# О группоиде первичного автомата PRI, определяемом для одного конечного языка

Мэн Линцянь, Б. Ф. Мельников

***Аннотация*—В настоящей работе мы продолжаем тематику наших предыдущих статей, посвящённых важному отношению эквивалентности на множестве формальных языков – т. н. отношению эквивалентности в бесконечности; для его подробного изучения мы также рассматривали бинарное отношение покрытия. При исследовании этих отношений мы применяли бесконечные итерационные деревья морфизмов, для каждого из которых после объединения эквивалентных вершин получается детерминированный конечный автомат.**

**Среди нескольких похожих вариантов автоматов, соответствующих бесконечному итерационному дереву морфизма, наибольший интерес вызывает т. н. первичный автомат (автомат PRI), специальным образом определяемый для заданной пары конечных языков. Он задаётся на подмножествах множества префиксов одного из этих языков – и эти подмножества мы рассматриваем как состояния определяемого автомата.**

**В настоящей статье мы продолжаем рассмотрение первичного автомата, построенного для любой пары конечных языков, – но определяем ещё и новые объекты: автомат, построенный только для одного конечного языка, а также соответствующие этому автомату группоид и полугруппу преобразований.**

**Опишем основной предмет настоящей статьи. Если структуры, возникающие в автомате PRI, «объединить построчно», то мы получим результат перехода каждого состояния – в случае, когда входом является некоторый язык. Вследствие этого можем изучать первичный автомат, построенный на основе только одного языка – и этот язык можно называть базовым. И для множества состояний, и для входного алфавита мы будем использовать подмножество множества собственных префиксов базового языка – либо полученный с помощью некоторого инверсного морфизма его прообраз.**

**В статье рассмотрен получающийся при этом группоид и доказана теорема, что при этом можно определять полугруппу преобразований. Основная идея доказательства последнего факта такова. Если полугрупповое требование рассматривать с точки зрения конечного автомата, то про две части используемого в нём уравнения можно сказать следующее. Левая часть уравнения этого полугруппового требования описывает переход из текущего состояния в соответствии с первым подсловом (префиксом) входа в некоторое промежуточное состояние, а затем продолжение работы в соответствии со вторым подсловом (суффиксом) входа; то же самое мы можем сказать и для правой части уравнения – а строку-вход в обоих случаях можно считать одной и той же.**

**В настоящей статье также рассмотрены некоторые алгебраические свойства описанной полугруппы, в том числе приведены простые примеры. Более сложные примеры мы предполагаем рассмотреть в статье-продолжении.**

***Ключевые слова*—формальные языки, итерации языков, морфизмы, детерминированные конечные автоматы, группоиды, полугруппы, алгоритмы.**

Статья получена 12 мая 2023 г.

Мэн Линцянь, Университет МГУ – ППИ в Шэньчжэне (kostya45e@mail.ru).

Борис Феликсович Мельников, Университет МГУ – ППИ в Шэньчжэне (bormel@smbu.edu.cn).

## I. Введение и мотивация

В настоящей статье мы продолжаем тематику статей [1], [2], [3], [4] и др. В этих статьях рассматривали важное отношение эквивалентности на множестве формальных языков – отношение эквивалентности в бесконечности, для подробного изучения которого мы рассматривали также бинарное отношение покрытия. При исследовании этих отношений мы применяли бесконечные итерационные деревья морфизмов, в каждом из которых после объединения эквивалентных вершин получается детерминированный конечный автомат.

При этом среди нескольких похожих вариантов автоматов, соответствующих бесконечному итерационному дереву морфизма, наибольший интерес вызывает т. н. первичный автомат (автомат PRI), специальным образом определяемый *для заданной пары конечных языков*. Он определяется на подмножествах множества префиксов одного из этих языков – и эти подмножества мы рассматриваем как состояния определяемого автомата (отождествляем с ними).

В настоящей статье мы продолжаем рассмотрение первичного автомата, построенного для любой пары конечных языков, – но определяем ещё и новые объекты:

- автомат, построенный для *одного* конечного языка;
- соответствующий этому автомату группоид;
- также соответствующую этому автомату полугруппу преобразований [1].

Итак, мы выше кратко сформулировали *первую часть мотивации* к рассмотрению первичного автомата PRI и некоторых других объектов, которые строятся на его основе. Но независимо от неё можно назвать и *вторую часть мотивации*, практически не связанную с первой, – переходим к её описанию.

По нашему мнению, в литературе, посвящённой полугруппам и конечным автоматам (при этом автомат часто рассматривается без входов и без выходов, аналогично, например, [5]), а ещё точнее – в литературе, посвящённой связи этих объектов, давно утвердилось мнение о том, что такая связь уже показана и вряд ли заслуживает дополнительных рассмотрений. Однако возникает много вопросов – но сначала кратко опишем саму возможную связь.

Любой конечный автомат, как мы уже сказали, рассматривается со следующими упрощениями:

[1] Соответственно, немного прокомментируем название статьи. Употребляя слово «одного», мы имеем в виду не «какого-либо», а то, что строим автомат и полугруппу *по одному* языку (а не по двум, как в предыдущих статьях).

- как детерминированный,
- без входов и без выходов;

то есть в стандартных обозначениях теории формальных языков его можно считать заданным в виде как тройки

$$(Q, \Sigma, \delta),$$

в которой функцию переходов $\delta$ достаточно определять как

$$\delta : Q \times \Sigma \to Q.$$

При этом для перехода от автомата к полугруппе необходимо «уровнять» число состояний в множестве $Q$ и число букв в алфавите $\Sigma$ – и для этого применяется[2] добавление:

- либо неиспользуемых состояний в множество $Q$,
- либо неиспользуемых букв в алфавит $\Sigma$.

После этого получается структура

$$(Q, Q, \delta), \quad \text{где} \quad \delta : Q \times Q \to Q \qquad -$$

которую *при дополнительном требовании на выполнение условия ассоциативности операции* $\delta$ можно назвать полугруппой: в стандартных алгебраических обозначениях множеством является $Q$, а операцией – $\delta$.

Но ассоциативность может и *не выполняться* – и пример такого невыполнения (т. е. когда мы не можем простейшим способом построить по автомату полугруппу) приведён на следующем рисунке.

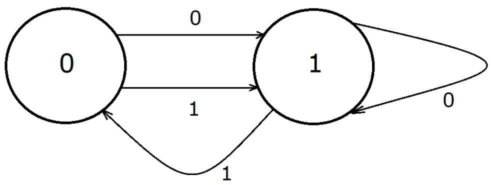


Рис. 1. Пример автомата (функции переходов $\delta$) без входов и выходов, соответствующего неассоциативной операции.

Действительно, получающийся группоид полугруппой не является – и чтобы показать последнее, достаточно привести хотя бы одну упорядоченную тройку

$$(a, b, c), \quad \text{где} \quad a, b, c \in Q = \{0, 1\},$$

такую что

$$(a\,\delta\,b)\,\delta\,c \neq a\,\delta\,(b\,\delta\,c).$$

Несложно убедиться, что в качестве такой упорядоченной тройки можно взять

$$a = 1, \quad b = 1, \quad c = 0,$$

поскольку

$$(1\,\delta\,1)\,\delta\,0 = 1, \quad \text{но} \quad 1\,\delta\,(1\,\delta\,0) = 0.$$

Обратный же переход – от полугруппы к автомату – очевиден: детерминированный автомат может быть очевидным образом построен *по любой* таблице группоида, и требование ассоциативности[3] при этом не является обязательным.

Однако перейдём к вопросам (и замечаниям), возникающим для подобной мини-теории о связи полугрупп и детерминированных конечных автоматов без входов и выходов – и мы уже начинали говорить об этих вопросах / замечаниях ранее.

- Во-первых, нам, вообще говоря, приходится добавлять в автомат состояния либо буквы – но ведь это меняет сам автомат!
- Во-вторых, мы как-то выбираем соответствие букв и состояний – но почему выбираем *именно такое* соответствие? Ведь даже при равном числе букв и числе состояний исходного автомата количество таких возможных соответствий, вообще говоря, равно $n!$.
- В-третьих, что имеется общего (у автоматов, у языков) для каждого из этих $n!$ соответствий?

Ни на один из этих вопросов [6] не отвечает – более того, ни в каких других известных авторам монографиях ([7], [8], [9], [10], [11] и др.) мы не нашли даже постановки подобных вопросов.

С точки зрения морфизмов первый из языков можно рассматривать в качестве входного алфавита. А исследованный в [1], [2], [3], [4] (детерминированный) первичный автомат $\mathrm{PRI}(A, B)$, множеством состояний которого является система подмножеств множества собственных префиксов второго языка (языка $B$), можно рассматривать в качестве *инициального автомата*, входным состоянием которого является язык-состояние, состоящий из единственного слова – пустого слова $\varepsilon$, т. е. язык $\{\varepsilon\}$.

В нашей работе мы будем чаще всего использовать табличное промежуточное представление. В подобных таблицах:

- каждый столбец соответствует входной букве[4], морфический образ которой является словом в первом языке;
- каждая строка соответствует состоянию, которое является подмножеством множества собственных префиксов второго языка;
- а в каждой клетке таблицы – состояние после выполненного соответствующего перехода.

При этом мы можем использовать для состояний морфизмо-подобное отображение – а именно, вместо самих состояний использовать их индексы.

Всё описанное выше являлось предметом предыдущих публикаций[5] – а теперь опишем предмет настоящей статьи (сначала кратко). Если вышеуказанные структуры «объединить построчно», то мы получим результат перехода каждого состояния – в случае, когда *входным сигналом* является некоторое *множество* слов, т. е. некоторый язык. Этот язык мы будем кратко обозначать в виде *некоторой новой буквы* – которая и является прообразом при рассматриваемом морфизме.

