МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»
Кафедра дискретной математики и информационных технологий

АВТОМАТНЫЕ МОДЕЛИ БЛОКЧЕЙН-СРЕДЫ, ХЭШИРОВАНИЕ НА ОСНОВЕ Т-ФУНКЦИЙ И ИХ ПРОЕКЦИИ

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студентки 2 курса 271 группы
направления 09.04.01 — Информатика и вычислительная техника
факультета КНиИТ
Преображенской Яны Алексеевны

Научный руководитель
доцент, к. ф.-м. н. _____ Л. Б. Тяпаев

Заведующий кафедрой
доцент, к. ф.-м. н. _____ Л. Б. Тяпаев

Саратов 2022

ВВЕДЕНИЕ

Моделирование функционирования смарт-контрактов необходимо для глубокого анализа функционирования цифровой экономики, которая в свою очередь строится на основе блокчейна с помощью автоматов с метками времени [1].

Функционирование смарт-контрактов в блокчейн-среде можно рассматривать как взаимодействие автоматов во времени, то есть автомат с метками времени можно рассматривать как релевантную модель описания такого взаимодействия. Моделирование функционирования смарт-контрактов в блокчейн-среде является одним из важных методов проверки стойкости против компрометации. Смарт-контракт использует входные данные от других смарт-контрактов и пользователей, а также о текущем времени и выдает выходные данные, которые используются другими пользователями и/или другими смарт-контрактами, а потому ошибочное функционирование одного смарт-контракта может привести к сбою в работе всех связанных с ним смарт-контрактов и узлов сети. Однако смарт-контракт может быть очень сложно устроен даже уже на уровне юридического документа, не говоря уже о его программной реализации, поэтому требуется тщательная проверка правильного функционирования смарт-контракта и в юридическом плане, и как компьютерной программы. Такую проверку достаточно сложно выполнить вручную, однако моделирование смарт-контракта позволяет поставить ряд машинных экспериментов для изучения поведения смарт-контракта как автомата при подаче на него тех или иных входных данных, то есть смоделировать его поведение в самых разных условиях.

В данной работе элементы цифровой экономики рассматриваются с использованием подхода к построению теоретико-автоматных моделей функционирования блокчейн-среды как автомата с временными метками, в котором физическое время представляется не действительными, а 2-адическими числами [2]. Итоговая модель представляет собой автомат в стандартном понимании этого термина, при этом задаваемое автоматом преобразование слов реализуется не в виде таблицы переходов состояний, а в виде программы без ветвления, представляющей собой последовательность стандартных команд процессора. К изучению таких автоматов (а значит, и к изучению функционирования смарт-контрактов в блокчейн-среде) применим развитый аппарат p-адического анализа и алгебраической динамики [3].

Практическая часть работы сфокусирована на процессе хэширования, ко-

торый является неотъемлемой частью построения блокчейна. В ракурсе теоретико-автоматного моделирования блокчейн-среды хэширование представляет собой последовательное вычисление хэш-значений T-функции. Итерирование T-функции порождает траектории (или орбиты динамической системы) в пространстве целых 2-адических чисел, к которым предъявляется ряд требований в криптографическом ракурсе, например, высокая скорость вычисления элементов траекторий, равномерное распределение траекторий, высокая линейная сложность траекторий.

Целью данной работы является анализ проекций T-функций в единичном квадрате евклидовой плоскости, на базе которых конструируются элементы блокчейн-среды. Данные проекции будут использованы для изучения распределения последовательностей пар вида $(x, f(x))$ с целью экспериментального наблюдения линейной сложности последовательностей порождаемых в процессе итерирования (например, при хэшировании) T-функции f.

