

ОБ ОДНОЙ ГИПОТЕЗЕ ТЕОРИИ ФОРМАЛЬНЫХ ЯЗЫКОВ. ЧАСТЬ I

© 2023 Б.Ф. Мельников¹

¹ Совместный университет МГУ – ППИ в Шэньчжэне
(Китай, 518172, провинция Гуандун, г. Шэньчжэнь, район Лунган,
Даюньсиньчэн, улица Гоцзидасюеюань, дом 1)
E-mail: bormel@mail.ru

Поступила в редакцию: 20.06.2023

Основной предмет статьи — рассмотрение задач, возникающих при исследовании необходимых условий равенства бесконечных итераций конечных языков. В предыдущих публикациях автором рассматривались примеры применения соответствующего этому равенству специального бинарного отношения эквивалентности на множестве конечных языков, причём рассматривались как примеры, описывающие необходимые условия его выполнения, так и примеры его использования. К одному из таких необходимых условий применены два варианта сведения рассматриваемой задачи: к конечным автоматам и к бесконечным итерационным деревьям. Также в статье приведены несколько вариантов важной гипотезы, формулируемой для множества конечных языков; ее исследование дает и иные варианты сведения рассматриваемой задачи к специальным задачам для недетерминированных конечных автоматов. При этом в случае выполнения сформулированной гипотезы некоторые из таких задач решаются за полиномиальное время, а некоторые не решаются; при продолжении работ по данной тематике последний факт может дать возможность переформулировки проблемы $P=NP$ в виде специальной задачи теории формальных языков.

Ключевые слова: формальные языки, итерации языков, бинарные отношения, морфизмы, инверсные морфизмы, алгоритмы, полиномиальные алгоритмы.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Мельников Б.Ф. Об одной гипотезе теории формальных языков. Часть I // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. X, № Y. С. Z1–Z2. DOI: 10.14529/cmse230X0Z.

Введение

Некоторые результаты предлагаемой работы уже были опубликованы, причем относительно недавно, в 2019 г. и позднее – см. [1–12]. Однако их желательно повторить (с добавлениями и изменениями) — по следующим причинам.

- Во-первых, необходима публикация на эту тематику именно в журнале, тематика которого в большей степени связана с прикладной математикой: в упомянутых выше публикациях большее внимание уделялось связи с теоретической информатикой¹.
- Во-вторых, приводятся подробные обозначения для применяемой в статье версии недетерминированных конечных автоматов (соответствующие определения и простейшие утверждения). Эти обозначения для конечных автоматов на русском языке уже были опубликованы – но в полном объеме лишь в монографии, [13].
- В-третьих, немного по-другому описываются результаты статьи [14] (1993 г.) — где было приведено *неполное* доказательство одного из рассматривавшихся там утверждений.

¹ Хотя, конечно, провести четкую границу между собственно математикой и теоретической информатикой вряд ли возможно.

- В-четвертых, хотя настоящую работу можно назвать обобщающей, — но в нее автор постарался включить только такой минимум из материала ранее опубликованных статей, который необходим для понимания рассматриваемых проблем. Здесь под этим минимумом в первую очередь понимается то, что необходимо для плана сведения равенства $P = NP$ (гипотезы $P = NP$) к специальной гипотезе теории формальных языков.
- Однако при этом («в-пятых»), такой минимально возможный объем материала отражает не только сами результаты работы, но и связь с другими ранее рассматривавшимися задачами. Более того, как автор надеется, при такой подаче, которая применена в настоящей обобщающей статье, более понятна мотивация к рассмотрению вспомогательных задач — а эту мотивацию при простом чтении в хронологическом порядке статей [1–12] можно и не увидеть.
- В-шестых, в настоящей статье добавлены некоторые новые результаты, полученные в последнее время.
- Однако самой важной является следующая причина («в-седьмых»): существенно изменяется сама структура изложения. А именно, в упомянутых выше работах основной целью было решение соответствующих вспомогательных задач, в частности — описание возможных полурешеток, полученных на основе двух заданных конечных языков, а также описание применения результатов, связанных с этими полурешетками, в прикладных задачах теории языков. А у настоящей статьи цель более глобальная, и поэтому приводится:
 - подробное описание самих рассматриваемых задач;
 - описание нескольких гипотез, возникающих для специальных множеств формальных языков (иными словами — для подмоноидов глобального надмоноида свободного моноида, [15]);
 - а также связь этих задач с упомянутой выше проблемой $P = NP$.

Все упомянутые выше задачи связаны с исследованием бинарного отношения эквивалентности в бесконечности, заданного на множестве конечных языков, — отношения \Leftrightarrow . В предыдущих публикациях уже были рассмотрены примеры применения как этого отношения, так и его частного случая — отношения \Leftarrow ; при этом примеры можно разделить на две группы:

- примеры, описывающие необходимые условия его выполнения
- и примеры его использования — в различных областях теории формальных языков, дискретной математики и абстрактной алгебры.

