

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

УДК 519.688

DOI 10.17223/20710410/67/4

ВАЛИДАЦИЯ ПОВЕДЕНИЯ ПЕРЕДАТЧИКА В КАНАЛЕ ЧАСТИЧНОГО СТИРАНИЯ В СЛУЧАЕ ОТСУТСТВИЯ АБСОЛЮТНО РАЗЛИЧИМЫХ СИМВОЛОВ

И. Б. Казаков

*Российский экономический университет им. Плеханова, г. Москва, Россия**Московский государственный технический университет им. Баумана, г. Москва, Россия***E-mail:** i_b_kazakov@mail.ru

Работа принадлежит циклу работ, посвящённых каналу частичного стирания, где были введены понятия структуры частичного стирания, канала частичного стирания, правильной функции и корректного протокола. Структура частичного стирания — это тройка, состоящая из алфавита A , семейства его разбиений и множества вероятностей, приписанных этим разбиениям. Символы $a_1, a_2 \in A$ называются абсолютно различимыми в структуре частичного стирания, если не существует такого разбиения, что они принадлежат одному его классу. Канал частичного стирания функционирует следующим образом: Алиса посылает Бобу символ $a \in A$, но Боб узнаёт только, какое выбрано разбиение и какому классу принадлежит отправленный Алисой символ. Поведение передатчика в канале частичного стирания задаётся функцией $F : S^* \rightarrow A^*$, где S — алфавит входной ленты, с которой передатчик считывает информацию. Функция F однозначно определяет детерминированную функцию $\hat{F} : S^* \rightarrow A^*$: для любого слова $\hat{s} \in S^*$, $\hat{s} = s_1 \dots s_m$, положим $\hat{F}(\hat{s}) = F(\Lambda)F(s_1)F(s_1s_2) \dots F(s_1 \dots s_m)$, где Λ — пустое слово. Функция $\hat{F}$ может быть представлена в виде автомата с входным алфавитом S и выходным алфавитом A^* . Автомат задаётся графом, вершины которого соответствуют состояниям, а рёбра имеют две подписи: $s \in S$ и $\alpha \in A^*$. Для существования корректного протокола, включающего в себя функцию F поведения передатчика, необходимо и достаточно принадлежности функции F к классу правильных. Функция является правильной тогда и только тогда, когда отсутствует аномалия 2-го рода, не являющаяся также и аномалией 1-го рода. Под аномалией 2-го рода понимается пара слов $(\hat{s}_1, \hat{s}_2)$, такая, что $|\hat{s}_2| = |\hat{s}_1| + 1$, $|\hat{F}(\hat{s}_2)| \leq |\hat{F}(\hat{s}_1)|$; под аномалией 1-го рода — пара слов $(\hat{s}_1, \hat{s}_2)$, для которой найдётся индекс i , что символы $\hat{F}(\hat{s}_1)[i]$ и $\hat{F}(\hat{s}_2)[i]$ абсолютно различимы в структуре частичного стирания. В важном частном случае отсутствия абсолютно различных символов аномалии 1-го рода не могут существовать и условие правильности упрощается до отсутствия аномалий 2-го рода. Показано, что проверка отсутствия аномалий 2-го рода сводится к проверке отсутствия путей (начинающихся в выделенной вершине) отрицательного веса в автомат-квадратном графе, которая может быть выполнена с помощью алгоритма Беллмана — Форда. Общая сложность такой проверки составляет $O(|Q|^4|S|^2)$ по времени и $O(|Q|^2|S|^2 \ln |Q| \ln L \ln |S|)$ по памяти, где $|Q|$ — количество состояний автомата, представляющего функцию $\hat{F}$, и L — максимальная длина передаваемого Алисой слова.

Ключевые слова: скрытые каналы, канал частичного стирания, абсолютно различимые символы, алгоритм Беллмана — Форда, проверка отсутствия аномалий 2-го рода.

VALIDATION OF THE TRANSMITTER'S BEHAVIOUR IN THE PARTIAL ERASURE CHANNEL IN THE CASE OF ABSENCE OF ABSOLUTELY DISTINGUISHABLE CHARACTERS

I. B. Kazakov

Plekhanov Russian University of Economics, Moscow, Russia

Bauman Moscow State Technical University, Moscow, Russia

This paper belongs to a series of papers devoted to the partial erasure channel, in which the concepts of partial erasure structure, partial erasure channel, correct function, and correct protocol have been introduced. The partial erasure structure is a triple consisting of the alphabet A , a family of partitions of this alphabet, and a set of probabilities attributed to the partitions. The characters $a_1, a_2 \in A$ are called absolutely distinguishable if there is no partition such that they belong to the same class of this partition. The partial erasure channel functions as follows. Alice sends Bob a symbol $a \in A$, but Bob receives only partial information about the symbol. Bob only knows which partition has been chosen and which partition class the symbol belongs to. The behavior of the transmitter in the partial erase channel can be described by specifying a function $F : S^* \rightarrow A^*$, where S is the alphabet of the input tape from which the transmitter reads information. The function F uniquely defines the deterministic function $\hat{F} : S^* \rightarrow A^*$ as follows: for any word $\hat{s} \in S^*$, $\hat{s} = s_1 \dots s_m$, let $\hat{F}(\hat{s}) = F(\Lambda)F(s_1)F(s_1s_2) \dots F(s_1 \dots s_m)$, where Λ is the empty word. The function $\hat{F}$ can be represented as an automaton with the input alphabet S and the output alphabet A^* . The automaton, in turn, is represented as a graph whose vertices correspond to states and edges have two signatures: the symbol $s \in S$ and the word $\alpha \in A^*$. For the existence of a correct protocol that includes the function F of the transmitter behavior, it is necessary and sufficient that the function F belongs to the class of correct functions. A function is correct if and only if there is no anomaly of the second kind, which is not an anomaly of the first kind. The anomaly of the second kind is a pair of words $(\hat{s}_1, \hat{s}_2)$ such that $|\hat{s}_2| = |\hat{s}_1| + 1$, $|\hat{F}(\hat{s}_2)| \leq |\hat{F}(\hat{s}_1)|$. The anomaly of the first kind is a pair of words $(\hat{s}_1, \hat{s}_2)$ such that there is an index i such that the characters $\hat{F}(\hat{s}_1)[i]$ and $\hat{F}(\hat{s}_2)[i]$ are absolutely distinguishable. In the important special case of the absence of absolutely distinguishable symbols, anomalies of the first kind cannot exist. Therefore, the correctness condition is simplified to the absence of anomalies of the second kind. In this paper, it is shown that checking for the absence of anomalies of the second kind is reduced to checking for the absence of paths (starting at the selected vertex) of negative weight in the so-called automaton-square graph. This absence can be detected by the Bellman — Ford algorithm. The total complexity of checking for the absence of anomalies of the second kind is $O(|Q|^4|S|^2)$ in time and $O(|Q|^2|S|^2 \ln |Q| \ln L \ln |S|)$ in memory, where $|Q|$ is the number of states of the automaton representing the function $\hat{F}$, L is the maximum length of the word Alice throws out.

Keywords: covert channels, partial erasure channel, absolutely distinguishable characters, Bellman — Ford algorithm, checking of the absence of an anomaly of the second kind.

Введение

В работе представлена процедура валидации, которая по формальному описанию поведения передатчика (традиционно называемого Алисой) в канале частичного стирания проверяет отсутствие аномалий 2-го рода для функции, описывающей данное поведение. В предыдущей работе [1] описан алгоритм проверки принадлежности функции к определённому в [2] классу правильных функций и установлен критерий правильности: функция поведения Алисы F является правильной тогда и только тогда, когда всякая её аномалия 2-го рода является также и аномалией 1-го рода.

Особый интерес представляет частный случай, в котором в структуре частичного стирания не имеется абсолютно различных символов и, следовательно, у функции поведения не может быть аномалий 1-го рода. В этом случае для проверки принадлежности функции поведения к классу правильных необходимо и достаточно проверить отсутствие аномалий 2-го рода. В настоящей работе излагается процедура, проверяющая принадлежность функции поведения Алисы к классу правильных в этом частном случае.

Представим структуру дальнейшего изложения. В п. 1 воспроизведены изложенные в [1, 2] сведения, задающие контекст настоящей работы. К ним относятся понятия структуры и канала частичного стирания, формализации поведения приемника и передатчика, определения аномалий 1-го и 2-го рода, а также определение правильной функции поведения передатчика.

Поведение передатчика представлено функцией $\hat{F}$, которая, в свою очередь, может быть задана графом, называемым автоматным. Этот граф является входными данными процедуры валидации. Построение автоматного графа по соответствующей функции поведения передатчика представлено в п. 2.

Процедура валидации, проверяющая отсутствие аномалий 2-го рода, описана в п. 3, доказательства сформулированных здесь утверждений приведены в п. 4 и 5.

