УДК 510.67

ОБ ОБЩЕМ МЕТОДЕ ДОКАЗАТЕЛЬСТВА РАЗРЕШИМОСТИ ЭЛЕМЕНТАРНЫХ ТЕОРИЙ

Шушунов А.С., Ососков А.В.
Кафедра информатики

Поступила в редакцию 25.04.2011, после переработки 05.05.2011.

В данной работе предлагается обобщение метода Блюменсата и Гределя [1] для исследования разрешимости теорий алгебраических систем. Для иллюстрации данного метода предлагаются новые классы формальных языков $\mathcal{S}$ и $\mathcal{SA}$. Выделяется класс МП-автоматов — SМП-автоматы, и доказывается, что они в точности распознают языки из класса $\mathcal{S}$. Доказывается замкнутость класса $\mathcal{SA}$ относительно пересечения, дополнения, проекций, цилиндрификаций и изоморфизмов, а также доказываются теоремы об алгоритмической разрешимости проблем вхождения и пустоты для языков класса $\mathcal{SA}$. На основе полученных результатов показана разрешимость некоторых арифметических теорий.

In this work we offer generalization of method of Blumensath A., Graedel E. to research arithmetic theories decidability. We offer new formal language classes $\mathcal{S}$ and $\mathcal{SA}$ to show this method. We distinguish a special class of pushdown automata — SMP-automata and prove they recognize the languages from class $\mathcal{S}$. We prove the class $\mathcal{SA}$ is closed under intersection, complement, projection, cylindrification and isomorphism, we also prove theorems about decidability of the occurring problem and the emptiness problem for languages from $\mathcal{SA}$. With the help of these results we show decidability of some arithmetic theories.

Ключевые слова: автоматная система, язык, МП-автомат.
Keywords: automatic structure, language, stack automata.

Введение

Исследование формальных языков является одной из важнейших задач теоретической информатики.

Одним из наиболее выдающихся результатов последнего времени является работа Блюменсата и Гределя [1]. В ней установлено, что свойства замкнутости класса регулярных языков можно использовать при исследовании разрешимости теорий алгебраических систем. В качестве примера в этой работе рассмотрена арифметика Пресбургера $\mathrm{Th}(\omega, <, +, 0, 1)$. Показано, что отношения $<$ и $+$ при подходящем кодировании натуральных чисел могут быть заданы при помощи конечных автоматов. Таким образом получено новое более простое доказательство разрешимости этой теории.

В данной работе предложено обобщение данного метода. Показано, что любой класс языков, обладающий необходимыми свойствами замкнутости, можно использовать аналогичным образом. Во второй части статьи приводится пример класса языков, обладающего свойствами из теоремы 2, но являющегося более широким, чем класс регулярных языков.

1. Определения

Определение 1. *Под i-ой* степенью *алфавита Σ, обозначаемой Σ^i, мы понимаем следующее определенное по индукции множество:*

$$\Sigma^0 = \{\epsilon\}, \text{ где } \epsilon \text{ — пустое слово;}$$
$$\Sigma^{i+1} = \{w \mid w = va, v \in \Sigma^i \text{ и } a \in \Sigma\}.$$

Если $w \in \Sigma^i$, то i называется длиной слова w или числом букв в слове w и обозначается через $|w|$. Выражением Σ^* мы будем обозначать множество любых слово в алфавите Σ:

$$\Sigma^* = \{w \mid \text{существует такое } i \geq 0, \quad \text{что } w \in \Sigma^i\}.$$

Определение 2 (описание набора слов)**.** *Пусть имеется набор слов $(\omega_1, \omega_2, \ldots, \omega_k)$ в алфавите Σ. Причём $\omega_i = \omega_{i1}\omega_{i2}\ldots\omega_{il_i}$. Введём специальный символ $\Box \notin \Sigma$, с помощью которого будем выравнивать длину слов набора. Тогда* описанием набора *$(\omega_1, \omega_2, \ldots, \omega_k)$ называется следующее слово $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k$ в алфавите $(\Sigma \cup \{\Box\})^k$ длины $l = max\{|\omega_1|, |\omega_2| \ldots |\omega_k|\}$:*

$$\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k = \begin{bmatrix} \omega'_{11} \\ \vdots \\ \omega'_{k1} \end{bmatrix} \cdots \begin{bmatrix} \omega'_{1l} \\ \vdots \\ \omega'_{kl} \end{bmatrix} \in \left((\Sigma \cup \{\Box\})^k\right)^*,$$

где $\omega'_{ij} = \omega_{ij}$, если $j \leq |\omega_i|$, и $\omega'_{ij} = \Box$, в остальных случаях.

Пример 1. Описанием набора слов $\omega_1 = 0011001, \omega_2 = 111, \omega_3 = 10010$ в алфавите $\{0, 1\}$ будет слово

$$\begin{matrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & \Box & \Box & \Box & \Box \\ 1 & 0 & 0 & 1 & 0 & \Box & \Box \end{matrix}$$

в алфавите $\{0, 1, \Box\}^3$.

Определение 3 (замкнутость семейства языков)**.** *Пусть $\mathcal{P}$ — семейство языков, $f^{(n)}$ — n-местная операция над языками, то есть отображение, которое языкам $L_1, L_2, \ldots, L_n$ ставит в соответствие язык $L = f(L_1, L_2, \ldots, L_n)$. Будем говорить, что* семейство $\mathcal{P}$ замкнуто *относительно операции f, если из того, что языки $L_1, L_2, \ldots, L_n$ принадлежат семейству $\mathcal{P}$, следует, что и язык $L = f(L_1, L_2, \ldots, L_n)$ принадлежит семейству $\mathcal{P}$. Если при этом для эффективно заданных языков $L_1, L_2, \ldots, L_n$ язык $f(L_1, L_2, \ldots, L_n)$ эффективно строится по $L_1, L_2, \ldots, L_n$, то будем говорить, что* семейство $\mathcal{P}$ эффективно замкнуто *относительно операции f.*

Определение 4. Проекцией *языка L в алфавите $\{a_1, a_2, \ldots, a_n\}^k$ по $i \le k$ называется такой язык $L^{pr(i)}$ в алфавите $\{a_1, a_2, \ldots, a_n\}^{k-1}$, что слово $w_1 w_2 \ldots w_l \in L$, где $w_j = (\delta_1^j, \ldots, \delta_{i-1}^j, \delta_i^j, \delta_{i+1}^j, \ldots, \delta_k^j)$, тогда и только тогда, когда $w'_1 w'_2 \ldots w'_l \in L^{pr(i)}$, где $w'_j = (\delta_1^j, \ldots, \delta_{i-1}^j, \delta_{i+1}^j, \ldots, \delta_k^j)$. Если $k = 1$, то $L^{pr(1)}$ - не определено. Менее формально, $L^{pr(i)}$ получается из L, если каждом слове вычеркнуть i-ый этаж.*

