УДК 004.75 DOI: 10.17587/it.26.360-367

И. М. Шилов, аспирант, инженер, e-mail: ilia.shilov@yandex.ru,
Д. А. Заколдаев, канд. техн. наук, доц., e-mail: d.zakoldaev@mail.ru,
Университет ИТМО, г. Санкт-Петербург

Многомерный блокчейн и его преимущества

Рассмотрены ключевые недостатки устойчивых распределенных реестров, основанных на технологии блокчейн. В качестве одного из потенциальных способов их устранения предложен новый подход к построению устойчивых распределенных реестров за счет модификации существующих решений — многомерный блокчейн. Проведена численная оценка ключевых характеристик предлагаемого решения: числа транзакций в единицу времени, периода генерации блоков и объема хранимой узлами сети информации. Полученные характеристики сопоставлены с аналогами.

Ключевые слова: *блокчейн, устойчивый распределенный реестр, многомерный блокчейн, масштабирование, адресация, транзакции, число транзакций в единицу времени, стойкость, целостность, доступность*

Введение

В настоящее время трудно переоценить значение криптографических валют и степень их влияния на развитие информационных технологий и, в частности, на построение децентрализованных систем. Ключевым преимуществом подобных систем является использование механизмов достижения взаимопонимания между отдельными узлами, не доверяющими друг другу и действующими в ненадежной среде [1, 2]. Неоднократно отмечалось, что использование блокчейна не ограничивается средой криптовалют, поскольку данная технология является лишь способом построения децентрализованной системы, а не конкретной реализацией прикладного протокола [3—5].

Развитию технологии и повсеместному применению препятствуют недостатки, среди которых наиболее критичны неограниченный рост объема блокчейна, неэффективные механизмы достижения консенсуса и наличие посредников при межсистемном обмене. Рост объема вследствие затруднения хранения всей базы данных транзакций влечет частичную централизацию, а наличие посредников негативно влияет на доступность системы и конфиденциальность информационных потоков.

Устранение приведенных недостатков явилось одним из основных направлений развития устойчивых распределенных реестров в последние годы. Математическое доказательство безопасности системы на основе доказательства доли владения открыло путь к замене энергозатратного механизма "доказательство работы" практически не требующим вычислительных ресурсов механизмом "доказательство доли владения" [6-7]. Неконтролируемые транзакции устраняются такими политиками, как KYC (Know Your Customer) и AML (Anti-Money Laundering), что позволяет привести системы на основе распределенных реестров в соответствие законодательству.

Наиболее существенным недостатком является увеличение объема хранимой узлами сети информации. Хотя попытки устранения данной проблемы предпринимаются постоянно, полноценное решение представлено не было. В табл. 1 приведены некоторые характеристики существующих решений. На рис. 1 приведены графики роста объема блокчейна для различных систем на начало 2020 г. Экстраполяция результатов не представляется возможной вследствие зависимости характеристик исследуемых систем от их популярности, что влечет нелинейный характер изменения размеров блокчейнов. Например, для блокчейна Bitcoin график приведен на рис. 2 [8].

Проблема перевода средств между фидуциарными денежными системами и криптова-

Таблица 1

Характеристики различных систем на основе блокчейна в апреле 2020 года

Характеристики	Блокчейн				
	Bitcoin	Ethereum	Bitcoin Cash	Litecoin	Dash
Размер блокчейна, Гб	270,14	667,1	165,09	20,23	8,05
Размер одного блока (средний), Кб	1100	23	646	18	24
Время создания одного блока (среднее)	10 мин	15 с	10 мин	2,5 мин	2,5 мин
Число транзакций в секунду	3,46	8,34	0,22	0,28	0,23

