Том 3, Выпуск 2 УДК 519.713
Стр. 32–57 (2025) MSC 68Q45

Связь регулярности двумерных и соответствующих им одномерных языков

Н.Н. Корнеева, Л.А. Гизатуллина

Аннотация. Доказаны соотношения между регулярностью двумерных и одномерных языков. Каждому двумерному языку ставятся в соответствие две последовательности одномерных языков, соответствующие строкам и столбцам двумерного языка. Для каждого из приведенных условий доказано существование как регулярных, так и нерегулярных двумерных языков: все строчные и все столбцовые языки регулярны; все строчные языки регулярны, все столбцовые языки нерегулярны; все столбцовые языки регулярны, все строчные языки нерегулярны; и все строчные, и все столбцовые языки нерегулярны.

Ключевые слова: регулярный язык, двумерный язык, конечный автомат, двумерный онлайн-автомат тесселяции.

DOI: 10.26907/2949-3919.2025.2.32-57

Введение

В данной работе изучается связь регулярности двумерных языков и одномерных языков, сопоставленных каждой строке и каждому столбцу двумерного языка. Двумерный язык – это язык, словами которого являются прямоугольные таблицы символов некоторого алфавита, т. е. двумерные слова. Каждому двумерному слову естественным образом сопоставляется некоторое множество одномерных слов, а именно, каждая строка и каждый столбец двумерного слова образуют обычные одномерные слова над тем же алфавитом. Поэтому каждому двумерному языку можно сопоставить две последовательности языков: первую образуют языки, состоящие из слов, соответствующих i-ым строкам двумерных слов, при $i = 1, 2, \ldots$, вторую – языки, состоящие из слов, соответствующих j-ым столбцам двумерных слов, при $j = 1, 2, \ldots$. В работе исследуется вопрос связи регулярности/нерегулярности двумерного языка и регулярности/нерегулярности одномерных языков, сопоставленных строкам и столбцам двумерного языка.

Напомним, что регулярный одномерный язык – это язык, который распознается детерминированным или недетерминированным конечным автоматом, или, эквивалентно, язык, заданный регулярным выражением.

Поступила: 19.09.2024. Принята: 02.07.2025. Опубликована: 16.07.2025.

Аналогично, с незначительными отличиями, можно определить и регулярные двумерные языки.

Один из способов определения регулярных двумерных языков – это их задание при помощи двумерных регулярных выражений. В этом случае возникают две возможные частичные операции конкатенации и итерации Клини, которые в работе называются горизонтальная и вертикальная конкатенация (в англоязычной литературе – column и row concatenation соответственно [1]) и горизонтальная и вертикальная итерация Клини (соответственно, column и row closure [1]). Двумерное регулярное выражение – это выражение, которое строится по индукции из атомарных выражений, т. е. $\emptyset$ и символов алфавита, при помощи операций горизонтальной и вертикальной конкатенации, горизонтальной и вертикальной итерации, объединения, пересечения и дополнения, см. [1]. В зависимости от выбора операций, с помощью которых строятся выражения, определяются различные классы языков.

Другой способ – определить регулярные двумерные языки как языки, распознаваемые конечными автоматами, работающими на двумерных объектах. В литературе предложено несколько типов таких автоматов. Первая модель двумерного автомата была предложена в [2] – это четырехсторонний автомат (four-way automata), в [3] предложена модель двумерного автомата, названная двумерный онлайн-автомат тесселяции (two-dimensional on-line tessellation automaton), которая является частным случаем клеточного автомата. В работах [3, 4] полностью описаны соотношения между классами языков, распознаваемых четырехсторонними автоматами и двумерными онлайн-автоматами тесселяции:

1) класс языков, распознаваемых детерминированными четырехсторонними автоматами, является собственным подмножеством класса языков, распознаваемых недетерминированными четырехсторонними автоматами, который является собственным подмножеством класса языков, распознаваемых двумерными недетерминированными онлайн-автоматами тесселяции;
2) класс языков, распознаваемых двумерными детерминированными онлайн-автоматами тесселяции, является собственным подмножеством класса языков, распознаваемых двумерными недетерминированными онлайн-автоматами тесселяции.

В дальнейшем вводились и другие типы автоматов, работающих на двумерных объектах. Приведем еще две модели двумерных автоматов. С одной стороны, для однозначной возможности распознавания слов двумерного языка определяется двумерный однозначный онлайн-автомат тесселяции (two-dimensional unambiguous on-line tessellation acceptor, [5]). Класс языков, распознаваемых данным типом двумерных автоматов, лежит строго между классами языков, распознаваемых двумерными детерминированными и двумерными недетерминированными онлайн-автоматами тесселяции [5]. С другой стороны, в [6] введено обобщение понятия двумерного недетерминированного онлайн-автомата тесселяции – двумерный приближенный онлайн-автомат тесселяции (two dimensional rough online tesselation automaton) – с использованием приближенных множеств в области значений его функции перехода. Очевидно, что класс языков, распознаваемых двумерными недетерминированными онлайн-автоматами тесселяции является подмножеством класса языков, распознаваемых двумерными приближенными онлайн-автоматами тесселяции (достаточно взять в качестве используемого отношения эквивалентности – отношение равенства). Однако вопрос о строгом включении или равенстве двух классов остается открытым, так как в указанной статье он не рассматривался.

Двумерный язык называется регулярным (см., например, [1]), если он распознается двумерным недетерминированным онлайн-автоматом тесселяции или, что эквивалентно, если он является проекцией языка, соответствующего регулярному выражению, в построении которого не участвует операция дополнения. Приведенное определение двумерного регулярного языка эквивалентно еще двум характеризациям двумерных языков, которые являются обобщениями известных характеризаций одномерных регулярных языков:

1) двумерные языки, распознаваемые при помощи плиточных систем (tiling system), см. [7, 8]. Этот подход является обобщением на двумерный случай характеризации одномерных регулярных языков как проекций локальных языков (см., например, [9]);
2) двумерные языки, определимые в экзистенциальной монадической логике второго порядка [10]. Это обобщение на двумерный случай результата Бюхи для одномерных регулярных языков [11].

В силу приведенных выше эквивалентных определений двумерных регулярных языков подход, основанный на двумерных недетерминированных онлайн-автоматах тесселяции, по сравнению с другими классами автоматов, кажется авторам наиболее актуальным с точки зрения установления соотношений между регулярностью двумерного языка и регулярностью соответствующих ему одномерных языков. В данной работе будем придерживаться именно этого определения регулярности двумерного языка. Однако, в большинстве случаев, для доказательства достаточно будет двумерных детерминированных онлайн-автоматов тесселяции.

В разделе 1 приводятся основные определения и обозначения, используемые в работе. В разделе 2 приводятся несколько алгоритмов построения регулярных двумерных языков из регулярных одномерных языков. В частности, в этом разделе доказано существование регулярных двумерных языков, у которых или все строчные, или все столбцовые языки регулярны. Причем в некоторых случаях в построенных двумерных языках также регулярны и все столбцовые или, соответственно, все строчные языки. Таким образом, в этом разделе доказано существование регулярного двумерного языка с регулярными как строчными, так и столбцовыми языками. В разделе 3 доказывается существование регулярных двумерных языков, у которых строчные и/или столбцовые языки нерегулярны. В разделе 4 приводятся примеры нерегулярных двумерных языков таких, что соответствующие им строчные и/или столбцовые языки могут быть и регулярными, и нерегулярными. В заключении в общем виде формулируются основные результаты работы.

1. Основные определения и обозначения

Приведем определения двумерного языка, операций на двумерных языках, которые являются естественными обобщениями аналогичных операций в одномерном случае. Эти определения можно найти, например, в обзорной статье [1].

Пусть Σ – конечный алфавит, т. е. конечное непустое множество символов (букв). Символы алфавита будем обозначать малыми латинскими буквами x, y, z, $x_1, x_2, \ldots$.

Определение 1. Двумерное слово p над алфавитом Σ – это прямоугольный массив элементов Σ, т. е.

$$p = \boxed{\begin{array}{ccc} x_{11} & \ldots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \ldots & x_{mn} \end{array}},$$

где $x_{ij} \in \Sigma$ ($i = \overline{1.m}$, $j = \overline{1.n}$).

Размером двумерного слова p считается пара (m, n), где m – число строк, n – число столбцов. Если двумерное слово имеет одну строку, то получаем обычное “одномерное” слово. Двумерное слово размера $(0, 0)$ называется пустым и обозначается λ (пустое одномерное слово также будем обозначать символом λ). Считается, что двумерные слова размера $(0, n)$ и $(m, 0)$, где $n > 0$, $m > 0$, не определены.

Множество всех двумерных слов размера (m, n) над алфавитом Σ обозначается $\Sigma^{m\times n}$, множество всех двумерных слов над алфавитом Σ – Σ^{**} (напомним, что в одномерном случае, множество всех одномерных слов над алфавитом Σ обозначается Σ^*, а множество всех непустых одномерных слов – Σ^+). Двумерный язык над Σ есть подмножество Σ^{**}.

Пусть L – двумерный язык над алфавитом Σ. Обозначим через $L_{(i)}$, $i \geq 1$, одномерный язык, состоящий из одномерных слов, находящихся в i-ой строке двумерных слов языка L; через $L^{(j)}$, $j \geq 1$, – одномерный язык, состоящий из одномерных слов, находящихся в j-ом столбце двумерных слов языка L. Таким образом, каждому двумерному языку сопоставляются две последовательности одномерных языков: $(L_{(i)})_{i\geq 1}$ и $(L^{(j)})_{j\geq 1}$.

Пусть p – двумерное слово над алфавитом Σ, $x \in \Sigma$. Как и в одномерном случае, через $|p|_x$ будем обозначать количество букв x в слове p.

Далее приведем определение частичных операций для двумерных слов.