Определение 5. Цилиндрификацией *языка W в алфавите Σ^k называется такой язык $Cy(W)$ в алфавите Σ^{k+1}, что слово $\omega_1\omega_2\ldots\omega_l$, где $\omega_i = (\omega_i^1, \omega_i^2, \ldots, \omega_i^k)$, тогда и только тогда принадлежит W, когда $\omega'_1\omega'_2\ldots\omega'_l$, где $\omega'_i = (\omega_i^1, \omega_i^2, \ldots, \omega_i^k, \omega_i^{k+1})$, ω_i^{k+1} — любая буква из Σ, принадлежит $Cy(W)$.*

Определение 6 (перестановка)**.** *Пусть имеется язык W в алфавите Σ. Отображение $\gamma_{a,b} : W \to W'$, где W' — язык в алфавите Σ называется* перестановкой *a* с *b ($a, b \in \Sigma$), если слово $\omega = \omega_1\omega_2\ldots\omega_l$ принадлежит W тогда и только тогда, когда слово $\omega' = \omega'_1\omega'_2\ldots\omega'_l$, где $\omega'_i = a$, если $\omega_i = b$, $\omega'_i = b$, если $\omega_i = a$, и $\omega'_i = \omega_i$ в остальных случаях, принадлежит W'.*

Так как $\gamma_{a,b}(t) = \gamma_{b,a}(t)$ для любого $t \in \Sigma^*$, то перестановки $\gamma_{a,b}$ и $\gamma_{b,a}$ совпадают для любых $a, b \in \Sigma$.

2. Описание структур классом языков

Определение 7. *Реляционная структура $\mathcal{S} = (A, R_1, \ldots, R_n)$ описывается классом языков Δ, если выполнены следующие условия:*

1. *Эффективно задан такой язык $W_A \subseteq \Sigma^*$, принадлежащей классу Δ, что существует сюръективная функция $\nu : W_A \to A$. W_A назовём нумерацией A.*

2. *Язык*
$$W_\epsilon = \{\omega \otimes \omega' : \nu(\omega) = \nu(\omega');\ \omega, \omega' \in W_A\}$$
задан эффективно и принадлежит классу Δ. Другими словами, эффективно задан язык из Δ, состоящий из описаний пар слов, имеющих одинаковый образ.

3. *Для каждого $i = 1 \ldots n$ язык*
$$W_R = \{\omega_1 \otimes \omega_2 \otimes \cdots \otimes \omega_r : (\nu(\omega_1), \nu(\omega_2), \ldots, \nu(\omega_r)) \in R_i\}$$
задан эффективно и принадлежит классу Δ. Другими словами, эффективно задан язык из Δ, состоящий из описаний наборов слов, образы которых принадлежат R_i.

Всюду далее мы будем рассматривать некоторую структуру $\mathcal{S}$ сигнатуры L, описываемую классом языков Δ, и формулы той же сигнатуры. Через W_A будем обозначать нумерацию основного множества структуры $\mathcal{S}$. Под словами «язык W описывает формулу $\Psi(\bar{x})$», надо понимать, что если для $\omega_1, \omega_2, \ldots, \omega_k \in W_A$ слово

$\omega_1 \otimes \omega_2 \otimes \cdots \otimes \omega_k$ принадлежит W, то $\Psi(\nu(\omega_1), \nu(\omega_2), \ldots, \nu(\omega_k))$ истинна в $\mathcal{S}$, и если $\Psi(a_1, a_2, \ldots, a_k)$ истинна в $\mathcal{S}$, то всевозможные слова $\delta_1 \otimes \delta_2 \otimes \cdots \otimes \delta_k$, где $\delta_1 \in \nu^{-1}(a_1), \delta_2 \in \nu^{-1}(a_2), \ldots, \delta_k \in \nu^{-1}(a_k)$, принадлежит W.

Рассмотрим язык W_A^k в алфавите $(\Sigma \cup \{\Box\})^k$. Слово $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k$ принадлежит W_A^k тогда и только тогда, когда $\omega_1, \omega_2, \ldots, \omega_{k-1}$ — любые слова в алфавите $\Sigma \cup \{\Box\}$, а ω_k — слово из W_A. Другими словами, W_A^k - это язык, состоящий из всех k- этажных слов, на k-ом этаже которых записано слово из W_A.

Лемма 1. *Если класс языков Δ замкнут относительно цилиндрификации и перестановки, то язык W_A^k принадлежит классу Δ.*

Доказательство. Рассмотрим язык $W_A' = \underbrace{Cy(Cy(\ldots(Cy(}_{k-1 \text{ раз}}W_A))\ldots))$.

Полученный язык принадлежит Δ, так как по условиям леммы класс языков Δ замкнут относительно цилиндрификации. Слово из W_A' представляет собой описание набора k слов, первое из которых принадлежит W_A, а остальные — любые слова в алфавите $\Sigma \cup \{\Box\}$. Проведём всевозможные перестановки a с b, где $a = (o_1, o_2, \ldots, o_k)$, а $b = (o_2, o_3, \ldots, o_k, o_1)$. Очевидно, что полученный язык представляет собой описания наборов k слов, последние из которых принадлежит W_A, а остальные — любые слова в алфавите $\Sigma \cup \{\Box\}$, то есть W_A^k. Так как по условиям леммы класс языков Δ замкнут относительно перестановки, то W_A^k принадлежит Δ. $\Box$

Лемма 2. *Пусть язык W в алфавите $(\Sigma \cup \{\Box\})^k$ описывает формулу $\Psi(\bar{x})$. Тогда язык I — проекция языка W по i — описывает формулу $\Omega(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k)$, где $\Omega(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k)$ есть $(\exists x_i)\Psi(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_k)$.*

Доказательство. Если слово

$$\omega_1 \otimes \ldots \otimes \omega_{i-1} \otimes \omega_i \otimes \omega_{i+1} \otimes \ldots \otimes \omega_k$$

принадлежит W, то

$$\Psi(\nu(\omega_1), \ldots, \nu(\omega_{i-1}), \nu(\omega_i), \nu(\omega_{i+1}), \ldots, \nu(\omega_k))$$

истинна. Следовательно,

$$\Omega(\nu(\omega_1), \ldots, \nu(\omega_{i-1}), \nu(\omega_{i+1}), \ldots, \nu(\omega_k))$$

тоже истинна. А по определению проекции, слово $\omega_1 \otimes \ldots \otimes \omega_{i-1} \otimes \omega_{i+1} \otimes \ldots \otimes \omega_k$ принадлежит I.