1. Канал частичного стирания

В работе [2] введены понятия структуры и канала частичного стирания.

Структура частичного стирания — это тройка, состоящая из алфавита A , семейства определённых на данном алфавите разбиений и набора вероятностей, приписанных разбиениям. Канал частичного стирания функционирует следующим образом. Алиса отправляет Бобу символ $a \in A$, а Боб получает только часть информации: какое выбрано разбиение и какому классу этого разбиения принадлежит отправленный символ. Множество пар, состоящих из разбиения и его класса, т. е. множество возможных ответов, которые может получить Боб, обозначается как алфавит Боба B .

Полагаем, что у Алисы имеется входная лента, с которой она читает символы некоторого конечного алфавита S . У Боба, в свою очередь, имеется выходная лента, на которую он может печатать символы алфавита S , а также специально зарезервированный (не входящий в S) символ стирания «*».

Канал частичного стирания изучается в связи со скрытым каналом, использующим блуждания по плоскости. Описание скрытого канала блужданий по плоскости, а также того, каким образом он связан с каналом частичного стирания, представлено в [3–5].

В работе [2] рассмотрены формализованные схемы организации передачи информации от Алисы к Бобу, названные протоколами передачи информации в канале частичного стирания. Воспроизведём соответствующие понятия.

Поведение Алисы описывается функцией $F : S^* \rightarrow A^*$, а поведение Боба — функцией $G : B^* \rightarrow S \cup \{*\}$, где A^* , B^* , S^* — множества всех слов (т. е. конечных последователь-

ностей символов, включая последовательность нулевой длины Λ , называемую «пустым словом») алфавитов A, B, S соответственно. Протокол — это пара функций (F, G) . Длину слова $\alpha \in A^*$ будем обозначать как $|\alpha|$. Функция F однозначно определяет детерминированную функцию $\hat{F} : S^* \rightarrow A^*$ следующим образом: для любого слова $\hat{s} \in S^*$, $\hat{s} = s_1 \dots s_m$, положим $\hat{F}(\hat{s}) = \hat{F}(s_1 \dots s_m) = F(\Lambda)F(s_1)F(s_1s_2) \dots F(s_1 \dots s_m)$.

Представим содержательную интерпретацию, относящуюся к функциям F и G . Пусть Алиса только что прочитала с входной ленты очередной символ $s \in S$, предварительно уже считав слово $\hat{s} = s_1 \dots s_m$. Тогда считаем, что на протяжении последующих $|F(\hat{s}s)|$ тактов Алиса посимвольно отправляет по каналу частичного стирания слово $F(\hat{s}s)$. Что касается Боба, то на каждом такте он получает символ $b \in B$ и должен принять решение о том, что печатать на выходной ленте: или какой-нибудь символ из множества $S \cup \{*\}$, или ничего не печатать. Пусть $\beta = b_1 \dots b_n \in B^*$ — слово, состоящее из символов, последовательно полученных Бобом на предыдущих (включая текущий) тактах. Полагаем, что если $G(\beta) \in S \cup \{*\}$, то Боб печатает символ $G(\beta)$. Иначе, т. е. если $G(\beta) = \Lambda$, Боб ничего не печатает.

Общей целью Алисы и Боба является печать на выходной ленте того же содержимого, которое изначально имеется на входной, при этом, может быть, с заменой значащих символов (т. е. символов алфавита S) на символ стирания «*». Протоколы, соответствующие этой цели, в [2] названы корректными. Если пара (F, G) является корректным протоколом, то будем говорить, что функция поведения Боба G согласована с функцией поведения Алисы F .

Основным вопросом, исследованным в [2], является вопрос о том, для каких функций поведения Алисы F существует согласованная с ней функция поведения Боба G . Ответом является теорема, что для этого необходимо и достаточно принадлежности функции поведения Алисы F к классу правильных. В [2] дано определение правильной функции.

В работе [1] приведён алгоритм проверки принадлежности функции F классу правильных и представлен критерий правильности, который можно взять за эквивалентное определение. Воспроизведём его формулировку:

Определение 1. Символы $a_1, a_2 \in A$ называются абсолютно различными в структуре частичного стирания, если в ней не найдётся такого разбиения, что a_1, a_2 принадлежат одному классу. Будем говорить, что слова α_1, α_2 имеют общую абсолютно различимую позицию, если найдётся такая позиция i , что символы $\alpha_1[i], \alpha_2[i]$ абсолютно различимы в соответствующей структуре частичного стирания.

Определение 2. Пара слов $(\hat{s}_1, \hat{s}_2)$ называется аномалией 1-го рода для функции F , если слова $\hat{F}(\hat{s}_1)$ и $\hat{F}(\hat{s}_2)$ имеют общую абсолютно различимую позицию.

Определение 3. Пара слов $(\hat{s}_1, \hat{s}_2)$ называется аномалией 2-го рода для функции F , если $(|\hat{s}_1| < |\hat{s}_2| \text{ и } |\hat{F}(\hat{s}_1)| \geq |\hat{F}(\hat{s}_2)|)$ или $(|\hat{s}_1| > |\hat{s}_2| \text{ и } |\hat{F}(\hat{s}_1)| \leq |\hat{F}(\hat{s}_2)|)$.

Обозначим через S_{pairs} множество всех пар слов $(\hat{s}_1, \hat{s}_2) \in (S^*)^2$, таких, что выполнено $|\hat{s}_1| = |\hat{s}_2| + 1$.

Определение 4. Функция поведения Алисы F называется правильной, если всякая её лежащая в S_{pairs} аномалия 2-го рода является также и аномалией 1-го рода.

Особый интерес представляет частный случай отсутствия абсолютно различных символов в структуре частичного стирания. В данном случае у функции поведения Алисы F не может быть аномалий 1-го рода и, следовательно, проверка на принадлежность функции классу правильных сводится к проверке отсутствия в S_{pairs} аномалий 2-го рода.

2. Автоматный граф

Представим функцию $\hat{F}$ в виде (не обязательно конечного) автомата, у которого S является входным алфавитом, а множество слов A^* — выходным: автомат на каждом шаге принимает символ $s \in S$ и выдаёт слово $\alpha \in A^*$. Действительно, рассмотрим бесконечный автомат, состояния которого находятся во взаимно однозначном соответствии с элементами множества S^* . Находясь в состоянии $\hat{s} \in S^*$ и приняв символ $s \in S$, автомат переходит в состояние $\hat{s}s$ и выдаёт слово $F(\hat{s}s)$. Подвергая автомат преобразованию, отождествляющему его неразличимые состояния, получаем приведённый автомат, представляющий функцию $\hat{F}$.

Множество состояний полученного автомата обозначим через Q . Автомат имеет функцию переходов $\phi : Q \times S \rightarrow Q$ и выходную функцию $\psi : Q \times S \rightarrow A^*$, а также начальное состояние q_Λ . Через $q_{\hat{s}}$ обозначим состояние, в которое автомат переходит, приняв на вход слово $\hat{s} = s_1 \dots s_m$, т.е. последовательно приняв символы $s_1 \dots s_m$. В соответствии с этим определением выполнено $\phi(q_{\hat{s}}, s) = q_{\hat{s}s}$.

Отметим, что все состояния автомата достижимы, т.е. для любого $q \in Q$ найдётся такое $\hat{s} \in S^*$, что $q = q_{\hat{s}}$. Достижимость всех состояний автомата следует из того, что он получен посредством отождествления неразличимых состояний первоначального бесконечного автомата, у которого все состояния достижимы по построению. Таким образом, функция, ставящая в соответствие слову $\hat{s} \in S^*$ состояние $q_{\hat{s}} \in Q$, является сюръекцией. Однако данная функция не обязательно является инъекцией: если состояния первоначального автомата, соответствующие словам $\hat{s}_1, \hat{s}_2$, неразличимы, то выполнено $q_{\hat{s}_1} = q_{\hat{s}_2}$.

Соответствующим образом определена и выходная функция $\psi(q_{\hat{s}}, s) = F(\hat{s}s)$. Значение $F(\Lambda)$, отсутствующее в описании автомата, фиксировано и хранится отдельно.

Автомат представим в виде ориентированного графа, вершины которого соответствуют элементам множества Q , а у каждого ребра есть две подписи: символ алфавита S и слово из A^* . Из каждой вершины $q \in Q$ выходят $|S|$ рёбер, каждое из которых подписано соответствующим символом $s \in S$ и словом $\psi(q, s)$. Граф имеет выделенную инициальную вершину q_Λ .

Определение 5. Граф, представляющий автомат и, следовательно, функцию поведения Алисы $\hat{F}$, будем называть автоматным графом функции $\hat{F}$.