Пусть p и q – двумерные слова над алфавитом Σ размера (m, n) и (m', n') соответственно, где $m, n, m', n' > 0$, т. е.

$$p = \boxed{\begin{array}{ccc} x_{11} & \ldots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \ldots & x_{mn} \end{array}}, \quad q = \boxed{\begin{array}{ccc} y_{11} & \ldots & y_{1n'} \\ \vdots & \ddots & \vdots \\ y_{m'1} & \ldots & y_{m'n'} \end{array}}.$$

Определение 2. Горизонтальная конкатенация непустых двумерных слов p и q – это частичная операция, определенная в случае $m = m'$ и сопоставляющая p и q двумерное слово:

$$p \oplus q = \begin{array}{|ccc|ccc|} \hline x_{11} & \ldots & x_{1n} & y_{11} & \ldots & y_{1n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{m1} & \ldots & x_{mn} & y_{m'1} & \ldots & y_{m'n'} \\ \hline \end{array}.$$

Определение 3. Вертикальная конкатенация непустых двумерных слов p и q – это частичная операция, определенная в случае $n = n'$ и сопоставляющая p и q двумерное слово:

$$p \ominus q = \begin{array}{|ccc|} \hline x_{11} & \ldots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \ldots & x_{mn} \\ \hline y_{11} & \ldots & y_{1n'} \\ \vdots & \ddots & \vdots \\ y_{m'1} & \ldots & y_{m'n'} \\ \hline \end{array}.$$

Кроме того, считается, что горизонтальная и вертикальная конкатенации (не пустого или пустого) двумерного слова p и пустого двумерного слова λ всегда определены и $p \oplus \lambda = \lambda \oplus p = p$ и $p \ominus \lambda = \lambda \ominus p = p$, т.е. λ является нейтральным элементом для этих операций.

Естественным образом эти операции переносятся на двумерные языки. Пусть L_1 и L_2 – двумерные языки над алфавитом Σ.

Определение 4. Горизонтальная конкатенация двумерных языков L_1 и L_2 – это двумерный язык

$$L_1 \oplus L_2 = \{p \oplus q \mid p \in L_1, q \in L_2\}.$$

Определение 5. Вертикальная конкатенация двумерных языков L_1 и L_2 – это двумерный язык

$$L_1 \ominus L_2 = \{p \ominus q \mid p \in L_1, q \in L_2\}.$$

По аналогии с “одномерным” случаем можно рассмотреть горизонтальные и вертикальные степени языка и горизонтальную и вертикальную итерации языка. Очевидно, что

$$L^{n\oplus} = L \oplus L^{(n-1)\oplus},$$
$$L^{n\ominus} = L \ominus L^{(n-1)\ominus},$$

причем $L^{1\oplus} = L^{1\ominus} = L$, $L^{0\oplus} = L^{0\ominus} = \{\lambda\}$.

Пусть теперь L – двумерный язык над алфавитом Σ.

Определение 6. Горизонтальная итерация двумерного языка L – это двумерный язык

$$L^{*\oplus} = \bigcup_{i \geq 0} L^{i\oplus}.$$

Определение 7. Вертикальная итерация двумерного языка L – это двумерный язык

$$L^{*\ominus} = \bigcup_{i \geq 0} L^{i\ominus}.$$

Следующая операция по двум одномерным языкам строит двумерный язык. Пусть L_1 и L_2 – одномерные языки над алфавитом Σ.

Определение 8. Сочетание (row-column combination) одномерных языков L_1 и L_2 – это двумерный язык, обозначаемый $L_1 \oplus L_2$, состоящий из двумерных слов $p \in \Sigma^{**}$ таких, что каждая строка слова p принадлежит L_1, а каждый столбец слова p принадлежит L_2.

Таким образом, $(L_1 \oplus L_2)_{(i)} \subseteq L_1$ и $(L_1 \oplus L_2)^{(j)} \subseteq L_2$ при любых $i, j \geq 1$.

В данной работе будут изучаться регулярные двумерные языки, т. е. такие двумерные языки, которые распознаются двумерными недетерминированными онлайн-автоматами тесселяции. Приведем определение автомата, следуя [1] (оно незначительно отличается от исходного определения [3]), и опишем его работу.

Определение 9. Двумерный недетерминированный (детерминированный) онлайн-автомат тесселяции – это совокупность $A = (\Sigma, S, I, F, \delta)$, где Σ – конечный алфавит, S – конечное множество состояний, $I \subseteq S$ $(I = \{s_0\} \subseteq S)$ – множество начальных состояний (выделенное начальное состояние), $F \subseteq S$ – множество допускающих состояний, $\delta : S \times S \times \Sigma \to 2^S$ $(\delta : S \times S \times \Sigma \to S)$ – функция перехода.

В данной работе не рассматриваются другие автоматы, работающие на двумерных словах, поэтому вместо "двумерный онлайн-автомат тесселяции" будем для краткости писать просто "двумерный автомат", уточняя при необходимости детерминированный или недетерминированный. Также сокращенно будем обозначать двумерный недетерминированный (детерминированный) онлайн-автомат тесселяции так, как принято в англоязычной литературе, – 2ota (2dota).

Рассмотрим работу лишь двумерного детерминированного онлайн-автомата тесселяции (в недетерминированном случае описание аналогичное, но необходимо работать не с состояниями, а с подмножествами состояний). Также как при работе на "одномерном" слове обычный "одномерный" детерминированный конечный автомат проходит последовательность состояний, так двумерный детерминированный онлайн-автомат тесселяции при работе на двумерном слове проходит прямоугольную таблицу состояний.

Пусть на вход двумерного детерминированного онлайн-автомата тесселяции A подано двумерное слово p размера (m, n), ограниченное специальным символом $\#$, т. е. в (i, j)-клетку прямоугольной таблицы записывается (i, j)-символ слова p – x_{ij} $(i = \overline{1.m}$, $j = \overline{1.n})$, а в 0 и $m+1$ строку и 0 и $n+1$ столбец – символы $\#$:

$$\boxed{\begin{array}{ccc} \# & \# & \# \\ \# & p & \# \\ \# & \# & \# \end{array}} = \boxed{\begin{array}{ccccc} \# & \# & \dots & \# & \# \\ \# & x_{11} & \dots & x_{1n} & \# \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \# & x_{m1} & \dots & x_{mn} & \# \\ \# & \# & \dots & \# & \# \end{array}}.$$

Двумерный детерминированный онлайн-автомат тесселяции A каждой (i,j)-позиции слова p сопоставляет некоторое состояние из множества S согласно правилам, определяемым функцией перехода δ. В начальный момент времени двумерный автомат находится в начальном состоянии s_0 (в случае двумерного недетерминированного автомата – в одном из начальных состояний) во всех позициях нулевой строки и нулевого столбца: $s_{i,0} = s_0$ при $i > 0$ и $s_{0,j} = s_0$ при $j > 0$. Действие функции перехода при чтении символа x_{ij} определяется состояниями, в которых находится двумерный автомат в $(i-1)$-ой строке и j-ом столбце $s_{i-1,j}$ и в i-ой строке и $(j-1)$-ом столбце $s_{i,j-1}$:

$$\begin{array}{c|cccccc} & s_0 & \dots & s_0 & s_0 & \dots & s_0 \\ \hline s_0 & & & & & & \\ \vdots & & & & & & \\ s_0 & & & & s_{i-1,j} & & \\ s_0 & & & s_{i,j-1} & x_{i,j} & & \\ \vdots & & & & & & \\ s_0 & & & & & & \end{array}.$$

Таким образом, в момент времени $t = 1$ двумерный автомат A прочитает символ x_{11} и сопоставит $(1,1)$-позиции состояние $\delta(s_0, s_0, x_{11}) = s_{11}$. В момент времени $t = 2$ двумерный автомат A одновременно читает символы x_{12} и x_{21} и сопоставляет $(1,2)$ и $(2,1)$-позициям состояния $\delta(s_0, s_{11}, x_{12}) = s_{12}$ и $\delta(s_{11}, s_0, x_{21}) = s_{21}$ соответственно. Двумерный автомат продолжает читать двумерное слово p двигаясь по побочным диагоналям, до тех пор пока в момент времени $t = m + n - 1$ не сопоставит позиции (m,n) состояние $s_{m,n}$.

Если полученное состояние $s_{m,n}$ допускающее, то двумерный автомат A распознает двумерное слово p, в противном случае – не распознает. В случае двумерного недетерминированного онлайн-автомата тесселяции, автомат A распознает двумерное слово p, если существует хотя бы один путь двумерного автомата A на слове p такой, что состояние $s_{m,n}$ – допускающее.

Также будем считать, что если $s_0 \in F$ (одно из начальных состояний двумерного недетерминированного автомата принадлежит F), то двумерный автомат A распознает пустое двумерное слово λ (это допущение согласуется с исходным определением двумерного онлайн-автомата тесселяции [3], согласно которому двумерный автомат в начальный момент времени находится в начальном состоянии s_0 (в недетерминированном случае, в одном из начальных состояний) во всех (i,j)-клетках прямоугольной таблицы, где записаны символы двумерного слова p).

Язык L распознается двумерным детерминированным онлайн-автоматом тесселяции $A = (\Sigma, S, s_0, F, \delta)$, если $p \in L \Leftrightarrow \delta(s_0, s_0, p) \in F$, и распознается двумерным недерминированным онлайн-автоматом тесселяции $A = (\Sigma, S, I, F, \delta)$, если $p \in L \Leftrightarrow \delta(s_0, s_0, p) \cap F \neq \emptyset$ для некоторого $s_0 \in I$, т. е. для некоторого $s_0 \in I$ хотя бы один путь $\delta(s_0, s_0, p)$ двумерного автомата A на слове p приводит в допускающее состояние.

Классы двумерных языков, распознаваемых двумерными недетерминированными и детерминированными онлайн-автоматами тесселяции, обозначаются $\mathcal{L}(2\mathrm{ota})$ и $\mathcal{L}(2\mathrm{dota})$ соответственно, также будем писать $\mathcal{L}_\Sigma(2\mathrm{ota})$ и $\mathcal{L}_\Sigma(2\mathrm{dota})$, если необходимо указать алфавит, над которым рассматриваются языки.

Определение 10. Двумерный язык L называется регулярным, если он распознается двумерным недетерминированным онлайн-автоматом тесселяции.

В литературе в зависимости от используемого определения регулярности двумерного языка используют разные обозначения для данного класса, однако для четырех приведенных во введении эквивалентных определений также используется общее обозначение REC (recognizable two-dimensional languages, см. [1]). В данной работе необходимо отличать какой “размерности” регулярный язык рассматривается, поэтому класс одномерных регулярных языков над алфавитом Σ будем обозначать $REG_1(\Sigma)$ или REG_1, если алфавит понятен из контекста, а класс двумерных регулярных языков – $REG_2(\Sigma)$ или REG_2. Таким образом, $REC = REG_2 = \mathcal{L}(2\mathrm{ota})$.

