102].

ОСНОВНЫЕ ВЫВОДЫ И ИХ ОБСУЖДЕНИЕ

1. В основе практически всех языковых систем, как естественных, так и формальных лежит понятие повтора в самом широком смысле (прямые и симметричные, разнесенные и тандемные, совершенные и несовершенные, с точностью до переименования или агрегирования элементов алфавита и т. п.). Значимые проявления повторности наблюдаются на разных иерархических уровнях и, как правило, имеют содержательную (эволюционно-функциональную) интерпретацию. Формальным отражением степени насыщенность текста повторами является такая его характеристика как сложность. Текст тем сложнее, чем меньше в нем повторов. Единого определения сложности не существует. Различные подходы к определению сложности отличаются по типу учитываемой повторности.

2. Колмогоровская сложность [6] конечной последовательности – это длина кратчайшей порождающей ее (с помощью универсального вычислительного устройства типа машины Тьюринга) программы. С этой точки зрения просматривается тесная связь между алгоритмами сжатия текстов и оценивания их сложности. Однако главной целью при практическом использовании сложностных оценок является не сжатие, как таковое, а выявление и интерпретация закономерностей, на которых оно основано. Поэтому некоторые определения сложности допускают потерю информации необходимой для однозначного восстановления последовательности.

3. Существуют два режима оценивания сложности: для последовательности в целом и в скользящем окне фиксированной длины. Первый режим (последовательность в целом) интересен в плане выявления крупноблочной структуры текста (аномально длинные тандемные и разнесенные повторы, повторяющиеся (в том числе и в разной ориентации) гены, зоны обширной гомологии и т.п.). Возможности сравнения по сложности последовательностей разной длины при этом существенно ограничены из-за того, что используемые нормировки не устраняют полностью зависимость сложности от длины текста. Весьма перспективен вариант представления одного текста в виде конкатенации фрагментов (общих повторов) из другого.

Второй режим (вычисление сложности в скользящем окне) является мощным эффективным средством обнаружения локальных структурных закономерностей, часто характеризующихся аномально низкими значениями сложности. Основная трудность заключается в интерпретации этих закономерностей, которая не всегда очевидна. Размер окна характеризует разрешающую способность сложностного анализа. Вопрос о рациональном выборе системы окон разной длины, наилучшим образом отражающих иерархическую структуру текста, остается открытым.

Сравнение текстов разной длины по сложности можно проводить по усредненным для каждого текста значениям сложности в окне фиксированного размера D (усреднением проводится по всем $N - D + 1$ возможным положениям окна, при этом предполагается, что $D << N$). Результаты сравнения при разных значениях D обычно согласуются друг с другом. Редкие исключения требуют специального рассмотрения.

4. Различные меры сложности, используемые в режиме скользящего окна, дают коррелированные результаты. Это объясняется наличием взаимосвязи между числом разных подслов в окне анализа (см. лингвистическую меру сложности), частотой их встречаемости (см. статистические меры сложности) и длинами повторов (см. структурные меры сложности). Так, уменьшение разнообразия подслов приводит к снижению сложности во всех трех типах мер.

Несмотря на значительную корреляцию между разными мерами сложности, можно, тем не менее, исходя из общих соображений, предполагать, что меры, использующие лишь повторы фиксированной длины, будут несколько уступать по чувствительности мерам, учитывающим полный спектр повторов, а меры использующие информацию лишь о размере алфавита подслов – мерам, использующим частоты. ДНК-специфичные меры должны выявлять больше аномалий в ДНК-последовательностях, чем универсальные. В смысле интерпретируемости и наглядности получаемых результатов предпочтение следует отдать методам, указывающим на конкретные структуры (тандемы, палиндромно-шпилечные структуры, локальные фракталы и т. п.) – этим характеризуются меры серии LZ. Выявление иерархической структуры текста – прерогатива грамматических мер сложности.

5. Сложностной анализ геномов позволяет сделать ряд выводов интересных в эволюционном отношении. Можно отметить, в частности:

- бóльшую сложность в среднем кодирующих фрагментов по сравнению с некодирующими;
- как возможное следствие первого – бóльшую сложность прокариотических последовательностей по сравнению с эукариотическими;
- отсутствие аномально длинных симметричных повторов, за исключением фрагментов типа $(CT)^n$, $(ACA)^n$, $(GAAG)^n$ и т.п., когда симметрия формально возникает внутри тандемной периодичности, но не имеет «собственного» механизма порождения;
- отсутствие достаточно длинных участков с аномально высокой сложностью;
- тяготение участков аномально низкой сложности к границам достаточно крупных структурных элементов (например, экзон-интрон, конец одного гена – начало другого и т. п.);

- наличие интересных классов структур, практически не обсуждаемых в литературе. Сюда относятся локальные фракталы со свойством «усиления закономерности»; составные структуры типа шпилька с симметричной петлей, разнесенная симметрия с комплементарным палиндромом внутри, периодичность со сменой длины периода, блочная симметрия, дупликативные знаки пунктуации (например, промоторы) и другие [72, 75].