Вследствие описанного «построчного объединения» мы можем изучать первичный автомат, построенный на основе *одного* языка – и этот язык можно называть *базовым*. У него и для множества состояний, и для входного алфавита мы будем использовать подмножество множества собственных префиксов базового языка – либо полученный с помощью некоторого инверсного морфизма

[2] По-видимому, всегда, т. е. во всех источниках.

[3] То есть тот возможный факт, что группоид является полугруппой.

[4] Заанее отметим, что при рассмотрении таблицы группоида (полугруппы) мы будем использовать другие соглашения: каждый столбец будет соответствовать некоторому языку. Ср. со следующим пунктом.

[5] Однако важно отметить, что в настоящей статье употреблён совершенно новый вариант описания проделанных ранее действий. Этот вариант можно назвать «алгебраическим».

его прообраз. Вследствие этого сразу видно, что в качестве применяемого преобразования можно рассматривать используемую в автомате PRI индексацию.

Однако подобный переход – совсем не тривиален: кроме рассматриваемых автоматов мы одновременно рассматриваем соответствующие ему группоид и специальную полугруппу, вследствие чего в статье применяются некоторые вопросы, характерные для теории полугрупп. А именно, мы, используя конечность рассматриваемых автоматов [6], а также естественную связь между конечной полугруппой и конечным автоматом, можем доказать корректность проводимых нами действий – то есть тот факт, что в результате рассмотрения множества преобразований получается ассоциативная операция и, следовательно, полугруппа. После этого мы получаем возможность непосредственно провести исследование свойств и основных элементов такой полугруппы – после чего можно, например, вернуться к рассмотрению первичного автомата PRI и продолжить исследование его [7].

Основная идея доказательства того факта, что для множества преобразований получается *именно полугруппа* (а не какой-то группоид), заключается в следующем.

Если мы полугрупповое требование

$$\big(\forall a, b, c \in Q\big) \ \big((a\,\delta\,b)\,\delta\,c = a\,\delta\,(b\,\delta\,c)\big)$$

рассмотрим с точки зрения конечного автомата, то про две части используемого в нём уравнения можно сказать следующее.

- Во-первых, используемые элементы полугруппы $a$, $b$ и $c$ можно, не ограничивая общности рассуждений, рассматривать как результат работы *нескольких* операций автомата [8].
- Во-вторых, левая часть уравнения описывает переход из текущего состояния в соответствии с первым подсловом (префиксом) входа в некоторое промежуточное состояние, а затем продолжение работы в соответствии со вторым подсловом (суффиксом) входа; а «опускаемая нами» часть уравнения (относящаяся к префиксу плюс суффиксу вместе) заключается в том, что мы непосредственно принимаем всю строку-вход, описывая всю работу автомата, т. е. весь его переход из начального состояния в конечное – игнорируя, какие состояния были пройдены «посередине».

  То же самое мы можем сказать и для правой части уравнения – а строку-вход в обоих случаях можно считать одной и той же.

(Таким образом, мы здесь очень кратко изложили, почему в нашем случае получается именно полугруппа – более подробно далее.)

Приведём содержание статьи по разделам.

*В разделе II* («Предварительные све́дения …») приведены некоторые основные понятия, связанные со следующими областями: формальными языками, конечными автоматами и полугруппами. К сожалению, некоторые термины и обозначения этих предметных областей конфликтуют между собой – и мы постарались по возможности рассматривать «неконфликтующую» систему обозначений.

*В разделе III* мы продолжаем рассматривать предварительные све́дения и приводим те основные понятия, которые связаны с ранее определённым автоматом $\mathrm{PRI}(A, B)$; специально отметим, что определяется такой автомат для заданной *пары* конечных языков. Но, кроме «старых» основных понятий, мы в этом же разделе вводим определения основного объекта настоящей статьи – полугруппы, специальным образом построенной для *одного* заданного конечного языка [9].

*В разделе IV* доказана теорема, отражающее основное свойство рассматриваемого объекта – то, что это действительно полугруппа. Эта теорема связывает первичные автоматы, полученные из бесконечного итерационного дерева морфизма, определённого и исследованного в [1], [2] и др., – с соответствующей полугруппой преобразований, введённой в разделе III.

*В разделе V* рассмотрены основные алгебраические свойства определённой здесь полугруппы автомата PRI. *А в разделе VI* рассмотрены несложные пример полугрупп автомата PRI.

В статье-продолжении мы предполагаем рассмотреть:

- существенно более сложный пример полугруппы автомата PRI;
- текст компьютерной программы, необходимой для построения полугруппы автомата PRI по заданному конечному языку $B$;
- текст программы для исследования свойств этой полугруппы;
- некоторые полученные результаты работы этих программ –

и об этом кратко сказано в заключении настоящей статьи (*раздел VII*).

## II. ПРЕДВАРИТЕЛЬНЫЕ СВЕ́ДЕНИЯ: ОПРЕДЕЛЕНИЯ И ОБОЗНАЧЕНИЯ ДЛЯ ПОЛУГРУПП

В настоящем разделе приведены некоторые основные понятия, связанные со следующими областями:

- формальными языками;
- конечными автоматами;
- полугруппами.

К сожалению, некоторые термины и обозначения этих предметных областей конфликтуют между собой – однако в настоящей работе таких ситуаций мало (и мы будем их отмечать). Также отметим, что для первых двух предметных областей в предыдущих работах ([1], [2], [3], [4] и др.) были введены соответствующие термины и обозначения – поэтому здесь эту информацию мы повторять не будем. Отметим только некоторые обозначения, не указанные в упомянутых работах.

Свободный язык $A^+$ – это множеством всех возможных последовательностей слов языка $A$. При этом среди «алгебраических» обозначений имеются соответствующие похожие понятия: над алфавитом $\Sigma$ рассматриваются свободная полугруппа $\Sigma^+$ и свободный моноид $(\Sigma^+)^1$

[6] То есть не бесконечное число их элементов: все элементы в пятёрке определения автомата являются конечными множествами.

[7] Дополнительно пользуясь полученными фактами о рассмотренной полугруппе преобразований.

[8] Для полугруппы без единицы – не менее чем 1 операции. Но если мы переходим к моноиду (об этом ниже) – то можно говорить и о 0 операций.

[9] Условно – для объекта, обозначаемому нами в других формулах, прежде всего в предыдущих статьях на эту тему, как язык $B$.

([6, с. 28]). В настоящей работе мы постоянно используем эти два понятия, тесно связаны с рассматриваемой нами темой, – поэтому будем называть их просто полугруппой и моноидом.

Произведением (или прямым произведением, или конкатенацией) двух языков $A \cdot B$ является язык

$$\mathfrak{C} = \{u \cdot v \mid u \in A,\ v \in B\},$$

т. е. все возможные конкатенации слов, принадлежащих этим двух языкам. $\mathcal{P}(A)$ представляет собой систему (множество) подмножеств множества (языка) $A$.

Теперь приведём некоторые основные понятия, относящиеся к полугруппам. Большинство из этих понятий являются стандартными – но важно определить конкретные обозначения, которые мы будем использовать в дальнейшем.

*Полугруппой* является структура $(S, \cdot)$, для которой выполнено следующее:

- $S$ – некоторое множество;
- на этом множестве $S$ определена бинарная операция $\cdot$, которая:
  - замкнута относительно множества $S$, т. е.
    $$(\forall\, a, b \in S)\ (a \cdot b \in S).$$
  - обладает свойством ассоциативности, т. е.
    $$(\forall\, a, b, c \in S)\ ((a \cdot b) \cdot c = a \cdot (b \cdot c)).$$

Обычно будем писать кратко «полугруппа $S$», что не будет вызывать путаницы (поскольку всегда мы будем рассматривать только одну полугрупповую операцию). Эта операция часто называется «умножением», но надо отметить, что в приведённом определении нет требования коммутативности. Число элементов множества $S$ может быть бесконечным. Кроме того, обычно мы не будем различать полугруппу и соответствующий ей моноид.

В последних определениях формально участвуют всевозможные упорядоченные тройки элементов (то есть для конечной полугруппы, содержащей $N$ элементов, чисто формально в худшем случае надо проверить $N^3$ равенств, и, следовательно, выполнить $4\,N^3$ умножений) – но сформулированное условие можно в определённом смысле «ослабить», для чего сначала нужно ввести понятие подполугруппы; мы продолжим обсуждение этих вопросов в разделе V.

*Подполугруппой* полугруппы $S$ называется структура $A$ – или, точнее, $(A, \cdot)$ – где $A \subseteq S$ и

$$(\forall x, y \in A)\,(x \cdot y \in A)$$

(равенство $x = y$ допускается). Очевидно, что подполугруппа также является полугруппой, так как условие ассоциативности не перестаёт быть истинным. Отметим, что ответ на вопрос о понимании ассоциативности над множеством, имеющим 2 элемента (или даже 1) на самом деле прост. Более того, в некоторых монографиях допускается даже возможность $S = \varnothing$.

Пока мы рассматривали основные понятия теории полугрупп, а также *возможное* получение конкретной интересующей нас полугруппы как группоида переходов автомата. Теперь введём понятия полугруппы преобразований – которые несколько сложнее, чем указанные выше основные понятия, и предоставляет собой более сложный способ построения *связанной с заданным автоматом* полугруппы. Однако с применением такого метода полугруппа на основе автомата PRI получается всегда.