2. Регулярные двумерные языки, построенные из регулярных одномерных языков

Цель данного раздела – привести несколько конструкций построения регулярных двумерных языков из регулярных одномерных языков. Однако приводимые здесь результаты тесно связаны с вопросами замкнутости классов двумерных языков, распознаваемых различными типами двумерных автоматов, относительно приведенных в разделе 1 языковых операций. Известно, например, что

1) классы языков $\mathcal{L}(4\mathrm{dfa})$ и $\mathcal{L}(4\mathrm{fa})$, распознаваемых четырехсторонними детерминированными и недетерминированными автоматами, не замкнуты относительно горизонтальной и вертикальной конкатенации и горизонтальной и вертикальной итерации [12];
2) класс языков $REG_2 = \mathcal{L}(2\mathrm{ota})$ замкнут относительно горизонтальной и вертикальной конкатенации и горизонтальной и вертикальной итерации [1];
3) класс языков $\mathcal{L}(2\mathrm{uota})$, распознаваемых двумерными однозначными онлайн-автоматами тесселяции, не замкнут относительно горизонтальной и вертикальной конкатенации и горизонтальной и вертикальной итерации [5];
4) класс языков $\mathcal{L}(2\mathrm{rota})$, распознаваемых двумерными приближенными онлайн-автоматами тесселяции, замкнут относительно горизонтальной и вертикальной

конкатенации [6], вопрос замкнутости относительно горизонтальной и вертикальной итерации в статье не рассматривался.

Для класса $\mathcal{L}(2\mathrm{dota})$ подобные результаты, по предположению авторов, не верны, однако если применять указанные операции к одномерным языкам (для удобства на них наложены дополнительные условия, значение которых понятно из замечания 13), то соответствующие результаты справедливы (см. теоремы 11 и 17).

Теорема 11. *Если* $L_1, L_2, \ldots, L_k \in REG_1(\Sigma)$, $\lambda \in L_1 \cap L_2 \cap \cdots \cap L_k$ *или* $\lambda \notin L_1 \cup L_2 \cup \cdots \cup L_k$, *то* $L = L_1 \ominus L_2 \ominus \cdots \ominus L_k \in \mathcal{L}_\Sigma(2\mathrm{dota})$.

Доказательство. Доказывать теорему будем для случая $k = 2$, так как в общем случае доказательство аналогично. Пусть $L = L_1 \ominus L_2$. Непустые двумерные слова языка L будут содержать ровно две строки. Язык L будет содержать пустое двумерное слово λ только в случае, если оба языка L_1 и L_2 содержат пустые одномерные слова.

Поскольку L_1 и L_2 – регулярные одномерные языки, они распознаются детерминированными конечными автоматами. Пусть $A_1 = (\Sigma, S_1, s_0^1, F_1, \delta_1)$ и $A_2 = (\Sigma, S_2, s_0^2, F_2, \delta_2)$ являются детерминированными конечными автоматами, распознающими языки L_1 и L_2 соответственно. Будем считать, что $S_1 \cap S_2 = \emptyset$. Построим двумерный детерминированный автомат $A = (\Sigma, S, s_0, F, \delta)$, распознающий язык $L = L_1 \ominus L_2$. Множеством состояний автомата будет $S = (S_1 \cup S_2) \times \{0,1\} \cup \{(s_0, \gamma), (s', 0)\}$, где $s_0, s' \notin S_1 \cup S_2$, $\gamma \in \{0, 1\}$. Значение γ зависит от принадлежности/не принадлежности пустых одномерных слов языкам L_1 и L_2: если $\lambda \in L_1 \cap L_2$, то $\gamma = 1$, если $\lambda \notin L_1 \cap L_2$, то $\gamma = 0$.

Определим функцию перехода (во всех приводимых ниже формулах $x \in \Sigma$, $s_1 \in S_1$, $s_2 \in S_2$, $\alpha, \beta \in \{0, 1\}$, χ_T – характеристическая функция множества T):

$\delta((s_0, \gamma), (s_0, \gamma), x) = (\delta_1(s_0^1, x), \chi_{F_1}(\delta_1(s_0^1, x)))$,
$\delta((s_0, \gamma), (s_1, \alpha), x) = (\delta_1(s_1, x), \chi_{F_1}(\delta_1(s_1, x)))$,
$\delta((s_1, \alpha), (s_0, \gamma), x) = (\delta_2(s_0^2, x), \alpha \cdot \chi_{F_2}(\delta_2(s_0^2, x)))$,
$\delta((s_1, \alpha), (s_2, \beta), x) = (\delta_2(s_2, x), \alpha \cdot \chi_{F_2}(\delta_2(s_2, x)))$,
$\delta((s_2, \alpha), (s_0, \gamma), x) = (s', 0)$,
$\delta((s_2, \alpha), (s', 0), x) = (s', 0)$,
$\delta((s', 0), (s_0, \gamma), x) = (s', 0)$,
$\delta((s', 0), (s', 0), x) = (s', 0)$.

Первые две команды отвечают работе двумерного автомата на первой строке двумерного слова, причем вторая компонента состояния говорит о принадлежности/непринадлежности полученного состояния множеству допускающих состояний F_1. Третья и четвертая команды отвечают работе двумерного автомата на второй строке двумерного слова, причем вторая компонента состояния говорит о принадлежности/непринадлежности полученного состояния множеству допускающих состояний F_2, при условии, что соответствующее состояние над ним принадлежит F_1. Иными словами, во второй строке вторая компонента будет равна единице, если прочитанное к этому моменту слово второй строки принадлежит L_2, а соответствующее ему слово первой строки принадлежит L_1. Пятая и шестая команды отвечают работе двумерного автомата на третьей строке двумерного слова, последние две команды – на четвертой и последующих строках.

Определим множество допускающих состояний: $F = \{(s, 1) \mid s \in S_2 \cup \{s_0\}\}$. Таким образом, возможны два случая:

1) если оба языка L_1 и L_2 содержат пустое одномерное слово, то

$$F = \{(s_2, 1) \mid s_2 \in S_2\} \cup \{(s_0, 1)\};$$

2) если оба языка L_1 и L_2 не содержат пустого одномерного слова, то

$$F = \{(s_2, 1) \mid s_2 \in S_2\},$$

так как $\gamma = 0$.

Покажем, что построенный двумерный детерминированный автомат A распознает язык $L = L_1 \ominus L_2$. Очевидно, что на пустом двумерном слове двумерный автомат A работает корректно в обоих случаях, поэтому рассмотрим лишь случай непустых двумерных слов. Непустое двумерное слово должно распознаваться построенным двумерным автоматом тогда и только тогда, когда оно имеет ровно две строки и слово в первой строке принадлежит L_1, а слово во второй строке принадлежит L_2.

Пусть $p = p_1 \ominus p_2 \ominus \cdots \ominus p_l$ и $\delta((s_0, \gamma), (s_0, \gamma), p) = (s, \xi)$, где $(s, \xi) \in S$, $\xi \in \{0, 1\}$.

Если $l = 1$, то $s \in S_1$ и $(s, \xi) \notin F$, следовательно, $p \notin L$.

Если $l \geq 3$, то $s = s'$ и $(s, \xi) = (s', 0) \notin F$, следовательно, $p \notin L$.

Если $l = 2$ и $p = p_1 \ominus p_2$: $(s, \xi) \in F$ тогда и только тогда, когда $s \in S_2, \xi = 1$, причем $s = \delta_2(s_0^2, p_2)$, а $\xi = \alpha \cdot \chi_{F_2}(s)$, где $\alpha = \chi_{F_1}(\delta_1(s_0^1, p_1))$. Приведенные условия эквивалентны следующим: $\chi_{F_2}(\delta_2(s_0^2, p_2)) = 1$ и $\chi_{F_1}(\delta_1(s_0^1, p_1)) = 1$ или, другими словами, $\delta_2(s_0^2, p_2) \in F_2$ и $\delta_1(s_0^1, p_1) \in F_1$. Таким образом

$$\delta((s_0, \gamma), (s_0, \gamma), p) \in F \;\Leftrightarrow\; p_1 \in L_1, p_2 \in L_2 \;\Leftrightarrow\; p = p_1 \ominus p_2 \in L_1 \ominus L_2. \qquad \square$$

Замечание 12. Если рассматривать только состояния, достижимые из начального, то обозначение состояний в доказательстве теоремы 11 можно было упростить, отождествив состояния из $S_1 \times \{0, 1\}$ с соответствующими состояниями из S_1, так как из двух состояний $(s_1, 0)$ и $(s_1, 1)$ для $s_1 \in S_1$ в двумерном автомате A будет лишь одно; (s_0, γ) – с s_0, поскольку из двух состояний $(s_0, 0)$ и $(s_0, 1)$ будет лишь одно; и $(s', 0)$ – с s'. Последнее состояние $(s', 0)$ – это состояние, которое обычно в теории автоматов называют мусорным состоянием. Однако приведенные в теореме более громоздкие обозначения удобны для обоснования корректной работы построенного двумерного автомата.

Замечание 13. Очевидно, что для языка $L = L_1 \ominus L_2 \ominus \cdots \ominus L_k \subseteq \Sigma^{**}$, построенного в теореме 11, выполнены соотношения: $L_{(1)} \subseteq L_1$, $L_{(2)} \subseteq L_2$, $\ldots$, $L_{(k)} \subseteq L_k$. Ниже доказано, что одномерные языки $L_{(1)}, L_{(2)}, \ldots, L_{(k)}$ также регулярны и

$$L_1 \ominus L_2 \ominus \cdots \ominus L_k = L_{(1)} \ominus L_{(2)} \ominus \cdots \ominus L_{(k)}.$$

Напомним стандартное определение гомоморфизма. Отображение $h : \Sigma^* \to \Sigma'^*$ называется гомоморфизмом (морфизмом), если $h(pq) = h(p)h(q)$ для любых $p, q \in \Sigma^*$. Очевидно, что достаточно задавать гомоморфизм h на однобуквенных словах, так как если $p = x_1x_2\ldots x_l$, то $h(p) = h(x_1)h(x_2)\ldots h(x_l)$.

Теорема 14. *Если* $L_1, L_2, \ldots, L_k \in REG_1(\Sigma)$, $\lambda \in L_1 \cap L_2 \cap \cdots \cap L_k$ *или* $\lambda \notin L_1 \cup L_2 \cup \cdots \cup L_k$, *и* $L = L_1 \ominus L_2 \ominus \cdots \ominus L_k$, *тогда* $L_{(1)}, L_{(2)}, \ldots, L_{(k)} \in REG_1(\Sigma)$ *и*

$$L_1 \ominus L_2 \ominus \cdots \ominus L_k = L_{(1)} \ominus L_{(2)} \ominus \cdots \ominus L_{(k)}.$$

Доказательство. Пусть $\Sigma' = \{a\}$ – однобуквенный алфавит. Рассмотрим гомоморфизм $h : \Sigma^* \to \Sigma'^*$, который каждую букву алфавита Σ отображает в букву $a \in \Sigma'$. Известно (см., например, [13]), что гомоморфизм и обратный гомоморфизм регулярного одномерного языка являются регулярными одномерными языками. Отсюда следует, что $h(L_1), h(L_2), \ldots, h(L_k) \in REG_1(\Sigma')$. Тогда $L' = h(L_1) \cap h(L_2) \cap \cdots \cap h(L_k) \in REG_1(\Sigma')$ как конечное пересечение регулярных одномерных языков. Следовательно, $h^{-1}(L') \in REG_1(\Sigma)$. Слова одномерного языка $h^{-1}(L')$ имеют такую длину, что существует двумерное слово, принадлежащее $L_1 \ominus L_2 \ominus \cdots \ominus L_k$, с соответствующим числом столбцов.