6. Использование сложностных разложений с разными типами операций копирования (векторная мера [76] и мера C_2) позволяет выявлять фрагменты текста с наложением структур разного типа. Их можно интерпретировать как многозначные слова в естественном языке. Смысл слова (т. е. представление о том, какая структура реально работает) зависит от контекста и внешних условий (ситуативной обстановки). Анализу многозначности в естественных языках уделяется большое внимание. Актуальна эта проблема и применительно к генетическому языку [103, 63]. Функциональная перегруженность какого-либо участка текста может послужить причиной его эволюционной перестройки (соответствующий пример, выявленный при сопоставлении геномов родственных бактериофагов φx174 и g4, описан в [75]). Сложностной анализ предлагает адекватный инструмент для выявления и исследования участков текста, где имеет место «суперпозиция кодов» (в терминологии Трифонова [103]).

7. Техника сложностного анализа естественным образом переносится с одного текста на пары и группы текстов при разных определениях сложности, что позволяет сравнивать тексты друг с другом без предварительного их выравнивания (см., например, [14, 81, 104–108]) по проявлениям того или иного типа повторности. Это актуально для сильно разошедшихся последовательностей с явными проявлениями генетических перестановок (genetic shuffling) [82–84] и других типов крупноблочных перестроек [86, 109–111]. Именно эта техника активно применяется для сопоставления геномов в целом [112].

8. Кроме подходов, описанных выше, существуют и другие, связанные с рассматриваемой тематикой, но выходящие (по разным причинам) за рамки данного обзора. Достаточно много работ посвящено определению и анализу сложности бесконечных символьных последовательностей, получаемых по какой-либо формальной схеме типа итерации морфизмов [113]. Они интересны различными модификациями в определении сложности [114–118], а также тем, что для отдельных классов последовательностей оценки сложности могут быть получены в аналитической форме. Следует, однако, заметить, что до практического использования многих мер сложности дело не доходит из-за трудностей, связанных с их интерпретацией.

Отдельная группа работ посвящена изучению близких и дальних корреляций в символьных (в том числе и в ДНК) последовательностях [119–124]. Хотя термин «сложность» при этом в явном виде не используется, речь идет о выявлении закономерностей в организации текста, учет которых может способствовать более компактному его описанию. Для изучения корреляций используется аппарат корреляционных функций (они измеряют только линейные зависимости) или оценки взаимной информации (обнаруживают любые типы зависимостей). Преобразование Фурье от автокорреляционной функции дает спектр мощности (power spectra). Утверждается [47, 125], что по виду спектральной картинки можно различать ДНК-последовательности: а) гомогенные в смысле базисной композиции; б) с «простой» гетерогенностью (как у бактериофага λ); со «сложной» гетерогенностью иерархического типа (как в первой хромосоме генома дрожжей *Saccharomyces cerevisiae*). Дальние корреляционные связи обнаружены как в кодирующих, так и в некодирующих областях, хотя они оцениваются как относительно слабые. Природа их пока дебатируется в

научных кругах, но в качестве одной из причин дальних связей в кодирующих областях указывается на неравномерность использования кодонов [120].

Третья группа работ связана с введением для сравнения последовательностей метрик (и мер близости) отличных от метрики Левенштейна (синонимичные термины: редакционное или эволюционное расстояние), используемой при выравнивании последовательностей [104–106, 112, 125]. Определение трансформационного расстояния [106] созвучно определению относительной сложности [81]. Обе меры обладают возможностью учета сложных блочных перестроек и повторов общего вида. Сравнение последовательностей без их предварительного выравнивания становится все более востребованным в практике анализа биологических текстов [107].

Итак, основное внимание в обзоре уделено трем типам определений мер сложности – статистическим, словарным и структурным. Первые учитывают частоту встречаемости отдельных символов или коротких *l*-грамм, вторые оценивают размер словаря, т.е. разнообразие *l*-грамм всевозможной длины, третьи основаны на представлении последовательностей в терминах повторов (как правило, максимально длинных).

Работа выполнена в рамках государственного задания ИМ СО РАН (проект № 0314-2019-0015)