Из сказанного выше следует, что основным предметом статьи можно считать рассмотрение разных задач, возникающих при исследовании необходимых условий равенства бесконечных итераций конечных языков (иными словами — условий выполнения бинарного отношения \Leftrightarrow). И, по-видимому, главной «составляющей» такой мотивации является план доказательства возможности сведения равенства $P = NP$ к специальной гипотезе теории формальных языков, сформулированной ранее в предыдущих публикациях. Стоит отметить по этому поводу, что первая версия этого плана была опубликована в 2011 г., см. [16] — а здесь приведена измененная версия, причем со значительно большим числом комментариев. Однако основная идея при этом та же самая — но здесь она уже оформлена конкретно, и решения многих вспомогательных подзадач, необходимых для осуществления указанного плана, за прошедшее время уже опубликованы.

Кратко этот план можно сформулировать следующим образом. По некоторому недетерминированному конечному автомату специальным образом строится пара конечных языков — и для этой пары показывается, что если бы была выполнена вышеупомянутая гипотеза, то существовал бы и алгоритм проверки выполнения отношения эквивалентности в бесконечности за полиномиальное время. Но, с другой стороны, взяв произвольный недетерминированный конечный автомат и рассматривая его как определенный в предыдущих публикациях т. н. автомат NSPRI (мы его строго определим и будем рассматривать начиная с раздела 6 настоящей статьи), мы за полиномиальное время могли бы построить соответствующую этому автомату пару конечных языков — причём такую пару, которая удовлетворяет отношению \Leftrightarrow тогда и только тогда, когда язык автомата NSPRI совпадает с универсальным языком над заданным алфавитом (т. е. языком, содержащим все возможные слова). Таким образом, в случае осуществления плана, кратко описанного здесь, мы покажем, что выполнение вышеупомянутой основной гипотезы влечет выполнение равенства $P = NP$.

В конце данного раздела отметим следующее. Мы всюду пишем «сведение равенства $P = NP$ к одной из гипотез теории формальных языков» (и т. п.) — но, по-видимому, еще точнее было бы говорить «сведение гипотезы $P = NP$ к одной из гипотез теории формальных языков»; однако последний вариант звучит хуже (из-за двух вхождений слова «гипотеза»). На сказанное здесь можно было бы возразить, что любая (или почти любая) вычислительная проблема может рассматриваться именно как проблема теории формальных языков (см. [17, 18] и мн. др.); однако «возражение на возражение» заключается в том, что для получения необходимой информации нужно решить *только одну* проблему, или, другими словами, здесь речь идет о формулировке задачи про задачи, можно сказать — «о формулировке метазадачи».

Следует также особо отметить, что даже после возможного завершения реализации плана, приведенного выше, о возможности решения проблемы $P = NP$ ничего утверждаться не будет: будет показана только возможность сведения этой проблемы к проблеме теории формальных языков.

Приведем содержание статьи по разделам.

В разделе 1 приводятся основные обозначения и обсуждаются соглашения об их использовании; некоторые из этих обозначений являются нестандартными. Одним из наиболее важных понятий, определенных в этом разделе, является бинарное отношение \Leftrightarrow , определенное на множестве конечных языков над рассматриваемым алфавитом; можно сказать, что с этим бинарным отношением связан весь материал настоящей статьи.

В разделе 2 рассматриваются *бесконечные деревья особого вида*, которые возникают в различных задачах теории формальных языков. Отличительной особенностью этих деревьев является то, что все их ребра отмечены буквами заданного алфавита, а вершины делятся на некоторые классы эквивалентности, причём (бесконечные) *поддеревья, соответствующие вершинам одного и того же класса, эквивалентны*. Разделом 2 заканчивается часть I настоящей статьи, последующие разделы составят части II–IV.

В разделе 3 обсуждается несложное *преобразование*, строящее по заданному бесконечному итерационному дереву специальный детерминированный конечный автомат. При этом с точки зрения алгебры мы переходим от работы с элементами определенного множества к работе с классами эквивалентности, на которые это множество делится с помощью бинарного отношения, фактически рассмотренного в предыдущем разделе 2.

Таким образом, в разделе 3 рассматривается общий (абстрактный) вариант преобразования произвольного бесконечного итерационного дерева в детерминированный конечный автомат; а в *разделах 4 и 5* — конкретное дерево, получаемое не «абстрактно», как в разделе 3 и ранее, а специально описанным образом по двум заданным конечным языкам; соответственно, получается связанный с этими языками детерминированный автомат. Эти дерево и автомат предназначены для проверки упомянутого выше бинарного отношения \diamond ; точнее, используется другое бинарное отношение \triangleleft , двойное применение которого и дает требуемое отношение \diamond .

В разделе 6 описывается совершенно иной объект (теперь — недетерминированный конечный автомат) — хотя при этом стоит отметить, что построение этого автомата может быть выполнено с использованием тех же самых алгоритмов (и соответствующих компьютерных программ, [5, 6]), что и для предыдущего раздела. Итак, в разделе 6 рассматривается недетерминированный автомат, отвечающий на тот же самый вопрос. Важно отметить, что последний автомат определен на конечном множестве слов — а не на конечном множестве языков (конечных множеств слов).