Если слово $\omega_1 \otimes \ldots \otimes \omega_{i-1} \otimes \omega_{i+1} \otimes \ldots \otimes \omega_k$ принадлежит I, то в W найдется слово $\omega_1 \otimes \ldots \otimes \omega_{i-1} \otimes \omega_i \otimes \omega_{i+1} \otimes \ldots \otimes \omega_k$, где $\omega_i \in W_A$. Следовательно, $\Psi(\nu(\omega_1), \ldots, \nu(\omega_{i-1}), \nu(\omega_i), \nu(\omega_{i+1}), \ldots, \nu(\omega_k))$ истинна. Тогда, $\Omega(\nu(\omega_1), \ldots, \nu(\omega_{i-1}), \nu(\omega_{i+1}), \ldots, \nu(\omega_k))$ тоже истинна. $\Box$

Лемма 3. *Пусть язык W в алфавите $(\Sigma \cup \{\Box\})^k$ описывает формулу $\Psi(\bar{x})$. Тогда язык I — пересечение цилиндрификации языка W с W_A^{k+1} — описывает формулу $\Omega(\bar{x}, y)$, причем $\Omega(\bar{x}, y)$ истинна в $\mathcal{S}$ тогда и только тогда, когда $\Psi(\bar{x})$ истинна в $\mathcal{S}$.*

Доказательство. По определению цилиндрификация языка W представляет собой описания наборов $k+1$ слов вида $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k \otimes \omega_{k+1}$, где ω_{k+1} — любое слово в алфавите $\Sigma \cup \{\square\}$, а $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k$ принадлежит W. Пересечение цилиндрификации языка W с W_A^{k+1} представляет собой описания наборов $k+1$ слов вида $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k \otimes \omega_{k+1}$, где ω_{k+1} — любое слово из W_A, а $\omega_1 \otimes \omega_2 \otimes \ldots \otimes \omega_k$ принадлежит W. Если I описывает формулу $\Omega(\bar{x}, y)$, то значение этой формулы не зависит от y. Действительно, $\Omega(x_1, \ldots, x_k, a) = \Omega(x_1, \ldots, x_k, b)$, для любых a и b, так как если слово

$$\nu^{-1}(x_1) \otimes \cdots \otimes \nu^{-1}(x_k) \otimes \nu^{-1}(a)$$

принадлежит I, то слово

$$\nu^{-1}(x_1) \otimes \cdots \otimes \nu^{-1}(x_k) \otimes \nu^{-1}(b)$$

также принадлежит I. Следовательно, $\Omega(\bar{x}, y)$ истинна тогда и только тогда, когда истинна $\Psi(\bar{x})$. $\square$

Следствие 1. *Если класс языков Δ замкнут относительно пересечения, цилиндрификации и перестановки, и существует язык из Δ, который описывет формулу $\Psi(\bar{x})$, то язык I, описывающий формулу $\Omega(\bar{x}, y)$, принадлежит Δ. Здесь $\Omega(\bar{x}, y)$ истинна тогда и только тогда, когда $\Psi(\bar{x})$ истинна.*

Лемма 4. *Пусть язык W в алфавите $(\Sigma \cup \{\square\})^k$ описывает формулу $\Psi(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$. Тогда язык I, получаемый всевозможными перестановками a с b языка W, где $a = (o_1, \ldots, o_i, \ldots, o_j, \ldots, o_k)$, а $b = (o_1, \ldots, o_j, \ldots, o_i, \ldots, o_k)$, описывает формулу $\Omega(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_k)$, причём $\Omega(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_k)$ истинна в $\mathcal{S}$ тогда и только тогда, когда $\Psi(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$ истинна в $\mathcal{S}$.*

Доказательство. Заметим, что если слово $\omega_1 \otimes \cdots \otimes \omega_i \otimes \cdots \otimes \omega_j \otimes \cdots \otimes \omega_k$ принадлежит W, то слово $\omega_1 \otimes \cdots \otimes \omega_j \otimes \cdots \otimes \omega_i \otimes \cdots \otimes \omega_k$ принадлежит I. Если $\Omega(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_k)$ истинна тогда и только тогда, когда $\Psi(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$ истинна, то из того, что слово

$$\omega_1 \otimes \cdots \otimes \omega_i \otimes \cdots \otimes \omega_j \otimes \cdots \otimes \omega_k$$

принадлежит W, следует, что слово

$$\omega_1 \otimes \cdots \otimes \omega_j \otimes \cdots \otimes \omega_i \otimes \cdots \otimes \omega_k$$

принадлежит языку, описывающему формулу $\Omega(\bar{x})$. Следовательно I описывает формулу $\Omega(\bar{x})$. $\square$

Следствие 2. *Если класс языков Δ замкнут относительно перестановки, и существует язык из Δ, описывающий формулу $\Psi(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$, то язык I, описывающий формулу $\Omega(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_k)$, принадлежит Δ. Здесь $\Omega(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_k)$ истинна тогда и только тогда, когда $\Psi(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$ истинна.*

Замечание 1. Мы можем считать, что для любой формулы найдется эквивалентная, в которой все аргументы отношений, в нее входящих, являются переменными и различны. Действительно, формула $R(x_1, \ldots, y, \ldots, y, \ldots, x_n)$ выполнима, тогда и только тогда, когда выполнима формула $(R(x_1, \ldots, z, \ldots, y, \ldots, x_n) \& y = z)$.

Теорема 1. *Если класс языков Δ замкнут относительно объединения, пересечения, дополнения, проекции, цилиндрификации и перестановки, то по любым незамкнутой формуле $\Psi(\bar{x})$ сигнатуры L и описываемой классом языков Δ структуре $\mathcal{S}$ той же сигнатуры можно построить язык из Δ, описывающий эту формулу.*

Доказательство. Рассмотрим структуру $\mathcal{S}$ сигнатуры L и формулу Φ той же сигнатуры. Возможны следующие случаи:

1. $\Phi(\bar{x})$ есть атомная формула. Так как мы рассматриваем только реляционные структуры, то Φ имеет вид: $x = y$, либо $R(x_1, x_2, \ldots .x_r)$. В первом случае существует язык $L_\epsilon \in \Delta$, состоящий из всех описаний наборов, состоящих из пар одинаковых элементов; во втором — существует $L_R \in \Delta$, состоящий из всех описаний наборов, принадлежащих $R^{\mathcal{S}}$.

2. $\Phi(\bar{z})$ есть $(\Psi(\bar{x}) \circ \Theta(\bar{y}))$, где $\circ \in \{\wedge, \vee\}$. Существует язык $L_\Psi \in \Delta$ в алфавите Σ^k, описывающий формулу $\Psi(\bar{x})$, и существует язык $L_\Theta \in \Delta$ в алфавите Σ^l, описывающий формулу $\Theta(\bar{y})$. Заметим, что k — количество переменных в наборе $\bar{x}$ и l — количество переменных в наборе $\bar{y}$. Пусть i — количество общих переменных в наборах $\bar{x}$ и $\bar{y}$. Согласно следствию из леммы 4 существуют:

   - Язык L'_Ψ из Δ, описывающий формулу $\Psi'(o_1, \ldots, o_i, \bar{q})$, где $(o_1, \ldots, o_i, \bar{q})$ — некоторая перестановка $\bar{x}$, $(o_1, \ldots, o_i)$ — общие переменные в наборах $\bar{x}$ и $\bar{y}$. Причём $\Psi'(o_1, \ldots, o_i, \bar{q})$ истинна тогда и только тогда, когда $\Psi(\bar{x})$ истинна.
   - Язык L'_Θ из Δ, описывающий формулу $\Theta'(o_1, \ldots, o_i, \bar{s})$, где $(o_1, \ldots, o_i, \bar{s})$ — некоторая перестановка $\bar{y}$, $(o_1, \ldots, o_i)$ — общие переменные в наборах $\bar{x}$ и $\bar{y}$. Причём $\Theta'(o_1, \ldots, o_i, \bar{s})$ истинна тогда и только тогда, когда $\Theta(\bar{y})$ истинна.