В настоящей работе, как и в [1], рассматриваются только ограниченно-детерминированные функции поведения Алисы $\hat{F}$, т.е. только такие функции, автоматный граф которых содержит конечное число вершин.

3. Процедура валидации

Представим процедуру, проверяющую отсутствие лежащих в S_{pairs} аномалий 2-го рода. Её работа состоит из двух этапов. На первом этапе по автоматному графу строится взвешенный граф, далее называемый автомат-квадратным. На втором этапе проверяется наличие непустых путей неположительного веса в построенном автомат-квадратном графе, эта проверка осуществляется алгоритмом Беллмана — Форда [6, 7].

Построение автомат-квадратного графа описано в п. 3.1, а алгоритм Беллмана — Форда — в п. 3.2. В п. 3.1 и 3.2 также приведены теоремы 2 и 3 соответственно.

В соответствии с теоремой 2 наличие лежащих в S_{pairs} аномалий 2-го рода равносильно наличию в автомат-квадратном графе начинающихся в инициальной вершине непустых путей неположительного веса. По теореме 3, алгоритм Беллмана — Форда выдаёт ответ «да» тогда и только тогда, когда в автомат-квадратном графе не име-

ется начинающихся в инициальной вершине непустых путей неположительного веса. Таким образом, выполнено утверждение о корректности работы процедуры проверки:

Теорема 1. Процедура проверки выдаёт ответ «да» тогда и только тогда, когда представленная автоматным графом функция поведения Алисы не имеет аномалий 2-го рода.

Заключительный п. 3.3 посвящён оценкам сложности алгоритма по времени и по памяти.

3.1. Автомат-квадратный граф

На основе автоматного графа построим взвешенный ориентированный граф, каждое ребро которого, кроме числовой метки (т. е. веса ребра), имеет буквенную метку из множества $(S \cup \{*\})^2$. Опишем вершины, рёбра и метки, тем самым определим граф, далее называемый автомат-квадратным графом функции поведения Алисы.

Определение 6. Автомат-квадратным графом для функции F называется ориентированный граф, вершины, рёбра, буквенные и числовые метки которого определены следующим образом:

- 1) Имеется $|Q|^2 + 1$ вершин: $|Q|^2$ вершин, соответствующих элементам множества Q^2 , и одна дополнительная вершина, далее называемая инициальной.
- 2) Из каждой неинициальной вершины $(q_1, q_2) \in Q^2$ выходит $|S|^2$ рёбер, каждое из которых подписано буквенной меткой $(s_1, s_2) \in S^2$. Из инициальной вершины выходит $|S|$ рёбер, каждое из которых подписано буквенной меткой $(s, *)$, $s \in S$.
- 3) Ребро, выходящее из вершины $(q_1, q_2) \in Q^2$ и имеющее буквенную метку $(s_1, s_2) \in S^2$, ведёт в вершину $(q'_1, q'_2) \in Q^2$, где $q'_1 = \phi(q_1, s_1)$, $q'_2 = \phi(q_2, s_2)$, и подписано также числовой меткой $|\psi(q_1, s_1)| - |\psi(q_2, s_2)|$. Ребро, выходящее из инициальной вершины и имеющее буквенную метку $(s, *)$, ведёт в вершину $(\phi(q_\Lambda, s), q_\Lambda)$ и подписано также числовой меткой $|\psi(q_\Lambda, s)|$.

Множество вершин автомат-квадратного графа обозначим через V , множество рёбер — через E . Множество путей, начинающихся в инициальной вершине, обозначим через P , а через P_0 — множество путей из P , не содержащих циклы. В множества P и P_0 включим также пустой путь Λ длины 0. Весом пути будем называть сумму весов составляющих его рёбер.

Теорема 2. Следующие условия равносильны:

- 1) существует пара $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$, являющаяся аномалией 2-го рода;
- 2) в автомат-квадратном графе существует путь $\pi \in P \setminus \{\Lambda\}$ неположительного веса.

Доказательство теоремы приведено в п. 4.

Таким образом, проверка функции F на отсутствие лежащих в S_{pairs} аномалий 2-го рода (и, следовательно, на принадлежность классу правильных функций в случае отсутствия абсолютно различных символов) сводится к проверке автомат-квадратного графа на отсутствие пути $\pi \in P \setminus \{\Lambda\}$ неположительного веса.

Вес рёбер автомат-квадратного графа может быть отрицательным. Возможно также наличие цикла отрицательного веса. Все пути, которые требуется проверять на положительность веса, начинаются в инициальной вершине. Для решения поставленной задачи подходит алгоритм Беллмана — Форда.

3.2. Алгоритм Беллмана — Форда

Отметим, что в инициальную вершину автомат-квадратного графа не входят рёбра.

Проверим отсутствие начинающихся в инициальной вершине непустых путей неположительного веса с помощью алгоритма Беллмана — Форда. Сложность данного алгоритма составляет $O(|V||E|)$ по времени. Кроме весов рёбер, алгоритм хранит в памяти массив из $|V|$ числовых меток, приписанных вершинам графа, сложность по памяти составляет $O(|V|)$. В соответствии с описанием автомат-квадратного графа числовые метки рёбер являются целыми числами.

Числовые метки вершин могут быть как целыми числами, так и элементами множества $\{+\infty, -\infty\}$. Введём обозначения: для $e \in E$ через $d(e)$ будем обозначать вес ребра, для $v \in V$ через $d(v)$ — числовую метку, приписанную вершине v ; $d(\pi)$ — вес пути π , т. е. сумма весов рёбер, составляющих этот путь.

Алгоритм 1 проверяет отсутствие начинающихся в инициальной вершине непустых путей неположительного веса.

Алгоритм 1. Алгоритм Беллмана — Форда

- 1: Выполнить процедуру инициализации: инициальной вершине присвоить метку 0, а всем остальным вершинам — метку $+\infty$.
 - 2: $|V| - 1$ раз выполнить процедуру регулярного обхода:
 - 3: Для всех рёбер $e \in E$, $e : v \rightarrow w$,
выполнить процедуру регулярной обработки:
 - 4: Если $d(v) + d(e) < d(w)$, то
 $d(w) := d(v) + d(e)$.
 - 5: Процедура финального обхода:
 - 6: Для всех рёбер $e \in E$, $e : v \rightarrow w$,
выполнить процедуру финальной обработки:
 - 7: Если $d(v) + d(e) < d(w)$, то
 $d(w) := -\infty$.
 - 8: Проверить, есть ли неинициальная вершина с неположительной числовой меткой.
Если такая вершина найдётся, то выдать ответ «нет». Иначе выдать ответ «да».
-

Теорема 3. Следующие условия равносильны:

- 1) алгоритм 1 выдаёт ответ «да»;
- 2) не существует пути $\pi \in P \setminus \{\Lambda\}$ неположительного веса.

Доказательство теоремы приведено в п. 5.

Несмотря на то, что алгоритм Беллмана — Форда общеизвестен, считаем полезным представить подробное доказательство его корректности, т. е. доказать теорему 3. В работах [6, 7] указано, что если при исполнении последней итерации длина кратчайшего пути до какой-либо вершины строго уменьшилась, то в графе есть отрицательный цикл, достижимый из инициальной вершины, однако анализ этого обстоятельства не является достаточно подробным.

3.3. Оценки сложности

Каждое ребро автоматного графа является сущностью, состоящей из входной и выходной вершин, символьной метки $s \in S$ и метки $\alpha \in A^*$. Для построения автомат-квадратного графа требуются не сами слова из множества A^* , являющиеся метками, а только их длины. Поэтому на практике вместо метки ребра автоматного графа, являющейся словом, в памяти можно хранить числовую метку, равную длине слова. Следовательно, для хранения ребра автоматного графа требуется $O(\ln |Q| \ln |L| \ln |S|)$ бит

памяти, где L обозначает максимальную длину слов из множества A^* , являющихся метками рёбер автоматного графа.

Каждое ребро автомат-квадратного графа, в свою очередь, является сущностью, состоящей из входной и выходной вершин, двух символьных меток $s_1, s_2 \in S \cup \{*\}$, а также числовой метки, значение которой лежит в интервале $[-L, L]$. Следовательно, для хранения ребра требуется $O(\ln |Q| \ln |2L + 1| \ln |S + 1|) = O(\ln |Q| \ln |L| \ln |S|)$ бит памяти.

Граф хранится в памяти в виде списка рёбер. Таким образом, для автоматного графа требуется $O(|Q||S| \ln |Q| \ln |L| \ln |S|)$ бит памяти, а для автомат-квадратного — $O((|Q|^2|S|^2 + |S|) \ln |Q| \ln |L| \ln |S|) = O(|Q|^2|S|^2 \ln |Q| \ln |L| \ln |S|)$ бит памяти.