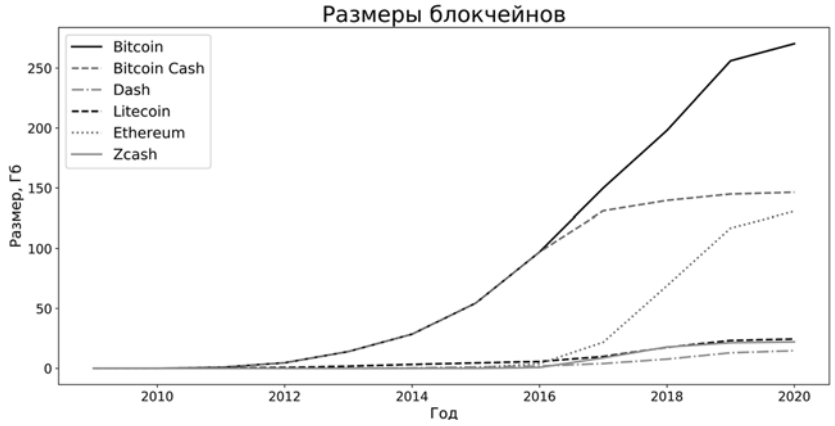

Рис. 1. Зависимость объемов некоторых блокчейнов от времени

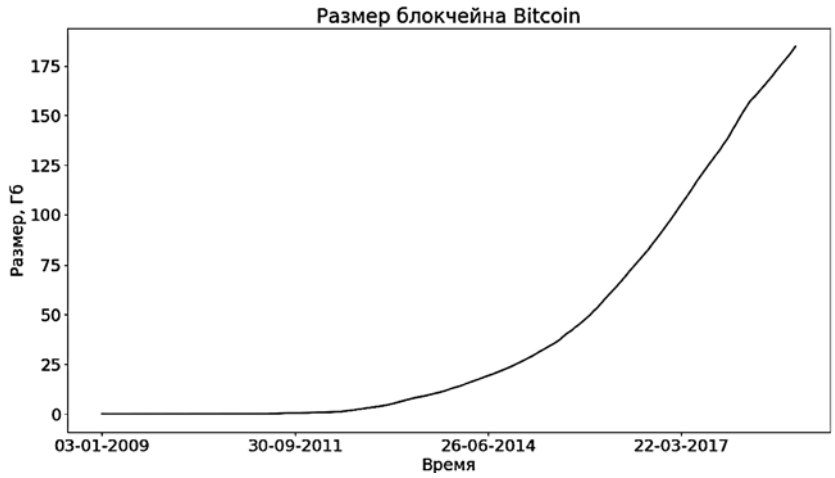

Рис.2. Размер блокчейна Bitcoin

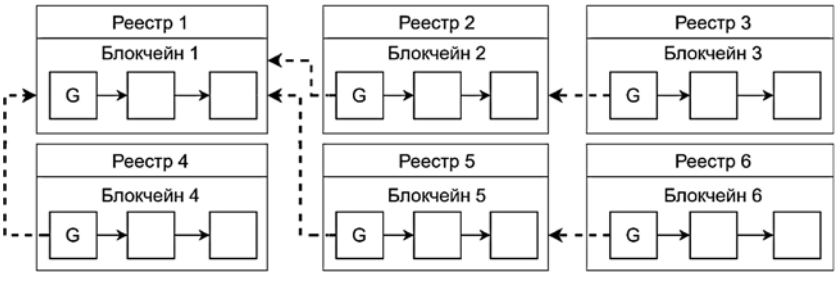

Рис. 3. Многомерный блокчейн

лютами или разными криптовалютами решается с привлечением третьих лиц или с использованием сайдчейнов, которые не представляют собой обобщенный подход и функционируют исключительно как средство обмена между двумя отдельными системами [9].

В работе предложен подход к решению некоторых описанных проблем, призванный снизить негативное влияние на конфиденциальность, целостность и доступность систем, построенных на базе устойчивых распределенных реестров. Предлагаемый подход позволяет реализовать безопасный межсистемный обмен, масштабирование и объединение отдельных систем в надсистему.

Структура многомерного блокчейна

Многомерной блокчейн представляет собой множество блокчейнов, в котором все блокчейны, за исключением первого, проходят процедуру регистрации в одном из существующих блокчейнов. Регистрация подразумевает внесение информации о генезис-блоке (первом блоке нового блокчейна) и базовых свойствах блокчейна в другой блокчейн. Понятие распределенного реестра при этом становится двояким: с одной стороны, многомерный блокчейн реализует распределенный реестр, с другой стороны каждый блокчейн в пределах многомерного блокчейна сам реализует распределенный реестр.

В зависимости от архитектуры возможны два режима функционирования многомерного блокчейна: режим блоков и режим состояний. Обобщенное представление многомерного блокчейна приведено на рис. 3. Каждый блокчейн в пределах многомерного блокчейна реализует распределенный реестр. Это предположение позволяет не затрагивать особенности функционирования отдельного блокчейна [10, 11].

Режим блоков подразумевает использование понятия типа блока. Типом называется набор полей структуры данных и ассоциированных с ней методов. Данная особенность является ключевым отличием решения от сайдчейнов, в которых пользователи одной из систем могут не обладать информацией о заморозке средств для использования в сайдчейне. Многомерный блокчейн в данной модели представляет собой разреженную матрицу, в которой каждый генезис-блок, кроме первого, помещается в существующий блокчейн в виде блока особого типа. Факт регистрации остается прозрачным для нового блокчейна, поскольку в дальнейшем он может функционировать самостоятельно, не прибегая к использованию функциональности многомерного блокчейна.