   Согласно следствию из леммы 3 существуют:

   - Язык L''_Ψ из Δ, описывающий формулу $\Psi''(\bar{o}, \bar{q}, \bar{s})$, причём $\Psi''(\bar{o}, \bar{q}, \bar{s})$ истинна тогда и только тогда, когда $\Psi'(\bar{o}, \bar{q})$ истинна.
   - Язык L''_Θ из Δ, описывающий формулу $\Theta''(\bar{o}, \bar{s}, \bar{q})$, причём $\Theta''(\bar{o}, \bar{s}, \bar{q})$ истинна тогда и только тогда, когда $\Theta'(\bar{o}, \bar{s})$ истинна.

   Заметим, что $(\bar{o}, \bar{s}, \bar{q})$ и $(\bar{o}, \bar{q}, \bar{s})$ есть некоторые перестановки $\bar{z}$. Согласно следствию из леммы 4 существуют языки:

   - L'''_Ψ из Δ, описывающий формулу $\Psi'''(\bar{z})$, причём $\Psi'''(\bar{z})$ истинна тогда и только тогда, когда $\Psi''(\bar{o}, \bar{q}, \bar{s})$ истинна.
   - L'''_Θ из Δ, описывающий формулу $\Theta'''(\bar{z})$, причём $\Theta'''(\bar{z})$ истинна тогда и только тогда, когда $\Theta''(\bar{o}, \bar{s}, \bar{q})$ истинна.

Если $\circ$ есть $\wedge$, то язык, описывающий формулу $\Phi(\bar{z})$, есть пересечение языков L'''_Θ и L'''_Ψ.

Если $\circ$ есть $\vee$, то язык, описывающий формулу $\Phi(\bar{z})$, есть объединение языков L'''_Θ и L'''_Ψ.

3. $\Phi(\bar{x})$ есть $\neg\Psi(\bar{x})$. Пусть i — количество переменных в наборе $\bar{x}$. Рассмотрим язык $K = Cy(\dots(Cy(Cy(W_A)\bigcap W_A^2)\bigcap W_A^3)\cdots\bigcap W_A^i)$. Полученный язык является описанием всевозможных наборов длины i слов из W_A. Кроме того существует язык $L_\Psi \in \Delta$, описывающий формулу $\Psi(\bar{x})$. Тогда язык, описывающий формулу $\Phi(\bar{x})$, есть пересечение K с дополнением языка L_Ψ.

4. $\Phi(\bar{x})$ есть $(\Psi(\bar{x}) \to \Theta(\bar{x}))$. Тогда $\Phi(\bar{x}) \equiv (\neg\Psi(\bar{x}) \vee \Theta(\bar{x}))$.

5. $\Phi(\bar{x})$ есть $(\exists y)\Psi(\bar{x}, y)$. Существует язык $L_\Psi \in \Delta$, описывающий формулу $\Psi(\bar{x}, y)$. Тогда согласно лемме 2, язык, описывающий формулу $\Phi(\bar{x})$, есть проекция языка L_Ψ.

6. $\Phi(\bar{x})$ есть $(\forall y)\Psi(\bar{x})$. Тогда $\Phi(\bar{x}) \equiv \neg(\exists y)\neg\Psi(\bar{x})$.

Так как языки класса Δ замкнуты относительно объединения, пересечения, дополнения, проекции, цилиндрификации и перестановки, то для любой формулы структуры $\mathcal{S}$ существует язык из Δ, описывающий эта формулу. $\square$

Теорема 2. *Если класс языков Δ эффективно замкнут относительно объединения, пересечения, дополнения, проекции, цилиндрификации и перестановки, и существует алгоритм проверки пустоты языков из Δ, то элементарная теория любой структуры, описываемой Δ, разрешима.*

Доказательство. Пусть мы имеем некоторую замкнутую формулу Φ структуры $\mathcal{S} \in \mathcal{K}_\Delta$. Можем считать, что Φ предваренная. Тогда имеют место два случая:

1. Φ есть $(\exists x)\Psi(x)$. Тогда формула Φ будет истинна тогда, и только тогда, когда язык, описывающий формулу $\Psi(x)$ не является пустым.

2. Φ есть $(\forall y)\Psi(\bar{x}) \equiv \neg(\exists y)\neg\Psi(\bar{x})$. Тогда формула Φ будет истинна тогда и только тогда, когда язык, описывающий формулу $\neg\Psi(x)$, пуст.

Следовательно, проверка истинности формулы Φ в $\mathcal{S}$ сводится к проверке пустоты языка из Δ. $\square$

3. Класс языков $\mathcal{SA}$

Определение 8. *Напомним, что через $|a|$ обозначается число букв в слове a.*

Пусть L_1 и L_2 — регулярные языки в алфавите Σ. Тогда псевдосимметричным *языком будем называть язык S следующего вида: $S = \{\omega_1\omega_2 \big| |\omega_1| = |\omega_2|, \omega_1 \in L_1, \omega_2 \in L_2\}$. Будем говорить, что языки L_1 и L_2 задают язык S.*

Определение 9. *Язык называется $\mathcal{SA}$-языком, если он образован объединением псевдосимметричного языка и автоматного языка.*

Для доказательства свойств касса языков $\mathcal{SA}$ введем понятие SМП-автомата.