Для достижения поставленной цели были выделены следующие задачи:

— изучение механизма и способов хэширования, в том числе хэширования в контексте технологии блокчейн, псевдослучайных последовательностей и линейной сложности бинарных последовательностей;
— изучение теории автоматов, а именно классических автоматов, автоматов с метками времени и асинхронных автоматов;
— изучение технология блокчейн, в том числе представление блокчейна автоматами;
— изучение смарт-контрактов, в том числе представление смарт-контрактов автоматами, способы моделирования смарт-контрактов, а также изучение в контексте смарт-контрактов автоматов с непрерывным временем и возможности продолжения автоматных функций на пространство действительных чисел;
— разработка приложения для построения проекций T-функций в единичном квадрате евклидовой плоскости.

Выпускная квалификационная работа состоит из введения, 5 разделов, заключения, списка использованных источников и 3 приложений. Общий объем работы — 64 страницы, из них 60 страниц — основное содержание, включая 20 рисунков, список использованных источников информации — 42 наименования.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Хэширование информации» посвящен основной теории по вопросам хэширования информации. Данный раздел состоит из 4 подразделов.

Первый подраздел «Хэширование в криптографии» посвящен описанию процесса хэширования, хэш-функций и сопутствующих элементов в ракурсе криптографии.

В настоящем времени криптографические хэш-функции играют фундаментальную роль. Хэширование — это важный вид криптографического преобразования и имеет огромную область применения. Однако стоит отметить, что построение качественной хэш-функции является довольно сложной задачей.

Хэш-функция $h(x)$ — это функция, которая в качестве аргумента на вход принимает строку M, которая имеет произвольную длину, и выдает на выход строку, которая также имеет фиксированную длину, в качестве результата.

Второй подраздел «Хэширование в технологии блокчейн» посвящен описанию процесса хэширования и хэш-функций в ракурсе технологии блокчейн. Также здесь описана значимость и практическое применение хэш-функций.

Хэширование в технологии блокчейн является важной и неотъемлемой частью. Хэширование применяется для адресации, формирования электронной цифровой подписи, а также для добычи криптомонет.

Хэш-функция — это отображение Ξ множества всех сообщений W, которое можно представить как множество всех конечных непустых слов над алфавитом $\mathbb{F}_2 = \{0, 1\}$, во множество $\mathbb{F}_2^m$ всех слов фиксированной длины m над этим алфавитом.

Третий подраздел «Псевдослучайные генераторы» посвящен описанию псевдослучайных генераторов как автоматов, описанию их построения и работы, также здесь описано какие псевдослучайные генераторы являются хорошими.

Одной из моделей псевдослучайных генераторов является конечный автомат

$$\mathfrak{A} =< \mathcal{S}, \mathcal{O}, S, O, s_0 >,$$

где $\mathcal{S}$ — конечное число состояний, $\mathcal{O}$ — конечный алфавит, $S : \mathcal{S} \to \mathcal{S}$ — функция перехода, $O : \mathcal{S} \to \mathcal{O}$ — функция выхода, $s_0 \in \mathcal{S}$ — начальное состояние.

Четвертный подраздел «Линейная сложность» посвящен описанию линейной сложности и линейного ранга с l ошибками.

Одной из важных мер криптографической сложности бинарной последовательности является линейный ранг или линейная сложность. Линейная сложность бинарной последовательности — число ячеек линейного регистра сдвига наименьшей длины, порождающего данную последовательность [4]. Линейная сложность дает оценку размерности линейной системы уравнений, которую необходимо решить для получения начального состояния.

Второй раздел «Автоматные модели» посвящен необходимой в контексте данной работы теории автоматов. Данный раздел состоит из 2 подразделов.

Первый подраздел «Классические автоматы» посвящен теории классических автоматов. Здесь рассматривается дискретная система с дискретным временем, детерминированный автомат с дискретным временем, автоматы-определители, автоматы-преобразователи, автоматы Мили, синхронные и асинхронные автоматы.

Теория автоматов — это раздел теории управляющих систем, которая изучает математические модели преобразователей дискретной информации, которые называются автоматами. Теория автоматов берет свое начало в середине 20-ого столетия в связи с изучением свойств конечных автоматов [5].

Второй подраздел «Автоматы с метками времени» посвящен описанию автоматов с метками времени, слов с метками времени и таблицы переходов с метками времени.