Докажем, что $L_{(i)} = L_i \cap h^{-1}(L')$. Действительно, $p_i \in L_{(i)}$ тогда и только тогда, когда существуют слова $p_1 \in L_{(1)}, \ldots, p_{i-1} \in L_{(i-1)}$, $p_{i+1} \in L_{(i+1)}, \ldots, p_k \in L_{(k)}$ такие, что

$$p_1 \ominus p_2 \ominus \cdots \ominus p_i \ominus \cdots \ominus p_k \in L.$$

Тогда очевидно, что $h(p_1) = h(p_2) = \cdots = h(p_i) = \cdots = h(p_k)$, следовательно, $h(p_i) \in L'$. Поскольку $L_{(i)} \subseteq L_i$, то из $p_i \in L_{(i)}$ следует $p_i \in L_i \cap h^{-1}(L')$. Обратно, если $p_i \in L_i \cap h^{-1}(L')$, то существуют слова $p_1 \in L_1, \ldots, p_{i-1} \in L_{i-1}$, $p_{i+1} \in L_{i+1}, \ldots, p_k \in L_k$ такие, что

$$h(p_1) = h(p_2) = \cdots = h(p_i) = \cdots = h(p_k),$$

значит, $p_1 \ominus p_2 \ominus \cdots \ominus p_i \ominus \cdots \ominus p_k \in L$, следовательно, $p_i \in L_{(i)}$. Отсюда $L_{(i)} \in REG_1(\Sigma)$ как пересечение регулярных одномерных языков.

Из доказанного выше соотношения следует, что

$$\begin{aligned} L_{(1)} \ominus L_{(2)} \ominus \cdots \ominus L_{(k)} &= (L_1 \cap h^{-1}(L')) \ominus (L_2 \cap h^{-1}(L')) \ominus \cdots \ominus (L_k \cap h^{-1}(L')) \\ &= L_1 \ominus L_2 \ominus \cdots \ominus L_k, \end{aligned}$$

так как для других слов не применима операция вертикальной конкатенации. □

Следствие 15. *Если* $L \in REG_1(\Sigma)$, *то* $L^{k\ominus} \in \mathcal{L}_\Sigma(\text{2dota})$ *для любого* $k \geq 1$. *Более того,* $\left(L^{k\ominus}\right)_{(i)} = L$ *при любом* $i = \overline{1.k}$.

Аналогично можно строить регулярные двумерные языки, в которых каждый столбцовый язык является регулярным одномерным языком. Поскольку обычно в одномерных языках слова записываюся в строчку, введем естественную операцию "транспонирования", чтобы получить одномерные языки, в которых слова записываются в столбец.

Пусть $p = \boxed{\begin{matrix} x_{11} & \dots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \dots & x_{mn} \end{matrix}}$, тогда $p^T = \boxed{\begin{matrix} x_{11} & \dots & x_{m1} \\ \vdots & \ddots & \vdots \\ x_{1n} & \dots & x_{mn} \end{matrix}}$.

Обозначим $L^T = \{p^T \mid p \in L\}$. Легко доказывается следующий результат, из которого потом следуют транспонированные версии теоремы 14 и следствия 15:

Лемма 16.

1) *Если* $L \in \mathcal{L}(\text{2ota})$, *то* $L^T \in \mathcal{L}(\text{2ota})$.
2) *Если* $L \in \mathcal{L}(\text{2dota})$, *то* $L^T \in \mathcal{L}(\text{2dota})$.

Доказательство. Очевидно, что для построения двумерного недетерминированного или детерминированного автомата, распознающего двумерный язык L^T, необходимо "транспонировать" все команды двумерного недетерминированного или детерминированного автомата, распознающего двумерный язык L. Пусть $A = (\Sigma, S, I, F, \delta)$ – двумерный автомат, распознающий L, тогда двумерный автомат $A' = (\Sigma, S, I, F, \delta')$, где $\delta'(s,t,x) = \delta(t,s,x)$ при любых $s,t \in S$, $x \in \Sigma$, распознает L^T. □

Теорема 17. *Если* $L_1, L_2, \dots, L_k \in REG_1(\Sigma)$, $\lambda \in L_1 \cap L_2 \cap \dots \cap L_k$ *или* $\lambda \notin L_1 \cup L_2 \cup \dots \cup L_k$, *то* $L = L_1^T \oplus L_2^T \oplus \dots \oplus L_k^T \in \mathcal{L}_\Sigma(\text{2dota})$, *причем* $L^{(1)}, L^{(2)}, \dots, L^{(k)} \in REG_1(\Sigma)$ *и*

$$L_1^T \oplus L_2^T \oplus \dots \oplus L_k^T = \left(L^{(1)}\right)^T \oplus \left(L^{(2)}\right)^T \oplus \dots \oplus \left(L^{(k)}\right)^T.$$

Следствие 18. *Если* $L \in REG_1(\Sigma)$, *то* $\left(L^T\right)^{k\oplus} \in \mathcal{L}_\Sigma(\text{2dota})$ *для любого* $k \geq 1$. *Более того,* $\left(\left(L^T\right)^{k\oplus}\right)^{(j)} = L$ *при любом* $j = \overline{1.k}$.

В действительности верно также утверждение, обратное к теоремам 14 и 17: каждому регулярному двумерному языку с фиксированным числом строк или столбцов можно сопоставить регулярный одномерный язык над декартовой степенью исходного алфавита.

Теорема 19. *Если* $L \in \mathcal{L}_\Sigma(\text{2ota})$ *и все слова языка* L *содержат ровно* $k \geq 1$ *строк (или* $k \geq 1$ *столбцов), тогда* $L \in REG_1\left((\Sigma^k)^T\right)$ *(соответственно,* $L \in REG_1\left(\Sigma^k\right)$*).*

Доказательство. При $k = 1$ утверждение теоремы очевидно, так как исходный язык L одномерный. Будем доказывать теорему для случая $k = 2$ строк, так как в общем случае и в случае k столбцов доказательство аналогично. Непустые двумерные слова языка L будут содержать ровно две строки (пустых двумерных слов язык L не содержит).

Пусть L распознается двумерным недетерминированным автоматом $A = (\Sigma, S, I, F, \delta)$. Как и в "одномерном" случае, можно преобразовать двумерный автомат A таким образом, чтобы он имел лишь одно начальное состояние, т. е. можно считать, что $I = \{s_0\}$.

Будем рассматривать язык L как одномерный язык над алфавитом $(\Sigma^2)^T$. "Одномерный" недетерминированный конечный автомат $A' = ((\Sigma^2)^T, (S^2)^T, s_0', F', \delta')$ определим следующим образом:

- алфавит $(\Sigma^2)^T = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \middle| \ x, y \in \Sigma \right\}$;
- множество состояний $(S^2)^T = \left\{ \begin{pmatrix} s \\ t \end{pmatrix} \middle| \ s, t \in S \right\}$;
- начальное состояние $s_0' = \begin{pmatrix} s_0 \\ s_0 \end{pmatrix}$;
- множество допускающих состояний $F' = \left\{ \begin{pmatrix} s \\ t \end{pmatrix} \middle| \ s \in S, t \in F \right\}$;
- функция перехода $\delta'\left(\begin{pmatrix} s \\ t \end{pmatrix}, \begin{pmatrix} x \\ y \end{pmatrix} \right) = \left\{ \begin{pmatrix} s' \\ t' \end{pmatrix} \middle| \ s' \in \delta(s_0, s, x), t' \in \delta(s', t, y) \right\}$.

Таким образом построенный одномерный автомат на первой и второй строках "одномерного" слова над алфавитом $(\Sigma^2)^T$ работает как исходный двумерный автомат на соответствующем двумерном слове над алфавитом Σ. Следовательно, A' распознает все слова языка L, рассматриваемые как одномерные слова над алфавитом $(\Sigma^2)^T$, и только их. $\square$

Теорема 20. *Если* $L \in REG_1(\Sigma)$, *то* $L^{*\ominus}, \left(L^T\right)^{*\oplus} \in \mathcal{L}_\Sigma(2\text{dota})$. *Более того,* $\left(L^{*\ominus}\right)_{(i)} = L$ *при любом* $i \geq 1$ *и* $\left(\left(L^T\right)^{*\oplus}\right)^{(j)} = L$ *при любом* $j \geq 1$.

Доказательство. Докажем регулярность первого двумерного языка, регулярность второго двумерного языка будет следовать из леммы 16 и очевидного соотношения $\left(L^{*\ominus}\right)^T = \left(L^T\right)^{*\oplus}$.

Пусть $A = (\Sigma, S, s_0, F, \delta)$ – детерминированный конечный автомат, распознающий язык L. Построим двумерный детерминированный автомат $A' = (\Sigma, S', (s_0', 1), F', \delta')$, распознающий двумерный язык $L^{*\ominus}$, с множеством состояний $S' = S \times \{0, 1\} \cup \{(s_0', 1)\}$, где $s_0' \notin S$, и множеством допускающих состояний $F' = \{(s, 1) \mid s \in F\} \cup \{(s_0', 1)\}$ (в действительности, можно отождествить $(s_0', 1)$ с s_0'). Осталось определить функцию перехода (во всех приводимых ниже формулах $x \in \Sigma$, $s_1, s_2 \in S$, $\alpha \in \{0, 1\}$):

$\delta'((s_0', 1), (s_0', 1), x) = (\delta(s_0, x), \chi_F(\delta(s_0, x)))$,

$\delta'((s_0', 1), (s_1, \alpha), x) = (\delta(s_1, x), \chi_F(\delta(s_1, x)))$,

$\delta'((s_1, \alpha), (s_0', 1), x) = (\delta(s_0, x), \alpha \cdot \chi_F(\delta(s_0, x)))$,

$\delta'((s_1, \alpha), (s_2, \beta), x) = (\delta(s_2, x), \alpha \cdot \chi_F(\delta(s_2, x)))$.