Определение 10. *SМП-автоматом будем называть МП-автомат* $\langle Q, \Sigma, \Gamma, \delta, q^0, \dagger, F\rangle$*, в котором*

1. Q *— множество состояний автомата;*
2. Σ *— входной алфавит;*
3. $\Gamma = \{\dagger, \diamond\}$ *— магазинный алфавит;*
4. δ *— отображение множества* $Q \times \Sigma \times \Gamma$ *в множество конечных подмножеств множества* $Q \times \Gamma^*$*;*
5. $q^0 \in Q$ *— начальное состояние;*
6. $\dagger$ *— символ, находящийся в магазине в начальный момент;*
7. $F \subseteq Q$ *— множество заключительных состояний;*
8. *слово принимается автоматом, если он перешел в одно из заключительных состояний* F *и в магазине остается ровно один символ* $\dagger$*;*
9. *если* $\delta(q, a, \dagger) = (q', m)$*, то* m *есть* $\{\diamond\}^*\dagger$*; если* $\delta(q, a, \diamond) = (q', m)$*, то* m *есть* $\{\diamond\}^*$*, то есть символ* $\dagger$ *обозначает дно магазина;*
10. *на каждом шаге МП-автомат считывает на входной ленте ровно одну букву;*
11. *все состояния* Q *делятся на две группы: в состояниях из первой группы автомат увеличивает стек ровно на одну букву, а в состояниях из второй группы автомат уменьшает стек;*
12. *не существует переходов из состояний второй группы в состояния первой.*

Теорема 3. *Язык является псевдосимметричным тогда и только тогда, когда он распознается некоторым SМП-автоматом.*

Доказательство. Пусть нам дан псевдосимметричный язык S. Он определяется языками L_1 и L_2, которые задаются автоматами $K_1 = \langle Q_1, \Sigma, \delta_1, q_1^0, F_1\rangle$ и $K_2 = \langle Q_2, \Sigma, \delta_2, q_2^0, F_2\rangle$ соответственно. Построим SМП-автомат M, который будет задавать язык S.

$M = \langle Q_1 \cup Q_2 \cup \{q^0\}, \Sigma, \Gamma, \delta, q^0, \dagger, F'\rangle$, где:

- q^0 — новое состояние, причем $q^0 \notin Q_2 \cup Q_1$;
- $\Gamma = \{\dagger, \diamond\}$;
- $\delta(q^0, a, \dagger) = (\delta_1(q_1^0, a), \diamond\dagger)$ для $a \in \Sigma$;
- $\delta(q_1, a, \diamond) = (\delta_1(q_1, a), \diamond\diamond)$ для $q_1 \in Q_1 \backslash F_1, a \in \Sigma$;
- $\delta(q_1, a, \diamond) = (\delta_1(q_1, a), \diamond\diamond) \cup (\delta_2(q_2^0, a), \epsilon)$ для $q_1 \in F_1, a \in \Sigma$;
- $\delta(q_2, a, \diamond) = (\delta_2(q_2, a), \epsilon)$ для $q_2 \in Q_2, a \in \Sigma$;

- $F' = \begin{cases} F_2 \cup \{q^0\}, & \text{если } \epsilon \in (L_1 \cap L_2), \\ F_2 & \text{иначе.} \end{cases}$

Мене формально, сначала M эмулирует работу автомата K_1, увеличивая стек. Если первая часть слова принималась K_1, то прочитав её M, продолжает работать эмулируя K_2. При этом стек уменьшается. Если вторая часть слова принималась K_2, то прочитав всё слово M, перейдет в одно из заключительных состояний. В стеке останется один символ † только при равенстве первой и ворой части.

Докажем, что все слова, которые принимает M', принадлежат S. Рассмотрим работу автомата на слове $\omega_1\omega_2$, где ω_1 оканчивается, когда первый раз получается состояние из Q_2, причем первая буква ω_2 есть a. Сначала автомат M' работает как K_1. После прочтения каждой буквы стек увеличивается ровно на один символ. Если автомат, после прочтения слова ω_1 окажется в состоянии $q \in F_1$, то слово $\omega_1 \in L_1$. Далее он переходит в состояние $\delta_2(q_2^0, a)$ и продолжает работать как K_2. После прочтения каждой следующей буквы стек уменьшается ровно на один символ. Слово будет принято автоматом M', если после прочтения всего слова он окажется в состоянии $q' \in F_2$ и в стеке будет ровно один символ. Из первого следует, что $\omega_2 \in L_2$, а из второго, что $|\omega_1| = |\omega_2|$. Следовательно, M' будет принимать только слова из S.

Теперь докажем, что M' принимает все слова, которые принадлежат S. Рассмотрим работу автомата на слове $\omega_1\omega_2$, причем первая буква ω_2 есть a, $|\omega_1| = |\omega_2|$, $\omega_1 \in L_1$ и $\omega_2 \in L_2$. Сначала автомат M' работает как K_1. После прочтения каждой буквы стек увеличивается ровно на один символ. После прочтения слова ω_1 автомат оказывается в состоянии $q \in F_1$, ибо $\omega_1 \in L_1$. Далее он переходит в состояние $\delta_2(q_2, a)$ и продолжает работать как K_2. После прочтения каждой следующей буквы стек уменьшается ровно на один символ. После прочтения всего слова он окажется в состоянии $q' \in F_2$, т.к. $\omega_2 \in L_2$. Из того, что $|\omega_1| = |\omega_2|$, следует, что в стеке останется ровно один символ. Следовательно, M' примет слово $\omega_1\omega_2$.

Докажем обратное. Пусть дан SMП-автомат $M = \langle Q, \Sigma, \Gamma, \delta, q^0, \dagger, F\rangle$, который задает язык S. Докажем, что S — псевдосимметричный. Разделим состояния M на две части: Q_1 — состояния в которых стек растет, Q_2 — в которых уменьшается. Построим два конечных автомата $K_1 = \langle Q_1 \cup \{q_f, q_1^0\}, \Sigma, \delta_1, q_1^0, F_1\rangle$ и $K_2 = \langle Q_2 \cup \{q_2^0\}, \Sigma, \delta_2, q_2^0, F_2\rangle$, где:

- q_2^0, q_1^0, q_f — новые состояния, не входящие в Q;
- $(Q_1 \cap \delta(q^0, a, \dagger)) = (\delta_1(q_1^0, a) \setminus \{q_f\})$ для каждого $a \in \Sigma$;
- для каждого $a \in \Sigma$ тогда и только тогда $q_f \in \delta_1(q_1^0, a)$, когда $(\delta(q^0, a, \dagger) \cap Q_2) \neq \emptyset$;
- $(Q_1 \cap \delta(q, a, \diamond)) = (\delta_1(q, a) \setminus \{q_f\})$ для каждых $a \in \Sigma$ и $q \in Q_1$;
- для каждых $a \in \Sigma$ и $q \in Q_1$ тогда и только тогда $q_f \in \delta_1(q, a)$, когда $(\delta(q, a, \diamond) \cap Q_2) \neq \emptyset$;
- $F_1 = \begin{cases} \{q_f, q_1^0\}, & \text{если } \epsilon \in S, \\ \{q_f\} & \text{иначе;} \end{cases}$
- $\delta_2(q, a) = \delta(q, a, \diamond)$ для каждых $a \in \Sigma$ и $q \in Q_2$;

- $\delta_2(q_2^0, a) = \{q|\ q_2 \in (Q_2 \cap \delta(q_1, b, g)), q \in \delta(q_2, a, \diamond), q_1 \in Q_1,\ g \in \Gamma, b \in \Sigma\}$ для каждого $a \in \Sigma$;
- $F_2 = \begin{cases} \{Q_2 \cap F\} \cup \{q_2^0\}, & \text{если } \epsilon \in S, \\ Q_2 \cap F & \text{иначе.} \end{cases}$

Менее формально, автомат K_1 эмулирует работу автомата M на первой части слова, при чтении которой стек увеличивался. А K_2 эмулирует работу на второй части слова, при чтении которой стек уменьшался.