На втором этапе дополнительно используется $O((|Q|^2 + 1) \ln |L|) = O(|Q|^2 \ln |L|)$ бит памяти для хранения числовых меток вершин автомат-квадратного графа.

Построение каждого ребра автомат-квадратного графа занимает $O(1)$ времени, следовательно, сложность по времени первого этапа составляет $O(|Q|^2|S|^2 + |S|) = O(|Q|^2|S|^2)$.

В соответствии с описанием в п. 3.2 сложность второго этапа по времени равна $O(|V||E|) = O((|Q|^2 + 1)(|Q|^2|S|^2 + |S|)) = O(|Q|^4|S|^2)$.

Таким образом, общая сложность процедуры валидации по памяти составляет

$$\begin{aligned} O(|Q||S| \ln |Q| \ln |L| \ln |S|) + O(|Q|^2|S|^2 \ln |Q| \ln |L| \ln |S|) + O(|Q|^2 \ln |L|) = \\ = O(|Q|^2|S|^2 \ln |Q| \ln |L| \ln |S|). \end{aligned}$$

Общая сложность по времени: $O(|Q|^2|S|^2) + O(|Q|^4|S|^2) = O(|Q|^4|S|^2)$.

Сравним полученные оценки с оценками сложности алгоритма, представленного в [1]: его сложность по памяти составляет $O(L^2|Q|^3|S|^2 \ln |Q| \ln |A|)$, а по времени — $O(L|Q|^3|S|^4)$. Таким образом, при не слишком большом числе $|Q|$ состояний проверка отсутствия лежащих в S_{pairs} аномалий 2-го рода существенно менее затратна, чем проверка того, что всякая лежащая в S_{pairs} аномалия 2-го рода является также аномалией 1-го рода.

4. Доказательство теоремы 2

В п. 4.1 построим взаимно однозначное соответствие между элементами множества S_{pairs} и начинающимися в инициальной вершине путями (т. е. элементами множества $P \setminus \{\Lambda\}$) в соответствии с буквенными метками рёбер автомат-квадратного графа.

Далее в п. 4.2 докажем, что вес пути $\pi((\hat{s}_1, \hat{s}_2))$ равен $|\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)|$, и тем самым установим, что аномалиям 2-го рода соответствуют начинающиеся в инициальной вершине непустые пути неположительного веса.

4.1. Б и е к ц и я $\hat{\pi} : S_{\text{pairs}} \rightarrow P \setminus \{\Lambda\}$

Определим функцию $\hat{\pi} : S_{\text{pairs}} \rightarrow P \setminus \{\Lambda\}$.

Пусть $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$ — пара, для которой $|\hat{s}_2| = n$. Определим путь $\pi((\hat{s}_1, \hat{s}_2))$ как последовательность смежных рёбер $e_1, \dots, e_{n+1}$.

В качестве e_1 возьмём ребро, исходящее из инициальной вершины и имеющее буквенную метку $(\hat{s}_1[1], *)$. Пусть построены рёбра $e_1, \dots, e_k$. Обозначим вершину, в которую входит ребро e_k , как v_k . Так как в инициальную вершину не входят рёбра, то v_k — неинициальная. В качестве e_{k+1} возьмём ребро, выходящее из v_k и имеющее буквенную метку $(\hat{s}_1[k+1], \hat{s}_2[k])$.

Утверждение 1. Функция $\hat{\pi} : S_{\text{pairs}} \rightarrow P \setminus \{\Lambda\}$ является сюръекцией.

Доказательство. Зафиксируем путь $\pi \in P \setminus \{\Lambda\}$, т.е. произвольный непустой путь, начинающийся в инициальной вершине. Пусть путь π является последовательностью смежных рёбер $e_1, \dots, e_n$ и рёбра $e_1, \dots, e_n$ имеют буквенные метки $(s_{1,1}, s_{1,2}), \dots, (s_{n,1}, s_{n,2})$, где $s_{1,1}, \dots, s_{n,1}, s_{1,2}, \dots, s_{n,2} \in S \cup \{*\}$. Так как ребро e_1 исходит из инициальной вершины, то $s_{1,2} = *$. Рёбра $e_2, \dots, e_n$ не исходят из инициальной вершины, поэтому все остальные символы $s_{i,j}$ принадлежат множеству S .

Положим $\hat{s}_1 = s_{1,1} \dots s_{n,1} \in S^*$, т.е. $\hat{s}_1[i] = s_{i,1}$ для $i = 1, \dots, n$.

Определим слово $\hat{s}_2 \in S^*$ следующим образом. Если $n = 1$, то $\hat{s}_2 = \Lambda$. Если $n > 1$, то $\hat{s}_2 = s_{2,2} \dots s_{n,2}$, т.е. $\hat{s}_2[i] = s_{i+1,2}$ для $i = 1, \dots, n-1$. По построению $|\hat{s}_1| = n$, $|\hat{s}_2| = n-1$, т.е. $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$.

Положим $\pi' = \hat{\pi}((\hat{s}_1, \hat{s}_2)) \in P \setminus \{\Lambda\}$, пусть π' является последовательностью смежных рёбер $e'_1, \dots, e'_n$. Докажем по индукции, что $\pi' = \pi$, т.е. что для всех $i = 1, \dots, n$ выполнено $e'_i = e_i$.

Рёбра e_1, e'_1 исходят из инициальной вершины. Ребро e_1 имеет буквенную метку $(s_{1,1}, s_{1,2}) = (s_{1,1}, *)$. По определению функции $\hat{\pi}$, а также слова $\hat{s}_1$ ребро e'_1 имеет буквенную метку $(\hat{s}_1[1], *) = (s_{1,1}, *)$. Так как рёбра e_1, e'_1 исходят из одной и той же вершины и имеют тождественные буквенные метки, они совпадают.

Предположим, что $e_k = e'_k$. Докажем, что из этого следует $e_{k+1} = e'_{k+1}$.

Рёбра e_{k+1}, e'_{k+1} начинаются в вершинах, в которых заканчиваются рёбра e_k, e'_k , а так как $e'_k = e_k$, то рёбра e_{k+1}, e'_{k+1} начинаются в одной и той же вершине.

Ребро e_{k+1} имеет буквенную метку $(s_{k+1,1}, s_{k+1,2})$. По определению функции $\hat{\pi}$, а также слов $\hat{s}_1, \hat{s}_2$ ребро e'_{k+1} имеет буквенную метку $(\hat{s}_1[k+1], \hat{s}_2[k]) = (s_{k+1,1}, s_{k+1,2})$. Так как рёбра e_{k+1}, e'_{k+1} выходят из одной и той же вершины и имеют тождественные буквенные метки, они совпадают.

Таким образом, $\hat{\pi}((\hat{s}_1, \hat{s}_2)) = \pi' = \pi$, т.е. путь π имеет прообраз. В силу произвольности выбора $\pi \in P \setminus \{\Lambda\}$ это означает сюръективность функции $\hat{\pi}$. ■

Утверждение 2. Функция $\hat{\pi} : S_{\text{pairs}} \rightarrow P \setminus \{\Lambda\}$ является инъекцией.

Доказательство. Зафиксируем пары $(\hat{s}_1, \hat{s}_2), (\hat{s}'_1, \hat{s}'_2) \in S_{\text{pairs}}$, такие, что $\hat{\pi}(\hat{s}_1, \hat{s}_2) = \hat{\pi}(\hat{s}'_1, \hat{s}'_2) = \pi \in P \setminus \{\Lambda\}$. Пусть путь π является последовательностью смежных рёбер $e_1, \dots, e_n$. В соответствии с определением функции $\hat{\pi}$ выполнено $|\hat{s}_1| = |\hat{s}'_1| = n$, $|\hat{s}_2| = |\hat{s}'_2| = n-1$.

Так как $\hat{\pi}((\hat{s}_1, \hat{s}_2)) = \pi$, ребру e_1 приписана метка $(\hat{s}_1[1], *)$. С другой стороны, так как $\hat{\pi}((\hat{s}'_1, \hat{s}'_2)) = \pi$, ребру e_1 приписана метка $(\hat{s}'_1[1], *)$. Таким образом, $\hat{s}_1[1] = \hat{s}'_1[1]$.

Пусть $k < n$. Так как $\hat{\pi}((\hat{s}_1, \hat{s}_2)) = \pi$, ребру e_{k+1} приписана метка $(\hat{s}_1[k+1], \hat{s}_2[k])$. С другой стороны, так как $\hat{\pi}((\hat{s}'_1, \hat{s}'_2)) = \pi$, ребру e_{k+1} приписана метка $(\hat{s}'_1[k+1], \hat{s}'_2[k])$. Таким образом, для всех $k < n$ выполнено $\hat{s}_1[k+1] = \hat{s}'_1[k+1]$, а также $\hat{s}_2[k] = \hat{s}'_2[k]$.