Первые две команды определяют работу двумерного автомата на первой строке двумерного слова, где вторая компонента состояния говорит о принадлежности/непринадлежности полученного состояния множеству допускающих состояний F. Третья и четвертая команды отвечают работе двумерного автомата на второй и последующих строках двумерного слова, причем вторая компонента состояния говорит о принадлежности/непринадлежности полученного состояния множеству допускающих состояний F, при условии, что все полученные строго над ним состояния принадлежат F. Иными словами, вторая компонента будет равна единице, если прочитанное к этому моменту слово принадлежит L, и все слова над ним также принадлежат L.

Покажем, что построенный двумерный детерминированный автомат A' распознает язык $L^{*\ominus}$. Очевидно, что на пустом двумерном слове двумерный автомат A' работает корректно. Рассмотрим работу двумерного автомата A' на непустых двумерных словах.

Пусть $p = p_1 \ominus p_2 \ominus \cdots \ominus p_l$, где $|p_1| = |p_2| = \cdots = |p_l|$, и $\delta'((s_0', 1), (s_0', 1), p) = (s, \xi)$, где $(s, \xi) \in S'$, $\xi \in \{0, 1\}$. Тогда, очевидно, что $s = \delta(s_0, p_l)$ и

$$\xi = \alpha_1 \cdot \alpha_2 \cdots \alpha_{l-1} \cdot \chi_F(s) = \chi_F(\delta(s_0, p_1)) \cdot \chi_F(\delta(s_0, p_2)) \cdots \chi_F(\delta(s_0, p_{l-1})) \cdot \chi_F(\delta(s_0, p_l)).$$

Отсюда видно, что для непустого двумерного слова $(s, \xi) \in F'$ тогда и только тогда, когда $s \in F, \xi = 1$, т. е. тогда и только тогда, когда

$$\chi_F(\delta(s_0, p_1)) = 1, \quad \chi_F(\delta(s_0, p_2)) = 1, \quad \ldots, \quad \chi_F(\delta(s_0, p_l)) = 1$$

или, другими словами, $\delta(s_0, p_1) \in F, \delta(s_0, p_2) \in F, \ldots, \delta(s_0, p_l) \in F$. Последние условия эквивалентны следующим: $p_1 \in L, p_2 \in L, \ldots, p_l \in L$ или $p = p_1 \ominus p_2 \ominus \cdots \ominus p_l \in L^{*\ominus}$.

Справедливость условия $\left(L^{*\ominus}\right)_{(i)} = L$ при любом $i \geq 1$ очевидна. □

Замечание 21. Теоремы 11, 17 и 20 и следствия 15, 18 можно обобщить на случай, когда используемые в них исходные языки не одномерные, а двумерные с фиксированным числом строк или столбцов (доказательства аналогичны доказательствам указанных теорем, только дополнительно нужно считать количество строк или столбцов первого и второго двумерных языков):

1) если $L_1, L_2 \in \mathcal{L}_\Sigma(\text{2dota})$ и все слова двумерного языка L_1 имеют m_1 строк, а все слова двумерного языка L_2 имеют m_2 строк, где $m_1, m_2 \geq 1$, то $L = L_1 \ominus L_2 \in \mathcal{L}_\Sigma(\text{2dota})$;
2) если $L_1, L_2 \in \mathcal{L}_\Sigma(\text{2dota})$ и все слова двумерного языка L_1 имеют n_1 столбцов, а все слова двумерного языка L_2 имеют n_2 столбцов, где $n_1, n_2 \geq 1$, то $L = L_1 \varobar L_2 \in \mathcal{L}_\Sigma(\text{2dota})$;
3) если $L \in \mathcal{L}_\Sigma(\text{2dota})$ и все слова двумерного языка L имеют m строк, где $m \geq 1$, то $L^{k\ominus}, L^{*\ominus} \in \mathcal{L}_\Sigma(\text{2dota})$ для любого $k \geq 1$;
4) если $L \in \mathcal{L}_\Sigma(\text{2dota})$ и все слова двумерного языка L имеют n столбцов, где $n \geq 1$, то $L^{k\varobar}, L^{*\varobar} \in \mathcal{L}_\Sigma(\text{2dota})$ для любого $k \geq 1$.

Таким образом, в указанных частных случаях справедлива замкнутость класса языков $\mathcal{L}(\text{2dota})$ относительно операций горизонтальной и вертикальной конкатенации и горизонтальной и вертикальной итерации.

Замечание 22. В теоремах 11, 14 и 20 (17 и 20) показано, как построить регулярный двумерный язык, каждый строчный (соответственно, столбцовый) язык которого регулярен, но ничего не говорится о регулярности столбцовых (соответственно, строчных) языков. Однако, легко видеть, что соответствующие столбцовые (соответственно, строчные) языки также регулярны, так как состоят из слов длины k или пустого слова, т. е. образуют конечные языки, которые регулярны.

Теорема 23. *Если* $L_1, L_2 \in REG_1(\Sigma)$, *то язык* $L_1 \oplus L_2 \in \mathcal{L}_\Sigma(\text{2dota})$.

Доказательство. Пусть $A_1 = (\Sigma, S_1, s_0^1, F_1, \delta_1)$ и $A_2 = (\Sigma, S_2, s_0^2, F_2, \delta_2)$ – детерминированные конечные автоматы, распознающие языки L_1 и L_2 соответственно. Будем считать, что $S_1 \cap S_2 = \emptyset$. Построим двумерный детерминированный автомат $A = (\Sigma, S, s_0, F, \delta)$, распознающий язык $L = L_1 \oplus L_2$. Множеством состояний автомата будет

$$S = S_1 \times \{0,1\} \times S_2 \times \{0,1\} \cup \{(s_0, \gamma)\},$$

где $s_0 \notin S_1 \cup S_2$, $\gamma \in \{0,1\}$. Значение γ зависит от принадлежности/не принадлежности пустых одномерных слов языкам L_1 и L_2, так как пустое двумерное слово принадлежит L только в случае, когда оба языка L_1 и L_2 содержат пустые одномерные слова. Таким образом, если $\lambda \in L_1 \cap L_2$, то $\gamma = 1$, если $\lambda \notin L_1 \cap L_2$, то $\gamma = 0$ (в действительности, (s_0, γ) можно отождествить с s_0).

Определим функцию перехода (во всех приводимых ниже формулах $x \in \Sigma$, $s_1, s_1' \in S_1$, $s_2, s_2' \in S_2$, $\alpha, \beta, \alpha', \beta' \in \{0,1\}$):

$\delta((s_0,\gamma),(s_0,\gamma),x) = (\delta_1(s_0^1,x), \chi_{F_1}(\delta_1(s_0^1,x)), \delta_2(s_0^2,x), \chi_{F_2}(\delta_2(s_0^2,x)))$,

$\delta((s_0,\gamma),(s_1,\alpha,s_2,\beta),x) = (\delta_1(s_1,x), \chi_{F_1}(\delta_1(s_1,x)), \delta_2(s_0^2,x), \beta\cdot\chi_{F_2}(\delta_2(s_0^2,x)))$,

$\delta((s_1,\alpha,s_2,\beta),(s_0,\gamma),x) = (\delta_1(s_0^1,x), \alpha\cdot\chi_{F_1}(\delta_1(s_0^1,x)), \delta_2(s_2,x), \chi_{F_2}(\delta_2(s_2,x)))$,

$\delta((s_1,\alpha,s_2,\beta),(s_1',\alpha',s_2',\beta'),x) = (\delta_1(s_1',x), \alpha\cdot\chi_{F_1}(\delta_1(s_1',x)), \delta_2(s_2,x), \beta'\cdot\chi_{F_2}(\delta_2(s_2,x)))$.

Первые две команды отвечают работе двумерного автомата на первой строке двумерного слова, третья и четвертая команды отвечают работе двумерного автомата на второй и последующих строках двумерного слова, причем третья команда отвечает работе на первом столбце.

Определим множество допускающих состояний. Возможны два случая:

1) оба языка L_1 и L_2 содержат пустое одномерное слово, тогда

$$F = \{(s_1, 1, s_2, 1) \mid s_1 \in S_1, s_2 \in S_2\} \cup \{(s_0, 1)\}$$

(в этом случае, $\gamma = 1$);

2) хотя бы один из языков L_1 и L_2 не содержат пустого одномерного слова, тогда

$$F = \{(s_1, 1, s_2, 1) \mid s_1 \in S_1, s_2 \in S_2\}$$

(в этом случае, $\gamma = 0$).

Покажем, что построенный двумерный детерминированный автомат A распознает двумерный язык $L = L_1 \oplus L_2$. Очевидно, что на пустом двумерном слове двумерный автомат A работает корректно в обоих случаях, поэтому рассмотрим лишь случай непустых двумерных слов.

Пусть $p = p_1 \ominus p_2 \ominus \cdots \ominus p_l = q_1 \oplus q_2 \oplus \cdots \oplus q_k$ и $\delta((s_0,\gamma),(s_0,\gamma),p) = (s_1,\xi,s_2,\eta)$, где $s_1 \in S_1$, $s_2 \in S_2$, $\xi,\eta \in \{0,1\}$, причем

$$\xi = \alpha_1 \cdot \alpha_2 \cdots \alpha_{l-1} \cdot \chi_{F_1}(s_1) = \chi_{F_1}(\delta_1(s_0^1,p_1)) \cdot \chi_{F_1}(\delta_1(s_0^1,p_2)) \cdots \chi_{F_1}(\delta_1(s_0^1,p_{l-1})) \cdot \chi_{F_1}(\delta_1(s_0^1,p_l)),$$
$$\eta = \beta_1 \cdot \beta_2 \cdots \beta_{k-1} \cdot \chi_{F_2}(s_2) = \chi_{F_2}(\delta_2(s_0^2,q_1)) \cdot \chi_{F_2}(\delta_2(s_0^2,q_2)) \cdots \chi_{F_2}(\delta_2(s_0^2,q_{k-1})) \cdot \chi_{F_2}(\delta_2(s_0^2,q_k)).$$

Тогда для непустого двумерного слова $(s_1,\xi,s_2,\eta) \in F$ тогда и только тогда, когда $s_1 \in S_1$, $s_2 \in S_2$, $\xi = 1$, $\eta = 1$, т. е. тогда и только тогда, когда

$$\chi_{F_1}(\delta_1(s_0^1,p_1)) = 1, \ \ \chi_{F_1}(\delta_1(s_0^1,p_2)) = 1, \ \ \ldots, \ \ \chi_{F_1}(\delta_1(s_0^1,p_l)) = 1$$

и

$$\chi_{F_2}(\delta_2(s_0^2,q_1)) = 1, \ \ \chi_{F_2}(\delta_2(s_0^2,q_2)) = 1, \ \ \ldots, \ \ \chi_{F_2}(\delta_2(s_0^2,q_k)) = 1$$

или, другими словами,

$$\delta_1(s_0^1,p_1) \in F_1, \ \ \delta_1(s_0^1,p_2) \in F_1, \ \ \ldots, \ \ \delta_1(s_0^1,p_l) \in F_1,$$
$$\delta_2(s_0^2,q_1) \in F_2, \ \ \delta_2(s_0^2,q_2) \in F_2, \ \ \ldots, \ \ \delta_2(s_0^2,q_k) \in F_2.$$

Последние условия эквивалентны следующим:

$$p_1 \in L_1, p_2 \in L_1, \ldots, p_l \in L_1 \quad \text{и} \quad q_1 \in L_2, q_2 \in L_2, \ldots, q_k \in L_2,$$

т. е. $p \in L_1 \oplus L_2$. □

Замечание 24. Очевидно, что $(L_1 \oplus L_2)_{(i)} \subseteq L_1$ и $(L_1 \oplus L_2)^{(j)} \subseteq L_2$ при любых $i,j \geq 1$. Однако, в общем случае, $(L_1 \oplus L_2)_{(i)} \neq (L_1 \oplus L_2)_{(i')}$ при $i \neq i'$ и $(L_1 \oplus L_2)^{(j)} \neq (L_1 \oplus L_2)^{(j')}$ при $j \neq j'$. Поэтому языки L_1 и L_2 нельзя заменить на подходящие строчные и столбцовые языки для $L_1 \oplus L_2$, как это делалось в теоремах 14 и 17.