СПИСОК ЛИТЕРАТУРЫ

1. Knuth D.E. *The Art of Computer Programming:* Vol. 2. *Seminumerical Algorithms*. Addison-Wesley Publishing Company, 1969.
2. Jermann W.H. Redundancy in deterministic sequences. *IEEE Trans. on Syst. Sci. and Cybernetics.* 1970. V. 6. № 4. doi: 10.1109/TSSC.1970.300313
3. Shannon C. A mathematical theory of communication. *Bell System Techn. J*. 1948. V. 27. № 3. P. 379–423. doi: 10.1002/j.1538-7305.1948.tb01338.x
4. Shannon C. A mathematical theory of communication. *Bell System Techn. J*. 1948. V. 27. № 4. P. 623–656. doi: 10.1002/j.1538-7305.1948.tb00917.x
5. Резникова Ж.И., Рябко Б.Я. Анализ языка муравьев методами теории информации. *Проблемы передачи информации*. 1986. Т. XXII. № 3. С. 103–108.
6. Колмогоров А.Н. Три подхода к определению понятия «количество информации». *Проблемы передачи информации*. 1965. Т. 1. № 1. С. 3–11.
7. Solomonoff R. *A Preliminary Report on a General Theory of Inductive Inference*. Cambridge, Ma.: Zator Co., 1960.
8. Solomonoff R. A. Formal theory of inductive inference. Part I. *Information and Control* . 1964. V. 7(1). P. 1–22. doi: 10.1016/S0019-9958(64)90223-2
9. Chaitin G. Information-theoretic limitations of formal systems. *Journal of the ACM*. 1974. V. 21(3). P. 403–424. 10.1145/321832.321839
10. Левин Л.А. О различных мерах сложности конечных объектов (аксиоматическое описание). *Доклады АН СССР*. 1976. Т.227 (4). С.804–807.
11. Salamon P., Konopka A.K. A maximum entropy principle for the distribution of local complexity in naturally occurring nucleotide sequences. *Computers Chem*. 1992. V. 16. № 2. P. 117–124. doi: 10.1016/0097-8485(92)80038-2.
12. Román-Roldán R., Bernaola-Galván P., Oliver J.L. Sequence compositional complexity of DNA through an entropic segmentation method, *Physical Review Letters*. 1998. V. 80. P. 1344–1347. doi: 10.1103/PhysRevLett.80.1344
13. Trifonov E.N. Making sense of the human genome. In: *Structure & Methods*. Eds. Sarma R.H., Sarma M.H. Adenine Press, 1990. V. 1. P. 69–77.
14. Crochemore M., Verin R. Zones of low entropy in genomic sequences. *Computers and Chemistry*. 1999. V. 23. P. 275–282. doi: 10.1016/S0097-8485(99)00009-1

15. Gabrielian A.E., Bolshoy A. Sequence complexity and DNA curvature. *Comput. Chem*. 1999. V. 23. P. 263–274. doi: 10.1016/S0097-8485(99)00007-8
16. Troyanskaya O.G., Arbell O., Koren Y., Landau G.M., Bolshoy A. Sequence complexity profiles of prokaryotic genomic sequences: a fast algorithm for calculating linguistic complexity. *Bioinformatics*. 2002. V. 18. № 5. P. 679–688. doi: 10.1093/bioinformatics/18.5.679
17. Grumbach S., Tahi F. Compression of DNA sequences. *Proc. IEEE Symp. on Data Compression*. 1993. P. 340–350. doi: 10.1109/DCC.1993.253115
18. Grumbach S., Tahi F. A new challenge for compression algorithms: genetic sequences. *J. Information Processing and Management*. 1994. V 30. № 6. P. 875–866. doi: 10.1016/0306-4573(94)90014-0
19. Pratas D., Hosseini M., Silva J.M., Pinho A.J. A reference-free lossless compression algorithm for DNA sequences using a competitive prediction of two classes of weighted models. *Entropy***.** 2019. V. 21. № 11. P. 1074. doi: 10.3390/e21111074
20. Brandon M.C., Wallace D.C., Baldi P. Data structures and compression algorithms for genomic sequence data. *Bioinformatics*. 2009. V. 25. № 14. P. 1731–1738. doi: 10.1093/bioinformatics/btp319
21. Deorowicz S., Grabowski S. Robust relative compression of genomes with random access. *Bioinformatics*. 2011. V. 27. № 21. P. 2979–2986. doi: https://doi.org/10.1093/bioinformatics/btr505
22. Pavlichin D.S., Weissman T, Yona G. The human genome contracts again. *Bioinformatics*. 2013. V. 29. № 17. P. 2199–2202. doi: 10.1093/bioinformatics/btt362
23. Bakr N.S., Sharawi A.A. DNA lossless compression algorithms: Review. *American Journal of Bioinformatics Research*. 2013. V. 3. № 3. P. 72–81. doi: 10.5923/j.bioinformatics.20130303.04
24. Zhu Z., Zhang Y., Ji Z., He S., Yang X. High-throughput DNA sequence data compression. *Briefings in Bioinformatics*. 2015. V. 16. № 1. P. 1–15. doi: 10.1093/bib/bbt087
25. Hosseini M., Pratas D., Pinho A. A survey on data compression methods for biological sequences. *Information*. 2016. V. 7. № 4. P. 56. doi: 10.3390/info7040056
26. Сметанин Ю.Г., Ульянов М.В., Пестова А.С. Энтропийный подход к построению меры символьного разнообразия слов и его применение к кластеризации геномов растений. *Математическая биология и биоинформатика*. 2016. Т. 11. № 1. С. 114–126 . doi: 10.17537/2016.11.114
27. Shannon C. Prediction and entropy of printed English. *Bell System Techn. J*. 1951. V. 30. № 1. P. 50–64. doi: 10.1002/j.1538-7305.1951.tb01366.x
28. Herzel H. Complexity of symbol sequences. *Systems Analysis Modelling Simulation*. V. 5. № 5. 1988. P. 435–444.
29. Ebeling W., Nicolis G. Word frequency and entropy of symbolic sequences: a dynamical perspecrive. *Chaos, Solitons and Fractals*. 1992. V. 2. № 6. P. 635–650. doi: 10.1016/0960-0779(92)90058-U
30. Schmitt A.O., Herzel H. Estimating the entropy of DNA sequences. *J. Theor. Biol*. 1997. V. 188. P. 369–377. doi: 10.1006/jtbi.1997.0493
31. Weiss O., Jiménes-Montaño M.A., Herzel H. Information content of protein sequences. *J. Theor. Biol*. 2000. V. 206. P. 379–386. doi: 10.1006/jtbi.2000.2138
32. Farach M., Noordewier M., Savari S., Shepp L., Syner A., Ziv J. On the entropy of DNA: algorithms and measurements based on memory and rapid convergence. In: *Proceedings of the 6th ACM-SIAM Symposium on Discrete Algorithms*. New-York: ACM, Inc., 1995. P. 48–57.
33. Loewenstern D., Yianilos P.N., Significantly lower entropy estimates for natural DNA sequences. *J. Comput. Biol*. 1999. V. 6. P. 125–142. doi: 10.1089/cmb.1999.6.125