Из предыдущего следует, что следующие два способа задания функции переходов автомата фактически описывают одно и то же:

- в первом способе мы рассматриваем множество переходов из заданного состояния по заданной букве;
- а в другом способе мы рассматриваем множество букв, соответствующих переходам между двумя заданными состояниями.

В класической теории полугрупп считается, что понятие полугруппы неразрывно связано с понятием детерминированного конечного автомата ([6, с. 21 и с. 174], [7, с. 28] и мн. др.). Действительно, для автомата

$$\mathfrak{A} = (Q, \Delta, \delta, S, F)$$

любую входную букву $a \in \Delta$ можно считать соответствующей некоторому преобразованию (функции)

$$\phi_a \in \mathcal{I}_r(Q) : Q \to Q\,;$$

последнее отображает текущее состояние в некоторое новое состояние, обозначаемое иногда $q\phi_a = \delta(q, a)$. При этом любая конкатенация букв – т. е. любая последовательность

$$a_1 \cdot a_2 \cdots a_n\,, \quad \text{где} \quad (\forall\, i \in \overline{1,n})\,(a_i \in \Delta)\ \ -$$

представляет собой суперпозицию подобных функций

$$\phi_{a_1} \circ \phi_{a_2} \circ \cdots \circ \phi_{a_n}.$$

Поэтому если мы будем рассматривать заданный входной алфавит как множество преобразований

$$\Delta \to \mathcal{I}_r(Q),$$

а соединение букв-входов – как вышеуказанную суперпозицию функций, то, согласно свойству свободного моноида, такое отображение может быть продолжено до гомоморфизма

$$\tau_{\mathfrak{A}} : \Delta^* \to \mathcal{I}_r(Q).$$

При этом мы получаем соответствующую автомату полугруппу преобразований или, согласно другой терминологии, просто *полугруппу автомата* $\mathfrak{A}$.

При наличии любого моноида $M$ можно построить инициальный автомат

$$\mathfrak{A} = (M, M, \delta, S, F),$$

для которого *множество элементов моноида и входной алфавит совпадают*. Также отметим, что в литературе [10] в подобных формулах употребляются точки – это означает любой возможный вариант (состояния, буквы и пр.).

Чисто формально рассматриваемый тип полугруппы описывается как

$$(\mathcal{I}_r(Q), \circ),$$

(т. е. как некоторый *синтаксический моноид* [6, с. 175] – или, другими словами, как автомат без входных и выходных состоянй с «единым» множеством для букв и состояний), где:

[10] Прежде всего – в «алгебраических» источниках.

- отображение $\tau_{\mathfrak{A}} : \Delta^* \to \mathfrak{I}_r(Q)$ задаётся;
- элемент полугруппы – образ входного символа для заданного отображения, т. е.
$$(\forall a \in \Delta)\ (\exists \phi_a \in \mathfrak{I}_r(Q))\ (a \xrightarrow{\tau_{\mathfrak{A}}} \phi_a)\,;$$
- $\circ$ – суперпозиция функций.

Таким образом, мы продолжаем полугруппу до моноида с помощью добавления тождественного отображения – которое можно считать соответствующим пустой последовательности входных сигналов. По отношению к автомату пустый вход представляет собой тривиальную ситуацию – следовательно можно считать, что основным объектом рассмотрения является полугруппа, причём *над множеством преобразований*, [6, с. 22 и с. 177].

Таким образом, с учётом естественных определений автомата над полугруппой преобразований, выполняются следующие соотношения:
$$(\forall q \in Q)\,(\forall a \in \Delta)\,(a \xrightarrow{\tau_{\mathfrak{A}}} \phi_a, q\phi_a \in Q),$$
а также
$$(\forall q \in Q)\ \ (\forall x, y \in \Delta :\ x \xrightarrow{\tau_{\mathfrak{A}}} \phi_x, y \xrightarrow{\tau_{\mathfrak{A}}} \phi_y)$$
$$((q\phi_x)\phi_y = q(\phi_x \circ \phi_y)).$$

Некоторые специальные свойства полугрупп мы опишем далее, в разделе V: для полного понимания возможности применения этих свойств к полугруппам, связанным с рассматриваемыми нами конечными автоматами, желательно подробно описать сами эти полугруппы – что мы и сделаем в следующих двух разделах.

## III. АВТОМАТ PRI И ПОСТРОЕННЫЕ НА ЕГО ОСНОВЕ ГРУППОИД И ПОЛУГРУППА

В этом разделе мы приводим те основные понятия, которые связаны с ранее определённым автоматом $\mathrm{PRI}(A, B)$; специально отметим, что определяется такой автомат для заданной *пары* конечных языков. Но, кроме «старых» основных понятий, мы здесь же вводим определения основных объектов настоящей статьи – группоида и полугруппы преобразований, специальным образом построенным для *одного* заданного конечного языка (условно – для объекта, обозначаемому нами в других формулах, прежде всего в предыдущих статьях на эту тему, как язык $B$).

Итак, после введения общетеоретических обозначений, кратко описанных в предыдущем разделе, мы можем описать автомат PRI, соответствующий автомату – основному объекту исследований в [1, разд. IV]. Мы обычно называем его первичным (primary) детерминированным конечным автоматом для бесконечных итерационных деревьев морфизмов (а кратко -–просто первичным автоматом).

Прежде всего, рассмотрим наиболее важную функцию, которую, *среди прочих возможных определений*, можно рассматривать как функцию переходов автомата PRI. Поскольку алгоритм вычисления каждого этой функции [11] был подробно описан в одной из предыдущих статей [1, разд. III], мы просто введем её формальное определение, а затем кратко рассмотрим некоторые её свойства и характеристики, а также её работу (т. е. применение в различных прикладных задачах).

Итак, из предыдущего следует, что $\underset{A-B}{\Phi}(u)$ является языком – т. е. конечным подмножеством полного множества $\Sigma^*$; конкретнее,
$$\underset{A-B}{\Phi}(u) = \big\{\, z \in \mathrm{opref}(B) \ \big|$$
$$(\exists \tilde{b} \in B^+,\ w \in A)\ (wu = w\bar{u}z,\ \tilde{b} = w\bar{u})\big\}.$$

Всё это естественным образом распространяется и на языки – то есть функцию $\Phi$ можно определить от трёх аргументов-языков; записывать обычно удобно следующим образом:
$$\underset{A-B}{\Phi}(C) = \bigcup_{u \in C} \underset{A-B}{\Phi}(u),$$
но иногда удобнее применять «функциональную» форму записи:
$$\underset{A-B}{\Phi}(C) = \Phi_B(C, A).$$

Можно заметить, что последнее обозначение может быть непосредственно использовано для записи операции при рассмотрении группоида (в частных случаях – полугруппы), поскольку оно «принимает на вход» два языка и «возвращает на выходе» один язык – что и является первым требованием для полугрупповой операции.

Комментарии к алгоритму построения функции $\Phi$ *в виде рисунков* были приведены в [12] – здесь мы приведём несколько иные рисунки-комментарии; в отличие от предыдущего варианта, мы здесь:

- работаем непосредственно с языками – а не со словами;
- рассматриваем конкретные значения объектов (языков, участвующих в формулах).

Использование на рисунке трёх цветов показывает слова в первом, втором и третьем языках соответственно – причём «основным» из них стóит считать язык B: это соответствует тому факту, что мы иногда называли его базовым. Более того, сами *действия*, связаные с этим базовым языком, мы также всюду помечаем синим цветом. Итак:

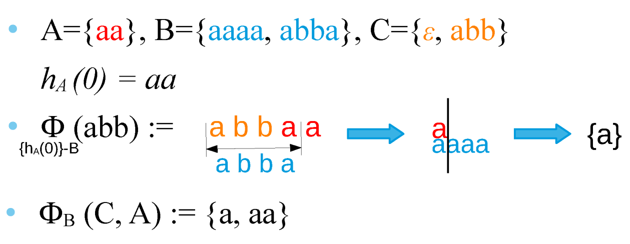


Рис. 2. Пример формирования одного значения функции $\Phi$ – как функции переходов.

Здесь для конкатенации двух языков мы *отрезаем* от него элементы базового языка – но не удаляем слова, а *отсекаем* эти части в элементах, а затем формируем множество подходящих оставшихся частей. Таким образом, мы привели объяснение функции перехода, которое мы можем получить *с точки зрения процесса её работы*.

[11] Отметим, что этот алгоритм не является тривиальным. В частности, требуется особое вспомогательное доказательства того факта, что *каждое значение* этой функции может быть вычислено за *полиномиальное время*. Обсуждать вопрос о возможном вычислении за полиномиальное время *всех* значений этой функции мы пока не будем.

Получение группоида на основе функции $\Phi$ очевидно – и мы теперь подробно опишем, как для рассматриваемого автомата PRI построить полугруппу. Однако поясним ещё раз, что речь идёт не о полугруппе переходов автомата – а о том, *как преобразовать один автомат или класс автоматов в соответствующую полугрупповую структуру*.

Структура множества автоматов с операциями изучена почти полностью, и в соответствующей теории в основном рассматриваются операции над множествами языков, соответствующих автоматам, – то есть можно сказать, что основным объектом исследования при этом являются языки и соответствующие им грамматики. В нашем же случае некоторая новизна заключается в том, что мы рассматриваем полугруппу для автомата, используя множества состояний, входов и выходов, а также, конечно, функцию переходов.