Действительно, пусть $L_1 = aa^* + bb^*, L_2 = a(ba)^*(b+\lambda)$. Тогда $L_1 \oplus L_2$ состоит из двумерных слов вида

$$\boxed{\begin{array}{ccccc} a & a & a & a & \ldots \\ b & b & b & b & \ldots \\ a & a & a & a & \ldots \\ b & b & b & b & \ldots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{array}}\,.$$

Тогда $(L_1 \oplus L_2)_{(2i-1)} = aa^*$, $(L_1 \oplus L_2)_{(2i)} = bb^*$ и $(L_1 \oplus L_2)^{(j)} = a(ba)^*(b+\lambda) = L_2$ при $i,j \geq 1$. Очевидно, что сочетание одномерных языков aa^* и L_2 дает двумерный язык aa^*, а сочетание одномерных языков bb^* и L_2 дает пустой двумерный язык. В обоих случаях, полученные двумерные языки не совпадают с $L_1 \oplus L_2$.

Замечание 25. Для построенного в теореме 23 регулярного двумерного языка $L_1 \oplus L_2$ вопрос о регулярности соответствующих ему строчных и столбцовых одномерных языков остался открытым.

Замечание 26. Принадлежность двумерных языков над алфавитом Σ, указанных в следствиях 15 и 18 и теореме 20, классу $\mathcal{L}_\Sigma(\text{2dota})$ теперь можно получить как следствие теоремы 23.

Действительно, если $L \in REG_1(\Sigma)$, то $L^{k\ominus}, \left(L^T\right)^{k\oplus}, L^{*\ominus}, \left(L^T\right)^{*\oplus} \in \mathcal{L}_\Sigma(\text{2dota})$, так как

$$\begin{aligned} L^{k\ominus} &= L \oplus (\Sigma^k \cup \{\lambda\}), & \left(L^T\right)^{k\oplus} &= \left(\Sigma^k \cup \{\lambda\}\right) \oplus L, \\ L^{*\ominus} &= (L \cup \{\lambda\}) \oplus \Sigma^*, & \left(L^T\right)^{*\oplus} &= \Sigma^* \oplus (L \cup \{\lambda\}). \end{aligned}$$

3. Регулярные двумерные языки, построенные из нерегулярных одномерных языков

Известно, что язык $\{a^n b^n \mid n \geq 1\}$ – нерегулярный одномерный язык (см., например, [13]). Приведем "аналог" указанного языка в двумерном случае. В отличие от одномерного случая он будет регулярным.

Рассмотрим двумерный язык

$$L = \left\{ \begin{array}{|c|c|} \hline p & q \\ \hline q & p \\ \hline \end{array} \;\middle|\; \exists n > 0 \left(p \in \{a\}^{n\times n},\ q \in \{b\}^{n\times n}\right) \right\} \tag{1}$$

над алфавитом $\Sigma = \{a, b\}$.

Предложение 27. $L \in \mathcal{L}(\text{2dota})$.

Доказательство. Построим двумерный детерминированный автомат $A = (\Sigma, S, s_0, F, \delta)$, распознающий двумерный язык (1) над алфавитом $\Sigma = \{a, b\}$, где $S = \{0, 1, \ldots, 13\}$, $s_0 = 0$, $F = \{13\}$. Двумерные слова, принадлежащие двумерному языку (1), состоят из четырех квадратных блоков одинаковых размеров, поэтому двумерный автомат A должен уметь проверять, из каких букв состоит блок, блок – квадратный, размеры разных блоков совпадают.

Определим функцию перехода.

I) На левом верхнем блоке:
$\delta(0, 0, a) = \delta(2, 3, a) = 1$,
$\delta(0, 1, a) = \delta(0, 2, a) = \delta(2, 1, a) = \delta(2, 2, a) = 2$,
$\delta(1, 0, a) = \delta(3, 0, a) = \delta(1, 3, a) = \delta(3, 3, a) = 3$.

II) На правом верхнем блоке:
$\delta(0, 2, b) = \delta(5, 6, b) = 4$,
$\delta(0, 4, b) = \delta(0, 5, b) = \delta(5, 4, b) = \delta(5, 5, b) = 5$,
$\delta(4, 1, b) = \delta(4, 2, b) = \delta(6, 1, b) = \delta(6, 2, b) = \delta(4, 6, b) = \delta(6, 6, b) = 6$.

III) На левом нижнем блоке:
$\delta(3,0,b) = \delta(8,9,b) = 7,$
$\delta(1,7,b) = \delta(3,7,b) = \delta(1,8,b) = \delta(3,8,b) = \delta(8,7,b) = \delta(8,8,b) = 8,$
$\delta(7,0,b) = \delta(9,0,b) = \delta(7,9,b) = \delta(9,9,b) = 9.$

IV) На правом нижнем блоке:
$\delta(6,8,a) = \delta(6,10,a) = \delta(10,8,a) = \delta(10,10,a) = 10,$
$\delta(4,10,a) = \delta(11,10,a) = 11,$
$\delta(10,7,a) = \delta(10,12,a) = 12,$
$\delta(11,12,a) = 13.$

V) На блоке размера (2,2):
$\delta(0,1,b) = 4,\ \delta(1,0,b) = 7,\ \delta(4,7,a) = 13.$

В определении функции перехода не указаны те пары состояний, которые будут приводить в мусорное состояние (оно также не добавлено в множество состояний двумерного автомата A).

При работе на двумерных словах двумерного языка (1) двумерный автомат A проходит следующую прямоугольную таблицу состояний:

$$\begin{array}{c|ccccc|ccccc|}
 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & \ldots & 0 \\
\hline
0 & 1 & 2 & 2 & \ldots & 2 & 4 & 5 & 5 & \ldots & 5 \\
0 & 3 & 1 & 2 & \ldots & 2 & 6 & 4 & 5 & \ldots & 5 \\
0 & 3 & 3 & 1 & \ldots & 2 & 6 & 6 & 4 & \ldots & 5 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 3 & 3 & 3 & \ldots & 1 & 6 & 6 & 6 & \ldots & 4 \\
\hline
0 & 7 & 8 & 8 & \ldots & 8 & 10 & 10 & 10 & \ldots & 11 \\
0 & 9 & 7 & 8 & \ldots & 8 & 10 & 10 & 10 & \ldots & 11 \\
0 & 9 & 9 & 7 & \ldots & 8 & 10 & 10 & 10 & \ldots & 11 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 9 & 9 & 9 & \ldots & 7 & 12 & 12 & 12 & \ldots & 13 \\
\hline
\end{array}
\quad \text{или} \quad
\begin{array}{c|cc|}
 & 0 & 0 \\
\hline
0 & 1 & 4 \\
0 & 7 & 13 \\
\hline
\end{array}\,.$$

Функция перехода, работая на состояниях 1, 2, 3, определяет левый верхний квадратный блок из букв a, на состояниях 4–6 и 7–9 – определяет квадратные блоки из букв b такого же размера, что и левый верхний блок из букв a, на состояниях 10–13 – проверяет, что нижний правый блок состоит из букв a и его размеры определяются последним столбцом и последней строкой правого верхнего и левого нижнего блоков из букв b соответственно. Отдельно выделены значения функции перехода, которая распознает двумерное слово размера (2,2).

Легко видеть, что построенный двумерный автомат правильно работает на двумерных словах, состоящих из одной строки и одного столбца (соответствующие двумерные слова не распознаются построенным двумерным автоматом). Очевидно также, что построенный двумерный автомат правильно работает на двумерных словах размера (2,2).

Покажем, что он правильно работает на двумерных словах большего размера, т. е. двумерных словах, содержащих не менее двух строк и не менее двух столбцов. Для того, чтобы прийти в допускающее состояние 13, двумерный автомат должен прийти в состояние 10. Это возможно только в том случае, если встретилась команда $\delta(6, 8, a)$, что возможно лишь в случае, если левый верхний блок из букв a – квадратный. Действительно, если левый верхний блок из букв a прямоугольный, но не квадратный, то в случае, когда число строк меньше числа столбцов, встретятся команды $\delta(2, 8, x)$ или $\delta(2, 7, x)$, которые приводят в мусорное состояние, в случае, когда число строк больше числа столбцов, встретятся команды $\delta(6, 3, x)$ или $\delta(4, 3, x)$, которые приводят в мусорное состояние (здесь x – произвольная буква алфавита Σ). В случае если нарушено "блочное" строение двумерного слова, то построенный двумерный автомат также придет в мусорное состояние. Таким образом, в обоих случаях соответствующее двумерное слово распознаваться не будет. Далее, состояние 13 может быть получено только как $\delta(11, 12, a)$, которое определяется по последнему столбцу правого верхнего и последней строке левого нижнего блоков из букв b. Эти блоки из букв b являются квадратными и их размеры совпадают с размерами левого верхнего блока из букв a, следовательно, и правый нижний блок из букв a также квадратный и имеет такие же размеры, как три других блока. □

На самом деле, выше уже доказана следующая

Теорема 28. *Существует* $L \in \mathcal{L}(\text{2dota})$ *такой, что* $L_{(i)}, L^{(j)} \notin REG_1$ *при любых* $i, j \geq 1$.