Основанная на модели функционирования Ethereum модель состояний является более совершенным и перспективным подходом. Ключевым ее преимуществом является принципиальная возможность верификации транзакций на основе информации только из последнего блока, что является следствием использования концепции виртуальных состояний. Также в этой модели реализованы Тьюринг-полные языки программирования, позволяющие осуществлять взаимодействие нескольких участников с применением смарт-контрактов. Состояние многомерного блокчейна определяется совокупностью состояний отдельных его компонентов, причем каждая система самостоятельно осуществляет дискретные переходы между состояниями. Периодичность переходов между состояниями определяется каждым вложенным блокчейном самостоятельно, а период перехода многомерного блокчейна равен наибольшему общему делителю всех периодов перехода вложенных блокчейнов.

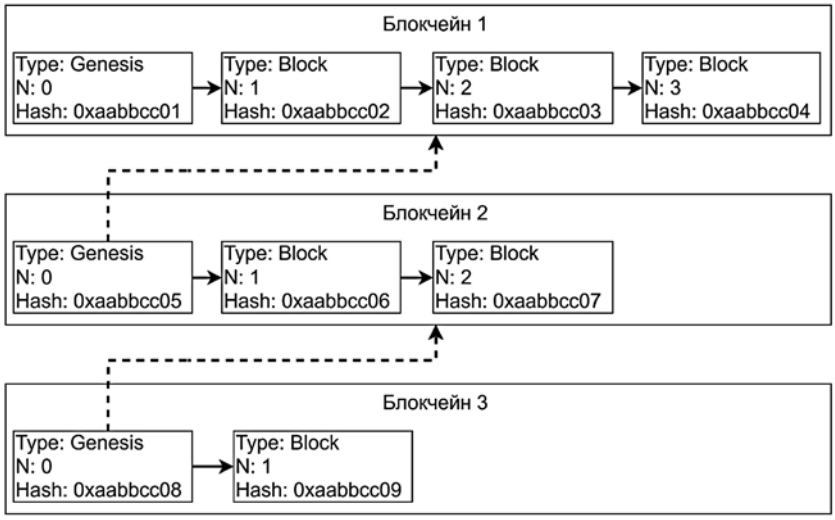

Рис. 4. Пример адресации в многомерном блокчейне в режиме машины состояний

Адресация в многомерном блокчейне

Наиболее существенным преимуществом многомерного блокчейна является адресация, которая оказывает влияние на принципы построения приложений. Она используется для передачи средств между блокчейнами и формирования логической структуры системы. Адресация осуществляется иерархически и напоминает порядок адресации файлов и директорий в иерархических файловых системах. Форма адресации задается для каждого приложения произвольно. Можно выделить два типа адресации в наиболее общем приближении: абсолютная адресация — относительно первого генезис-блока, и относительная адресация — в пределах текущего блокчейна. При этом также возможны варианты записи адреса — в форме номера (метки) или в форме хэш-суммы. В режиме блоков и режиме машины состояний адресация работает по-разному.

В режиме блоков дочерние блокчейны адресуются номерами блоков, в которых осуществлена регистрация, и их хэш-суммами. Подход с использованием хэш-сумм предпочтителен, поскольку позволяет избежать полного обхода всей цепочки блоков.

В режиме машины состояний каждый блокчейн регистрируется в структуре данных родительского блокчейна. Ссылка на дочерний блокчейн осуществляется с помощью хэш-суммы генезис-блока. Вероятность появления коллизии в криптографических хэш-функциях пренебрежимо мала. Это позволяет гарантировать уникальность хэш-суммы генезис-блока в пределах многомерного блокчейна с вероятностью, близкой к единице, при условии уникальности генезис-блока.

Адрес в пределах многомерного блокчейна представляется в текстовой форме. При этом адрес строится следующим образом:

$$DM.A.E.T.V\{DM.A.E.T.V\}^* \qquad (1)$$

Рассмотрим адресацию на примере блока 0xaabbcc08 в блокчейне с режимом работы на основе состояний (рис. 4):