Следовательно, $(\hat{s}_1, \hat{s}_2) = (\hat{s}'_1, \hat{s}'_2)$, что и означает инъективность функции $\hat{\pi}$. ■

Следствие 1. Функция $\hat{\pi} : S_{\text{pairs}} \rightarrow P \setminus \{\Lambda\}$ является биекцией.

4.2. В е с п у т и $\hat{\pi}((\hat{s}_1, \hat{s}_2))$

Утверждение 3. Пусть $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$. Тогда вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ равен

$$|\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)|.$$

Доказательство. Пусть путь $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ является последовательностью смежных рёбер $e_1, \dots, e_n$, где $|\hat{s}_1| = n$, $|\hat{s}_2| = n-1$. В соответствии с определением функции $\hat{\pi}$ ребро e_1 имеет буквенную метку $(\hat{s}_1[1], *)$, при всех $k < n$ ребро e_{k+1} имеет буквенную метку $(\hat{s}_1[k+1], \hat{s}_2[k])$.

Ребро e_1 выходит из инициальной вершины, далее обозначаемой как v_0 . Вершину, из которой выходит ребро e_{k+1} , обозначим как v_k . Таким образом, ребро e_k идет из вершины v_{k-1} в вершину v_k , и путь $\pi((\hat{s}_1, \hat{s}_2))$ последовательно проходит через вершины $v_0, v_1, \dots, v_n$.

Обозначим префикс слова $\hat{s}_1$ длины k через $\hat{s}_1^k$, т.е. $\hat{s}_1^k = \Lambda$ при $k = 0$, $\hat{s}_1^k = \hat{s}_1[1] \dots \hat{s}_1[k]$ при $k > 0$. По определению функции $\hat{F}$ выполнено

$$\hat{F}(\hat{s}_1) = F(\Lambda)F(\hat{s}_1[1])F(\hat{s}_1[1]\hat{s}_1[2]) \dots F(\hat{s}_1[1] \dots \hat{s}_1[n]) = F(\hat{s}_1^0) \dots F(\hat{s}_1^n)$$

и, следовательно, $|\hat{F}(\hat{s}_1)| = \sum_{k=0}^n |F(\hat{s}_1^k)|$.

Обозначим префикс слова $\hat{s}_2$ длины k как $\hat{s}_2^k$. Тогда

$$\hat{F}(\hat{s}_2) = F(\Lambda)F(\hat{s}_2[1])F(\hat{s}_2[1]\hat{s}_2[2]) \dots F(\hat{s}_2[1] \dots \hat{s}_2[n-1]) = F(\hat{s}_2^0) \dots F(\hat{s}_2^{n-1})$$

и, следовательно, $|\hat{F}(\hat{s}_2)| = \sum_{k=0}^{n-1} |F(\hat{s}_2^k)|$.

Докажем по индукции, что для всех k , таких, что $1 \leq k \leq n$, выполнено $v_k = (q_{\hat{s}_1^k}, q_{\hat{s}_2^{k-1}}) \in Q^2$.

База индукции: $k = 1$. Ребро e_1 исходит из инициальной вершины и имеет буквенную метку $(\hat{s}_1[1], *)$. Следовательно, ребро e_1 входит в вершину $(\phi(q_\Lambda, \hat{s}_1[1]), q_\Lambda)$. Однако ребро e_1 входит в вершину v_1 . Таким образом, $v_1 = (\phi(q_\Lambda, \hat{s}_1[1]), q_\Lambda) = (q_{\hat{s}_1[1]}, q_\Lambda) = (q_{\hat{s}_1^1}, q_{\hat{s}_2^0})$, где символ $\hat{s}_1[1]$ рассматривается как однобуквенное слово.

Предположим, что $v_k = (q_{\hat{s}_1^k}, q_{\hat{s}_2^{k-1}})$. Ребро e_{k+1} исходит из вершины v_k и имеет буквенную метку $(\hat{s}_1[k+1], \hat{s}_2[k])$. Следовательно, ребро e_{k+1} входит в вершину $(\phi(q_{\hat{s}_1^k}, \hat{s}_1[k+1]), \phi(q_{\hat{s}_2^{k-1}}, \hat{s}_2[k])) = (q_{\hat{s}_1^k \hat{s}_1[k+1]}, q_{\hat{s}_2^{k-1} \hat{s}_2[k]}) = (q_{\hat{s}_1^{k+1}}, q_{\hat{s}_2^k})$. Однако ребро e_{k+1} входит в вершину v_{k+1} . Таким образом, $v_{k+1} = (q_{\hat{s}_1^{k+1}}, q_{\hat{s}_2^k})$.

Ребро e_1 исходит из инициальной вершины и имеет буквенную метку $(\hat{s}_1[1], *)$. Следовательно, ребро e_1 имеет числовую метку $|\psi(q_\Lambda, \hat{s}_1[1])| = |F(\hat{s}_1[1])| = |F(\hat{s}_1^1)|$, где символ $\hat{s}_1[1]$ рассматривается как однобуквенное слово.

Зафиксируем $1 \leq k < n$. Ребро e_{k+1} исходит из вершины $v_k = (q_{\hat{s}_1^k}, q_{\hat{s}_2^{k-1}})$ и имеет буквенную метку $(\hat{s}_1[k+1], \hat{s}_2[k])$. Следовательно, ребро e_{k+1} имеет числовую метку $|\psi(q_{\hat{s}_1^k}, \hat{s}_1[k+1])| - |\psi(q_{\hat{s}_2^{k-1}}, \hat{s}_2[k])| = |F(\hat{s}_1^k \hat{s}_1[k+1])| - |F(\hat{s}_2^{k-1} \hat{s}_2[k])| = |F(\hat{s}_1^{k+1})| - |F(\hat{s}_2^k)|$.

Вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ равен сумме числовых меток рёбер $e_1, \dots, e_n$, т.е. $|F(\hat{s}_1^1)|$ в случае $n = 1$ и $|F(\hat{s}_1^1)| + \sum_{k=2}^n (|F(\hat{s}_1^k)| - |F(\hat{s}_2^{k-1})|)$ в случае $n > 1$.

Преобразуем выражение для веса пути:

— в случае $n = 1$:

$$|F(\hat{s}_1^1)| = |F(\Lambda)| + |F(\hat{s}_1^1)| - |F(\Lambda)| = |F(\hat{s}_1^0)| + |F(\hat{s}_1^1)| - |F(\hat{s}_2^0)| = |\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)|;$$

— в случае $n > 1$:

$$\begin{aligned} & |F(\hat{s}_1^1)| + \sum_{k=2}^n (|F(\hat{s}_1^k)| - |F(\hat{s}_2^{k-1})|) = \sum_{k=1}^n |F(\hat{s}_1^k)| - \sum_{k=1}^{n-1} |F(\hat{s}_2^k)| = \\ & = (|F(\Lambda)| + \sum_{k=1}^n |F(\hat{s}_1^k)|) - (|F(\Lambda)| + \sum_{k=1}^{n-1} |F(\hat{s}_2^k)|) = \sum_{k=0}^n |F(\hat{s}_1^k)| - \sum_{k=0}^{n-1} |F(\hat{s}_2^k)| = |\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)|. \end{aligned}$$

Таким образом, в обоих случаях вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ равен $|\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)|$. ■

Утверждение 4. Пара $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$ является аномалией 2-го рода тогда и только тогда, когда вес пути $\pi((\hat{s}_1, \hat{s}_2))$ неположителен.

Доказательство.

Необходимость. Пусть $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$ — пара, являющаяся аномалией 2-го рода, тогда $|\hat{F}(\hat{s}_1)| \leq |\hat{F}(\hat{s}_2)|$. В соответствии с утверждением 3 вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ равен $|\hat{F}(\hat{s}_1)| - |\hat{F}(\hat{s}_2)| \leq 0$.

Достаточность. Пусть $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$ — такая пара, что вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2))$ неположителен. Следовательно, $|\hat{F}(\hat{s}_1)| \leq |\hat{F}(\hat{s}_2)|$ и пара $(\hat{s}_1, \hat{s}_2)$ является аномалией 2-го рода.

Утверждение 4 доказано. ■

Доказательство теоремы 2 (формулировка на с. 85).

$1 \Rightarrow 2$:

Пусть $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$ — пара, являющаяся аномалией 2-го рода, тогда в соответствии с утверждением 4 вес пути $\hat{\pi}((\hat{s}_1, \hat{s}_2)) \in P \setminus \{\Lambda\}$ неположителен.

$2 \Rightarrow 1$:

Пусть $\pi \in P \setminus \{\Lambda\}$ — путь неположительного веса. По утверждению 1 функция $\hat{\pi}$ является биекцией, поэтому существует такая пара $(\hat{s}_1, \hat{s}_2) \in S_{\text{pairs}}$, что $\hat{\pi}((\hat{s}_1, \hat{s}_2)) = \pi$. Тогда в соответствии с утверждением 4 пара $(\hat{s}_1, \hat{s}_2)$ является аномалией 2-го рода.