Автоматы K_1 и K_2 задают языки L_1 и L_2 соответственно. Пусть $\omega \in S$. Очевидно, что ω имеет четную длину. Пусть $\omega = \omega_1\omega_2, |\omega_1| = |\omega_2|$. Рассмотрим работу автомата K_1 на слове ω_1. Прочитав первую букву, он перейдет в состояние $\delta(q^0, a, \dagger)$. Далее он работает как M. Прочитав последнюю букву слова ω_1, M должен перейти в одно из состояний Q_2. Следовательно, K_1 может перейти в состояние q_f. Таким образом, $\omega_1 \in L_1$.

Рассмотрим работу автомата K_2 на слове ω_2. Автомат M прочет слово ω_1 и первую букву слова ω_2 и перейдет в состояние q. По построению, автомат K_2, прочитав первую букву слова ω_2, тоже перейдет в состояние q. Далее он работает как M. Множество конечных состояний автомата M, в которых стек опустошается, совпадает с множеством конечных состояний K_2. Следовательно, прочитав слово ω_2 автомат K_2 окажется в одном из заключительных состояний. Таким образом, $\omega_2 \in L_2$.

Теперь докажем, что если $\omega_1 \in L_1$, $\omega_2 \in L_2$ и $|\omega_1| = |\omega_2| \neq 0$, то $\omega_1\omega_2 \in S$. Рассмотрим работу автоматов K_1 на слове ω_1 и M на слове $\omega_1\omega_2$. Автомат K_1 считывает первую букву ω_1 и попадает в состояние q. Если $|\omega_1| = 1$, то $q = q_f$. Следовательно автомат M перейдет в состояние $q_2 \in Q_2$. При этом в стеке будет находится одна буква $\diamond$. Если $|\omega_1| > 1$, то $q = q_1 \in Q_1$. Следовательно, автомат M тоже перейдет в состояние q_1. Далее автоматы работают одинаково. Прочитав последнюю букву, K_1 попадает в состояние q_f. По построению, M перейдет в состояние $q_2 \in Q_2$. При этом в стеке будет находится $|\omega_1|$ букв $\diamond$. Таким образом, прочитав первую половину слова ω, автомат M перейдет в конфигурацию $(q_2 \in Q_2, \omega_2, \underbrace{\diamond \cdots \diamond}_{|\omega_1|}\dagger)$.

Теперь рассмотрим работу автоматов K_2 на слове ω_2 и M на конфигурации $(q_2 \in Q_2, \omega_2, \underbrace{\diamond \cdots \diamond}_{|\omega_2|}\dagger)$. Автомат K_2, прочитав первую букву a слова ω_2, попадает в состояние q. Так как автомат M перешел в состояние q_2 из какого-то состояния $q \in Q$, то $q \in \delta_2(q_2^0, a)$. Следовательно, K_2 то же перейдет в состояние q. Далее автоматы работают одинаково. При этом, с каждым шагом стек автомата M уменьшается на один символ. Согласно построению, автомат K_2 никогда не возвращается в состояние q_2^0. Следовательно, на последнем шаге автомат K_2 окажется в состоянии $q' \in F$. Автомат M тоже перейдет в состояние q, оставив в стеке одну букву $\dagger$. Таким образом, $\omega_1\omega_2 \in S$.

Заметим что, M принимает пустое слово, тогда и только тогда, когда K_1 и K_2 тоже принимают пустое слово.

Из вышескзанного следует что, язык S — псевдосимметричный. Он определяется языками L_1 и L_2. $\square$

Следствие 3. *Каждый псевдосимметричный язык является контекстно-свободным.*

Доказательство. Пусть дан псевдосимметричный язык S. Его задает некоторый SМП-автомат $M = \langle Q, \Sigma, \Gamma, \delta, q^0, \dagger, F \rangle$. Добавим в M новое состояние q. И правила $\delta(q_f, \epsilon, \dagger) = q$, для всех $q_f \in F$. Полученный МП-автомат будет допускать язык S опустошением стека. Следовательно, S — контекстно-свободный. □

Теорема 4. *Пересечение псевдосимметричного языка с регулярным языком является псевдосимметричным языком.*

Доказательство. Пусть нам дан псевдосимметричный язык S, который задаётся SМП-автоматом $M = \langle Q_1, \Sigma, \Gamma, \delta_1, q_1^0, \dagger, F_1 \rangle$, и автоматный язык A, который задаётся конечным автоматом $K = \langle Q_2, \Sigma, \delta_2, q_2^0, F_2 \rangle$. Без потери общности можно считать, что $Q_1 \cap Q_2 = \emptyset$. Можно построить SМП-автомат M', который будет задавать язык $S \cap A$.

$M' = \langle Q_1 \times Q_2, \Sigma, \Gamma, \delta, (q_1^0, q_2^0), \dagger, F \rangle$, где

- $Q_1 \times Q_2$ — декартово произведение множеств состояний исходных автоматов;
- $F = \{(f_1, f_2) | f_1 \in F_1, f_2 \in F_2\}$;
- $((q_1', q_2'), T) \in \delta((q_1, q_2), a, t)$, если $\delta_2(q_2, a) = q_2'$ и $(q_1', T) \in \delta_1(q_1, a, t)$.

Несложно показать, что если после прочтения слова ω автомат M' перейдет в конфигурацию $((q_1, q_2), \alpha, \tau)$, то автомат M после прочтения слова ω перейдет в конфигурацию (q_1, α, τ), а автомат K перейдет в конфигурацию (q_2, α). Следовательно, слово будет приниматься автоматом M' тогда и только тогда, когда оно принимается автоматами M и K. □

Теорема 5. *Объединение двух псевдосимметричных языков является псевдосимметричным языком.*

Доказательство. Пусть нам даны два псевдосимметричных языка S_1 и S_2, которые распознаются SМП-автоматами $M_1 = \langle \Sigma, Q_1, \Gamma, \delta_1, q_1^0, \dagger, F_1 \rangle$ и $M_2 = \langle Q_2, \Sigma, \Gamma, \delta_2, q_2^0, \dagger, F_2, \rangle$ соответственно. Без потери общности можно считать, что $Q_1 \cap Q_2 = \emptyset$. Построим SМП-автомат M', который будет задавать язык $S_1 \cup S_2$.