Таблица 2

Нотация для адресации

Условное обозначение	Расшифровка	Множество допустимых значений	Описание
D	Delimiter	[/:\]'"*a* – *zA* – *Z*]	Разделитель, используется для перехода на следующий уровень иерархии (первый разделитель может быть опущен, если адресуется блок в текущем одномерном блокчейне)
M	Mode	[*sb*]	Режим работы блокчейна
A	Addressing mode	[*ar*]	Режим адресации — абсолютный и относительный
E	Entity	[*abct*]	Сущность
T	Type	[*NHLA*]	Тип адреса
V	Value	[a – zA – Z0 – 9]*	Значение

```
Функция CHECK-OUT
Входные параметры:      ExtTX - внешняя транзакция
Выходные параметры:     None
1. Для каждого получателя проверить допустимость выполнения
транзакции. При неудаче прекратить исполнение.
2. Проверить синтаксическую и семантическую корректность
транзакции. При неудаче прекратить исполнение.
3. Проверить допустимость совершения транзакции
пользователем-инициатором. При неудаче прекратить
исполнение.

Функция SEND-EXT-TX
Входные параметры:      ExtTX - внешняя транзакция
Выходные параметры:     None
1. Вызвать CHECK-OUT(ExtTX)
2. Применить транзакцию к реестру
3. Если флаг NOTIFY_ACCEPTORS установлен, то отправить
уведомление в реестр каждого получателя.

Функция CHECK-IN
Входные параметры:  ExtTX - внешняя транзакция
1. Для реестра-инициатора проверить допустимость
выполнения транзакции. При неудаче прекратить
исполнение.
2. Установить таймер приема внешней транзакции.
3. Для реестра-инициатора проверить наличие внешней
транзакции с использованием протокола поиска в
многомерном блокчейне. При неудаче прекратить
исполнение.
4. По истечении времени таймера проверить отсутствие
приема транзакции в текущем реестре. При неудаче
прекратить исполнение.

Функция SEND-EXT-TX
1. Получить транзакцию ExtTX от пользователя или из
реестра через уведомление.
2. Вызвать CHECK-IN(ExtTX).
3. Применить транзакцию к реестру.
```

Рис. 5. Псевдокод работы реестра-отправителя и реестра-получателя

/s.a.c.H.0xaabbcc05/s.a.c.H.0xaabbcc08/s.a.b.N.0

/s.a.c.H.0xaabbcc05/s.a.c.H.0xaabbcc08/s.a.b.H.0xaabbcc08

Обозначения раскрыты в табл. 2.

Внешние транзакции в многомерном блокчейне

Под транзакцией в распределенных системах понимается упорядоченный набор применяемых к состоянию преобразований. Сообщением является структурированный набор данных, пересылаемый в пределах сети передачи информации. В данной работе под внешней транзакцией подразумевается упорядоченная последовательность логических операций, охватывающая два и более распределенных реестра. Инициатором называется реестр, в котором начинается внешняя транзакция, а остальные реестры (в том числе, при необходимости, исходный) — получателями или акцепторами. Соответственно, любая внешняя транзакция включает две фазы: инициацию и акцепт. При этом получателей в любой внешней транзакции может быть много, однако инициатор всегда один.

Упрощенный порядок работы реестра-отправителя и реестра-получателя представлены на рис. 5.

Преимущества многомерного блокчейна

Можно выделить несколько преимуществ многомерного блокчейна перед аналогами:

1) децентрализованное принятие решений (аналогично одномерному блокчейну) [12, 13];

2) безопасный обмен информацией между системами;

3) масштабирование существующих решений;

4) увеличение скорости генерации информации при сохранении исходных значений параметров, влияющих на свойства безопасности.

Первое и второе преимущества следуют из принципов построения системы и порядка ее работы. Одноранговые сети и механизмы достижения консенсуса обеспечивают децентрализацию. Этому посвящено множество работ по доказательству работы, доказательству доли владения и византийским протоколам. Безопасный обмен информацией обеспечивается протоколом проверки наличия транзакций во внешнем реестре при условии сохранения протоколом свойств устойчивых распределенных реестров [14].

При замене одномерного блокчейна на многомерный аналог возможно достичь экономии памяти отдельными узлами вычислительной сети. Наиболее существенные факторы, влияющие на размер базы данных реестра, — размер блока и период генерации требуемого для формирования блока числа транзакций. Используются следующие условные обозначения:

- LV — объем реестра;
- N_L — число реестров;
- T — период генерации транзакций отдельным аккаунтом;
- T_0 — период генерации транзакции системой;
- v — частота генерации транзакций отдельным аккаунтом;
- v_0 — частота генерации транзакций системой;
- N_A — число аккаунтов;
- p_i — доля аккаунтов, перешедших в дочерний блокчейн;
- t — время;
- s — размер одной транзакции;
- S — размер блока;
- N_{TX} — число транзакций в блоке;
- T_s — продолжительность временного слота.

Масштабирование распределенного реестра с использованием многомерного блокчейна влечет разделение всех пользователей на группы. Количество генерируемой информации в каждом вложенном реестре оказывается меньше, чем количество генерируемой информации в исходной системе. Возможны два варианта изменения исходных параметров:

1) сохранение неизменного размера блоков при увеличении периода их генерации. Уменьшение числа аккаунтов приводит к снижению общего числа транзакций в единицу времени, следствием чего является уменьшение скорости создания блоков прежней величины;

2) сокращение размера блоков при сохранении скорости генерации блоков. Некоторые механизмы достижения консенсуса и прикладные приложения требуют поддержания постоянной скорости генерации блоков. Соблюдение этого требования повлечет уменьшение размера блоков при неизменном числе транзакций в единицу времени в рамках всей системы.