С точки зрения общей алгебры подобных структур может быть много. В предыдущем разделе мы представили возможный способ преобразования автомата в полугруппу, содержащий несколько вспомогательных понятий, при помощи понятия гомоморфизма полугруппы преобразований, зависящих от входов. Применяя такой способ, мы можем получить достаточно элегантную структуру, можно также перейти от полугруппы к моноиду – хотя при этом надо согласиться с возможным утверждением об излишней простоте этого способа.

Отметим, что из изложенного выше ещё не следует конечность получаемого группоида [12] – но её (конечность) можно считать следствием того, что каждый элемент этого группоида представляет собой некоторое подмножество множества собственных префиксов базового языка $B$ – а последнее множество конечное [13]. В предыдущих статьях [3], [4] задающие группоид (а также автомат PRI) языки в формулах обозначались $A$ и $B$. Алфавит $\Delta_A$ является вспомогательным алфавитом, определяемым для заданного языка $A$, соответствующий ему морфизм –

$$h_A : \Delta_A^* \to \Sigma^*.$$

Перейдём к основному определению автомата. Детерминированный автомат $\mathrm{PRI}(A,B)$ для заданных конечных языков $A, B \subseteq \Sigma^*$ определяется следующим образом:

$$\mathrm{PRI}(A,B) = \big(\, Q_B,\, \Delta_A,\, \delta_{A-B},\, \{\{\varepsilon\}\},\, \{\varnothing\}\big), \qquad (1)$$

где:

- $Q_B \subseteq \mathcal{P}(\mathrm{opref}(B))$;
- $\Delta_A = \{0,1,\dots,n{-}1\}$, где $n = |A|$ – число слов языка $A$;
- для некоторых $q_1, q_2 \in \overline{Q}_B$ и $d \in \Delta_A$ переход

  $$q_1 \xrightarrow[\delta_{A-B}]{d} q_2$$

  имеется тогда и только тогда, когда

  $$\Phi_B(q_1, \{h_A(d)\}) = q_2.$$

При этом мы определяем, что в автомате будет по одному входному и выходному состоянию – то есть $\{\{\varepsilon\}$ и $\{\varnothing\}$ соответственно, а все состояния мы определяем как некоторые подмножества – этим и вызваны применённые «сложные» обозначения. Из рассматриваемой полной системы подмножеств могут быть удалены недостижимые состояния – что мы обычно и делаем, получая *подгруппоид*.

Итак, на основе определённого первичного автомата (1) мы вводим *полугруппу (моноид) преобразований* как

$$\mathrm{Pri}(A,B) = (\mathcal{I}_r(Q_B), \circ)$$

(мы специально применяем «похожее на автомат» обозначение для полугруппы), где:

$$\big(\forall\, q \in Q_B\big)\ \big(\forall\, i \in \overline{0, |A|{-}1}\big)\ \big(\exists\, \phi_{h_A(i)} \in \mathcal{I}_r(Q_B)\big)$$
$$\big(i \xrightarrow{h_A} a_i \xrightarrow{\tau_{\mathfrak{A}}} \phi_{h_A(i)},\ q\phi_{h_A(i)} = \delta_{A-B}(q, a_i) \in Q_B\big).$$

Если мы имеем дело с «плохими» автоматами PRI – т. е. такими с автоматами, чей группоид относительно их основной операции полугруппой не является – то мы можем использовать описанный нами способ получения полугруппы, *жертвуя понятийной простотой ради достижения структурного совершенства*. Именно такой способ будет использоваться в качестве основного – для дальнейших исследований, связанных с полугруппой автомата PRI. При этом построенная полугруппа обладает различными интересными свойствами – и некоторые из них мы рассмотрим в настоящей статье.

Кроме того, надо отметить, что мы строим не автомат по некоторой полугруппе (как делается, по видимому, в большинстве алгебраических приложений), а проводим действия, которые можно назвать обратными – строим полугруппе по некоторому автомату.

Но, формулируя другими словами сказанное во введении, можно сказать, что основная цель – получить упрощение автомата $\mathrm{PRI}(A,B)$, то есть описать полугрупповую структуру вида $(Q_B, \Phi_B)$, не зависящую от первого языка, обычно обозначаемого $A$ (в предыдущих публикациях и выше в настоящей статье). То есть рассматривая автомат $\mathrm{PRI}(B,B)$ – или просто $\mathrm{PRI}(B)$ – который имеет вид

$$\mathrm{PRI}(B) = \big(\mathcal{P}(\mathrm{opref}(B)), \Delta_{\mathcal{P}(\mathrm{opref}(B))}, \Phi_B, \{\{\varepsilon\}\}, \{\varnothing\}\big),$$

мы можем сказать, что он соответствует группоиду

$$\mathrm{Pri}(B) = \big(\mathcal{P}(\mathrm{opref}(B)), \Phi_B\big),$$

являющемуся в частных случаях полугруппой. Очевидно, что при этом операция определена на «подходящем» множестве:

$$\Phi_B : Q_B \times Q_B \to Q_B.$$

Свойство замкнутости, очевидно, выполнено – и поэтому для того, чтобы ответить а вопрос о том, является ли полугруппой группоид автомата [14], нам остаётся проверить ассоциативность рассматриваемой операции.

## IV. ОСНОВНАЯ ТЕОРЕМА

В этом разделе доказана теорема, отражающее основное свойство рассматриваемого объекта – то, что он действительно является полугруппой. Как мы уже отмечали, эта теорема связывает:

[12] То есть то, что соответствующее *бесконечное итерационное дерево* содержит конечное число классов эквивалентности.

[13] И, конечно, в частных случаях это может быть проверено с помощью компьютерной программы – описать которую мы предполагаем в статье-продолжении.

[14] Ещё раз отметим, что он не является полугруппой преобразований автомата – это совершенно иная структура.

- первичные автоматы, полученные из бесконечного итерационного дерева морфизма, определённого и исследованного в [1], [2] и др., –
- с соответствующей полугруппой преобразований, введённой в предыдущем разделе.

Действительно, (детерминированный) конечный автомат в нашем случае однозначно определяет конечную полугруппу [15] – но подобное доказательство для нашего конкретного автомата требует дополнительных построений.

Сначала приведём теорему, которая фактически является формулировкой результатов процитированных ранее работ.

**Теорема 1:** Для любой пары заданных конечных языков $A$ и $B$ автомат $\mathrm{PRI}(A,B)$ является детерминированным конечным автоматом. □

Теперь опишем основную теорему настоящей статьи.

**Теорема 2 (основная):** Для любого конечного языка $B \subseteq \Sigma^*$ группоид

$$(\mathcal{J}_r(Q_B), \circ)$$

является конечной полугруппой.

*Доказательство.* Первое условие, которое необходимо выполнить для доказательства, состоит в том, чтобы установить отображение из некоторого *свободного* входного алфавита в полугруппу преобразований множества состояний. Для этого мы, например, должны определить, обладает ли функция перехода, введённая в [1], [3], замкнутостью над множеством состояний. Тогда, согласно свойству прямого произведения языка, соответствующего суперпозиции нескольких преобразований ([6, сс. 18, 22, 177, 178]), а также его продолжения на моноид, мы получаем полугруппу автомата.

Сначала опишем проверку замкнутости. Заметим, что в [1] согласно алгоритму формирования функции перехода $\delta_{A-B}$ результат вычисления для любой пары языков является подмножеством множества собственных префиксов языка $B$; это подмножество, согласно определению первичного автомата PRI, принадлежит множеству состояний – так как оно состоит из достижимых состояний при выполнении функции перехода. Тогда для $Q_B$ и $q_0 = \{\varepsilon\}$ мы для автомата $\mathrm{PRI}(A,B)$ получаем следующее:

$$\left(\forall q \in Q_B\right) \left(\exists \bar{a} \in \Delta_A^*\right) \left(q_0 \xrightarrow[\delta_{A-B}]{\bar{a}} q\right).$$

Следовательно, функция переходов замкнута относительно множества состояний – и одновременно можно однозначно определить отображение из свободного входного алфавита в множество преобразований, которое замкнуто относительно множества состояний.

Вследствие доказанного выполнения свойства замкнутости мы можем перейти к доказательству выполнения свойства ассоциативности [16].

Условие на полугруппу (моноид) автомата, обладающую этим свойством, можно записать в виде

$$(q\phi_x)\phi_y = q(\phi_x \circ \phi_y),$$
$$\text{где} \quad q = \{\beta_1, \beta_2, \cdots, \beta_s\} \subseteq \mathrm{opref}(B).$$

Сразу можно заметить, что входом здесь является слово произвольной длины из свободного входного алфавита; такая произвольность длины влечёт возможность увеличения числа членов в последовательности суперпозиции функций – и это ассоциативности не нарушает. Поэтому мы, не ограничивая общности, для простоты рассмотрим только тот случай, когда вход является 1-буквенным словом.

Для левой части равенства, т. е. для записи $(q\phi_x)\phi_y$, мы видим, что при обработке первой части ввода автомат переходит в некоторое промежуточное состояние, которое по определению также является множеством собственных префиксов языка $B$. Далее будем рассматривать состояния в некотором определённом порядке – например, в лексикографическом порядке для рассматриваемого свободного алфавита; т. е. если применим те же самые обозначения

$$(q\phi_x)\phi_y = q(\phi_x \circ \phi_y),$$
$$\text{для} \quad B = (b_1, b_2, \cdots, b_s),$$

то после выполнения рассматриваемого процесса полученный результат также будет иметь некоторый фиксированный порядок – соответствующий рассматриваемому. Это можно записать следующим образом:

$$\tilde{q} = (\beta_{11}, \ldots, \beta_{1k_1}, \beta_{21}, \ldots, \beta_{2k_2}, \ldots, \beta_{s1}, \ldots, \beta_{sk_s}),$$
$$\text{где} \ \underset{\{x\}-B}{\Phi}(\beta_i) = (\beta_{i1}, \ldots, \beta_{ik_i}) \ \text{ для } \ i \in \overline{1,s};$$

приведём два связанных замечания к последней формуле:

- согласно сказанному выше, мы не различаем множества и упорядоченные множества;
- лексикографический порядок, вообще говоря, нарушен – но это не влияет на изложение доказательства.