34. Kisliuk O.S., Borovina T.A., Nazipova N.N. Estimation of redundancy of genetic texts by the high frequency component of the *L*-gram graph. *Biophysics*. 1999. V. 44 (4). P. 621–630.
35. Фано Р. *Передача информации. Статистическая теория связи*. М.: Мир, 1965. 438 с.
36. Huffman D. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*. 1952. V. 40. № 9. P. 1098–1101. doi: 10.1109/JRPROC.1952.273898
37. Knuth D.E. Dynamic Huffman Coding. *Journal of Algorithms*. 1985. V. 6. № 2. P. 163–180. doi: 10.1016/0196-6774(85)90036-7
38. Рябко Б.Я. Быстрый алгоритм адаптивного кодирования. *Пробл. передачи информ.* 1990. Т. 26. № 4. С. 24–37.
39. Gilbert E.N., Moore E.F. Variable-length binary encodings. *Bell System Technical Journal*. 1959. V. 38. № 4. P. 933–967. doi: 10.1002/j.1538-7305.1959.tb01583.x
40. Рябко Б.Я. Сжатие данных с помощью стопки книг. *Пробл. передачи информ.* 1980. Т. 16. № 4. С. 16–21.
41. Nigel G., Martin N. Range encoding: An algorithm for removing redundancy from a digitized message. *Video & Data Recording Conference.* Southampton, UK, 1979.
42. Said A. Introduction to Arithmetic Coding Theory and Practice. In: *Lossless Compression Handbook.* Ed. Sayood K. Elsevier Inc., 2003. P. 101–152.
43. Barron A., Rissanen J., Yu B. The minimum description length principle in coding and modeling. *IEEE Transactions on Information Theory*. 1998. V. 44. № 6. doi: 10.1109/18.720554
44. Orlov Y.L., Filippov V.P., Potapov V.N., Kolchanov N.A. Construction of stochastic context trees for genetic texts. *In Silico Biology*. 2002. V. 2. № 3. P. 233–247.
45. Konopka A.K. Sequences and codes: fundamentals of biomolecular cryptology. In: *Biocomputing: Informatics and Genome Projects.* Ed. Smith D.W. New York: Academic Press, 1994. P. 119–174. doi: 10.1016/B978-0-08-092596-7.50008-3
46. Wan H., Wootton J.C. A global compositional complexity measure for biological sequences: AT-rich and CG-rich genomes encode less complex proteins. *Computers and Chem.* 2000. V. 24. № 1. P. 71–94. doi: 10.1016/S0097-8485(00)80008-X
47. Hartley R.V.L. Transmission of Information. *Bell Syst Techn J.* 1928. V. 7. № 3. P. 535–563. doi: 10.1002/j.1538-7305.1928.tb01236.x
48. Wootton J.C., Federhen S. Statistics of local complexity in amino acid sequences and sequence databases. *Computers &. Chemistry*. 1993. V. 17. № 2. P. 149–163. doi: 10.1016/0097-8485(93)85006-X
49. Wootton J.C., Federhen S. Analysis of compositionally biased regions in sequence databases. *Methods in Enzymology*. 1996. V. 266. 554–571. doi: 10.1016/S0076-6879(96)66035-2
50. Bernaola-Galvan P., Román-Roldán R., Oliver J.L. Compositional segmentation and long-range fractal correlation in DNA sequences. *Phys. Rev. E.* 1996. V. 53. № 5. P. 5181–5189. doi: 10.1103/PhysRevE.53.5181
51. Li W. The complexity of DNA: the measure of compositional heterogeneity in DNA sequences and measures of complexity. *Complexity*. 1997. V. 3. № 2. P. 33–37. doi: 10.1002/(SICI)1099-0526(199711/12)3:2%3C33::AID-CPLX7%3E3.0.CO;2-N
52. Oliver J.L., Román-Roldán R., Pérez J., Bernaola-Galván P. SEGMENT: identifying compositional domains in DNA sequences. *Bioinformatics*. 1999. V. 15. № 2. P. 974–979.
53. Lin J. Divergence measure based on the Shannon entropy. *IEEE Transactions on Information Theory*. 1991. V. 37. P. 145–151.
54. Tautz D., Trick M., Dover G.A. Cryptic simplicity in DNA is major source of genetic variation. *Nature*. 1986. V. 322. P. 652–656.