В разделе 7 и последующих разделах рассматривается специальная гипотеза теории формальных языков, которую автор считает очень важной; по ней и названа вся статья. Очень кратко эту гипотезу можно сформулировать следующим образом: бесконечные итерации конечных языков одинаковы тогда и только тогда, когда эти языки могут быть представлены как один и тот же морфизм расширенных максимальных префиксных кодов, определенных над некоторым вспомогательным алфавитом (едином для обоих языков). Непосредственно в разделе 7 рассматриваются результаты предыдущих публикаций, относящиеся к упомянутой гипотезе в частности и к материалу настоящей статьи вообще. Далее в разделе 8 рассматриваются несколько основных вариантов формулировки этой гипотезы — а именно, такие варианты, которые не использующие автоматы и деревья.

В разделе 9 кратко обсуждаются недетерминированные конечные автоматы особого вида — так называемые лепестковые автоматы (иногда в английской литературе встречается термин “semiflower automata”). Мы приводим результаты одной из предыдущих работ, эти результаты можно сформулировать следующим образом: для любого регулярного языка и его таблицы бинарного отношения $\#$ (подробнее об этом отношении см. раздел 1) существует алгоритм построения лепесткового автомата, имеющего либо ту же самую таблицу, либо ее же с ровно одним добавленным столбцом (т. е. с ровно одним добавленным состоянием канонического автомата для зеркального языка). Конкретно в настоящей статье лепестковые автоматы используются для еще двух переформулировок основной рассматриваемой в статье гипотезы — см. *раздел 10*. Поэтому название раздела 10 таково: «Вспомогательные варианты формулировки гипотезы — с использованием автоматов и деревьев». Для таких вариантов и используются лепестковые конечные автоматы, а также бесконечные итерационные деревья.

Упомянутый выше план сведения равенства $P = NP$ к более простой формулировке приведен в *разделе 11*. Более детальное название этого раздела могло бы быть таким: «О плане доказательства возможности сведения равенства $P = NP$ к гипотезе теории формальных языков».

Последний раздел — заключение; в нем формулируются возможные направления дальнейших работ по тематике настоящей статьи.

1. Предварительные сведения

В этом разделе приводятся основные обозначения и соглашения об их использовании. Следует отметить, что некоторые из этих обозначений являются нестандартными. Одним из наиболее важных понятий, приводимых в этом разделе, является бинарное отношение \trianglelefteq , определенное на множестве конечных языков над рассматриваемым алфавитом, с ним связан весь материал этой статьи. Перейдем к самим обозначениям.

Чаще всего слова и языки будут рассматриваться над главным алфавитом Σ ; вспомогательным же алфавитом обычно будет Δ , иногда с нижними индексами; все алфавиты всегда конечны.

Для заданного слова $u \in \Sigma^*$ и языка $A \subseteq \Sigma^*$:

- язык $\text{pref}(u)$ – это множество префиксов слова u (включая само u);
- $\text{opref}(u) = \text{pref}(u) \setminus \{u\}$;
- $\text{pref}(A) = \bigcup_{u \in A} \text{pref}(u)$;
- $\text{opref}(A)$ определяется аналогично.

Если для двух конечных языков A и B (в предыдущих работах они чаще всего рассматриваются над главным алфавитом Σ) выполняется условие

$$(\forall u \in A^*) (\exists v \in B^*) (u \in \text{opref}(v)),$$

то будем писать $A \trianglelefteq B$ (либо $B \trianglerighteq A$). Если условия $A \trianglelefteq B$ и $A \trianglerighteq B$ выполняются одновременно, то будем писать $A \trianglelefteq B$.

(Специально отметим еще раз разницу в обозначениях: с одной стороны, в недавних работах [1–12], и с другой стороны, в [14] и некоторых других работах 1990-х годов. Конечно, в настоящей статье будут использоваться только обозначения, приведенные в этом разделе.)

Теперь приведем некоторые обозначения, связанные с недетерминированными конечными автоматами. Пусть

$$K = (Q, \Sigma, \delta, S, F) \quad (1)$$

некоторый автомат. δ – его функция переходов вида $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$, где обозначение $\mathcal{P}(Q)$ обозначает надмножество (степень множества, множество подмножеств) множества Q ; таким образом, будут рассматриваться только автоматы без ϵ -переходов. Иногда некоторая дуга графа переходов автомата $\delta(q, a) \ni r$ будет записываться в виде $q \xrightarrow[a]{a} r$, или, если это не вызывает неоднозначности, просто в виде $q \xrightarrow{a} r$.

Зеркальный автомат для автомата (1), т. е. $(Q, \Sigma, \delta^R, F, S)$, где $q' \xrightarrow[\delta^R]{a} q''$ тогда и только тогда, когда $q'' \xrightarrow[\delta]{a} q'$, будем обозначать K^R ; заметим, что K^R всегда определяет зеркальный язык L^R .