Эти условия не могут выполняться одновременно, поскольку одновременное изменение обоих параметров не допускается современными механизмами достижения консенсуса.

Уменьшение периода генерации блоков. Пусть исходный реестр разделен на N_L реестров. Доля перемещенных в каждый реестр аккаунтов определяется следующим соотношением:

$$p_i = \frac{N_A^{(i)}}{N_A}. \tag{2}$$

Поскольку добавления новых аккаунтов не происходит, то

$$\sum_i p_i = \frac{\sum_i N_A^{(i)}}{N_A} = 1. \tag{3}$$

Частота генерации транзакций представляет собой число транзакций, генерируемых в единицу времени, что для исходного реестра определяется соотношением

$$v = \frac{1}{T} \Rightarrow v_0 = N_A v = \frac{N_A}{T}. \tag{4}$$

Следовательно, период генерации транзакции системой равен

$$T_0 = \frac{1}{v_0} = \frac{T}{N_A}. \tag{5}$$

Объем реестра определяется по формуле

$$LV = \left[\frac{t}{T_0}\right] S, \tag{6}$$

где [.] — целая часть числа.

Число аккаунтов в каждом новом реестре меньше исходного, при этом выполняется следующее соотношение:

$$N_A^{(i)} = p_i N_A. \tag{7}$$

Соответственно увеличивается период генерации транзакций при уменьшении числа транзакций в единицу времени:

$$T_0^{(i)} = \frac{T}{N_A^{(i)}} = \frac{T}{N_A}\frac{1}{p_i} = \frac{T_0}{p_i}; \tag{8}$$

$$v_0^{(i)} = \frac{p_i}{T_0}. \tag{9}$$

Объем отдельного реестра удовлетворяет соотношению

$$LV^{(i)} = \left[\frac{t}{T_0^{(i)}}\right] S = \left[\frac{t}{T_0}\right] p_i S = LV \cdot p_i. \tag{10}$$

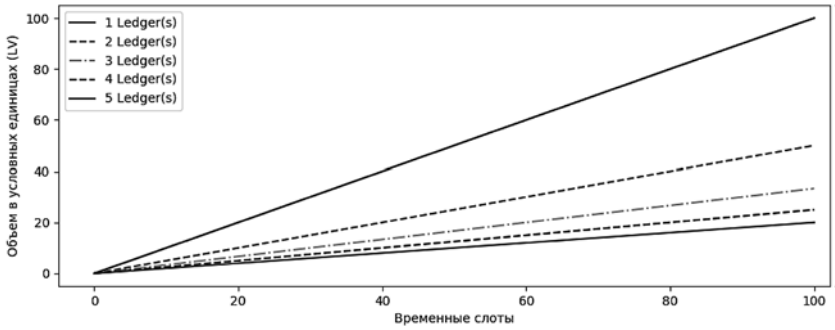

Рис. 6. Объем информации, хранимой узлами многомерного блокчейна

Следовательно, среднее значение объема реестра в произвольный момент времени равно

$$\overline{LV} = \frac{\sum\limits_i LV^{(i)}}{N_L} = \frac{LV\sum\limits_i p_i}{N_L} = \frac{LV}{N_L}. \qquad (11)$$

В среднем объем отдельного реестра при применении многомерного блокчейна оказывается меньше в N_L раз. Пример для линейного роста объема реестра приведен на рис. 6.

Уменьшение размеров блоков. Пусть неизменным остается период генерации блоков в каждом реестре. Тогда размер каждого блока равен

$$S = sN_{TX} = sN_A vT_s. \qquad (12)$$

Уменьшение числа аккаунтов приводит к снижению числа транзакций и, соответственно, размеров блока:

$$S^{(i)}) = sN_{TX}^{(i)} = sN_A^{(i)} vT_s = \frac{S}{N_a} N_A^{(i)} = Sp_i. \qquad (13)$$

Тогда объем отдельного реестра составляет

$$LV^{(i)} = \left[\frac{t}{T}\right] S^{(i)} = \left[\frac{t}{T}\right] Sp_i = LVp_i. \qquad (14)$$

Среднее значение объема реестра в произвольный момент времени равен

$$\overline{LV} = \frac{\sum\limits_i LV^{(i)}}{N_L} = \frac{LV\sum\limits_i p_i}{N_L} = \frac{LV}{N_L}. \qquad (15)$$

Следовательно, уменьшение размера блоков и увеличение периода генерации транзакций оказывают сопоставимое влияние на объем информации, хранимой каждым поддерживающим систему узлом. Стоит отметить, что данная оценка не учитывает применение таких механизмов, как шардинг, который позволяет еще более сократить объем хранимой информации.