Теорема 2 доказана. ■

5. Доказательство теоремы 3

В п. 5.1 описаны вспомогательные свойства, которыми может обладать граф во время выполнения алгоритма. Соотношения между данными свойствами установлены в п. 5.2. В п. 5.3 и 5.4 представлены состояния графа после выполнения шага 2 и шага 5 соответственно. Описанию результата заключительной проверки посвящён п. 5.5.

5.1. Свойства графа

Введём некоторые вспомогательные свойства графа. Поскольку числовые метки вершин графа изменяются по ходу выполнения алгоритма, то все указанные свойства определены для некоторого момента времени.

Определение 7. Будем говорить, что вершина графа $v \in V$ является вершиной достижимой разметки, если выполнено одно из следующих условий:

- 1) $d(v) = +\infty$;
- 2) $d(v) \in \mathbb{Z}$ и существует путь $\pi \in P$, оканчивающийся в вершине v , такой, что $d(\pi) = d(v)$;
- 3) $d(v) = -\infty$ и для любого $M \in \mathbb{Z}$ существует путь $\pi \in P$, оканчивающийся в вершине v , такой, что $d(\pi) < M$.

Определение 8. Будем говорить, что граф является графом достижимой разметки, если все вершины графа являются вершинами достижимой разметки и для инициальной вершины v_0 выполнено $d(v_0) = 0$.

Определение 9. Будем говорить, что путь $\pi \in P$, заканчивающийся в вершине v , обработан в графе, если $d(v) \leq d(\pi)$.

Определение 10. Будем говорить, что граф полностью обработан, если он является графом достижимой разметки и в нём обработаны все пути из множества P .

Определение 11. Будем говорить, что граф почти обработан, если он является графом достижимой разметки и в нём обработаны все пути из множества P_0 .

Определение 12. Будем говорить, что граф является уменьшаемым, если в нём найдётся такое ребро $e : v \rightarrow w$, что $d(v) + d(e) < d(w)$.

Введём обозначения для конкатенации пути с ребром и для конкатенации двух путей. Пусть путь π заканчивается в той же вершине, из которой исходит ребро e . Тогда путь, полученный добавлением ребра e в конец пути π , будем обозначать $\pi + e$. Пусть теперь путь π_1 заканчивается в той же вершине, в которой начинается путь π_2 . Тогда путь, полученный склеиванием конца пути π_1 с началом пути π_2 , будем обозначать $\pi_1 + \pi_2$.

5.2. Соотношения свойств

Утверждение 5. Пусть граф не содержит цикла отрицательного веса, достижимого из инициальной вершины. Тогда для любой вершины $v \in V$ и для каждого пути $\pi \in P$, заканчивающегося в вершине v , найдётся путь $\pi_0 \in P_0$, также заканчивающийся в вершине v , для которого $d(\pi_0) \leq d(\pi)$.

Доказательство. Индукция по длине пути π .

База индукции. Путь длины 0, т. е. пустой путь, уже лежит в множестве P_0 .

Шаг индукции. Предположим, что утверждение выполнено для всех путей из множества P длины меньше k . Докажем, что тогда оно верно для любого пути $\pi \in P$ длины k ; пусть путь π заканчивается в вершине v .

Если путь π не содержит циклов, то $\pi \in P_0$ и можно положить $\pi_0 = \pi$.

Иначе все циклы, содержащиеся в пути π , имеют неотрицательный вес; посредством «вырезания» какого-нибудь цикла можно получить путь $\pi' \in P$, заканчивающийся в вершине v и имеющий длину меньше k . Для пути π' выполнено $d(\pi') \leq d(\pi)$ и по предположению индукции существует заканчивающийся в вершине v путь $\pi_0 \in P_0$, для которого $d(\pi_0) \leq d(\pi')$. Следовательно, $d(\pi_0) \leq d(\pi)$. ■

Утверждение 6. Пусть $\pi \in P$ — путь, содержащий цикл отрицательного веса и заканчивающийся в вершине v . Тогда для любого $M \in \mathbb{Z} \cup \{+\infty\}$ существует путь $\pi' \in P$, заканчивающийся в вершине v , для которого $d(\pi') < M$.

Доказательство. Предположим обратное: существует $M \in \mathbb{Z} \cup \{+\infty\}$, такое, что для любого пути $\pi' \in P$, заканчивающегося в вершине v , выполнено $d(\pi') \geq M$.

Обозначим множество путей из P , содержащих цикл отрицательного веса и заканчивающихся в вершине v , как P' . По условию $\pi \in P'$, т. е. множество P' непусто. Для любого $\pi' \in P'$ выполнено $d(\pi') \geq M$.

Так как множество P' непусто и множество весов путей из P' ограничено снизу, то можно выбрать путь $\pi' \in P'$, который имеет минимальный вес в множестве P' . Путь π' содержит цикл отрицательного веса, т. е. он представим в виде $\pi' = \pi_1 + c + \pi_2$, где π_1 — путь из инициальной вершины в некоторую вершину w ; c — цикл отрицательного веса, начинающийся и заканчивающийся в вершине w ; π_2 — путь из вершины w в вершину v .

Рассмотрим путь $\pi'' = \pi_1 + c + c + \pi_2$. Он содержит цикл отрицательного веса и заканчивается в вершине v , т. е. $\pi'' \in P'$ и $d(\pi'') = d(\pi') + d(c) < d(\pi')$, что противоречит минимальности веса пути π' . ■

Утверждение 7. Если граф является графом достижимой разметки, то следующие условия равносильны:

- 1) граф является полностью обработанным;
- 2) граф не является уменьшаемым.

Доказательство.

$1 \Rightarrow 2$:

Предположим обратное. Тогда найдётся такое ребро $e : v \rightarrow w$, что выполнено $d(v) + d(e) < d(w)$. Формально возможны три случая: $d(v) = +\infty$, $d(v) \in \mathbb{Z}$, $d(v) = -\infty$.

Так как $d(v) + d(e) < d(w)$, то $d(v) \neq +\infty$.

Рассмотрим случай $d(v) \in \mathbb{Z}$, докажем, что он невозможен.

Так как вершина v является вершиной достижимой разметки, то существует такой путь $\pi \in P$, что $d(\pi) = d(v)$ и π заканчивается в вершине v . Рассмотрим путь $\pi' = \pi + e \in P$. Он заканчивается в вершине w и $d(\pi') = d(\pi) + d(e) = d(v) + d(e) < d(w)$. Так как $d(\pi') < d(w)$, то путь $\pi' \in P$ не является обработанным, что противоречит тому, что граф является полностью обработанным.

Рассмотрим случай $d(v) = -\infty$, докажем, что он невозможен.

Выберем произвольное $M \in \mathbb{Z}$. Так как граф является графом достижимой разметки, то существует заканчивающийся в вершине v путь $\pi \in P$, что $d(\pi) < M - d(e)$. Рассмотрим путь $\pi' = \pi + e \in P$. Он заканчивается в вершине w и $d(\pi') = d(\pi) + d(e) < M - d(e) + d(e) = M$.

Таким образом, для любого $M \in \mathbb{Z}$ существует заканчивающийся в вершине w путь $\pi' \in P$, для которого $d(\pi') < M$. Из того, что граф является графом достижимой разметки, следует, что $d(w) = -\infty$.

Получили противоречие: $-\infty = -\infty + d(e) = d(v) + d(e) < d(w) = -\infty$.

$2 \Rightarrow 1$:

База индукции: пустой путь является обработанным в графе.

Действительно, пустой путь $\Lambda \in P$ заканчивается в инициальной вершине v_0 , для которой в силу того, что граф является графом достижимой разметки, выполнено $d(v_0) \leq 0$.

Шаг индукции. Предположим, что обработаны все пути длины k из множества P . Докажем, что обработаны все пути длины $k + 1$ из множества P .

Предположим обратное. Тогда существует путь $\pi \in P$ длины $k + 1$, не являющийся обработанным в графе. Построим путь $\pi' \in P$ длины k удалением последнего ребра $e : v \rightarrow w$ пути π . По построению $d(\pi') + d(e) = d(\pi)$. По предположению индукции, путь π' , заканчивающийся в вершине v , является обработанным. Следовательно, $d(v) \leq d(\pi')$.

Так как граф не является уменьшаемым, то $d(w) \leq d(v) + d(e) \leq d(\pi') + d(e) = d(\pi)$. Так как путь $\pi \in P$ заканчивается в вершине w и не является обработанным, то $d(w) > d(\pi)$. Противоречие.