Автомат с метками времени (Т-автомат, timed automaton) — это конечный автомат, который имеет временные метки, которые представлены конечным множеством. При работе автомата значения меток времени увеличиваются с одинаковой скоростью в соответствии с увеличением реального времени. Метки времени можно сравнивать с целыми числами и сбрасывать. Сравнение временных меток дает возможность ограничивать поведение автомата, включая и отключая переходы. На переходах автомата таймеры также могут обнуляться. Таким образом, автомат с метками времени может отслеживать абсолютное время наступления отдельных событий и относительное время между событиями. Автоматы с метками времени являются подклассом гибридных автоматов [6].

На содержательном уровне автоматы с метками времени — это автоматы с двумя входами и двумя выходами, причем один из входов/выходов — это

действительные числа, а второй вход/выход — символы соответствующих конечных алфавитов [7].

Третий раздел «Технология блокчейн» посвящен описанию технологии блокчейн. Данный раздел состоит из 2 подразделов.

Первый подраздел «Историческая справка» содержит некоторые исторические факты о технологии блокчейн, описаны поколения данной технологии, принципы ее работы. Также здесь описаны криптовалюты и их характеристики.

В 1991 году ученые-исследователи Стюарт Хабер и У. Скотт Шторнетта внедрили вычислительно-практическое решение для цифровых документов с штампом времени. Это делалось для того, чтобы эти документы не могли подделываться или оформляться задним числом. Именно тогда была первые описана технология блокчейн.

Со временем технология блокчейн стала приобретать всю большую известность, в связи с чем, ее дальнейшее развитие было поделено на поколения. В настоящее время существует 3 поколения, а 4 находится в стадии разработки.

Второй подраздел «Определение блокчейна» посвящен определению блокчейна в контексте теории автоматов, то есть представление блокчейна как автомата. Также здесь рассматриваются основные составляющие блокчейна и его характеристики.

Блокчейн в общем случае — это конечная последовательность двоичных слов B_i, $i = 0, 1, 2, \ldots$, эти слова называются блоками, в которой каждый блок B_i, есть конкатенация

$$B_i = \begin{cases} w_i \circ t_i \circ \Xi(B_{i-1}) \circ \Xi(w_i \circ t_i \circ \Xi(B_{i-1})), \text{ если } i > 0; \\ w_0 \circ t_0 \circ \Xi(w_0 \circ t_0), \text{ если } i = 0, \end{cases}$$

где w_i (при $i > 0$) — слово длины $L - k - 2m$ (список транзакций, информационный блок), w_0 — случайное слово длины $L - k - m$, t_i — слово длины k (метка времени), Ξ — хэш-функция.

Четвертый раздел «Смарт-контракт» посвящен рассмотрению смарт-контрактов. Данный раздел состоит из 5 подразделов.

Первый подраздел «Определение смарт-контракта» посвящен неформальному определению смарт-контракта. Здесь приведены некоторые исторические справки по развитию смарт-контрактов, рассматриваются их достоинства, при-

менение в реальной жизни и перспективы развития. Также рассматривается определение распределенных реестров и их принципы.

В 1996 году ученым-программистом Ником Сабо впервые был озвучен термин «смарт-контракт». Он считал, что можно достаточно сильно улучшить бумажные юридические контракты с помощью смарт-контрактов, которые разрабатываются с помощью механизмов цифровой безопасности.

Второй подраздел «Представление смарт-контракта как автомата» посвящен изучению вопроса представления смарт-контракта как автомата. Также рассматриваются опасности, которые могут возникнуть при их использовании и задачи, которые необходимо решить для устранения этих опасностей.

Любой юридически правильно составленный контракт можно представить в виде автомата с метками времени [8].

Смарт-контракт — это программная реализация автомата с метками времени. Отличается он лишь тем, что никакие юридические споры по исполнению смарт-контракта не предусмотрены — все действия по контракту осуществляются автоматически. Все транзакции могут быть записаны в общий блокчейн.