55. Hancock J.M., Armstrong J.S. SIMPLE34: an improved and enhanced implementation for VAX and Sun computers of the SIMPLE algorithm for analysis of clustered repetitive motifs in nucleotide sequences. *Comput. Appl. Biosci*. 1994. V. 10. P. 67–70.
56. Alba M. Mar, Laskowski R.A., Hancock J.M. Detecting cryptically simple protein sequences using the SIMPLE algorithm. *Bioinformatics.* 2002. V. 5. P. 672–678. doi: 10.1093/bioinformatics/18.5.672.
57. Promponas V.J., Enright A.J., Tsoka S., Kreil D.P., Leroy C., Hamodrakas S., Sander C., Ouzounis C.A. CAST: an iterative algorithm for the complexity analysis of sequence tracts. *Bioinformatics*. 2000. V. 16. № 10. P. 915–922. doi: 10.1093/bioinformatics/16.10.915.
58. Benson G. Tandem repeats finder: a program to analyze DNA sequences. *NAR*. 1999. V. 22. № 2. P. 573–580. doi: 10.1093/nar/27.2.573
59. Чалей М.Б., Кутыркин В.А., Тюльбашева Г.Э., Теплухина Е.И., Назипова Н.Н. Исследование феномена скрытой периодичности в геномах эукариотических организмов. *Математическая биология и биоинформатика*. 2013. Т. 8(2). С. 480–501. doi: 10.17537/2013.8.480
60. Lothaire M. *Combinatorics on Words*. Reading, MA: Addison-Wesley, 1983.
61. Ferenczi S. Complexity of sequences and dynamical systems. *Discrete Mathematics*. 1999. V. 206. № 1–3. P. 145–154. doi: 10.1016/S0012-365X(98)00400-2
62. Bolshoy A. DNA sequence analysis linguistic tools: contrast vocabularies, compositional spectra and linguistic complexity. *Applied Bioinformatics*. 2003. V. 2. № 2. P. 103–112.
63. Bolshoy A., Shapiro K., Trifonov E.N., Ioshikhes I. Enhancement of the nucleosomal pattern in sequences of lower complexity. *Nucl. Acids Res*. 1997. V. 25. P. 3248–3254. doi: 10.1093/nar/25.16.3248
64. Ukkonen E. On-line constructing of suffix trees. *Algorithmica*. 1995. V. 14. P. 249–260. doi: 10.1007/BF01206331
65. Blumer A., Blumer J., Ehrenfeucht A., Haussler D., McConnel R. Building the minimal DFA for the set of all subwords of a word on-line in linear time. *Lect. Notes in Comput. Sci.* 1984. V. 172. P. 109–118. doi: 10.1007/3-540-13345-3_9
66. Lempel A., Ziv J. On the complexity of finite sequences. *IEEE Trans. Inform. Theory*. 1976. V. IT-22. № 1. P. 75–81. doi: 10.1109/TIT.1976.1055501
67. Ziv J., Lempel A. A universal algorithm for sequential data compression. *IEEE Trans. Inform. Theory*. 1977. V. IT-23. № 3. P. 337–343. doi: 10.1109/TIT.1977.1055714
68. Ziv J., Lempel A. Compression of individual sequences via variable-rate coding. *IEEE Trans. Inform. Theory*. 1978. V. IT-24. № 5. P. 530–536. doi: 10.1109/TIT.1978.1055934
69. Chen X., Li M., Ma B., Tromp J. DNACompress: fast and effective DNA sequence compression. *Bioinformatics.* 2002. V. 18. № 12. P. 1696–1698. doi: 10.1093/bioinformatics/18.12.1696
70. Mishra K.N., Aaggarwal A., Abdelhadi E., Srivastava D. An efficient horizontal and vertical method for online dna sequence compression. *Int. J. Comput. Appl.* 2010. V. 3. № 1. P. 39–46. doi: 10.5120/757-954
71. Гусев В.Д., Мирошниченко Л.А., Чужанова Н.А. Выявление фракталоподобных структур в ДНК-последовательностях. *Information Science & Computing. International Book Series*, № 8: *Classification, Forecasting, Data Mining*. Sofia: ITHEA, 2009. P. 117−123.
72. Гусев В.Д., Куличков В.А., Чупахина О.М. Сложностной анализ генетических текстов (на примере фага λ): препринт. Новосибирск: ИМ СОАН СССР, 1989. № 20. 49 стр.
73. Гусев В.Д., Куличков В.А., Чупахина О.М. Сложностной анализ геномов. I Меры сложности и классификация выявляемых закономерностей. *Молекулярная биология*. 1991. Т. 25. № 3. С. 825−833.