Для рассматриваемого языка L его канонический автомат будет обозначаться записью \tilde{L} . Пусть автоматы \tilde{L} и \tilde{L}^R для заданного регулярного языка L таковы:

$$\tilde{L} = (Q_\pi, \Sigma, \delta_\pi, \{s_\pi\}, F_\pi) \quad \text{и} \quad \tilde{L}^R = (Q_\rho, \Sigma, \delta_\rho, \{s_\rho\}, F_\rho).$$

Более того, для языков конечных автоматов не будет допускаться возможность $L = \emptyset$, поэтому оба этих автомата действительно имеют стартовые состояния (т. е. приведенные обозначения не могут содержать противоречия).

Для автомата (1) выходной язык состояния q (он обозначается записью $\mathcal{L}_K^{out}(q)$) — это язык автомата $(Q, \Sigma, \delta, \{q\}, F)$. Аналогично, входной язык состояния q (обозначается $\mathcal{L}_K^{in}(q)$) — это язык автомата $(Q, \Sigma, \delta, S, \{q\})$.

Также напомним определение бинарного отношения $\#$, подробнее см. [19] и др. Отношение $\# \subseteq Q_\pi \times Q_\rho$ определяется для пар состояний автоматов \tilde{L} и \tilde{L}^R следующим образом: $A \# X$ тогда и только тогда, когда

$$(\exists uv \in L) (u \in \mathcal{L}_{\tilde{L}}^{in}(A), v^R \in \mathcal{L}_{\tilde{L}^R}^{in}(X))$$

(в последней записи для каждого из канонических автоматов применено обозначение входного языка состояния). Важно отметить, что именно такой вариант определения неконструктивен; однако, например, [19] содержит эквивалентный конструктивный вариант подобного определения (однако, как обычно в подобных ситуациях, более сложно формулируемый).

Важно также отметить, что детерминированные конечные автоматы всегда рассматриваются как *частный случай недетерминированных автоматов*, определенных в соответствии с (1), — аналогично большинству источников, но в отличие, например, от [20]. В статье не приводятся более подробные определения для детерминированных автоматов, они являются более-менее стандартными; отметим только, что будут специально использоваться так называемые всюду определенные автоматы (“total automata”), т. е. такие, для которых выполняется следующее условие:

$$(\forall q \in Q) (\forall a \in \Sigma) (|\delta(q, a)| = 1).$$

В то же время во многих публикациях подобная такая «полнота» даже рассматривается в качестве обязательного свойства детерминированного автомата — однако в настоящей статье применяется другой подход, т. е. всюду определенные автоматы рассматриваются в качестве собственного подмножества автоматов детерминированных. В настоящей статье будут использоваться и так называемые лепестковые конечные автоматы (“semiflower automata”); подробности см. в разделе 10, где кроме них определены еще и автоматы типа $\mathcal{K}(A)$ для заданного конечного языка $A \subseteq \Sigma^*$. Дополнительная информация о связанных с такими автоматами обозначениях приведена в [21].

Перейдем к специальным обозначениям, связанным с морфизмами языков. Для этого сначала специально отметим, что все рассматриваемые конечные языки не являются пустыми и не содержат пустого слова ε . Для некоторого языка

$$A \subseteq \Sigma^*, \quad A = \{u_1, u_2, \dots, u_n\}$$

(не ограничивая общности можно предполагать, что слова языка каким-то образом упорядочены), мы рассматриваем алфавит $\Delta_A = \{d_1, d_2, \dots, d_n\}$ (если это не вызывает неоднозначности, обычно пишем просто Δ). Для этого алфавита рассматривается морфизм вида $h_A : \Delta_A^* \rightarrow \Sigma^*$, определенный следующим образом:

$$h_A(d_1) = u_1, \quad h_A(d_2) = u_2, \quad \dots, \quad h_A(d_n) = u_n.$$

При этом мы, как обычно, для каждого слова $d_1 d_2 \dots d_n \in \Delta^*$ предполагается, что

$$h(d_1 d_2 \dots d_n) = h(d_1) h(d_2) \dots h(d_n).$$

Однако, как может быть понятно из введения, предмет настоящей статьи связан (частично) с более важной и более сложной проблемой (т.е. более сложной, чем построение морфизма), а именно — с проблемой построения *инверсного* морфизма. Отметим, что в [8–10] приведено только начало рассмотрения этой проблемы.

Сформулируем его определение более подробно. Во-первых, дадим такое естественное определение. Для данного конечного языка A , морфизма h_A и некоторого слова $u \in \Sigma^*$ рассмотрим *язык*

$$h_A^{-1}(u) = \{ u_\Delta \in \Delta^* \mid h_A(u_\Delta) = u \};$$

специально подчеркнем, что этот объект представляет собой множество (а не единственный элемент), причем, возможно, \emptyset . Такую же конструкцию можно рассмотреть для некоторого языка (вместо слова u ; пусть это язык B), нас будут интересовать только конечные языки. A именно,

$$h_A^{-1}(B) = \bigcup_{u \in B} h_A^{-1}(u).$$

Теперь рассмотрим другое определение, практически не связанное с предыдущим. Причем, в отличие от других определений настоящей статьи, алфавит Δ в нем используется в качестве главного. Множество максимальных префиксных кодов (как множество языков) над этим алфавитом будет обозначаться $\text{mp}(\Delta)$.