График изменения скорости роста объема реестра при последовательном увеличении числа дочерних реестров приведен на рис. 7. В данном случае предполагается линейная зависимость количества хранимой информации от времени и равномерное распределение узлов по реестрам. При правильном выборе момента для создания нового реестра средний объем хранимой отдельным узлом вычислительной системы информации может поддерживаться относительно постоянным.

Данная модель не учитывает того факта, что объем информации в начале работы системы больше единицы. Многомерный блокчейн подразумевает постепенное создание новых вложенных блокчейнов. Одним из возможных направлений для исследования может стать оптимизационная задача поиска оптимального момента времени для разделения блокчейна и требуемого числа вложенных блокчейнов.

Рассмотрим увеличение числа аккаунтов. Предложенная ранее модель основана на соотношении (2). Число аккаунтов в системе должно оставаться неизменным, что не отражает реального положения дел, когда для каждого реестра характерно увеличение не только числа транзакций, но и числа аккаунтов. В этом случае соотношение (2) более не выполняется и заменяется следующим соотношением:

$$\sum_i p_i = \frac{\sum\limits_i N_A^{(i,k)}}{N_A} = \frac{\sum\limits_i N_A^{(i)} + k}{N_A} = 1 + \frac{k}{N_A}. \qquad (16)$$

Иными словами, совокупное число аккаунтов увеличивается на k. Средняя нагрузка на узлы вычислительной системы в многомерном блокчейне определяется сотношением

$$\overline{LV'} = \frac{\sum\limits_i LV_i}{N_L} = \frac{\sum\limits_i p_i \cdot LV}{N_L} = \frac{LV}{N_L}\left(1 + \frac{k}{N_A}\right), \qquad (17)$$

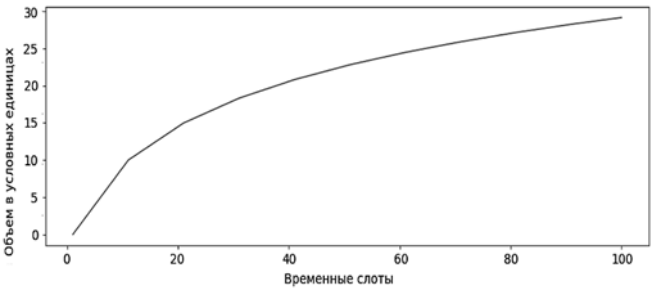

Рис. 7. Объем информации при последовательном увеличении числа реестров

тогда как для одномерного блокчейна — соотношением

$$T_0 = \frac{T}{N_A + k} \Rightarrow \overline{LV} = LV = \frac{t}{T_0} S = \\ = \frac{t(N_A + k)}{T} S = LV\left(1 + \frac{k}{N_A}\right). \quad (18)$$

Средняя нагрузка на узлы вычислительной системы в случае многомерного блокчейна оказывается меньше в N_L раз, причем эффект более выражен в случае равномерного распределения узлов по вложенным блокчейнам. График зависимости объема хранимой отдельными узлами информации в данном случае не отличается от случая с постоянным числом пользователей (см. рис. 5), поскольку для произвольного числа реестров множитель $\left(1 + \frac{k}{N_A}\right)$ является константой.

Рассмотрим число транзакций в единицу времени. Данный показатель является ключевым при описании преимуществ блокчейна перед иными способами построения реестров. Измерение точных значений затруднено необходимостью точной оценки числа узлов в вычислительной системе, что невозможно для распределенных систем без ограничений, накладываемых на участие. Однако оценка изменения числа транзакций (TPS) в единицу времени в многомерном блокчейне относительно одномерного аналога возможна. Эта оценка приведена на рис. 8. В каждом временном слоте число блокчейнов возрастает на единицу. На рис. 8, *а* демонстрируется рост общего числа транзакций в единицу времени при условии сохранения прироста числа транзакций в каждом блокчейне. Рис. 8, *б* отражает рост числа транзакций в единицу времени в системе в целом при неизменной нагрузке на каждый блокчейн. На рис. 8, *в* отражено снижение нагрузки на каждый блокчейн в отдельности при росте числа блокчейнов с учетом ограниченности их ресурсов и в условиях постоянного роста числа транзакций (до определенного граничного значения — в данном случае 0,75). Рис. 8, *г* отражает снижение нагрузки на каждый блокчейн в отдельности при постоянной нагрузке на систему.