74. Гусев В.Д., Куличков В.А., Чупахина О.М. Сложностной анализ геномов. II. Зоны обширной гомологии в бактериофаге λ. *Молекулярная биология*. 1991. Т. 25. № 4. С. 1080–1089.
75. Gusev V.D., Kulichkov V.A., Chupakhina O.M. The Lempel–Ziv complexity and local structure analysis of genomes. *Biosystems*. 1993. V. 30. № 1–3. P. 183–200.
76. Gusev V.D., Nemytikova L.A., Chuzhanova N.A. On the complexity measures of genetic sequences. *Bioinformatics.* 1999. V. 15. № 12. P. 994–999.
77. Гусев В.Д., Мирошниченко Л.А. Использование сложностных разложений в задачах анализа символьных последовательностей. В: *Доклады 8-й Международной конференции "Интеллектуализация обработки информации"* (*ИОИ-2010*) (*Кипр, Пафос, 17-24 октября 2010*). 2010. С. 469–472.
78. Гусев В.Д., Мирошниченко Л.А. Поиск комбинированных структур в ДНК-последовательностях. В: *Доклады всероссийской конференции ММРО-13* «*Математические методы распознавания образов*» (*Ленинградская обл., г. Зеленогорск. 30 сентября-6 октября 2007г.*). М.: Макс-Пресс, 2007. С. 473−476.
79. Гусев В.Д. Сложностные профили символьных последовательностей. Методы обработки символьных последовательностей и сигналов. В: *Вычислительные системы*. Вып. 132. Новосибирск, 1989. С. 35−63.
80. Orlov Yu.L., Gusev V.D., Miroshnichenko L.A. LZcomposer: decomposition of genomic sequences by repeat fragments. *Biophisics*. 2003. V. 48. Suppl. 1. P. S7−S16.
81. Гусев В.Д., Немытикова Л.А., Чужанова Н.А. Быстрый метод выявления взаимосвязей в подборках функционально и/или эволюционно близких биологических текстов. *Молекулярная биология*. 2001. Т. 35. № 6. С. 1015–1022.
82. Chuzhanova N.A., Krawczak M., Nemytikova L.A., Gusev V.D., Cooper D.N. Promoter shuffling has occurred during the evolution of the vertebrate growth hormone gene. *Gene*. 2000. V. 254. P. 9–18. doi: 10.1016/S0378-1119(00)00308-5
83. Surguchov A. Migration of promoter elements between genes: a role in transcriptional regulation and evolution. *Biomed. Sci*. 1991. V. 2. P. 22–28.
84. Chuzhanova N.A., Krawczak M., Thomas N., Nemytikova L.A., Gusev V.D., Cooper D.N. The evolution of the vertebrate beta–globin gene promoter. *Evolution*. 2002. V. 56. № 2. P. 224–232. doi: 10.1111/j.0014-3820.2002.tb01333.x
85. Orlov Yu.L., Potapov V.N. Estimation of stochastic complexity of genetical texts. *Computational technologies* (*Novosibirsk*). 2000. V. 5. Special issue. P. 5–15.
86. Kiknadze I.I., Gunderina L.I., Istomina A.G., Gusev V.D., Nemytikova L.A. Similarity analysis of inversion banding sequences in chromosomes of *Chironomus* species (breakpoint phylogeny). In: *Bioinformatics of Genome Regulation and Structure*. Eds. N. Kolchanov, R. Hofestaedt. Boston, MA: Springer, 2004. P. 245–254. doi: 10.1007/978-1-4419-7152-4_26
87. Григорьева А.Н. Меры сложности слов на основе предиката вхождения и редакционного расстояния. *Зап. научн. семинаров ЛОМИ АН СССР*. 1981. Т. 105. С. 18–24.
88. Allison L., Edgoose T., Dix T.I. Compression of strings with approximate repeats. In: *Intelligent Systems in Molecular Biology* (*ISMB'98*) (*Montreal, 28 June-1 July 1998*). 1998. P. 8–16.
89. Chen X., Kwong S., Li M. A compression algorithm for DNA sequences and its applications in genome comparison. *Genome informatics. International Conference on Genome Informatics*. 1999. V. 10. P. 51–61.
90. Ma B., Tromp J., Li M. PatternHunter: Faster and more sensitive homology search. *Bioinformatics.* 2002. V. 18. № 3. P. 440–445. doi: 10.1093/bioinformatics/18.3.440
91. Мерекин Ю.В. Нижняя оценка сложности для схем конкатенации слов. *Дискретн. анализ и исслед. опер.* 1996. Т. 3. № 1. С. 52–56.