(Не будем давать подробного определения максимального префиксного кода. Вполне достаточно сказать, что это — код, который является максимальным и префиксным, причем все эти понятия присутствуют в «обычных студенческих курсах», [22] и мн. др. То есть можно сказать, что *само название уже содержит определение.*)

Множество языков, каждый из которых содержит некоторый максимальный префиксный код в качестве подмножества (возможно, несобственного), будет обозначаться записью $\text{mp}^+(\Delta)$. Таким образом, $\text{mp}(\Delta) \subset \text{mp}^+(\Delta)$, и, конечно же, для любого алфавита такое включение является собственным. Мы будем называть каждый из этих языков *расширенным максимальным префиксным кодом*.

Вернемся к рассмотрению главного алфавита Σ , а также морфизмов вида $h_A : \Delta_A^* \rightarrow \Sigma^*$. Пусть у нас будет какой-нибудь язык $A_\Delta \in \text{mp}(\Delta)$; тогда считаем, что выполняется следующее условие: $h_A(A_\Delta) \in \text{mp}(A)$; именно так определяется множество языков $\text{mp}(A)$ над алфавитом Σ . Аналогично, для некоторого языка $A_\Delta \in \text{mp}^+(\Delta)$ считаем, что $h_A(A_\Delta) \in \text{mp}^+(A)$. Следовательно, выполняются такие альтернативные варианты определений:

- $\text{mp}^+(\Delta)$ — это множество языков, каждый из которых содержит некоторый максимальный префиксный код над Δ в качестве подмножества;
- $\text{mp}^+(\Delta)$ — это множество языков, каждый из которых является A -морфизмом некоторого языка множества $\text{mp}^+(\Delta)$; т.е. каждый из таких языков является специальным морфизмом некоторого расширенного максимального префиксного кода.

2. Бесконечные итерационные деревья

В этом разделе рассматриваются *бесконечные деревья специального вида*, возникающие в различных прикладных задачах теории формальных языков. Отличительной особенностью этих деревьев является то, что все их ребра помечены буквами заданного алфавита, вершины разделены на некоторые классы эквивалентности, а бесконечные поддеревья, соответствующие вершинам одного и того же класса, эквивалентны. Число полученных классов эквивалентности обычно (но не всегда) конечно.

ным образом.) См. рис. 1, на приведенном на нем примере разными цветами показаны вершины, принадлежащие разным классам эквивалентности. Отметим также, что условие, сформулированное выше в этом абзаце, эквивалентно другому, более простому: достаточно потребовать, чтобы для каждого ребра, принадлежащего одному и тому же классу эквивалентности, и для всех букв $a \in \Sigma$ ребра, выходящие из этих вершин, также вели к вершинам одного и того же класса эквивалентности (вообще говоря, к вершинам какого-то нового класса).

Заранее отметим, что во всех рассматриваемых примерах количество классов эквивалентности будет конечным. Немного более подробные комментарии по этому поводу будут приведены в следующем разделе. На рис. 1 такие классы эквивалентности расположены более или менее произвольно, хотя важно отметить, что все свойства и условия, сформулированные выше, выполняются.

Далее рассмотрим пример алфавита из 2 букв (будем алфавит обозначать $\{a, b\}$ вместо приведенного на рисунке $\{0, 1\}$). Конкретный пример также может быть получен следующим образом (будем условно называть такие деревья деревьями простейшего уровня). Во-первых, начинаем с дерева, уже рассмотренного на рис. 1, у которого, однако, вершины еще не выделены цветами. Для дальнейших построений задаем язык (пусть это $A = \{a, ab, bb\}$), после чего для этого языка среди вершин дерева отмечаем те, которые соответствуют словам языка-итерации A^* . Более того, на рис. 2 используются следующие цвета для выделения слов:

- красный — для тех слов языка A^* , для которых — если их представить именно в виде итераций слов языка A , — последним словом может быть a ;
- аналогично синий — для возможного последнего слова ab ;
- и зеленый — для возможного последнего слова bb .