Третий подраздел «Способы моделирования смарт-контрактов» посвящен изучению вопроса моделирования смарт-контрактов. Здесь рассматриваются такие способы как моделирование с помощью автоматов с метками времени и с помощью Т-функций, рассматриваются преимущества данных подходов.

Если рассматривать правильно составленные с юридической точки зрения контракты с точки зрения математического описания и моделирования, то их можно представить как конечный автомат. При этом также известны методы получения описания смарт-контрактов из функционирующей во времени блокчейн-среды как конечных автоматов. Исходя из этого можно сделать вывод, что функционирование смарт-контрактов в блокчейн-среде можно рассматривать как взаимодействие автоматов во времени, то есть автоматы с метками времени являются релевантной моделью описания такого взаимодействия [2].

Также можно использовать абсолютно другой подход к построению теоретико-автоматных моделей функционирования блокчейн-среды, в том числе, функционирования смарт-контрактов в этой среде, как автомата с метками времени, в котором физическое время представляется не действительными, а 2-адическими числами.

Автоматы-определители и автоматы-преобразователи могут быть сведе-

ны один к другому [9]. Таким образом, в случае $p = 2$ задачи о d-автоматах и распознаваемых ими языках могут быть сведены к задачам о функциях, удовлетворяющих 2-адическому условию Липшица с константой 1. Такие функции также называются функциями треугольного вида, двоичными совместимыми функциями, T-функциями.

Четвертый подраздел «Автоматы с непрерывным временем» посвящен изучению вопроса отличия непрерывного времени от дискретного.

Отличие дискретного времени от непрерывного состоит в том, что для любого данного момента дискретного времени существует момент, непосредственно следующий за ним.

Непрерывность времени означает, что время представляет собой метрическое пространство $\mathbb{T}$ с метрикой τ такой, что для любого действительного $\epsilon > 0$ и любого $t \in \mathbb{T}$ найдется $t_1 \in \mathbb{T}$ такое, что $0 < \tau(t, t_1) < \epsilon$.

Пятый подраздел «Автоматные функции, продолженные на пространство действительных чисел» посвящен изучению вопроса отображения рациональных чисел в рациональные.

Поскольку множество $\mathbb{Z}_p \cap \mathbb{Q}$ рациональных целых p-адических чисел всюду плотно в $\mathbb{Z}_p$, 1-Липшицева функция $f : \mathbb{Z}_p \to \mathbb{Z}_p$ однозначно задается своим ограничением на $\mathbb{Z}_p \cap \mathbb{Q}$. Автоматная функция $f_{\mathfrak{A}}$, которая задается конечным автоматом отображает рациональное целое p-адическое число в рациональное целое p-адическое. Это значит, что $f_{\mathfrak{A}}(\mathbb{Z}_p \cap \mathbb{Q}) \subset \mathbb{Z}_p \cap \mathbb{Q}$. Обратное неверно. В общем случае, автоматные функции f такие, что $f(\mathbb{Z}_p \cap \mathbb{Q}) \subset \mathbb{Z}_p \cap \mathbb{Q}$ не описаны.

Пятый раздел «Построение проекций T-функций» посвящен практической части данной работы. Данный раздел содержит 3 подраздела.

Первый подраздел «Необходимая теория» содержит теорию, которая не была освящена раньше, но понадобится при построении проекций T-функций, а именно расчет координат точек, по которым будут строиться графики.

Смарт-контракт базируется на блокчейне, который в свою очередь использует хэширование своих блоков. Хэширование осуществляется с помощью хэш-функции, и она, на самом деле, является T-функцией, то есть является автоматной или 1-Липшицевой. T-функции также применяются в криптографических примитивах, при синтезе поточных шифраторов и в генераторах псевдослучайных чисел.

Второй подраздел «Программная реализация» посвящен описанию разра-

ботанной программы. Здесь подробно описаны все существующие в программе классы и их взаимодействие.