Доказательство. Примером такого языка является двумерный язык (1) из предложения 27, поскольку согласно построению, одномерные языки $L_{(i)}$ и $L^{(j)}$ $(i, j \geq 1)$ – это нерегулярные языки $\{a^n b^n \mid n \geq 1\}$ или $\{b^n a^n \mid n \geq 1\}$. □

Замечание 29. Аналогично доказательству предложения 27 и теоремы 28 можно получить следующие результаты:

1) существует $L \in \mathcal{L}(\text{2dota})$ такой, что $L_{(i)} \notin REG_1$, $L^{(j)} \in REG_1$ при любых $i, j \geq 1$;
2) существует $L \in \mathcal{L}(\text{2dota})$ такой, что $L_{(i)} \in REG_1$, $L^{(j)} \notin REG_1$ при любых $i, j \geq 1$.

Действительно, примерами таких двумерных языков будут языки над алфавитом $\Sigma = \{a, b\}$ следующего вида:

- для условия 1):

$$L' = \left\{ \boxed{\begin{array}{c|c} p & q \end{array}} \;\middle|\; \exists n > 0 \left(p \in \{a\}^{n \times n},\ q \in \{b\}^{n \times n} \right) \right\};$$

- для условия 2):

$$L'' = \left\{ \boxed{\begin{array}{c} p \\ \hline q \end{array}} \;\middle|\; \exists n > 0 \left(p \in \{a\}^{n \times n},\ q \in \{b\}^{n \times n} \right) \right\}.$$

Двумерные детерминированные автоматы, распознающие приведенные двумерные языки, аналогичны двумерному детерминированному автомату из предложения 27. Приведем двумерный автомат лишь для двумерного языка L'. Он должен проверять, является ли блок из букв a квадратным и имеет ли блок из букв b такие же размеры, что и блок из букв a. Пусть $A = (\Sigma, S, s_0, F, \delta)$, где $S = \{0, 1, \dots, 8\}$, $s_0 = 0$, $F = \{8\}$.

Функция перехода:

I) На левом блоке:
$\delta(0,0,a) = \delta(2,3,a) = 1$,
$\delta(0,1,a) = \delta(0,2,a) = \delta(2,1,a) = \delta(2,2,a) = 2$,
$\delta(1,0,a) = \delta(3,0,a) = \delta(1,3,a) = \delta(3,3,a) = 3$.

II') На правом блоке:
$\delta(0,2,b) = \delta(5,6,b) = 4$,
$\delta(0,4,b) = \delta(0,5,b) = \delta(5,4,b) = \delta(5,5,b) = 5$,
$\delta(4,2,b) = \delta(6,2,b) = \delta(4,6,b) = \delta(6,6,b) = 6$,
$\delta(4,1,b) = \delta(6,1,b) = \delta(6,7,b) = \delta(4,7,b) = 7$,
$\delta(5,7,b) = 8$.

V') На блоке размера (1,2):
$\delta(0,1,b) = 8$.

Приведенные команды с незначительными изменениями повторяют команды двумерного автомата из предложения 27, состояния 7 и 8 в данном случае отслеживают последнюю строку в двумерном слове.

Двумерный автомат для двумерного языка L'' строится аналогично с подходящим видоизменением команд I), III), V).

Для приведенных языков $L'_{(i)} = L''^{(j)} = \{a^n b^n \mid n \geq 1\} \notin REG_1$ при любых $i, j \geq 1$ и $L'^{(j)}, L''_{(i)} \in REG_1$, поскольку это один из двух языков aa^* или bb^* при любых $i, j \geq 1$.

4. Нерегулярные двумерные языки и соответствующие им одномерные языки

Для доказательства основных результатов этого раздела потребуется очевидная лемма, которая приведена в [3] без доказательства, поскольку ее доказательство повторяет известные доказательства для “одномерных” конечных автоматов. Приведем доказательство для недетерминированного случая, в детерминированном случае проверка аналогична.

Лемма 30.

1) *Если* $L_1, L_2 \in \mathcal{L}(\text{2ota})$, *то* $L_1 \cap L_2, L_1 \cup L_2 \in \mathcal{L}(\text{2ota})$.
2) *Если* $L_1, L_2 \in \mathcal{L}(\text{2dota})$, *то* $L_1 \cap L_2, L_1 \cup L_2 \in \mathcal{L}(\text{2dota})$.

Доказательство. Пусть двумерные недетерминированные автоматы $A_1 = (\Sigma, S_1, I_1, F_1, \delta_1)$ и $A_2 = (\Sigma, S_2, I_2, F_2, \delta_2)$ распознают двумерные языки L_1 и L_2 соответственно. Построим декартово произведение этих двумерных автоматов, т. е. двумерный недетерминированный автомат $A = A_1 \times A_2 = (\Sigma, S, I, F, \delta)$, с подходящими настройками для объединения и пересечения. В двумерном автомате A множество состояний $S = S_1 \times S_2$, множество начальных состояний $I = I_1 \times I_2$, функция перехода $\delta((s_1, s_2), (s_1', s_2'), x) = \delta_1(s_1, s_1', x) \times \delta_2(s_2, s_2', x)$.

Построенный двумерный автомат A с множеством допускающих состояний $F = \{(s_1, s_2) \mid s_1 \in F_1, s_2 \in F_2\}$ распознает $L_1 \cap L_2$. Действительно,

$$\delta((I_1, I_2), (I_1, I_2), p) \cap F \neq \emptyset \;\Leftrightarrow\; \delta_1(I_1, I_1, p) \times \delta_2(I_2, I_2, p) \cap F \neq \emptyset \;\Leftrightarrow$$
$$\Leftrightarrow\; \delta_1(I_1, I_1, p) \cap F_1 \neq \emptyset \text{ и } \delta_2(I_2, I_2, p) \cap F_2 \neq \emptyset \;\Leftrightarrow\; p \in L_1 \text{ и } p \in L_2 \;\Leftrightarrow\; p \in L_1 \cap L_2.$$

Построенный двумерный автомат A с множеством допускающих состояний $F = \{(s_1, s_2) \mid s_1 \in F_1 \text{ или } s_2 \in F_2\}$ распознает $L_1 \cup L_2$, так как

$$\delta((I_1, I_2), (I_1, I_2), p) \cap F \neq \emptyset \;\Leftrightarrow\; \delta_1(I_1, I_1, p) \times \delta_2(I_2, I_2, p) \cap F \neq \emptyset \;\Leftrightarrow$$
$$\Leftrightarrow\; \delta_1(I_1, I_1, p) \cap F_1 \neq \emptyset \text{ или } \delta_2(I_2, I_2, p) \cap F_2 \neq \emptyset \;\Leftrightarrow\; p \in L_1 \text{ или } p \in L_2 \;\Leftrightarrow\; p \in L_1 \cup L_2. \quad \square$$

Замечание 31. Очевидно, что если $L \in \mathcal{L}(2\text{dota})$, то $\overline{L} \in \mathcal{L}(2\text{dota})$. Действительно, в исходном двумерном детерминированном автомате нужно заменить допускающие состояния на не допускающие и наоборот. Однако, как показано в [4] и в [14] существует $L \in \mathcal{L}(2\text{ota})$ такой, что $\overline{L} \notin \mathcal{L}(2\text{ota})$.

Теорема 32. *Существует $L \notin REG_2$ такой, что $L_{(i)}, L^{(j)} \in REG_1$ при любых $i, j \geq 1$.*

Доказательство. Пусть $L = \{p \mid |p|_a = |p|_b\} \subseteq \{a, b\}^{**}$ – двумерный язык над алфавитом $\Sigma = \{a, b\}$, состоящий из двумерных слов, в которых количество букв a совпадает с количеством букв b. Двумерный язык L нерегулярен. Действительно, если бы $L \in REG_2$, то $L \cap \Sigma^* \in REG_2$ как пересечение регулярных двумерных языков. Но $L \cap \Sigma^*$ состоит из одномерных слов, т. е. $L \cap \Sigma^* \in REG_1$, с одинаковым количеством букв a и b, что противоречит его регулярности.

Очевидно, что $L_{(i)} = L^{(j)} = \Sigma^* \in REG_1$ при любых $i, j \geq 1$. $\square$

Теорема 33. *Существует $L \notin REG_2$ такой, что $L_{(i)}, L^{(j)} \notin REG_1$ при любых $i, j \geq 1$.*

Доказательство. Пусть $L_1 = \{p \mid |p|_a = |p|_b\} \subseteq \{a, b\}^*$ – одномерный язык над алфавитом $\Sigma = \{a, b\}$, состоящий из одномерных слов, в которых количество букв a совпадает с количеством букв b.

Рассмотрим двумерный язык $L = (L_1)^{*\ominus} \cap \left((L_1)^T\right)^{*\oplus}$, в котором

$$L_{(i)} = L^{(j)} = L_1 \notin REG_1$$

при любых $i, j \geq 1$. Докажем, что двумерный язык L нерегулярен. Если бы $L \in REG_2$, то $L_2 = L \cap (\Sigma^+ \ominus \Sigma^+) \in REG_2$ как пересечение двух регулярных двумерных языков. Двумерные слова двумерного языка L_2 содержат ровно две строки, причем в каждой строке и каждом столбце количество букв a совпадает с количеством букв b. Следовательно, вторая строка двумерного слова двумерного языка L_2 получена из первой строки заменой буквы a на b и буквы b на a. Двумерному языку L_2 над алфавитом Σ взаимно-однозначно сопоставляется одномерный язык L_2 над алфавитом $\left\{\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} b \\ a \end{pmatrix}\right\}$, состоящий из непустых слов с одинаковым количеством букв $\begin{pmatrix} a \\ b \end{pmatrix}$ и $\begin{pmatrix} b \\ a \end{pmatrix}$. Согласно теореме 19, $L_2 \in REG_1\left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} b \\ a \end{pmatrix}\right)$, что противоречит его нерегулярности. □

Замечание 34. Аналогично доказательству теоремы 33 можно получить следующие результаты:

1) существует $L \notin REG_2$ такой, что $L_{(i)} \notin REG_1$, $L^{(j)} \in REG_1$ при любых $i, j \geq 1$;
2) существует $L \notin REG_2$ такой, что $L_{(i)} \in REG_1$, $L^{(j)} \notin REG_1$ при любых $i, j \geq 1$.