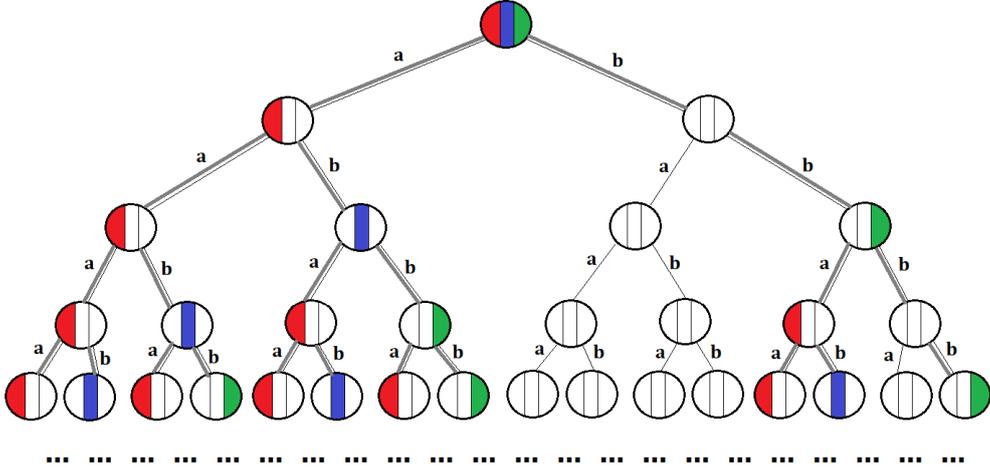


Рис. 2. Пример *конкретного* бесконечного итерационного дерева морфизма для алфавита $\Sigma = \{a, b\}$ и языка $A = \{a, ab, bb\}$ (показаны уровни от 0-го до 4-го).

Отметим, что в нашем примере рассматриваемый язык A является (однозначным) кодом, поэтому все выделенные слова — за исключением пустого слова ϵ — выделены только одним цветом; однако в общем случае такое выделение одного слова только одним цветом, вообще говоря, необязательно.

сложным алгоритмом, который в то же время удовлетворяет приведенному выше описанию-определению. В этом разделе такие деревья рассматриваются очень кратко (более подробно — в следующих разделах), однако при этом приводится начало существенно более сложного примера — который будет далее рассматриваться для нескольких разных целей. Для подобных примеров нам нужны два непустых конечных языка (т. е. упорядоченная пара языков) над заданным алфавитом Σ (а не только один язык, как в примерах, рассматривавшихся ранее); в предыдущих публикациях такие два языка почти всегда обозначались как A (первый язык) и B (второй). Вершины такого дерева соответствуют не словам алфавита Σ (в предыдущих примерах ε соответствует слову 0-го уровня, все 1-буквенные слова соответствуют словам 1-го уровня и так далее), а словам нового, специально введенного алфавита Δ ; значение этого нового алфавита такое же, как и при введении алфавита того же названиее «для организации морфизмов». (В нашей ситуации это A -морфизм; его определение см. в разделе 1, а его использование — далее, например в разделе 9.)

При таком способе определения бесконечного дерева предполагается, что пометкой каждой его вершины является подмножество множества префиксов языка B . О методе выбора такой разметки, а также о разметке ребер этого дерева см. ниже. Отметим лишь, что «выходящие» ребра каждого куста дерева желательно обозначать числами от 0 до 3, и эти ребра можно считать соответствующими A -морфизму из алфавита $\Delta = \{0, 1, 2, 3\}$ в алфавит $\Sigma = \{a, b\}$.

Итак, в этом разделе приводится только начало примера (см. рис. 4); пример полностью (т. е. описание всего бесконечного итерационного дерева, причем не только одно описание, но и подробные варианты применения) будет рассматриваться в дальнейших частях статьи — применительно к нескольким разным дискретным моделям.

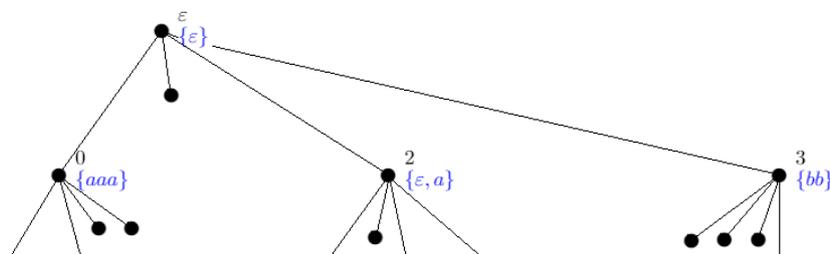


Рис. 4. Начало построения бесконечного итерационного дерева морфизма для пары заданных языков $A = \{aaa, aabba, abba, bb\}$ и $B = \{aaaa, abb, abba, bbb\}$.

Как уже было отмечено, само *применение* бесконечных итерационных деревьев для рассматриваемых в статье задач, а также сведение этих деревьев к конечным автоматам будет рассмотрено в частях II–IV настоящей статьи.

Работа частично поддержана грантом научной программы китайских университетов “Higher Education Stability Support Program” (раздел “Shenzhen 2022 — Science, Technology and Innovation Commission of Shenzhen Municipality”).

Литература

1. Melnikov B.F., Korabelshchikova S.Yu., Dolgov V.N. On the task of extracting the root from the language // International Journal of Open Information Technologies. 2019. Vol. 7, No. 3.

Р. 1–6.