92. Евдокимов А.А. Анализ, сложность и реконструкция символьных последовательностей. *Вестник ТГУ*. 2005. № 14. С. 4–12.
93. Ebeling W., Jiménes-Montaño M.A. On grammars, complexity, and information measures of biological macromolecules. *Math. Biosci*. 1980. V. 52. P. 53–71. doi: 10.1016/0025-5564(80)90004-8
94. Jiménes-Montaño M.A. On syntactic structure of protein sequences and the concept of grammar complexity. *Bull. Math. Biol*. 1984. V. 46. P. 641–659. doi: 10.1007/BF02459508
95. Jiménes-Montaño M.A., Pöschel T., Rapp P.E. A measure of the information content of neural spike trains. *Proc. Symp. on Complexity in Biology*. Eds. Mizraji E., Acerenza L., Alvares F., Pomi A. Montevideo, Uruguay: D.I.R.A.C., 1997. P. 113–142.
96. Charikar M., Lehman E., Liu D., Panigrahy R., Prabhakaran M., Sahai A., Shelat A. The smallest grammar problem. *IEEE Transactions on Information Theory.* 2005. V. 51. № 7. P. 2554–2576. doi: 10.1109/TIT.2005.850116
97. Nevill-Manning C.G., Witten I. H. Identifying hierarchical structure in sequences: a linear-time algorithm. *Journal of Artificial Intelligence Research*. 1997. V. 7. P. 67–82. doi: 10.1613/jair.374
98. Witten I.H. Adaptive text mining: inferring structure from sequences. *Journal of discrete algorithms*. 2004. V. 2. № 2. P. 137–159. doi: 10.1016/S1570-8667(03)00084-4
99. Carrascosa R., Coste F., Gallé M., Infante-Lopes G. Searching for smallest grammars on large sequences and application to DNA. *Journal of Discrete Algorithms*. 2012. V. 11. P. 62–72. doi: 10.1016/j.jda.2011.04.006
100. Nevill-Manning C.G., Witten I. H. Online and offline heuristics for inferring hierarchies of repetitions in sequences. *Proc IEEE*. 2000. V. 88. № 11. P. 1745–1755. doi: 10.1109/5.892710
101. Cherniavsky N., Ladner R. Grammar-based compression of DNA sequences. In: *Proceedings of the DIMACS Working Group on the Burrows-Wheeler Transform.* New Jersey, 2004.
102. Liu Q., Yang Yu., Chen C., Bu J., Zhang Y., Ye X. RNACompress: Grammar-based compression and informational complexity measurement of RNA secondary structure. *BMC Bioinformatics*. 2008. V. 9. P. 176. doi: 10.1186/1471-2105-9-176
103. Трифонов Э.Н. Генетическое содержание последовательности ДНК определяется суперпозицией многих кодов. *Молекулярная биология*. 1997. Т. 31. № 4. С. 759–767.
104. Bennett C.H., Glacs P., Li M., Vitányi P., Zurek W.H. Information Distance. *IEEE Trans. on Inf. Th*. 1998. V. 44. № 4. P. 1407–1423. doi: 10.1109/18.681318
105. Li M., Chen X., Li X., Ma B., Vitanyi P.M.B. The similarity metric. *IEEE Trans. on Inf. Th*. 2004. V. 50. № 12. P. 3250–3264. doi: 10.1109/TIT.2004.838101
106. Varré J.-S., Delahaye J.-P., Rivals E. Transformation distances: a family of dissimilarity measures based on movements of segments. *Bioinformatics*. 1999. V. 15. № 3. P. 194–202. doi: 10.1093/bioinformatics/15.3.194
107. Vinga S., Almeida J.S. Alignment-free sequence comparison - a Review. *Bioinformatics.* 2003. V. 19. № 4. P. 513–523. doi: 10.1093/bioinformatics/btg005
108. Wallace C.S., Boulton D.M. An information measure for classification. *Computer J*. 1968. V. 11. № 2. P. 185–194. doi: 10.1093/comjnl/11.2.185
109. Sankoff D., Leduc G., Antoine N., Paquin B., Lang B.F., Cedergren R. Gene order comparison for phylogenetic inference: Evolution of the mitochondrial genome. *PNAS USA*. 1992. V. 89. P. 6575–6579. doi: 10.1073/pnas.89.14.6575
110. Sankoff D., Nadeau J.H. Conserved synteny as a measure of genomic distance. *Discrete Appl. Math*. 1996. V. 71. P. 247–257. doi: 10.1016/S0166-218X(96)00067-4