$M' = \langle Q_2 \cup Q_1 \cup \{q^0\}, \Sigma, \Gamma, \delta, q^0, \dagger, F_2 \cup F_1 \rangle$, где q^0 — новое состояние, причем $q^0 \notin Q_2 \cup Q_1$, а функция переходов определяется равенствами:

- $\delta(q, a, t) = \delta_1(q, a, t)$, для любых $q \in Q_1, a \in \Sigma, t \in \Gamma$
- $\delta(q, a, t) = \delta_2(q, a, t)$, для любых $q \in Q_2, a \in \Sigma, t \in \Gamma$
- $\delta(q^0, a, t) = \delta_1(q_1^0, a, t) \cup \delta_2(q_2^0, a, t)$, для любых $a \in \Sigma, t \in \Gamma$

Таким образом, автомат M' на первом шаге как бы угадывает, какой из автоматов ему моделировать, а затем продолжает работать как M_1 или как M_2. Очевидно, что M' будет распознавать слова из $S_1 \cup S_2$. □

Теорема 6. *Класс языков $\mathcal{SA}$ замкнут относительно объединения.*

Доказательство. Пусть нам даны языки $L_1, L_2 \in \mathcal{SA}$, причем $L_1 = A_1 \cup S_1$ и $L_2 = A_2 \cup S_2$, где A_1 и A_2 — автоматные языки, S_1, S_2 — псевдосимметричные. Тогда $L_1 \cup L_2 = (A_1 \cup S_1) \cup (A_2 \cup S_2) = (A_1 \cup A_2) \cup (S_1 \cup S_2)$.

Так как регулярные языки замкнуты относительно объединения, то язык $(A_1 \cup A_2)$ является регулярным. Согласно теореме 5 язык $(S_1 \cup S_2)$ псевдосимметричный. Следовательно, язык $L_1 \cup L_2 \in \mathcal{SA}$. □

Теорема 7. *Класс языков $\mathcal{SA}$ замкнут относительно дополнения.*

Доказательство. Пусть дан язык $L = A \cup S$, где A автоматный, а S - псевдосимметричный язык.

Тогда $\bar{L} = \overline{A \cup S} = \bar{A} \cap \bar{S}$. Язык $\bar{A}$ автоматный, т.к. является дополнением автоматного.

Пусть язык S задается языками L_1 и L_2. Слово $\xi \notin S$ тогда и только тогда, когда выполняется одно из следующих условий:

1. длина слова ξ — нечетное число;
2. $\xi = \omega_1\omega_2$, причем $|\omega_1| = |\omega_2|$, $\omega_1 \in \bar{L_1}$ и $\omega_2 \in \bar{L_2}$.
3. $\xi = \omega_1\omega_2$, причем $|\omega_1| = |\omega_2|$, $\omega_1 \in L_1$ и $\omega_2 \in \bar{L_2}$.
4. $\xi = \omega_1\omega_2$, причем $|\omega_1| = |\omega_2|$, $\omega_1 \in \bar{L_1}$ и $\omega_2 \in L_2$.

Слова удовлетворяющее первому условию образуют автоматный язык A', а слова из остальных образуют псевдосимметричные языки. Согласно теореме 5 объединение языков из последних трех пунктов будет псевдосимметричным языком S'.

Тогда, $\bar{L} = \bar{A} \cap \bar{S} = \bar{A} \cap (S' \cup A') = (\bar{A} \cap S') \cup (\bar{A} \cap A')$

Согласно теореме 4, язык $\bar{A} \cap S'$ — псевдосимметричный, а язык $\bar{A} \cap A'$ является автоматным. Следовательно, $\bar{L} \in \mathcal{SA}$. □

Следствие 4. *Класс языков $\mathcal{SA}$ замкнут относительно пересечения.*

Определение 11. *Гомоморфизм $f : \Sigma \to \Sigma'$ называется равномерным, если f является эпиморфизмом и $|f(a)| = 1$ для всех $a \in \Sigma$.*

Теорема 8. *Класс языков $\mathcal{SA}$ замкнут относительно равномерных гомоморфизмов.*

Доказательство. Пусть дан равномерный гомоморфизм $f : \Sigma \to \Sigma'$ и язык $L \in \mathcal{SA}$, причем $L = A \cup S$, где A — автоматный, S — псевдосимметричный. Рассмотрим язык $L' = f(L) = f(A \cup S) = f(A) \cup f(S)$.

Пусть язык S задает SMП-автомат $M = \langle Q, \Sigma, \Gamma, \delta, q^0, \dagger, F, \rangle$. Построим SMП-автомат M', который будет задавать язык $f(S)$.

$M' = \langle Q, \Sigma', \Gamma, \delta', q^0, \dagger, F \rangle$, где $\delta'(q, f(a), t) = \bigcup_{f(b)=f(a)} \delta(q, b, t)$ для всех $q \in Q$, $a \in \Sigma$, $t \in \Gamma$. Очевидно, что если M принимает слово ω, то M' принимает слово $f(\omega)$, и если M' принимает слово σ', то M принимает слово σ, причем $\sigma \in f^{-1}(\sigma')$. Следовательно, язык $f(S)$ — псевдосимметричный. А так как автоматные языки замкнуты относительно гомоморфизмов, то язык $L' \in \mathcal{SA}$ □

Следствие 5. *Класс языков $\mathcal{SA}$ замкнут относительно перестановки.*

Следствие 6. *Класс языков $\mathcal{SA}$ замкнут относительно проекции.*

Доказательство. Рассмотрим проекцию языка $L \in \Sigma^k$ по i. Язык $L^{pr(i)}$ есть равномерный гомоморфизм. Следовательно, является автоматным. $\square$

Теорема 9. *Класс языков $\mathcal{SA}$ замкнут относительно цилиндрификации.*

Доказательство. Пусть язык $L \in \mathcal{SA}$, причем $L = A \cup S$, где A — автоматный, S — псевдосимметричный. Рассмотрим язык $L' = Cy(L) = Cy(A \cup S) = Cy(A) \cup Cy(S)$.

Пусть язык S задает SMП-автомат $M = \langle Q, \Sigma^k, \Gamma, \delta, q^0, \dagger, F, \rangle$. Построим SMП-автомат M', который будет задавать язык $Cy(S)$.

$M' = \langle Q, \Sigma^{k+1}, \Gamma, \delta', q^0, \dagger, F \rangle$, где $\delta'(q, (\omega_1, \ldots, \omega_{k+1}), t) = \delta(q, \omega_1, \ldots, \omega_k, t)$ для $q \in Q, \omega_i \in \Sigma, t \in \Gamma$.

Рассмотрим работу автоматов M и M' на словах $\omega \in (\Sigma^k)^*$ и $\omega' \in (\Sigma^{k+1})^*$, причем $\omega' \in Cy(\{\omega\})$ и $|\omega| = i$. Автомат M прочитывает слово ω, при этом он последовательно проходит состояния $q^0, q^1, \ldots q^i$. По построению δ', автомат M', работая на слове ω', пройдет эту же последовательность состояний. При этом он будет производить аналогичные операции со стеком. Следовательно, если слово ω воспринимается M, то слова из $Cy(\{\omega\})$ воспринимаются M'. Обратное утверждение доказывается аналогично.