Для построения проекций функций была написана программа на языке программирования Python [10]. Для построения и отображения графиков была использована библиотека matplotlib [11], а для реализации пользовательского интерфейса использовалась библиотека PyQt5 [12]. Проект состоит из 3 файлов: `Poly.py`, `Interface.py`, `Main.py`. Каждый файл содержит класс, который отвечает за определенную часть работы приложения.

Третий подраздел «Анализ проекций T-функций» содержит примеры проекций, которые были построены в результате работы программы. В качестве примеров были взяты различные функции с разной точностью (количеством младших разрядов) k. Получившиеся графики были проанализированы и в результате были сформулированы некоторые выводы.

При анализе полученных проекций было замечено, что некоторые проекции совпадают, не смотря на то, что функции, которые задают эти графики, отличаются. Совпадение графиков в ракурсе автоматного моделирования смарт-контрактов означает, что различные автоматы в процессе функционирования переходят в один и то же минимальный подавтомат.

Это означает, что при (возможно, длительном) функционировании различных смарт-контрактов цифровая экономика может перейти в такой режим (с точки зрения автоматной модели — в минимальный подавтомат), что логика функционирования смарт-контрактов будет неотличима. Более того, как только экономика перейдет в такой режим, выйти из этого режима уже будет не возможно. Решение этой проблемы, вероятно, следует искать в новом представлении автоматов, например, продолжив автоматы до отображений из $\mathbb{R}$ в $\mathbb{R}$.

Для хэширования нужна хорошая T-функция. Она должна:

— быть эргодической, а следовательно, транзитивной;

— быть быстровычислимой, а следовательно, полиномы степени выше 2 нам не подходят, и нужно уметь строить T-функцию из простых и быстровычислимых операций;

— иметь высокую линейную сложность, желательно бесконечную, однако даже для полиномов степени 2 окончательного понимания в этом вопросе до сих пор нет.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы была достигнута поставленная цель — анализ проекций T-функций в единичном квадрате евклидовой плоскости, на базе которых конструируются элементы блокчейн-среды: распределенные реестры, смарт-контракты, хэш-функции. Данные проекции были использованы для изучения распределения последовательностей пар вида $(x, f(x))$ с целью экспериментального наблюдения линейной сложности последовательностей порождаемых в процессе итерирования (например, при хэшировании) T-функции f.

Для достижения поставленной цели были решены следующие задачи:

- изучение механизма и способов хэширования, в том числе хэширования в контексте технологии блокчейн, псевдослучайных последовательностей и линейной сложности бинарных последовательностей;
- изучение теории автоматов, а именно классических автоматов, автоматов с метками времени и асинхронных автоматов;
- изучение технология блокчейн, в том числе представление блокчейна автоматами;
- изучение смарт-контрактов, в том числе представление смарт-контрактов автоматами, способы моделирования смарт-контрактов, а также изучение в контексте смарт-контрактов автоматов с непрерывным временем и возможности продолжения автоматных функций на пространство действительных чисел;
- разработка приложения для построения проекций T-функций в единичном квадрате евклидовой плоскости.

В ходе изучения способов моделирования смарт-контрактов мы пришли к выводу, что для моделирования функционирования блокчейн-среды и моделирования работы смарт-контрактов нет необходимости использовать сложные и ресурсоемкие модели, которые основаны на концепции автоматов с метками времени, достаточно ограничиться моделированием этой среды с помощью T-функций. Эти функции могут быть реализованы в виде программ без ветвления, выполненных как последовательности стандартных команд. Эти программы имеют относительно простую реализацию и высокое быстродействие.

Также на языке программирования Python была написана программа для построения проекций T-функций в единичном квадрате евклидовой плоскости.

Мы пришли к выводу, что для хэширования требуется эргодическая T-

функция с высокой линейной сложностью. А именно, нам нужны эргодические T-функции, порождающие последовательности, линейная сложность которых была бы не конечной, а желательно — бесконечной.