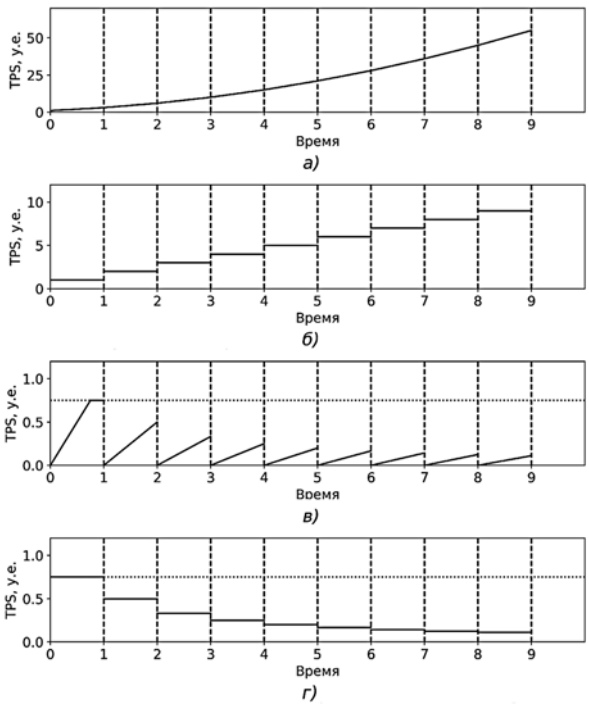

Рис. 8. Показатель TPS в многомерном блокчейне:
а — совокупный TPS при росте TPS в каждом блокчейне; *б* — совокупный TPS при неизменном TPS в каждом блокчейне; *в* — TPS в отдельном блокчейне при росте совокупного TPS; *г* — TPS в отдельном блокчейне при постоянном совокупном TPS

Выводы

Предложенная технология — многомерный блокчейн — является развитием концепции одномерного блокчейна, используемой в основе многих современных распределенных реестров. Решение проблем масштабирования и безопасного обмена информацией между реестрами позволит значительно ускорить внедрение технологии на практике. Применение многомерного блокчейна возможно в различных сферах — в банковском и финансовом секторах экономики, системах управления доменными именами, системах управления версиями программного обеспечения, системах государственного управления, системах управления производством и т.д.

В то же время, для многих существующих систем на основе блокчейна не была проведена оценка безопасности их функционирования, что теоретически может привести к их компрометации. Поэтому перед реализацией систем на основе многомерного блокчейна и их использованием требуется оценить безопасность предлагаемого решения с использованием теории вероятностей и фреймворка универсальной композиции [15].

Список литературы

1. **Cachin I., Guerraoui R., Rodrigues L.** Introduction to Reliable and Secure Distributed Programming. Springer-Verlag Berlin Heidelberg, 2011.

2. **Равал С.** Децентрализованные приложения. Технология Blockchain в действии (Серия "Бестселлеры O'Reilly"). СПб.: Питер, 2017. 240 с.

3. **Свон М.** Блокчейн. Схема новой экономики. М.: Олимп-Бизнес, 2015.

4. **Лекомцева М., Нестерова Н., Семенов В.** Анализ рисков информационной безопасности в банке // Научно-технический вестник информационных технологий, механики и оптики. 2006. № 29.

5. **Vukolic M.** Rethinking permissioned blockchains // Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts (BCC '17). 2017. P. 3—7. DOI: 10.1145/3055518.3055526.

6. **Kiayias A.** Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol / A. Kiayias, A. Russell, B. David, R. Oliynykov // In CRYPTO 2017, Part I (LNCS). 2017. Vol. 10401. Springer, Heidelberg, 357—388.

7. **David B. Ouroboros Praos**, **Gazi P., Kiayias A., Russell A.** An adaptively-secure, semi-synchronous proof-of-stake protocol // Cryptology ePrint Archive, Report 2017/573. 2017.

8. **Antonopoulos A.** Mastering Bitcoin. Sebastopol: O'Reilly Media, 2014. 298 p.

9. **Gazi P.**, **Kiayias A., Zindros D.** Proof-of-Stake Sidechains // 2019 IEEE Symposium on Security and Privasy (SP). 2019. Vol. 1. P. 677—694.

10. **Garay J., Kiayias A., Leonardos N.** The Bitcoin Backbone Protocol with Chains of Variable Difficulty // Advances in Cryptology — CRYPTI 2017. Lecture Notes in Computer Science. 2017. V. 10401. P. 291—323. DOI: 10.1007/978-3-319-63688-7_10.

11. **Garay J.**, **Kiayias A., Leonardos N.** The bitcoin backbone protocol: Analysis and applications // Advances in Cryptology — Eurocrypt 2015, Lecture Notes in Computer Science. 2015. Vol. 9057. P. 281—310. DOI: 10.1007/978-3-662-46803-6_10.

12. **Pease M.**, **Shostak R., Lamport L.** Reaching agreement in the presence of faults // Journal of the ACM (JACM). 1980. Vol. 27. N. 2. P. 228—234.

13. **Pass. R.**, **Seeman L., Shelat A.** Analysis of the Blockchain Protocol in Asynchronous Networks // Advances in Cryptology — EUROCRYPT 2017. Lecture Notes in Computer Science. Springer, Cham, 2017. Vol. 10211.