2. Melnikov B.F., Melnikova A.A. Some more on ω -finite automata and ω -regular languages. Part I: The main definitions and properties // International Journal of Open Information Technologies. 2020. Vol. 8, No. 8. P. 1–7.
3. Мельников Б.Ф., Мельникова А.А. Бесконечные деревья в алгоритме проверки условия эквивалентности итераций конечных языков. Часть I // International Journal of Open Information Technologies. 2021. Vol. 9, No. 4. P. 1–11.
4. Мельников Б.Ф., Мельникова А.А. Бесконечные деревья в алгоритме проверки условия эквивалентности итераций конечных языков. Часть II // International Journal of Open Information Technologies. 2021. Vol. 9, No. 5. P. 1–11.
5. Мельников Б.Ф. Варианты конечных автоматов, соответствующих бесконечным итерационным деревьям морфизмов. Часть I // International Journal of Open Information Technologies. 2021. Vol. 9, No. 7. P. 5–13.
6. Мельников Б.Ф. Варианты конечных автоматов, соответствующих бесконечным итерационным деревьям морфизмов. Часть II // International Journal of Open Information Technologies. 2021. Vol. 9, No. 10. P. 1–8.
7. Мельников Б.Ф., Мельникова А.А. Полиномиальный алгоритм построения конечного автомата для проверки равенства бесконечных итераций двух конечных языков // International Journal of Open Information Technologies. 2021. Vol. 9, No. 11. P. 1–10.
8. Мельников Б.Ф. Полурешетки подмножеств потенциальных корней в задачах теории формальных языков. Часть I. Извлечение корня из языка // International Journal of Open Information Technologies. 2022. Vol. 10, No. 4. P. 1–9.
9. Мельников Б.Ф. Полурешетки подмножеств потенциальных корней в задачах теории формальных языков. Часть II. Построение инверсного морфизма // International Journal of Open Information Technologies. 2022. Vol. 10, No. 5. P. 1–8.
10. Мельников Б.Ф. Полурешетки подмножеств потенциальных корней в задачах теории формальных языков. Часть III. Условие существования решетки // International Journal of Open Information Technologies. 2022. Vol. 10, No. 7. P. 1–9.
11. Мельников Б.Ф. Лепестковые конечные автоматы: основные определения, примеры и их связь с полными автоматами. Часть I // International Journal of Open Information Technologies. 2022. Vol. 10, No. 9. P. 1–11.
12. Мельников Б.Ф. Лепестковые конечные автоматы: основные определения, примеры и их связь с полными автоматами. Часть II // International Journal of Open Information Technologies. 2022. Vol. 10, No. 10. P. 1–10.
13. Мельников Б.Ф. Регулярные языки и недетерминированные конечные автоматы (монография). М.: Изд-во Российского государственного социального университета, 2018. 179 с.
14. Melnikov B.F. The equality condition for infinite catenations of two sets of finite words // International Journal of Foundations of Computer Science. 1993. Vol. 4, No. 3. P. 267–273.
15. Мельников Б.Ф. Описание специальных подмоноидов глобального надмоноида свободного моноида // Известия высших учебных заведений. Математика. 2004. № 3. С. 46–56.

16. Алексеева А.Г., Мельников Б.Ф. Итерации конечных и бесконечных языков и недетерминированные конечные автоматы // Вектор науки Тольяттинского государственного университета. 2011. № 3 (17). С. 30–33.
17. Hopcroft R., Motwani R., Ullman J. Introduction to Automata Theory, Languages, and Computation. Massachusetts: Addison-Wesley Publishing Company Reading, 2006. 530 p.
18. Hromkovič J. Theoretical Computer Science. Introduction to Automata, Computability, Complexity, Algorithmics, Randomization, Communication, and Cryptography. Berlin: Springer, 2003. 323 p.
19. Melnikov B.F. Once more on the edge-minimization of nondeterministic finite automata and the connected problems // Fundamenta Informaticae. 2010. Vol. 104, No. 3. P. 267–283.
20. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. Том 1. М.: Мир, 1978. 617 с.
21. Melnikov B.F. The complete finite automaton // International Journal of Open Information Technologies. 2017. Vol. 5, No. 10. P. 9–17.
22. Яблонский С.В. Введение в дискретную математику. М.: Высшая школа, 2003. 384 с.

Мельников Борис Феликсович, д.ф.-м.н., профессор, факультет Вычислительной математики и кибернетики, Совместный университет МГУ–ППИ в Шэньчжэне (Шэньчжэнь, Китай)

DOI: 10.14529/cmse230X0Z

ON A HYPOTHESIS OF THE THEORY OF FORMAL LANGUAGES. PART I

© 2023 B.F. Melnikov¹

¹ *Shenzhen MSU–BIT University (No. 1, International University Park Road, Dayun New Town, Longgang District, Shenzhen, Guangdong Province, 518172, China)*

E-mail: bormel@mail.ru

Received: 20.06.2023

The main subject of the article is the consideration of problems arising in the study of the necessary conditions for the equality of infinite iterations of finite languages. In previous publications, the author considered examples of the application of a special binary equivalence relation corresponding to this equality in a set of finite languages, and considered both examples describing the necessary conditions for its implementation and examples of its use. Further reducing the problem under consideration is applied to one of such necessary conditions: to finite automata and to infinite iterative trees. In addition, the article presents several variants of the formulation of an important hypothesis on the set of finite languages, its study also provides other options for reducing the problem under consideration to special problems of studying nondeterministic finite automata. At the same time, if the formulated hypothesis is fulfilled, some of these problems are solved in polynomial time, and some are not; with the continuation of work on this topic, the last fact may make it possible to reformulate the problem $P=NP$ in the form of a special problem of the theory of formal languages.