Утверждение 7 доказано. ■

Утверждение 8. Если граф почти обработан и не содержит вершин с меткой $-\infty$, то следующие условия равносильны:

- 1) граф содержит цикл отрицательного веса, достижимый из инициальной вершины;
- 2) граф является уменьшаемым.

Доказательство.

$1 \Rightarrow 2$:

Так как граф содержит цикл отрицательного веса, достижимый из инициальной вершины, то существует путь $\pi \in P$, содержащий цикл отрицательного веса. Вершину, в которой заканчивается путь π , обозначим через v .

Поскольку $d(v) \in \mathbb{Z} \cup \{+\infty\}$, по утверждению 6 существует путь $\pi' \in P$, заканчивающийся в вершине v , для которого $d(\pi') < d(v)$, следовательно, путь π' не обработан в графе.

Граф почти обработан, поэтому он является графом достижимой разметки. Так как путь $\pi' \in P$ не обработан, то граф не является полностью обработанным. Следовательно, в соответствии с утверждением 7 граф является уменьшаемым.

$2 \Rightarrow 1$:

Предположим обратное: граф не содержит цикла отрицательного веса, достижимого из инициальной вершины.

Выберем путь $\pi \in P$, вершину, в которой он заканчивается, обозначим как v .

Так как граф не содержит цикла отрицательного веса, достижимого из инициальной вершины, то по утверждению 5 существует заканчивающийся в вершине v ациклический путь $\pi_0 \in P_0$, для которого $d(\pi_0) \leq d(\pi)$.

Так как граф является почти обработанным, то путь π_0 обработан в графе, т.е. $d(v) \leq d(\pi_0)$. Следовательно, $d(v) \leq d(\pi_0) \leq d(\pi)$, т.е. путь π также обработан в графе. Поскольку это верно для любого пути $\pi \in P$, граф является полностью обработанным, а значит, он является графом достижимой разметки. В соответствии с утверждением 7 граф не является уменьшаемым, что противоречит условию.

Утверждение 8 доказано. ■

5.3. Регулярные обходы

Установим, что после выполнения шага 2 алгоритма выполнены условия утверждения 8, т.е. граф является почти обработанным и не содержит вершин с меткой $-\infty$ (см. далее утверждение 12).

Утверждение 9. Процедура регулярной обработки сохраняет свойство графа «быть графом достижимой разметки», т.е. если граф перед регулярной обработкой ребра $e : v \rightarrow w$ был графом достижимой разметки, то и после выполнения процедуры он остаётся графом достижимой разметки.

Доказательство. Процедура регулярной обработки ребра $e : v \rightarrow w$, может быть, меняет только метку вершины w . Так как в инициальную вершину ребра не входят, то вершина w не является инициальной. Таким образом, нужно проверить только сохранение свойства «быть графом достижимой разметки» в вершине w и только в том случае, если процедура меняет метку вершины w , т.е. если до её исполнения $d(v) + d(e) < d(w)$.

Формально возможны три случая: $d(v) = +\infty$, $d(v) \in \mathbb{Z}$, $d(v) = -\infty$.

Так как $d(v) + d(e) < d(w)$, то $d(v) \neq +\infty$.

Рассмотрим случай $d(v) \in \mathbb{Z}$, докажем, что после обработки ребра e вершина w останется вершиной достижимой разметки.

Так как граф до обработки ребра e является графом достижимой разметки, то существует заканчивающийся в вершине v путь $\pi \in P$, для которого $d(\pi) = d(v)$. Процедура обработки не меняет метки вершины v , поэтому и после неё $d(\pi) = d(v)$.

Положим $\pi' = \pi + e \in P$. Путь π' заканчивается в вершине w , по построению $d(\pi') = d(\pi) + d(e) = d(v) + d(e)$. После процедуры обработки $d(w) = d(v) + d(e) = d(\pi')$, следовательно, вершина w останется вершиной достижимой разметки.

Рассмотрим случай $d(v) = -\infty$.

После обработки ребра e выполнено $d(w) = -\infty$. Выберем произвольное $M \in \mathbb{Z}$.

Так как граф является графом достижимой разметки до обработки ребра e и $d(v) = -\infty$, то существует заканчивающийся в вершине v путь $\pi \in P$, такой, что $d(\pi) < M - d(e)$. Положим $\pi' = \pi + e \in P$. Путь π' заканчивается в вершине w . По построению $d(\pi') = d(\pi) + d(e) < M$.

Таким образом, для любого $M \in \mathbb{Z}$ существует заканчивающийся в вершине w путь $\pi' \in P$, для которого $d(\pi') < M$. Значит, после процедуры обработки вершина w останется вершиной достижимой разметки. ■

Утверждение 10. Пусть $\pi \in P$ — путь, заканчивающийся в вершине v . Если путь π обработан в графе перед исполнением процедуры регулярной обработки ребра $e : v \rightarrow w$, то после исполнения этой процедуры заканчивающийся в вершине w путь $\pi' = \pi + e \in P$ будет обработан в графе.

Доказательство. Так как путь $\pi \in P$ обработан в графе, то как перед, так и после обработки выполнено $d(v) \leq d(\pi)$.

Формально возможны два случая (до исполнения процедуры): $d(v) + d(e) < d(w)$ и $d(v) + d(e) \geq d(w)$.

Рассмотрим случай $d(v) + d(e) \geq d(w)$, докажем, что путь π' обработан в графе после исполнения процедуры.

Так как $d(v) + d(e) \geq d(w)$, то процедура не меняет метку вершины w . Следовательно, после исполнения процедуры $d(w) \leq d(v) + d(e) \leq d(\pi) + d(e) = d(\pi')$, где $\pi' \in P$, т.е. путь π' обработан в графе.

Рассмотрим случай $d(v) + d(e) < d(w)$. Так как $d(v) + d(e) < d(w)$, то после исполнения процедуры $d(w) = d(v) + d(e) \leq d(\pi) + d(e) = d(\pi')$, где $\pi' \in P$, т.е. путь π' обработан в графе. ■

Утверждение 11. Если в графе обработаны все пути длины не более k из множества P , то после выполнения процедуры регулярного обхода будут обработаны все пути длины не более $k + 1$ из множества P .

Доказательство. Процедура регулярной обработки рёбер не увеличивает меток вершин, поэтому она сохраняет обработанность путей, т.е. если путь $\pi \in P$ является обработанным в графе до процедуры регулярной обработки, то он останется обработанным и после неё.

Процедура регулярного обхода — это совокупность вызовов процедуры регулярной обработки, следовательно, она также сохраняет обработанность путей и после её исполнения все пути длины k будут обработаны в графе.

Выберем произвольный путь $\pi' \in P$ длины $k + 1$, последнее ребро пути π' обозначим как e . Таким образом, $\pi' = \pi + e$, где $\pi \in P$ — путь длины k .

По условию путь $\pi \in P$ обработан. Так как процедура регулярного обхода вызывает процедуру регулярной обработки ребра e , то по утверждению 10 после регулярного обхода путь $\pi' = \pi + e$ будет обработан. ■

Утверждение 12. После исполнения процедуры инициализации и $|V| - 1$ регулярных обходов граф является почти обработанным и не содержит вершин с меткой $-\infty$.

Доказательство. После процедуры инициализации граф является графом достижимой разметки, так как все вершины, кроме инициальной, являются вершинами достижимой разметки, поскольку имеют метку $+\infty$; инициальная вершина, имеющая метку 0, является вершиной достижимой разметки, поскольку в неё ведёт путь $\Lambda \in P$ нулевого веса и выполнено условие $d(v_0) = 0 \leq 0$.

После процедуры инициализации пустой путь $\Lambda \in P$ обработан в графе, т.е. обработаны все пути длины 0.

В соответствии с утверждением 11 после исполнения $|V| - 1$ регулярных обходов будут обработаны все пути из множества P длины не более $|V| - 1$. Так как все пути из множества P_0 не содержат циклов, то их длина не превышает $|V| - 1$. Таким образом, после исполнения регулярных обходов будут обработаны все пути из множества P_0 .

По утверждению 9 после регулярных обходов граф останется графом достижимой разметки; так как на этот момент в нём обработаны все пути из множества P_0 , граф является почти обработанным.

Процедура инициализации не приписывает никакой вершине метку $-\infty$. Процедура регулярной обработки и, следовательно, процедура регулярного обхода также не приписывает метки $-\infty$. Следовательно, после исполнения регулярных обходов нет вершин с меткой $-\infty$. ■

5.4. Ф и н а л ь н ы й о б х о д

Установим свойства графа, которыми он обладает на момент завершения шага 5.

Утверждение 13. Пусть граф не содержит циклов отрицательного веса, достижимых из инициальной вершины. Тогда на момент завершения шага 5 алгоритма выполнены следующие условия:

- 1) граф является полностью обработанным;
- 2) граф не содержит вершин с числовой меткой $-\infty$.