14. **Hirt M., Zikas V.** Adaptively secure Broadcast // Eurocrypt'2010, LNCS 6110. 2010. P. 466—485.

15. **Canetti R.** Universally composable security: A new paradigm for cryptographic protocols // Cryptology ePrint Archive, Report 2000/067, December 2000. Revised edition, July 2013.

I. M. Shilov, e-mail: ilia.shilov@yandex.ru, **D. A. Zakoldaev,** PhD, e-mail: d.zakoldaev@mail.ru,
ITMO University, Saint Petersburg, Russian Federation

Multidimensional Blockchain and its Advantages

The paper observes key disadvantages of robust ledgers based on blockchain technology. Most severe problems of such technologies include rapid increase of data for storage and the necessity to use mediators during information exchange between different systems. The purpose of this paper is solution of these problems which can be compatible with existing solutions. A new approach to building robust distributed ledgers based on existing solutions — multidimensional blockchain — is presented as one of potential mitigations. The paper contains description of possible modes of multidimensional blockchain functioning and its key peculiarities. The main advantages of multidimensional blockchain are excessive use of external transactions and system-wide addressing of transactions, blocks and accounts. External transactions are held in stages which mitigates double-spending attacks and preserves system independence. Initiating step is similar to sending usual transaction while accepting step implies usage of search and validation protocol. Numeric assessment of key features of multidimensional blockchain is presented. These include transactions per second, block generation period, amount of data stored by network nodes. Transactions per second are assessed under various circumstances: when amount of transactions generated in one time slot is static and when it is not. Those features are compared to analogs (currently — cryptocurrencies). The results can be used in building systems based on robust distributed ledgers and are a basis for formal security proof of the proposed solution.

Keywords: *blockchain, robust distributed ledger, multidimensional blockchain, scaling, addressing, transactions, transactions per second, availability, integrity, liveness, persistence*

DOI: 10.17587/it.26.360-367

References

1. **Cachin C., Guerraoui R., Rodrigues L.** Introduction to Reliable and Secure Distributed Programming, Springer-Verlag Berlin Heidelberg, 2011.

2. **Raval S.** Decentralized applications. Blockchain technology in action, SPb., Piter, 2017, 240 p. (in Russian).

3. **Swan M.** Blockchain: Blueprint for a New Economy. Moscow, Olimp-Business, 2015 (in Russian).

4. **Lekomtseva M., Nesterova N., Semenov V.** Analysis of information security risks in bank, *Nauchno-technicheskiy vestnik informatsionnyh technologiy, mechaniki i optiki*, 2006, vol. 29, pp. 172—174.

5. **Vukolic M.** Rethinking permissioned blockchains, *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, 2017, pp. 3—7.

6. **Kiayias A., Russel A., David B., Oliynykov R.** Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol, *CRYPTO 2017, Part I (LNCS)*, 2017, vol. 10401, Springer, Heidelberg, pp. 357—388.

7. **David B., Gazi P., Kiayias A., Russel A.** Ouroboros Praos: An adaptively-secure, semi-synchronous proof-of-stake protocol, Cryptology ePrint Archive, Report 2017/573, 2017.

8. **Antonopoulos A.** Mastering Bitcoin, Sebastopol, O'Reilly Media, 2014.

9. **Gazi P., Kiayias A, Zindros D.** Proof-of-Stake Sidechains. *2019 IEEE Symposium on Security and Privasy (SP)*, 2019, vol. 1, pp. 677—694.

10. **Garay J., Kiayias A., Leonardos N.** The Bitcoin Backbone Protocol with Chains of Variable Difficulty, *Advances in Cryptology — CRYPTI 2017. Lecture Notes in Computer Science*, 2017, vol. 10401, pp. 291—323.

11. **Garay J., Kiayias A., Leonardos N.** The bitcoin backbone protocol: Analysis and applications, *Advances in Cryptology — Eurocrypt 2015, Lecture Notes in Computer Science*, 2015, vol. 9057, pp. 281—310.

12. **Pease M, Shostak R., Lamport L.** Reaching agreement in the presence of faults, *Journal of the ACM (JACM)*, 1980, vol. 27, no. 2, pp. 228—234.

13. **Pass R., Seeman L., Shelat L.** Analysis of the Blockchain Protocol in Asynchronous Networks, *Advances in Cryptology — EUROCRYPT 2017. Lecture Notes in Computer Science*, 2017, vol. 10211.

14. **Hirt M. Zikas V.** Adaptively secure Broadcast, *Eurocrypt'2010, LNCS 6110*, 2010, pp. 466—485.

15. **Canetti R.** Universally composable security: A new paradigm for cryptographic protocols, Cryptology ePrint Archive, Report 2000/067, December 2000, Revised edition, July 2013.