После этого мы принимаем вторую часть ввода

$$(\beta_{11} \cdot y,\ \beta_{12} \cdot y,\ \cdots,\ \beta_{sk_s} \cdot y) \ -$$

и тем самым переходим от промежуточного состояния к итоговому.

А для правой части формулы, т. е. $q(\phi_x \circ \phi_y)$, мы в силу эквивалентности суперпозиции функций и прямого произведения языков напрямую применяем к начальному состоянию входной сигнал длины 2, т. е.

$$(\beta_1 \cdot xy,\ \beta_2 \cdot xy,\ \cdots,\ \beta_s \cdot xy).$$

В процессе работы алгоритма должна появиться конкатенация слов, полностью совпадающих со словами промежуточного состояния $\tilde{q}$, и второй входной буквы

$$\beta_{ij} \cdot y \ -$$

что доказывает теорему. □

## V. Основные алгебраические свойства полугруппы автомата PRI

Перечислим необходимые для дальнейшего изложения свойства полугрупп, а также некоторые возможные элементы полугрупп конкреного вида. При этом мы приводим такие определения, которые могут быть применены для описания применения к рассматриваемой полугруппе первичного автомата PRI.

Полугруппа $(S, \cdot)$ является *коммутативной*, если её элементы перестановочны:

$$(\forall x, y \in S)\ (x \cdot y = y \cdot x).$$

[15] В отличие от мини-примера, рассмотренного во введении – где рассматриваемый группоид полугруппой не является.

[16] Напомним, что краткое и неформальное описание этого доказательства было приведено во введении.

В *не* коммутативной полугруппе также могут быть некоторые перестановочные пары элементов (в этом случае говорят, что элементы этой пары *коммутируют*) – но обязательно имеется хотя бы одна некоммутирующая пара.

*Единицей* полугруппы $(S, \cdot)$ является некоторый её элемент $\iota$, такой что

$$(\forall x \in S)\ (x \cdot \iota = \iota \cdot x = x).$$

Легко доказать, при наличии единицы она единственна. Полугруппа с единицей называется *моноидом*.

*Идемпотентом* является такой элемент $e$ полугруппы, что $e \cdot e = e$. Множество всех идемпотентов полугруппы $S$ обозначается $E(S)$. Полугруппа $S$ называется *унипотентной*, если $|E(S)| = 1$; при этом такой её идемпотент обозначается $e_S$. В этом случае *порядком элемента* $g$ полугруппы $S$ называется такое наименьшее натуральное $t$, что $g^t = e_S$; если для выбранного элемента $g$ такого $t$ не существует, то считаем, что порядок $g$ равен бесконечности.

Согласно сказанному выше, для полугрупп преобразований идемпотент должен быть определён следующим образом: это такой элемент $e$, что

$$\left(\forall q \in Q_B,\ e \xrightarrow{\tau_{\mathfrak{A}}} \phi_e\right)\ \left(q\phi_e\phi_e = q\phi_e\right).$$

В качестве заключительного состояния автомата PRI мы были обязаны указать единственное – пустое множество[17] – и этот элемент является тривиальным идемпотентом. Понятно, что интерес вызывают только *не* тривиальные идемпотенты – однако часто полугруппы первичных автоматов унипотентны, т. е. они содержат только тривиальный идемпотент, а порядок их элементов (кроме идемпотента) обычно является бесконечностью.

Пусть $P$ – некоторое непустое подмножество множества $S$, причём $(S, \cdot)$ – полугруппа. Для любого $P$ существует *наименьшая подполугруппа в* $S$, *содержащая* $P$, которая обозначается $<P>$; она состоит из всех конечных произведений элементов из $P$. Если $<P> = S$, то множество $P$ называют *порождающим* или *системой образующих элементов полугруппы* $S$. Если $P$ содержит только один элемент $g$, то полугруппа $<P>$ (или $<g>$, опускаем фигурные скобки) называется *циклической* или *моногенной*, а элемент $g$ – *порождающим моногенную полугруппу* $<g>$.

В частности, для входного алфавита $A$ и некоторого его подмножества

$$\tilde{A} \subseteq A, \qquad \tilde{A} = \{a_1, \dots, a_r\} \quad \text{при} \quad r \in \mathbb{N}$$

получаем

$$<\tilde{A}> = \big\{\phi_p \in \mathcal{I}_r(Q_B)\ \big|\ (\forall q \in Q_B)\ \left(q\phi_p = q(\phi_{a_{k_1}} \circ \dots \circ \phi_{a_{k_i}}\right),\ \{k_1, \cdots, k_i\} \subseteq \{a_1, \cdots, a_r\}\big)\big\}.$$

Очевидно, что рассмотренное понятие наименьшей подполугруппы можно естественно продолжить до моноида; при этом должна быть добавлена возможность $i = 0$ – т. е.

$$\{k_1, \cdots, k_i\} = \varnothing.$$

[17] Этим для автомата PRI и была вызвана запись $\{\varnothing\}$ в качестве множества заключительных состояний.

И, как можно видеть, в случае свободного моноида и в других частных случаях с наименьшей подполугруппой совпадает множество, состоящее из входного алфавита[18].

Продолжим обсуждение материала раздела II. При наличии некоторого описания порождающей подполугруппы на ассоциативность можно проверять не все упорядоченные тройки элементов, а только те из них, которые входят в наименьшую порождающую подполугруппу.

Важным объектом исследования полугруппы, в первую очередь конечной, является ряд степеней элемента:

$$g, g^2, g^3, \cdots, g^i, \dots, g \in S. \tag{2}$$

А в силу ассоциативности полугруппы мы имеем корректность следующих равенств:

$$g^t \cdot g^r = g^{r+t}, \qquad (g^t)^r = g^{tr}.$$

Пусть в ряду (2) встречаются одинаковые элементы, а $t$ – такое наименьшее натуральное число, что $g^d = g^t$ при некотором натуральном $d < t$. Обозначим $n = t - d$ и перепишем последнее равенство в виде $g^d = g^{d+n}$. Для конечной циклической полугруппы $<g>$ это равенство называется *определяющим соотношением*, а пара

$$(g;\ g^d = g^{d+n}),$$

состоящая из порождающего элемента $g$ и определяющего соотношения – её *копредставлением*. Несложно показать, что равенство

$$g^{d+i} = g^{d+n+i}$$

выполнено при любом $i \in \mathbb{N}$.

При этом:

- число $d$ называется *циклической глубиной* или *индексом элемента* $g$; обозначение $\mathrm{dep}(g)$;
- число $n$ называется *периодом элемента* $g$; обозначение $\mathrm{per}(g)$.

Также будем говорить, что элемент $g$ полугруппы с циклической глубиной $d$ и с периодом $n$ имеет тип $(d; n)$ и порождает циклическую полугруппу $<g>$ типа $(d; n)$.

По отношению полугруппы автомата определяем член ряда как упорядоченный набор следующего вида:

$$g^r = \left(q_{1r}, q_{2r}, \dots, q_{|Q_B|r}\right),$$

где

$$q_{ir} = q_i\Big(\underbrace{\phi_g \circ \cdots \circ \phi_g}_{r}\Big), \quad 1 \le i \le |Q_B|.$$

Приведём ещё одно отношение, которое не только относится к свойствам полугруппы, но и в немалой степени связано с автоматами. На множестве, содержащем:

- все (неупорядоченные) пары различных элементов $Q_B$ (т. е. сочетания по 2 элемента этого множества),
- а также специальный элемент $\mathcal{F}$,

рассмотрим такое бинарное отношение $\succ$:

- $\{q_1, q_2\} \succ \{q_3, q_4\}$, где $q_1 \ne q_2$ и $q_3 \ne q_4$ – если для некоторого $a \in \Delta$ имеем $\Phi_B(q_1, a) = q_3$ и $\Phi_B(q_2, a) = q_4$;
- $\{q_1, q_2\} \succ \mathcal{F}$, если и только если из двух состояний $q_1 \ne q_2$ ровно одно финальное.

[18] Точнее, множество всех 1-буквенных слов.

Это бинарное отношение связано с алгоритмами построения канонического (минимального детерминированного) конечного автомата – и, конечно, легко связывается с рассматриваемыми нами полугруппами. (Про возможные подходы к построению *практических* алгоритмов для давно решённой проблемы канонизации конечного автомата см., например, [13].)

Некоторые примеры описанных здесь понятий мы рассмотрим в следующем разделе, а также предполагаем рассмотреть в статье-продолжении.