Действительно, примерами таких двумерных языков будут языки $L' = (L_1)^{*\ominus}$ и $L'' = \left((L_1)^T\right)^{*\oplus}$ (где L_1 – одномерный язык из доказательства теоремы 33). Первый двумерный язык состоит из двумерных слов, в каждой строке которых количество букв a совпадает с количеством букв b, второй – из двумерных слов, в каждом столбце которых количество букв a совпадает с количеством букв b. Таким образом, $L'_{(i)} = L''^{(j)} = L_1 \notin REG_1$ и $L'^{(j)} = L''_{(i)} = \Sigma^* \in REG_1$ при любых $i, j \geq 1$. Доказательства нерегулярности приведенных двумерных языков L_1 и L_2 аналогичны доказательству в теореме 33.

Заключение

Сформулируем в общем виде основные полученные в работе результаты. Из теоремы 28 и замечаний 22, 29 следует

Теорема 35. *Для каждого из приведенных условий существует $L \in \mathcal{L}(\text{2dota})$ (следовательно, $L \in REG_2$), который ему удовлетворяет*:

a) $L_{(i)}, L^{(j)} \in REG_1$ *при любых* $i, j \geq 1$;
b) $L_{(i)}, L^{(j)} \notin REG_1$ *при любых* $i, j \geq 1$;
c) $L_{(i)} \notin REG_1$, $L^{(j)} \in REG_1$ *при любых* $i, j \geq 1$;
d) $L_{(i)} \in REG_1$, $L^{(j)} \notin REG_1$ *при любых* $i, j \geq 1$.

Теоремы 32, 33 и замечание 34 в общем виде дают следующий результат:

Теорема 36. *Для каждого из приведенных условий существует $L \notin REG_2$, который ему удовлетворяет:*

a) $L_{(i)}, L^{(j)} \in REG_1$ *при любых* $i, j \geq 1$;
b) $L_{(i)}, L^{(j)} \notin REG_1$ *при любых* $i, j \geq 1$;
c) $L_{(i)} \notin REG_1$, $L^{(j)} \in REG_1$ *при любых* $j \geq 1$;
d) $L_{(i)} \in REG_1$, $L^{(j)} \notin REG_1$ *при любых* $j \geq 1$.

Таким образом, в работе показано, что могут быть реализованы восемь основных соотношений между регулярностью и нерегулярностью двумерных и соответствующих им одномерных языков.

Авторы выражают благодарность Селиванову Виктору Львовичу за вопросы, связанные с нерегулярными двумерными языками, благодаря которым работа приобрела законченный вид. Также авторы благодарят рецензента за то, что обратил их внимание на статью [6].

Список литературы

[1] D. Giammarresi, A. Restivo, *Two-dimensional languages*, in: G. Rozenberg, A. Salomaa (eds), Handbook of formal languages, Vol. 3, Springer, Berlin, 215–267 (1997).
DOI: https://doi.org/10.1007/978-3-642-59126-6_4

[2] M. Blum, C. Hewitt, *Automata on a 2-dimensional tape*, Proceedings of the 8th Annual Symposium on Switching and Automata Theory, IEEE Computer Society, Washington, 155–160 (1967).
DOI: https://doi.org/10.1109/FOCS.1967.6

[3] K. Inoue, A. Nakamura, *Some properties of two-dimensional on-line tessellation acceptors*, Inform. Sci. **13** (2), 95–121 (1977).
DOI: https://doi.org/10.1016/0020-0255(77)90023-8

[4] A. Szepietowski, *Two-dimensional on-line tessellation acceptors are not closed under complement*, Inform. Sci. **64** (1–2), 115–120 (1992).
DOI: https://doi.org/10.1016/0020-0255(92)90114-N

[5] M. Anselmo, D. Giammarresi, M. Madonia, A. Restivo, *Unambiguous recognizable two-dimensional languages*, RAIRO Theor. Inform. Appl. **40** (2), 277–293 (2006).
DOI: https://doi.org/10.1051/ita:2006008

[6] N. Vijayaraghavan, N. Jansirani, V.R. Dare, *Two dimensional rough online tesselation automaton and its properties*, J. Math. Comput. Sci. **11** (3), 3482-3495 (2021).
DOI: https://doi.org/10.28919/jmcs/5739

[7] D. Giammarresi, A. Restivo, *Recognizable picture languages*, Int. J. Pattern Recognit. Artif. Intell. **6** (2–3), 241–256 (1992).
DOI: https://doi.org/10.1142/S021800149200014X

[8] K. Inoue, I. Takanami, *A characterization of recognizable picture*, Int. J. Pattern Recognit. Artif. Intell. **8** (2), 501–508 (1994).
DOI: https://doi.org/10.1142/S0218001494000255

[9] S. Eilenberg, *Automata, languages and machines*, Vol. A, in: Pure Appl. Math. **58**, New York, Academic Press, 1974.

[10] D. Giammarresi, A. Restivo, S. Seibert, W. Thomas, *Monadic second order logic over rectangular pictures and recognizability by tiling systems*, Inform. and Comput. **125** (1), 32–45 (1996).
DOI: https://doi.org/10.1006/inco.1996.0018

[11] J.R. Büchi, *Weak second-order arithmetic and finite automata*, Z. Math. Logik Grundlagen Math. **6**, 66–92 (1960).
DOI: https://doi.org/10.1002/malq.19600060105

[12] K. Inoue, I. Takanami, A. Nakamura, *A note on two-dimensional finite automata*, Information Processing Lett. **7** (1), 49–52 (1978).
DOI: https://doi.org/10.1016/0020-0190(78)90040-6

[13] Д. Хопкрофт, Р. Мотвани, Дж. Ульман, *Введение в теорию автоматов, языков и вычислений*, Издательский дом “Вильямс”, М., 2008.

[14] K. Inoue, A. Nakamura, *Nonclosure properties of two-dimensional on-line tessellation acceptors and one way parallel sequential array acceptors*, Proc. IECE of Japan Trans. (E) **60** (9), 475–476 (1977).

Наталья Николаевна Корнеева
Казанский (Приволжский) федеральный университет,
ул. Кремлевская, д. 18, г. Казань, 420008, Россия,
e-mail: Natalia.Korneeva@kpfu.ru

Лилия Альбирусовна Гизатуллина
Казанский (Приволжский) федеральный университет,
ул. Кремлевская, д. 18, г. Казань, 420008, Россия,
e-mail: glilia99@mail.ru

Volume 3, Issue 2
pp. 32–57 (2025)

UDC 519.713
MSC 68Q45

Correlation of regularity of two-dimensional and corresponding one-dimensional languages

N.N. Korneeva, L.A. Gizatullina

Abstract. We prove relations between regularity of two-dimensional and one-dimensional languages. Each two-dimensional language is corresponded to two sequences of one-dimensional languages corresponding to rows and columns of the two-dimensional language. For each of the following conditions the existence of both regular and nonregular two-dimensional languages is proved: all row and all column languages are regular; all row languages are regular, all column languages are nonregular; all column languages are regular, all row languages are nonregular; both all row and all column languages are nonregular.

Keywords: regular language, two-dimensional language, finite automaton, two-dimensional on-line tessellation automaton.

DOI: 10.26907/2949-3919.2025.2.32-57

References

[1] D. Giammarresi, A. Restivo, *Two-dimensional languages*, in: G. Rozenberg, A. Salomaa (eds), Handbook of formal languages, Vol. 3, Springer, Berlin, 215–267 (1997).
DOI: https://doi.org/10.1007/978-3-642-59126-6_4

[2] M. Blum, C. Hewitt, *Automata on a 2-dimensional tape*, Proceedings of the 8th Annual Symposium on Switching and Automata Theory, IEEE Computer Society, Washington, 155–160 (1967).
DOI: https://doi.org/10.1109/FOCS.1967.6

[3] K. Inoue, A. Nakamura, *Some properties of two-dimensional on-line tessellation acceptors*, Inform. Sci. **13** (2), 95–121 (1977).
DOI: https://doi.org/10.1016/0020-0255(77)90023-8

[4] A. Szepietowski, *Two-dimensional on-line tessellation acceptors are not closed under complement*, Inform. Sci. **64** (1–2), 115–120 (1992).
DOI: https://doi.org/10.1016/0020-0255(92)90114-N

[5] M. Anselmo, D. Giammarresi, M. Madonia, A. Restivo, *Unambiguous recognizable two-dimensional languages*, RAIRO Theor. Inform. Appl. **40** (2), 277–293 (2006).
DOI: https://doi.org/10.1051/ita:2006008

Received: 19 September 2024. Accepted: 02 July 2025. Published: 16 July 2025.

[6] N. Vijayaraghavan, N. Jansirani, V.R. Dare, *Two dimensional rough online tesselation automaton and its properties*, J. Math. Comput. Sci. **11** (3), 3482-3495 (2021).
DOI: https://doi.org/10.28919/jmcs/5739

[7] D. Giammarresi, A. Restivo, *Recognizable picture languages*, Int. J. Pattern Recognit. Artif. Intell. **6** (2–3), 241–256 (1992).
DOI: https://doi.org/10.1142/S021800149200014X

[8] K. Inoue, I. Takanami, *A characterization of recognizable picture*, Int. J. Pattern Recognit. Artif. Intell. **8** (2), 501–508 (1994).
DOI: https://doi.org/10.1142/S0218001494000255

[9] S. Eilenberg, *Automata, languages and machines*, Vol. A, in: Pure Appl. Math. **58**, Academic Press, New York, 1974.

[10] D. Giammarresi, A. Restivo, S. Seibert, W. Thomas, *Monadic second order logic over rectangular pictures and recognizability by tiling systems*, Inform. and Comput. **125** (1), 32–45 (1996).
DOI: https://doi.org/10.1006/inco.1996.0018

[11] J.R. Büchi, *Weak second-order arithmetic and finite automata*, Z. Math. Logik Grundlagen Math. **6**, 66–92 (1960).
DOI: https://doi.org/10.1002/malq.19600060105

[12] K. Inoue, I. Takanami, A. Nakamura, *A note on two-dimensional finite automata*, Information Processing Lett. **7** (1), 49–52 (1978).
DOI: https://doi.org/10.1016/0020-0190(78)90040-6

[13] J.E. Hopcroft, R. Motwani, J.D. Ullman, *Introduction to automata theory, languages, and computation*, Addison-Wesley, MA, 2001.

[14] K. Inoue, A. Nakamura, *Nonclosure properties of two-dimensional on-line tessellation acceptors and one way parallel sequential array acceptors*, Proc. IECE of Japan Trans. (E) **60** (9), 475–476 (1977).

Natalia Nikolaevna Korneeva
Kazan Federal University,
18 Kremlyovskaya str., Kazan 420008, Russia,
e-mail: Natalia.Korneeva@kpfu.ru

Liliya Albirusovna Gizatullina
Kazan Federal University,
18 Kremlyovskaya str., Kazan 420008, Russia,
e-mail: glilia99@mail.ru