Keywords: formal languages, iterations of languages, binary relations, morphisms, inverse morphisms, algorithms, polynomial algorithms.

FOR CITATION

Melnikov B.F. On a hypothesis of the theory of formal languages. Part I. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. X, no. Y. P. Z1–Z2. (in Russian) DOI: 10.14529/cmse230X0Z.

This paper is distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Melnikov B.F., Korabelshchikova S.Yu., Dolgov V.N. On the task of extracting the root from the language // International Journal of Open Information Technologies. 2019. Vol. 7, No. 3. P. 1–6.
2. Melnikov B.F., Melnikova A.A. Some more on ω -finite automata and ω -regular languages. Part I: The main definitions and properties // International Journal of Open Information Technologies. 2020. Vol. 8, No. 8. P. 1–7.
3. Melnikov B.F., Melnikova A.A. Infinite trees in the algorithm for checking the equivalence condition of iterations of finite languages. Part I // International Journal of Open Information Technologies. 2021. Vol. 9, No. 4. P. 1–11. (in Russian).
4. Melnikov B.F., Melnikova A.A. Infinite trees in the algorithm for checking the equivalence condition of iterations of finite languages. Part II // International Journal of Open Information Technologies. 2021. Vol. 9, No. 5. P. 1–11. (in Russian).
5. Melnikov B.F. Variants of finite automata corresponding to infinite iterative morphism trees. Part I // International Journal of Open Information Technologies. 2021. Vol. 9, No. 7. P. 5–13. (in Russian).
6. Melnikov B.F. Variants of finite automata corresponding to infinite iterative morphism trees. Part II // International Journal of Open Information Technologies. 2021. Vol. 9, No. 10. P. 1–8. (in Russian).
7. Melnikov B.F., Melnikova A.A. A polynomial algorithm for constructing a finite automaton to check the equality of infinite iterations of two finite languages // International Journal of Open Information Technologies. 2021. Vol. 9, No. 11. P. 1–10. (in Russian).
8. Melnikov B.F. Semi-lattices of the subsets of potential roots in the problems of the formal languages theory. Part I. Extracting the root from the language // International Journal of Open Information Technologies. 2022. Vol. 10, No. 4. P. 1–9. (in Russian).
9. Melnikov B.F. Semi-lattices of the subsets of potential roots in the problems of the formal languages theory. Part II. Constructing an inverse morphism // International Journal of Open Information Technologies. 2022. Vol. 10, No. 5. P. 1–8. (in Russian).
10. Melnikov B.F. Semi-lattices of the subsets of potential roots in the problems of the formal languages theory. Part III. The condition for the existence of a lattice // International Journal of Open Information Technologies. 2022. Vol. 10, No. 7. P. 1–9. (in Russian).
11. Melnikov B.F. Petal finite automata: basic definitions, examples and their relation to complete automata. Part I // International Journal of Open Information Technologies. 2022. Vol. 10, No. 9. P. 1–11. (in Russian).

12. Melnikov B.F. Petal finite automata: basic definitions, examples and their relation to complete automata. Part II // International Journal of Open Information Technologies. 2022. Vol. 10, No. 10. P. 1–10. (in Russian).
13. Melnikov B.F. Regular languages and nondeterministic finite automata (monograph). Moscow: Russian State Social University Ed., 2018. (in Russian). 179 p.
14. Melnikov B.F. The equality condition for infinite catenations of two sets of finite words // International Journal of Foundations of Computer Science. 1993. Vol. 4, No. 3. P. 267–273. (in Russian).
15. Melnikov B.F. Description of special submonoids of the global supermonoid of the free monoid // News of higher educational institutions. Mathematics. 2004. No. 3. P. 46–56. (in Russian).
16. Alekseeva A.G., Melnikov B.F. Iterations of finite and infinite languages and nondeterministic finite automata // Vektor nauki of Togliatti State University. 2011. No. 3 (17). P. 30–33. (in Russian).
17. Hopcroft R., Motwani R., Ullman J. Introduction to Automata Theory, Languages, and Computation. Massachusetts: Addison-Wesley Publishing Company Reading, 2006. 530 p.
18. Hromkovič J. Theoretical Computer Science. Introduction to Automata, Computability, Complexity, Algorithmics, Randomization, Communication, and Cryptography. Berlin: Springer, 2003. 323 p.
19. Melnikov B.F. Once more on the edge-minimization of nondeterministic finite automata and the connected problems // Fundamenta Informaticae. 2010. Vol. 104, No. 3. P. 267–283.
20. Aho A., Ullman J. The Theory of Parsing, Translation, and Compiling. Vol. 1: Parsing NJ: Prentice Hall, 1972. 560 p.
21. Melnikov B.F. The complete finite automaton // International Journal of Open Information Technologies. 2017. Vol. 5, No. 10. P. 9–17.
22. Yablonskiy S.V. Introduction to Discrete Mathematics. Moscow: Vysshaya shkola, 2003. 384 p. (in Russian).