## VI. НЕСЛОЖНЫЕ ПРИМЕРЫ

**Пример 1:** Базовый язык $B_1 = \{aa\}$[19].

Получаем следующее:

$$\mathcal{P}(\mathrm{opref}(B_1)) = \big\{\varnothing, \{\varepsilon\}, \{a\}, \{\varepsilon, a\}\big\}.$$

Согласуясь с предыдущими договорёнными, обозначим

$$\Delta_{\mathcal{P}(\mathrm{opref}(B_1))} = \{0, 1, 2, 3\};$$

при этом морфизм такой[20]:

$$h(0) = \varnothing,\ h(1) = \{\varepsilon\},\ h(2) = \{a\},\ h(3) = \{\varepsilon, a\}.$$

После несложных вычислений получаем следующую таблицу для группоида $\mathrm{Pri}(B_1)$:

Таблица I
Группоид для примера 1.

| | | 0 | 1 | 2 | 3 |
|---|---|---|---|---|---|
| $\leftarrow$ | 0 | 0 | 0 | 0 | 0 |
| $\rightarrow$ | 1 | 0 | 1 | 2 | 3 |
| | 2 | 0 | 2 | 1 | 3 |
| | 3 | 0 | 3 | 3 | 3 |

Приведём то же самое в графическом виде:

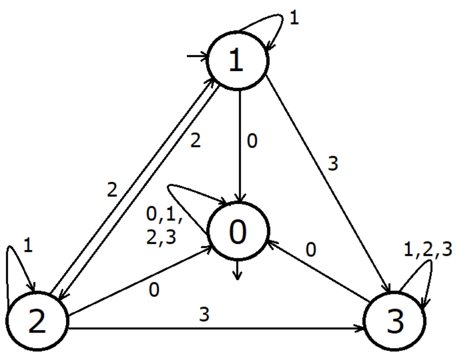


Рис. 3. Граф автомата для примера 1.

Можно показать, что ассоциативность выполняется, коммутативность тоже, единица есть – то есть полученный группоид $\mathrm{Pri}(B_1)$ является коммутативным моноидом. $\square$

[19] Пример может быть посчитан «вручную», без применения компьютерной программы.

[20] Ниже записи морфизма будем иногда опускать.

**Пример 2:** Базовый язык $B_2 = \{aa, aaaaaab\}$[21].

Мы видим, что

$$\big|\mathcal{P}(\mathrm{opref}(B_2))\big| = 2^7,$$

поэтому будем обозначать

$$\Delta_{\mathcal{P}(\mathrm{opref}(B_2))} = \{0, 1, \ldots, 127\}.$$

Небольшая часть множества значений результирующего группоида[22] приведена на рис. 4 ниже. Для всех приведённых пар значений выполняется коммутативность, а для всех приведённых упорядоченных троек – ассоциативность; согласно результатам работы программы, те же факты верны и для всех значений. Однако единичного элемента нет – таковым не является ни «естественный вариант» единицы $\{\varepsilon\}$, ни какой-либо иной элемент из показанных либо не показанных на рисунке. Поэтому рассматриваемый группоид $\mathrm{Pri}(B_2)$ является коммутативной полугруппой с нулём, но не является моноидом. А возможное дополнение этой полугруппы до моноида путём введения нового специального единичного элемента вряд ли представляет интерес для рассматриваемой нами предметной области – группоиде первичного автомата PRI.

Приведём ещё небольшие комментарии к рис. 4.

- Для обозначения «номеров» подмножеств мы применяем такой несложный алгоритм: пустое множество, далее множества с 1 элементом, потом с 2, с 3 и т. д. Множества с одинаковым количеством элементов упорядочиваем согласно лексикографического порядка. См. подробнее рис. 5.
- По-видимому, ассоциативность выполняется вследствие того, что все собственные префиксы всех слов заданного языка являются словами над 1-буквенным алфавитом[23]. $\square$

Итак, в обоих рассмотренных примерах мы получили полугруппы, причём коммутативные. Однако существуют достаточно простые примеры, для которых рассматриваемый в них группоид полугруппой не является.

**Пример 3:** Базовый язык $B_3 = \{aa, bb\}$[24].

После несложных вычислений получаем таблицу для группоида $\mathrm{Pri}(B_1)$, приведённую в таб. II.

Таблица II
Группоид для примера 3.

| | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $\leftarrow$ | $\varnothing$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | $\rightarrow$ | $\{\varepsilon\}$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | | $\{a\}$ | 0 | 2 | 1 | 0 | 4 | 2 | 1 | 4 |
| 3 | | $\{b\}$ | 0 | 3 | 0 | 1 | 3 | 5 | 1 | 5 |
| 4 | | $\{\varepsilon,a\}$ | 0 | 4 | 4 | 3 | 4 | 7 | 7 | 7 |
| 5 | | $\{\varepsilon,b\}$ | 0 | 5 | 2 | 5 | 7 | 5 | 7 | 7 |
| 6 | | $\{a,b\}$ | 0 | 6 | 1 | 1 | 7 | 7 | 1 | 7 |
| 7 | | $\{\varepsilon,a,b\}$ | 0 | 7 | 4 | 5 | 7 | 7 | 7 | 7 |

[21] Пример вряд ли может быть посчитан «вручную», применение компьютерной программы, по-видимому, необходимо.

[22] Примерно 5%.

[23] Но подробное рассмотрение этого вопроса, в частности – возможное описание необходимых и достаточных условий ассоциативности, в круг вопросов настоящей статьи не входит.

[24] Этот пример также может быть посчитан «вручную».

| | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ← | 0 | Ø | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| → | 1 | {ε} | 0 | 1 | 2 | 9 | 15 | 35 | 49 | 78 | 8 | 9 | 30 | 35 | 69 | 78 | 29 | 15 | 65 | 49 | 103 | 64 | 35 | 100 | 78 | 99 | 49 | 121 | 120 | 78 | 127 | 29 |
| | 2 | {a} | 0 | 2 | 9 | 15 | 35 | 49 | 78 | 49 | 29 | 15 | 65 | 49 | 103 | 49 | 64 | 35 | 100 | 78 | 100 | 99 | 49 | 121 | 49 | 120 | 78 | 120 | 127 | 49 | 127 | 64 |
| | 3 | {aa} | 0 | 9 | 15 | 35 | 49 | 78 | 49 | 78 | 64 | 35 | 100 | 78 | 100 | 78 | 99 | 49 | 121 | 49 | 121 | 120 | 78 | 120 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 99 |
| | 4 | {aaa} | 0 | 15 | 35 | 49 | 78 | 49 | 78 | 49 | 99 | 49 | 121 | 49 | 121 | 49 | 120 | 78 | 120 | 78 | 120 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 127 | 49 | 127 | 120 |
| | 5 | {aaaa} | 0 | 35 | 49 | 78 | 49 | 78 | 49 | 78 | 120 | 78 | 120 | 78 | 120 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 6 | {aaaaa} | 0 | 49 | 78 | 49 | 78 | 49 | 78 | 49 | 127 | 49 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 78 | 127 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 127 | 49 | 127 | 127 |
| | 7 | {aaaaaa} | 0 | 78 | 49 | 78 | 49 | 78 | 49 | 78 | 127 | 78 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 8 | {ε,a} | 0 | 8 | 29 | 64 | 99 | 120 | 127 | 127 | 29 | 64 | 99 | 120 | 127 | 127 | 64 | 99 | 120 | 127 | 127 | 99 | 120 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 64 |
| | 9 | {ε,aa} | 0 | 9 | 15 | 35 | 49 | 78 | 49 | 78 | 64 | 35 | 100 | 78 | 100 | 78 | 99 | 49 | 121 | 49 | 121 | 120 | 78 | 120 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 99 |
| | 10 | {ε,aaa} | 0 | 30 | 65 | 100 | 121 | 120 | 127 | 127 | 99 | 100 | 121 | 120 | 127 | 127 | 120 | 121 | 120 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 120 |
| | 11 | {ε,aaaa} | 0 | 35 | 49 | 78 | 49 | 78 | 49 | 78 | 120 | 78 | 120 | 78 | 120 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 12 | {ε,aaaaa} | 0 | 69 | 103 | 100 | 121 | 120 | 127 | 127 | 127 | 100 | 127 | 120 | 127 | 127 | 127 | 121 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 13 | {ε,aaaaaa} | 0 | 78 | 49 | 78 | 49 | 78 | 49 | 78 | 127 | 78 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 14 | {a,aa} | 0 | 29 | 64 | 99 | 120 | 127 | 127 | 127 | 64 | 99 | 120 | 127 | 127 | 127 | 99 | 120 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 99 |
| | 15 | {a,aaa} | 0 | 15 | 35 | 49 | 78 | 49 | 78 | 49 | 99 | 49 | 121 | 49 | 121 | 49 | 120 | 78 | 120 | 78 | 120 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 127 | 49 | 127 | 120 |
| | 16 | {a,aaaa} | 0 | 65 | 100 | 121 | 120 | 127 | 127 | 127 | 120 | 121 | 120 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 17 | {a,aaaaa} | 0 | 49 | 78 | 49 | 78 | 49 | 78 | 49 | 127 | 49 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 78 | 127 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 127 | 49 | 127 | 127 |
| | 18 | {a,aaaaaa} | 0 | 103 | 100 | 121 | 120 | 127 | 127 | 127 | 127 | 121 | 127 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 19 | {aa,aaa} | 0 | 64 | 99 | 120 | 127 | 127 | 127 | 127 | 99 | 120 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 120 |
| | 20 | {aa,aaaa} | 0 | 35 | 49 | 78 | 49 | 78 | 49 | 78 | 120 | 78 | 120 | 78 | 120 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 21 | {aa,aaaaa} | 0 | 100 | 121 | 120 | 127 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 22 | {aa,aaaaaa} | 0 | 78 | 49 | 78 | 49 | 78 | 49 | 78 | 127 | 78 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 23 | {aaa,aaaa} | 0 | 99 | 120 | 127 | 127 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 24 | {aaa,aaaaa} | 0 | 49 | 78 | 49 | 78 | 49 | 78 | 49 | 127 | 49 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 78 | 127 | 127 | 49 | 127 | 49 | 127 | 78 | 127 | 127 | 49 | 127 | 127 |
| | 25 | {aaa,aaaaaa} | 0 | 121 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 26 | {aaaa,aaaaa} | 0 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 27 | {aaaa,aaaaaa} | 0 | 78 | 49 | 78 | 49 | 78 | 49 | 78 | 127 | 78 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 49 | 127 | 127 | 78 | 127 | 78 | 127 | 49 | 127 | 127 | 78 | 127 | 127 |
| | 28 | {aaaaa,aaaaaa} | 0 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 |
| | 29 | {ε,a,aa} | 0 | 29 | 64 | 99 | 120 | 127 | 127 | 127 | 64 | 99 | 120 | 127 | 127 | 127 | 99 | 120 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 99 |
| | 30 | {ε,a,aaa} | 0 | 30 | 65 | 100 | 121 | 120 | 127 | 127 | 99 | 100 | 121 | 120 | 127 | 127 | 120 | 121 | 120 | 127 | 127 | 127 | 120 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 127 | 120 |