Следовательно, язык $Cy(S)$ — псевдосимметричный. А так как для автоматного языка его цилиндрификация является тоже автоматным языком, то язык $L' \in \mathcal{SA}$. $\square$

Теорема 10. *Класс $\mathcal{SA}$ содержит только контекстно-свободные языки.*

Доказательство. Согласно следствию 3 каждый язык из $\mathcal{SA}$ представляет собой объединение регулярного и контекстно-свободного языков. Следовательно, является контекстно-свободным. $\square$

Следствие 7. *Существует алгоритм, позволяющий по произвольному языку L из класса $\mathcal{SA}$ узнать, пуст ли он.*

Итак, мы показали, что класс языков $\mathcal{SA}$ полностью отвечает условиям теоремы 2. В следующем разделе мы докажем разрешимость некоторых теорий на основе класса языков $\mathcal{SA}$.

4. Примеры разрешимых теорий

Рассмотрим теорию $T_P = \mathrm{Th}(\mathbb{N}, P^{(1)})$. Предикат P определяется следующим образом: $x \in P$, если двоичное представление x есть $\{\underbrace{1\ldots1}_{n \text{ штук}}\ \underbrace{0\ldots0}_{n \text{ штук}}\}$.

Теорема 11. *Теория T_P разрешима.*

Доказательство. Рассмотрим язык W_A в алфавите $\{0,1\}$. Слово $x \in W_A$, если его последняя буква 1 или оно состоит из одной буквы 0. Язык L_δ является автоматным. Можно в явном виде построить конечный автомат G_δ, задающий этот язык. Язык $W_A \in \mathcal{SA}$, так как является автоматным.

Пусть $\nu : W_A \to N$ — отображение, которое каждому слову из W_A ставит в соответствие число, двоичной записью которого оно является, если считать первую букву слова наименьшим разрядом. Очевидно, что ν является изоморфизмом. То есть W_A является нумерацией $\mathbb{N}$.

Пусть W_ϵ — язык, состоящий из всех описаний пар слов, имеющих одинаковый образ. Так как отображение ν является изоморфизмом, то W_ϵ состоит из описаний пар одинаковых слов. Язык W_ϵ является автоматным. Можно в явном виде построить конечный автомат G_ϵ, задающий этот язык:

$K_\epsilon = (\{q_0, q_1\}, \Sigma, \delta, q_0, \{q_1\})$, где:

$\{q_0, q_1\}$ — множество состояний;

$\Sigma = \left\{ \begin{matrix} 0 \\ 0 \end{matrix}, \begin{matrix} 0 \\ 1 \end{matrix}, \begin{matrix} 1 \\ 0 \end{matrix}, \begin{matrix} 1 \\ 1 \end{matrix}, \begin{matrix} \square \\ 1 \end{matrix}, \begin{matrix} \square \\ 0 \end{matrix}, \begin{matrix} 0 \\ \square \end{matrix}, \begin{matrix} 1 \\ \square \end{matrix} \begin{matrix} \square \\ \square \end{matrix} \right\}$ — входной алфавит;

q_0 — начальное состояние;

$\{q_0\}$ — множество конечных состояний;

$$\begin{aligned} \delta(q_0, x) &= q_0 \\ \delta(q_0, y) &= q_1 \\ \delta(q_1, x) &= q_1 \\ \delta(q_1, y) &= q_1, \end{aligned}$$

где x — любое из $\left\{ \begin{matrix} 0 \\ 0 \end{matrix}, \begin{matrix} 1 \\ 1 \end{matrix} \right\}$, а y — любое из $\left\{ \begin{matrix} 0 \\ 1 \end{matrix}, \begin{matrix} 1 \\ 0 \end{matrix}, \begin{matrix} \square \\ 1 \end{matrix}, \begin{matrix} \square \\ 0 \end{matrix}, \begin{matrix} 0 \\ \square \end{matrix}, \begin{matrix} 1 \\ \square \end{matrix}, \begin{matrix} \square \\ \square \end{matrix} \right\}$.

Язык $W_\epsilon \in \mathcal{SA}$, так как является автоматным.

Пусть W_P — язык, состоящий из слов, образы которых принадлежат $P^{(1)}$. Для W_P не трудно построить SMП-автомат, распознающий этот язык. Значит, $W_P \in \mathcal{SA}$.

Следовательно, структура $(\mathbb{N}, P^{(1)})$ описывается классом языков $\mathcal{SA}$. Тогда, по теореме 2, T_P разрешима. $\square$

Добавим в теорию T_P предикат $<^{(2)}$. $(x, y) \in <^{(2)} \Leftrightarrow x < y$.

Теорема 12. *Теория* $T_{P'} = \mathrm{Th}(\mathbb{N}, P, <)$ *разрешима.*

Доказательство. Для того, чтобы доказать, что структура $(\mathbb{N}, P^{(1)}, <^{(2)})$ описывается классом языков $\mathcal{SA}$, надо закодировать предикат $<$. Пусть $W_<$ — язык, состоящий из всех описаний пар слов, прообраз первого из которых больше прообраза второго. Очевидно, что язык $W_<$ является автоматным.

Следовательно, по теореме 2, $T_{P'}$ разрешима. $\square$

Заключение

Итак, мы показали, что если класс языков замкнут относительно операций объединения, пересечения, дополнения, проекции, цилиндрификации и перестановки, и существует алгоритм проверки пустоты языков из рассматриваемого класса, то элементарная теория любой структуры, описываемой языками рассматриваемого класса, разрешима. Предложили новые классы языков $\mathcal{S}$ и $\mathcal{SA}$, показали, что проблемы пустоты и вхождения для языков класса $\mathcal{SA}$ алгоритмически разрешимы. Выделили класс МП-автоматов — SMП-автоматы, и доказали, что они распознают в точности языки из класса $\mathcal{S}$. Доказали, что класс $\mathcal{SA}$ замкнут относительно операций пересечения, дополнения, изоморфизмов, проекций и цилиндрификаций. Используя полученные результаты, мы доказали разрешимость некоторых арифметических теорий.

Список литературы

[1] Blumensath A., Graedel E. Automatic structures // Proc. 15th IEEE Symp. on Logic in Computer Science, 2000. P. 51–62.

[2] Ахо А., Ульман Дж. Теория синтаксического анализа перевода и компиляции. Том 1, Ситаксический анализ. М.: Мир, 1978.

[3] Гинзбург С. Математическая теория контекстно-свободных языков, перевод с английского, М.: Мир, 1970.

[4] Дудаков С.М., Тайцлин М.А. Трансляционные результаты для языков запросов теории баз данных //61:2, УМН, 2006. С. 3-66.

[5] Столбоушкин А.П., Тайцлин М.А. Математические основания информатики. Часть 1, Тверь: Тверской государственный университет, 1998.