111. Bafna V., Pevzner P.A. Sorting by reversals: genome rearrangements in plant organelles and evolutionary history of X chromosome. *Molecular Biology and Evolution*. 1995. V. 12. № 2. P. 239–246. doi: 10.1093/oxfordjournals.molbev.a040208
112. Li M., Badger J.H., Chen X., Kwong S., Kearney P., Zhang H. An information-based sequence distance and its application to whole mitochondrial genome phylogeny. *Bioinformatics*. 2001. V. 17. № 2. P. 149–154. doi: 10.1093/bioinformatics/17.2.149
113. Salomaa A. *Jewels of formal language theory*. Rockville: Computer Science Press, 1981.
114. Iványi. On the d-complexity of words. *Ann. Univ. Sci Budapest Sect Comput*. 1987. V. 8. P. 69–90.
115. Nakashima I., Tamura J., Yasutomi S., Modified complexity and *-Sturmian word. *Proc. Japan Acad. Ser. A Math. Sci.* 1999. V. 75. № 3. P. 26–28. doi: 10.3792/pjaa.75.26
116. Kamae T., Zamboni L. Sequence entropy and the maximal pattern complexity of infinite words. *Ergodic Theory Dynamical Systems*. 2002. V. 22. № 4. P. 1191–1199. doi: 10.1017/S014338570200055X
117. Restivo A., Salemi S. Binary patterns in infinite binary words. In: *Formal and Natural Computing. Lecture Notes in Computer Science*. Eds. Brauer W., Ehrig H., Karhumäki J., Salomaa A. Berlin, Heidelberg: Springer, 2002. V. 2300. P. 107–116. doi: 10.1007/3-540-45711-9_8
118. Frid A.E. Arithmetical complexity of symmetric DOL words. *Theoretic Computer Science*. 2003. V. 306. P. 535–542. doi: 10.1016/S0304-3975(03)00345-1
119. Herzel H., Grobe I. Measuring correlations in symbol sequences. *Phisica A*. 1995. V. 216. P. 518–542. doi: 10.1016/0378-4371(95)00104-F
120. Buldyrev S.V., Goldberger A.L., Havlin S., Mantegna R.N., Matsa M.E., Peng C.-K., Simons M., Stanley H.E. Long-range correlations properties of coding and non-coding DNA-sequences – GenBank analysis. *Physical Review E*. 1995. V. 51. P. 5084–5091. doi: 10.1103/PhysRevE.51.5084
121. Havlin S., Buldyrev S.V., Goldberger A.L., Mantegna R.N., Peng C.-K., Simons M., Stanley H.E. Statistical and linguistic features of DNA sequences. *Fractals*. 1995. V. 3. № 2. P. 269–284. doi: 10.1142/S0218348X95000229
122. Karlin S., Brendel V. Patchiness and correlations in DNA sequences. *Science*. 1993. V. 259. № 5095. P. 677–680. doi: 10.1126/science.8430316
123. Voss R.F. Long-range fractal correlations in DNA introns and exons. *Fractals*. 1994. V. 2. P. 1–6. doi: 10.1142/S0218348X94000831
124. Li W. The study of correlation structures of DNA sequences: a critical review. *Computer & Chem*. 1997. V. 21. № 4. P. 257–271. doi: 10.1016/S0097-8485(97)00022-3
125. Cormode G., Paterson M., Sahinalp S.C., Vishkin U. Communication complexity of document exchange. In: *Proc. Eleventh ACM-SIAM Symposium on Discrete Algorithms* (*SODA*). 2000. P. 197–206

Рукопись поступила в редакцию 23.10.2020, переработанный вариант поступил 14.11.2020.
Дата опубликования 30.11.2020.

========== **REVIEWS** ==========

The complexity of DNA sequences. Different approaches and definitions

Vladimir D. Gusev, Liubov A. Miroshnichenko

Sobolev Institute of Mathematics Siberian Branch of the Russian Academy of Sciences

Abstract. An important quantitative characteristic of symbolic sequence (texts, strings) is complexity, which reflects at the intuitive level the degree of their "non-randomness". A.N. Kolmogorov formulated the most general definition of complexity. He proposed measuring the complexity of an object (symbolic sequence) by the length of the shortest descriptions by which this object can be uniquely reconstructed. Since there is no program guaranteed to search for the shortest description, in practice, various algorithmic approximations considered in this paper are used for this purpose. Along with definitions of complexity, suggesting the possibility of reconstruction a sequence from its "description", a number of measures are considered that do not imply such restoration. They are based on the calculation of some quantitative characteristics. Of interest is not only a quantitative assessment of complexity, but also the identification and classification of structural regularities that determine its specific value. In one form or another, they are expressed in the demonstration of repetition in the broadest sense. The considered measures of complexity are conventionally divided into statistical ones that take into account the frequency of occurrence of symbols or short “words” in the text, “dictionary” ones that estimate the number of different “subwords” and “structural” ones based on the identification of long repeating fragments of text and the determination of relationships between them.

Most of the methods are designed for sequences of an arbitrary linguistic nature. The special attention paid to DNA sequences, reflected in the title of the article, is due to the importance of the object, manifestations of repetition of different types, and numerous examples of using the concept of complexity in solving problems of classification and evolution of various biological objects. Local structural features found in the sliding window mode in DNA sequences are of considerable interest, since zones of low complexity in the genomes of various organisms are often associated with the regulation of basic genetic processes.

Key words: *DNA sequences, algorithms, complexity, entropy, data compression, statistical measures, linguistic measure of complexity, structural measures of complexity.*

Acknowledgments. The study was carried out within the framework of the state contract of the Sobolev Institute of Mathematics (project no. 0314-2019-0015).