Рис. 4. Группоид для примера 2 – «верхняя левая» часть таблицы значений.

| | | | | | |
|---|---|---|---|---|---|
| 29 | {ε,a,aa} | 62 | {aaa,aaaaa,aaaaaa} | 95 | {aa,aaa,aaaa,aaaaaa} |
| 30 | {ε,a,aaa} | 63 | {aaaa,aaaaa,aaaaaa} | 96 | {aa,aaa,aaaaa,aaaaaa} |
| 31 | {ε,a,aaaa} | 64 | {ε,a,aa,aaa} | 97 | {aa,aaaa,aaaaa,aaaaaa} |
| 32 | {ε,a,aaaaa} | 65 | {ε,a,aa,aaaa} | 98 | {aaa,aaaa,aaaaa,aaaaaa} |
| 33 | {ε,a,aaaaaa} | 66 | {ε,a,aa,aaaaa} | 99 | {ε,a,aa,aaa,aaaa} |
| 34 | {ε,aa,aaa} | 67 | {ε,a,aa,aaaaaa} | 100 | {ε,a,aa,aaa,aaaaa} |
| 35 | {ε,aa,aaaa} | 68 | {ε,a,aaa,aaaa} | 101 | {ε,a,aa,aaa,aaaaaa} |
| 36 | {ε,aa,aaaaa} | 69 | {ε,a,aaa,aaaaa} | 102 | {ε,a,aa,aaaa,aaaaa} |
| 37 | {ε,aa,aaaaaa} | 70 | {ε,a,aaa,aaaaaa} | 103 | {ε,a,aa,aaaa,aaaaaa} |
| 38 | {ε,aaa,aaaa} | 71 | {ε,a,aaaa,aaaaa} | 104 | {ε,a,aa,aaaaa,aaaaaa} |
| 39 | {ε,aaa,aaaaa} | 72 | {ε,a,aaaa,aaaaaa} | 105 | {ε,a,aaa,aaaa,aaaaa} |
| 40 | {ε,aaa,aaaaaa} | 73 | {ε,a,aaaaa,aaaaaa} | 106 | {ε,a,aaa,aaaa,aaaaaa} |
| 41 | {ε,aaaa,aaaaa} | 74 | {ε,aa,aaa,aaaa} | 107 | {ε,a,aaa,aaaaa,aaaaaa} |
| 42 | {ε,aaaa,aaaaaa} | 75 | {ε,aa,aaa,aaaaa} | 108 | {ε,a,aaaa,aaaaa,aaaaaa} |
| 43 | {ε,aaaaa,aaaaaa} | 76 | {ε,aa,aaa,aaaaaa} | 109 | {ε,aa,aaa,aaaa,aaaaa} |
| 44 | {a,aa,aaa} | 77 | {ε,aa,aaaa,aaaaa} | 110 | {ε,aa,aaa,aaaa,aaaaaa} |
| 45 | {a,aa,aaaa} | 78 | {ε,aa,aaaa,aaaaaa} | 111 | {ε,aa,aaa,aaaaa,aaaaaa} |
| 46 | {a,aa,aaaaa} | 79 | {ε,aa,aaaaa,aaaaaa} | 112 | {ε,aa,aaaa,aaaaa,aaaaaa} |
| 47 | {a,aa,aaaaaa} | 80 | {ε,aaa,aaaa,aaaaa} | 113 | {ε,aaa,aaaa,aaaaa,aaaaaa} |
| 48 | {a,aaa,aaaa} | 81 | {ε,aaa,aaaa,aaaaaa} | 114 | {a,aa,aaa,aaaa,aaaaa} |
| 49 | {a,aaa,aaaaa} | 82 | {ε,aaa,aaaaa,aaaaaa} | 115 | {a,aa,aaa,aaaa,aaaaaa} |
| 50 | {a,aaa,aaaaaa} | 83 | {ε,aaaa,aaaaa,aaaaaa} | 116 | {a,aa,aaa,aaaaa,aaaaaa} |
| 51 | {a,aaaa,aaaaa} | 84 | {a,aa,aaa,aaaa} | 117 | {a,aa,aaaa,aaaaa,aaaaaa} |
| 52 | {a,aaaa,aaaaaa} | 85 | {a,aa,aaa,aaaaa} | 118 | {a,aaa,aaaa,aaaaa,aaaaaa} |
| 53 | {a,aaaaa,aaaaaa} | 86 | {a,aa,aaa,aaaaaa} | 119 | {aa,aaa,aaaa,aaaaa,aaaaaa} |
| 54 | {aa,aaa,aaaa} | 87 | {a,aa,aaaa,aaaaa} | 120 | {ε,a,aa,aaa,aaaa,aaaaa} |
| 55 | {aa,aaa,aaaaa} | 88 | {a,aa,aaaa,aaaaaa} | 121 | {ε,a,aa,aaa,aaaa,aaaaaa} |
| 56 | {aa,aaa,aaaaaa} | 89 | {a,aa,aaaaa,aaaaaa} | 122 | {ε,a,aa,aaa,aaaaa,aaaaaa} |
| 57 | {aa,aaaa,aaaaa} | 90 | {a,aaa,aaaa,aaaaa} | 123 | {ε,a,aa,aaaa,aaaaa,aaaaaa} |
| 58 | {aa,aaaa,aaaaaa} | 91 | {a,aaa,aaaa,aaaaaa} | 124 | {ε,a,aaa,aaaa,aaaaa,aaaaaa} |
| 59 | {aa,aaaaa,aaaaaa} | 92 | {a,aaa,aaaaa,aaaaaa} | 125 | {ε,aa,aaa,aaaa,aaaaa,aaaaaa} |
| 60 | {aaa,aaaa,aaaaa} | 93 | {a,aaaa,aaaaa,aaaaaa} | 126 | {a,aa,aaa,aaaa,aaaaa,aaaaaa} |
| 61 | {aaa,aaaa,aaaaaa} | 94 | {aa,aaa,aaaa,aaaaa} | 127 | {ε,a,aa,aaa,aaaa,aaaaa,aaaaaa} |

Рис. 5. Обозначение всех элементов группоида для примера 2 (элементов, не вошедших в рис. 4).

```
2,2,3  2,2,5  2,2,6  2,2,7  2,3,3  2,3,5  2,3,6  2,3,7  2,4,3  2,4,5
2,4,6  2,4,7  2,5,3  2,5,6  2,6,3  2,6,5  2,6,6  2,6,7  2,7,3  2,7,5
2,7,6  2,7,7  3,2,2  3,2,4  3,2,6  3,2,7  3,3,2  3,3,4  3,3,6  3,3,7
3,4,2  3,4,6  3,5,2  3,5,4  3,5,6  3,5,7  3,6,2  3,6,4  3,6,6  3,6,7
3,7,2  3,7,4  3,7,6  3,7,7  4,2,3  4,2,5  4,2,6  4,2,7  4,3,3  4,3,5
4,3,6  4,3,7  4,5,3  4,6,3  4,6,6  4,7,3  5,2,2  5,2,4  5,2,6  5,2,7
5,3,2  5,3,4  5,3,6  5,3,7  5,4,2  5,6,2  5,6,6  5,7,2  6,2,2  6,2,3
6,2,4  6,2,5  6,3,2  6,3,3  6,3,4  6,3,5  6,4,2  6,4,3  6,5,2  6,5,3
6,6,2  6,6,3  6,6,4  6,6,5  6,7,2  6,7,3  7,2,2  7,2,3  7,2,4  7,2,5
7,3,2  7,3,3  7,3,4  7,3,5  7,4,2  7,5,3  7,6,2  7,6,3  7,7,2  7,7,3
```

Рис. 6. Упорядоченные тройки, нарушающие ассоциативность в примере 3.

Как несложно видеть, нарушения ассоциативности имеются; одно из 100 таких нарушений – упорядоченная тройка $(2, 2, 3)$, поскольку

$$(2\,\delta\,2)\,\delta\,3 = 1\,\delta\,3 = 3,$$

но при этом

$$2\,\delta\,(2\,\delta\,3) = 2\,\delta\,0 = 0.$$

(Все 100 упорядоченных троек, нарушающих ассоциативность, приведены на рис. 6.)