Доказательство. В соответствии с утверждением 12, на момент завершения шага 2 граф почти обработан и не содержит вершин с меткой $-\infty$. Тогда по утверждению 8 граф не является уменьшаемым.

Так как граф почти обработан, он является графом достижимой разметки. Так как граф не является уменьшаемым, то в соответствии с утверждением 7 он полностью обработан на момент завершения шага 2.

Так как граф не является уменьшаемым, для всякого ребра $e \in E$, $e : v \rightarrow w$ выполнено $d(v) + d(e) \geq d(w)$ и, следовательно, процедура финальной обработки (а значит, и процедура финального обхода) не производит никаких действий на ребре e .

Таким образом, на момент завершения шага 5 граф является таким же, как и после шага 2, т. е. полностью обработанным и не содержащим вершин с меткой $-\infty$. ■

Утверждение 14. Пусть граф содержит цикл отрицательного веса, достижимый из инициальной вершины. Тогда на момент завершения шага 5 существует вершина с числовой меткой $-\infty$.

Доказательство. По утверждению 12 на момент завершения шага 2 граф является почти обработанным и не содержит вершин с числовой меткой $-\infty$. Так как граф содержит цикл отрицательного веса, достижимый из инициальной вершины, то в соответствии с утверждением 8 граф является уменьшаемым.

Предположим, что на момент завершения шага 5 граф не содержит вершин с числовой меткой $-\infty$.

Как на момент завершения шага 2, так и на момент завершения шага 5 граф не содержит вершин с числовой меткой $-\infty$. Следовательно, каждый вызов процедуры финальной обработки не производит никаких действий. Значит, для каждого ребра $e \in E$, $e : v \rightarrow w$, выполнено $d(v) + d(e) \geq d(w)$, т. е. граф не является уменьшаемым на момент завершения шага 2 — противоречие. ■

5.5. З а к л ю ч и т е л ь н а я п р о в е р к а

Опишем результат работы шага 8.

Утверждение 15. Следующие условия равносильны:

- 1) на момент завершения шага 5 существует неинициальная вершина с неположительной числовой меткой;
- 2) существует непустой путь $\pi \in P \setminus \{\Lambda\}$ неположительного веса.

Доказательство. Возможны два случая: граф содержит или не содержит цикл отрицательного веса, достижимый из инициальной вершины.

С л у ч а й 1: граф содержит цикл отрицательного веса, достижимый из инициальной вершины.

$1 \Rightarrow 2$:

По условию существует путь $\pi \in P$, содержащий отрицательный цикл. Вершину, в которой заканчивается путь π , обозначим через v . Путь π не пустой и, значит, вершина v не инициальная.

В соответствии с утверждением 6 существует путь $\pi' \in P$, также заканчивающийся в вершине v , для которого $d(\pi') < 0$, т.е. путь π' имеет неположительный вес; а так как путь π' заканчивается в неинициальной вершине v , то он не является пустым, т.е. $\pi' \in P \setminus \{\Lambda\}$.

$2 \Rightarrow 1$:

Так как граф содержит цикл отрицательного веса, достижимый из инициальной вершины, то в соответствии с утверждением 14 на момент завершения шага 5 существует вершина v с неположительной числовой меткой $-\infty$.

Вершина v не является инициальной, потому что на момент завершения шага 5 инициальная вершина имеет метку 0: процедура инициализации назначает ей метку 0, а процедуры регулярной и финальной обработки не изменяют метку, поскольку в инициальную вершину не входят рёбра.

С л у ч а й 2: граф не содержит цикла отрицательного веса, достижимого из инициальной вершины.

В соответствии с утверждением 13 на момент завершения шага 5 граф является полностью обработанным (а значит, является графом достижимой разметки) и не содержит вершин с числовой меткой $-\infty$.

$1 \Rightarrow 2$:

Пусть v — неинициальная вершина, для которой $d(v) \leq 0$. По условию $d(v) \neq -\infty$.

Так как $d(v) \in \mathbb{Z}$ и граф является графом достижимой разметки, то существует путь $\pi \in P$, заканчивающийся в вершине v , для которого $d(\pi) = d(v) \leq 0$, а так как вершина v не инициальная, то путь $\pi \in P$ не пустой.

$2 \Rightarrow 1$:

Пусть $\pi \in P \setminus \{\Lambda\}$ — путь, для которого $d(\pi) \leq 0$; вершину, в которой заканчивается путь π , обозначим через v ; очевидно, что она не является инициальной.

Так как граф является полностью обработанным, то в нём обработан путь $\pi \in P$, т.е. $d(v) \leq d(\pi) \leq 0$.

Утверждение 15 доказано. ■

Доказательство теоремы 3 (формулировка на с. 86).

В соответствии с описанием шага 8 алгоритм Беллмана — Форда выдаёт ответ «да» тогда и только тогда, когда к моменту завершения шага 5 не существует неинициальной вершины с неположительной числовой меткой.

По утверждению 15 к моменту завершения шага 5 существование неинициальной вершины с неположительной числовой меткой равносильно существованию непустого пути $\pi \in P \setminus \{\Lambda\}$ неположительного веса. ■

Заключение

В работе представлен алгоритм проверки отсутствия лежащих в S_{pairs} аномалий 2-го рода, т.е. решена частная задача, относящаяся к каналу частичного стирания.

Такие поставленные в работах [2, 4, 5] вопросы, как способы построения правильных функций поведения передатчика, построение поведения приемника по заданному поведению передатчика, а также оценки пропускной способности канала остаются за пределами настоящей работы и работы [1], поскольку они посвящены исключительно задаче проверки передатчика на предмет существования согласованного с ним приемника, которую требуется решить прежде иных упомянутых задач. Таким образом, указанные задачи являются предметом дальнейшего исследования.

ЛИТЕРАТУРА

1. Казаков И. Б. Алгоритм валидации ограниченно-детерминированного поведения передатчика в канале частичного стирания // Прикладная дискретная математика. 2023. № 59. С. 88–110.
2. Казаков И. Б. Критерий существования корректного протокола в канале частичного стирания // Чебышевский сборник. 2021. Т. 22. № 1. С. 133–151.
3. Казаков И. Б. Критерий надёжности канала с запрещениями // Интеллектуальные системы. Теория и приложения. 2019. Т. 23. № 2. С. 33–55.
4. Казаков И. Б. Передача информации в каналах, задаваемых структурами частичного стирания. Ч. 1 // Программная инженерия. 2020. Т. 11. № 5. С. 277–284.
5. Казаков И. Б. Передача информации в каналах, задаваемых структурами частичного стирания. Ч. 2 // Программная инженерия. 2020. Т. 11. № 6. С. 322–329.
6. Bellman R. E. On a routing problem // Quarterly Appl. Math. 1958. V. 16. P. 87–90.
7. Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C. Introduction to Algorithms. 2nd ed. Cambridge, Massachusetts: MIT Press and McGraw-Hill, 2001. 1203 p.

REFERENCES

1. Kazakov I. B. Algorithm validatsii ogranichenno-determinirovannogo povedeniya peredatchika v kanale chastichnogo stiraniya [A validation algorithm of the transmitter's boundedly deterministic behaviour in a partial erasure channel]. Prikladnaya Diskretnaya Matematika, 2023, no. 59, pp. 88–110. (in Russian)
2. Kazakov I. B. Kriteriy sushchestvovaniya korrektnogo protokola v kanale chastichnogo stiraniya [Criterion for the existence of a consistent protocol in a partial erasure channel]. Chebyshevskiy Sbornik, 2021, vol. 22, no. 1, pp. 133–151. (in Russian)
3. Kazakov I. B. Kriteriy nadezhnosti kanala s zapreshcheniyami [Reliability criterion for channels with prohibitions]. Intellektual'nye Sistemy. Teoriya i Prilozheniya, 2019, vol. 22, no. 2, pp. 33–55. (in Russian)
4. Kazakov I. B. Peredacha informatsii v kanalakh, zadavaemykh strukturami chastichnogo stiraniya. Ch. 1 [Transmission of information in channels specified by structures of partial erasure. P. 1]. Programmnaya Inzheneriya, 2020, vol. 11, no. 5, pp. 277–284. (in Russian)
5. Kazakov I. B. Peredacha informatsii v kanalakh, zadavaemykh strukturami chastichnogo stiraniya. Ch. 2 [Transmission of information in channels specified by structures of partial erasure. P. 2]. Programmnaya Inzheneriya, 2020, vol. 11, no. 6, pp. 322–329. (in Russian)
6. Bellman R. E. On a routing problem. Quarterly Appl. Math., 1958, vol. 16, pp. 87–90.
7. Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C. Introduction to Algorithms. 2nd ed. Cambridge, Massachusetts, MIT Press and McGraw-Hill, 2001. 1203 p.