При рассмотрении получившихся графиков можно обратить внимание на то, что все функции имеют линейную структуру и все точки попадают на конечное число параллельных прямых, из чего следует, что распределение пар является очень плохим. Также необходимо заметить, что некоторые функции имеют одинаковые проекции. В ракурсе автоматного моделирования смарт-контрактов означает, что различные автоматы в процессе функционирования переходят в один и то же минимальный подавтомат.

Такой переход означает, что при (возможно, длительном) функционировании различных смарт-контрактов цифровая экономика может перейти в такой режим (с точки зрения автоматной модели — в минимальный подавтомат), что логика функционирования смарт-контрактов будет неотличима. Более того, как только экономика перейдет в такой режим, выйти из этого режима уже будет не возможно. Решение этой проблемы, вероятно, следует искать в новом представлении автоматов, например, продолжив автоматы до отображений из $\mathbb{R}$ в $\mathbb{R}$.

Для обозначения того, что T-функция в некотором смысле является хорошей, были сформулированы следующие условия:

- T-функция должна быть эргодической, а следовательно, транзитивной;
- T-функция должна быть быстровычислимой, а следовательно, полиномы степени выше 2 нам не подходят, и нужно уметь строить T-функцию из простых и быстровычислимых операций;
- T-функция должна иметь высокую линейную сложность, желательно бесконечную.

К сожалению, задача поиска T-функции, которая бы отличалась быстродействием в техническом плане и гарантировала бы хорошее распределение пар, до сих пор остается до конца не решенной.

Основные источники информации:

1 *Andrychowicz, M.* Modelling bitcoin contracts by timed automata / M. Andrychowicz, S. Dziembowski, D. Malinowski, L. Mazurek // *Lecture notes in computer science ser. Springer*. — 2014. — no. 8711. — Pp. 7–22.

2 *Анашин, В. С.* О теоретико-автоматных моделях блокчейн-среды / В. С. Анашин // *Информатика и ее применения*. — 2019. — Т. 13, № 2. — С. 29–

36.

3 *Anashin, V.* Applied algebraic dynamics / V. Anashin, A. Khrennikov // *Gruyter expositions in mathematics*. — 2009. — Vol. 49. — 533 pp.

4 *Klimov, A.* A new class of invertible mappings / A. Klimov, A. Shamir // *Cryptographic Hardware and Embedded Systems 2002*. — 2002. — Vol. 2523. — Pp. 470–483.

5 *Кудрявцев, В. Б.* Элементы теории автоматов / В. Б. Кудрявцев, С. В. Алешин, А. С. Подколзин. — Москва: Издательство Московского университета, 1978. — 216 с.

6 *Dong, J. S.* Timed automata patterns / J. S. Dong, P. Hao, S. Qin, J. Sun, W. Yi // *IEEE Transactions on software engineering*. — 2008. — Vol. 34, no. 6. — Pp. 844–859.

7 *Alur, R.* The theory of timed automata / R. Alur, D. Dill // *Lecture Notes in Computer Science*. — 1992. — Vol. 600. — Pp. 45–73.

8 *Flood, M. D.* Contract as automaton: The computational representation of financial agreements / M. D. Flood, O. R. Goodenough // *Office of Financial Research, US Department of the Treasury*. — 2015. — no. 15–04. — 25 pp.

9 *Allouche, J.-P.* Automatic sequences. Theory, applications, generalizations / J.-P. Allouche, J. Shallit. — Cambridge: Cambridge University Press, 2003. — 583 pp.

10 Welcome to Python.org [Электронный ресурс]. — URL: https://www.python.org/ (Дата обращения 20.05.2022). Загл. с экр. Яз. англ.

11 Matplotlib — Visualization with Python [Электронный ресурс]. — URL: https://matplotlib.org/ (Дата обращения 20.05.2022). Загл. с экр. Яз. англ.

12 PyQt5 — PyPI [Электронный ресурс]. — URL: https://pypi.org/project/PyQt5/ (Дата обращения 20.05.2022). Загл. с экр. Яз. англ.