Таким образом, полученный группоид может быть назван коммутативным группоидом с нулём и единицей [25] – однако полугруппой он не является. □

## VII. Заключение

В намечаемой статье – продолжении данной мы в первую очередь сосредоточимся на более подробном описании и программной реализации алгоритмов:

- работы с автоматом PRI,
- получения соответствующего группоида
- и проверки его свойств –

поскольку нас интересуют не только теоретические аспекты, но и различные примеры, необхдмые для проверки формулируемых об автомате PRI и соответствующем ему группоиде Pri гипотез. Заранее отметим, что оценка сложности алгоритмов для подобных программ для нас большого интереса не представляет – однако немного материала, поясняющего сложность алгоритмов для применяемой нами в программах структуры объектов, мы привести предполагаем. Как мы уже отметили, построеный группоид – и, в частных случаях, полугруппа – обладают рядом интересных свойств, некоторые из которых мы уже рассмотрели в настоящей статье.

Также мы предполагаем привести более подробные исследования, связанные со следующими вопросами, рассмотрение которых в настоящей статье только началось.

- Нужно показать или опровергнуть сделанное выше предположение о том, что ассоциативность группоида Pri(B) выполняется вследствие того, что все собственные префиксы всех слов заданного языка B являются словами над 1-буквенным алфавитом. Понятно, что такое условие не является необходимым – но является ли оно достаточным?
- Желательно также исследовать «причины» нарушения ассоциативности в примере 3: во всех приведённых упорядоченных тройках элементов группоида, связанных с нарушением ассоциативности, присутствуют либо единственный элемент 2, либо единственный элемент 3, либо оба этих элемента без каких-либо иных; а оба эти элемента (2 и 3) во-первых, состоят из 1 буквы 2-буквенного слова, и, во-вторых, «не имеют прообразов» после применения инверсного морфизма. Является ли такая причина необходимым условием? является ли достаточным? (Говоря про подобный инверсный морфизм, сто́ит рассмотреть материал последней статьи [14].)
- Могут ли подобные элементы группоида Pri (аналогичные рассмотренным в предыдущем пункте) появляться при исследовании реальных *пар* языков – т. е. при рассмотрении автомата PRI(A, B)?

И, конечно, в статье-продолжении мы предполагаем привести результаты рассмотрения примеров «существенно большего размера».

[25] Эти объекты здесь «обычные», причём их необходимые свойства, а также указанная нами коммутативность, проверяются элементарно.

## VIII. Благодарности

Настоящая работа была частично поддержана грантом научной программы китайских университетов “Higher Education Stability Support Program” (раздел “Shenzhen 2022 – Science, Technology and Innovation Commission of Shenzhen Municipality”) – 深圳市 2022 年高等院校稳定支持计划资助项目.

## Список литературы

[1] 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.
[2] 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.
[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] Melnikov B. *2ω-finite automata and sets of obstructions of their language* // The Korean Journal of Computational and Applied Mathematics (Journal of Applied Mathematics and Computing). – 2000. – Vol. 6. No. 3. – P. 565–574.
[6] Скорняков Л. (ред.) *Общая алгебра. Том 2*. – М., Наука. – 1991. – 480 с.
[7] Ляпин Е. С. *Полугруппы*. – М., Физматлит. – 1960. – 592 с.
[8] Биркгоф Г., Барти Т. *Современная прикладная алгебра*. – М., Мир. – 1976. – 400 с.
[9] Лаллеман Ж. *Полугруппы и комбинаторные приложения*. – М., Мир. – 1985. – 440 с.
[10] Саломаа А. *Жемчужины теории формальных языков*. – М., Мир. – 1986. – 159 с.
[11] Pin J.-E. *Mathematical Foundations of Automata Theory*. – Berlin, Springer-Verlag. – 2012. – 310 p.
[12] Мельников Б. Ф., Мельникова А. А. *Применение лепестковых конечных автоматов для проверки выполнения частного случая гипотезы Zyu (для заданного конечного языка)* // International Journal of Open Information Technologies. – 2023. – Vol. 11. No. 3. – P. 1–11.
[13] Мельников Б. Ф., Долгов В. Н. *Упрощённые регулярные языки и специальное отношение эквивалентности на классе регулярных языков. Часть I* // International Journal of Open Information Technologies. – 2022. – Vol. 10. No. 9. – P. 12–20.
[14] Мельников Б. Ф. *Полиномиальный алгоритм построения оптимального инверсного морфизма* // International Journal of Open Information Technologies. – 2023. – Vol. 11. No. 6. – P. 1–10.

МЭН Линцянь,
бакалавр Университета МГУ – ППИ в Шэньчжэне
(`http://szmsubit.ru/`),
email$_1$: `1120190010@smbu.edu.cn`,
email$_2$: `kostya45e@mail.ru`,
ORCID: `orcidID=0009-0009-2933-5684`.

Борис Феликсович МЕЛЬНИКОВ,
профессор Университета МГУ – ППИ в Шэньчжэне
(`http://szmsubit.ru/`),
email$_1$: `bormel@smbu.edu.cn`,
email$_2$: `bf-melnikov@yandex.ru`,
mathnet.ru: `personid=27967`,
elibrary.ru: `authorid=15715`,
scopus.com: `authorId=55954040300`,
ORCID: `orcidID=0000-0002-6765-6800`.

# On the groupoid of the primary automaton PRI defined for the only finite language

Meng Lingqian, Boris Melnikov

***Abstract*—In this paper, we continue the theme of our previous papers devoted to the important equivalence relation in a set of formal languages, i.e., the equivalence relation in infinity; for its detailed study we also considered the binary coverage relation. In the study of these relations, we used infinite iterative morphism trees, for each of which, after combining its equivalent vertices, a deterministic finite automaton is obtained.**

**Among some similar variants of automata corresponding to an infinite iterative morphism tree, of greatest interest is the so-called primary automaton (PRI automaton), which is defined in a special way for a given pair of finite languages. It is defined on subsets of the set of prefixes of one of these languages, and we consider these subsets as states of the defined automaton.**

**In this paper, we continue to consider the primary automaton constructed for any pair of finite languages, but we also define new objects: an automaton constructed for only one finite language, as well as the groupoid and semigroup of transformations corresponding to this automaton.**

**Let us describe the main subject of this paper. If we "combine line by line" the structures arising in the PRI automaton, then we get the result of the transition of each state – in the case when the input is some language. As a result, we can study a primary automaton built on the basis of only one language, and this language can be called the basic one. For both the set of states and the input alphabet, we shall use a subset of the set of proper prefixes of the base language, or its prototype obtained using some inverse morphism.**

**The paper considers the resulting groupoid and proves the theorem that in this case, it is possible to define a semigroup of transformations. The main idea of proving this fact is as follows. If the semigroup requirement is considered from the point of view of a finite automaton, then on the two parts of the equation used in it, we can say the following. The left part of the equation of this semigroup requirement describes the transition from the current state in accordance with the first subword (the prefix) of the input to some intermediate state; then the continuation of work in accordance with the second subword (the suffix) of the input. We can say the same for the right side of the equation, and the input string in both cases can be considered the same.**

**This paper also discusses some algebraic properties of the described semigroup, including simple examples. We propose to consider more complex examples in the continuation paper.**

***Keywords*—formal languages, iterations of languages, morphisms, finite automata, groupoids, semigroups, algorithms.**

## References

[1] Melnikov B. *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).

[2] Melnikov B. *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).

[3] Melnikov B., Melnikova 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., Melnikova 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] Skornyakov L. (Ed.) *General Algebra. Vol. 2*. – Moscow, Nauka. – 1991. – 480 p. (in Russian).

[6] Melnikov B. *$2\omega$-finite automata and sets of obstructions of their language* // The Korean Journal of Computational and Applied Mathematics (Journal of Applied Mathematics and Computing). – 2000. – Vol. 6. No. 3. – P. 565–574.

[7] Lyapin E. *Semigrioups*. – Moscow, Fizmatlit. – 1960. – 592 p. (in Russian).

[8] Birkhoff G., Bartee T. *Modern Applied Algebra*. – NY, McGraw-Hill. – 1970. – 431 p.

[9] Lallement G. *Semigroups and Combinatorial Applications*. – NY, John Wiley & Sons. – 1979. – 376 p.

[10] Salomaa A. *Jewels of Formal Language Theory*. – Rockville, Maryland, Computer Science Press. – 1981. – 144 p.

[11] Pin J.-E. *Mathematical Foundations of Automata Theory*. – Berlin, Springer-Verlag. – 2012. – 310 p.

[12] Melnikov B., Melnikova A. *The use of petal finite automata to verify the fulfillment of a special case of the Zyu hypothesis (for a given finite language)* // International Journal of Open Information Technologies. – 2023. – Vol. 11. No. 3. – P. 1–11 (in Russian).

[13] Melnikov B., Dolgov V. *Simplified regular languages and a special equivalence relation on a class of regular languages. Part I* // International Journal of Open Information Technologies. – 2022. – Vol. 10. No. 9. – P. 12–20 (in Russian).

[14] Melnikov B. *A polynomial algorithm for constructing the optimal inverse morphism* // International Journal of Open Information Technologies. – 2023. – Vol. 11. No. 6. – P. 1–10 (in Russian).

MENG Lingqian,
Graduate student of Shenzhen MSU – BIT University, China
(`http://szmsubit.ru/`),
email$_1$: `1120190010@smbu.edu.cn`,
email$_2$: `kostya45e@mail.ru`,
ORCID: `orcidID=0009-0009-2933-5684`.

Boris MELNIKOV,
Professor of Shenzhen MSU – BIT University, China
(`http://szmsubit.ru/`),
email$_1$: `bormel@smbu.edu.cn`,
email$_2$: `bf-melnikov@yandex.ru`,
mathnet.ru: `personid=27967`,
elibrary.ru: `authorid=15715`,
scopus.com: `authorId=55954040300`,
ORCID: `orcidID=0000-0002-6765-6800`.