Bitbon System

АЛГОРИТМ COMMUNITY PROOF-OF-STAKE — СОВМЕСТНОЕ ДОКАЗАТЕЛЬСТВО ДОЛИ

ОБЕСПЕЧЕНИЕ КОНСЕНСУСА ПО АЛГОРИТМУ COMMUNITY POS В СИСТЕМЕ УЧЕТА ТОКЕНОВ РАСПРЕДЕЛЕННОГО РЕЕСТРА СИСТЕМЫ BITBON

(Редакция от 17 декабря 2021 года)

*Данный документ является специальной версией Приложения к Публичному контракту Системы **Bit**bon «Провайдинг в Системе **Bit**bon: экономико-правовая децентрализация Системы **Bit**bon», созданной для удобства понимания принципов обеспечения консенсуса по алгоритму Community PoS в системе учета токенов распределенного реестра Системы **Bit**bon.*

СОДЕРЖАНИЕ

Введение

Блокчейн является одним из видов распределенного реестра. Ключевой особенностью распределенного реестра является децентрализация, то есть отсутствие единого центра хранения и регистрации данных. При этом информация во всех узлах распределенного реестра должна быть валидна и актуальна, что возможно только посредством достижения консенсуса между всеми узлами такого реестра. Первый алгоритм достижения консенсуса, который был применен для сети блокчейн, представляет собой общий случай решения задачи византийских генералов, в котором количество узлов сети неограниченно и может динамически изменяться.

Для решения этой задачи может быть применен подход к организации сообществ, использующих блокчейн, в том смысле, что вместо спонтанных порядков должны быть предложены формы совместной деятельности, где участники сообществ создавали бы децентрализованные общественные организации, представляющие их интересы, и участвовали в развитии Социальной сети «Система **Bit**bon». Внедрение такого решения приводит к необходимости создания не только инструментов, которые позволят сообществу управлять системой учета токенов распределенного реестра, но и к созданию социально-правовой модели взаимоотношений пользователей и государственных органов.

Такой подход открывает возможность повысить производительность систем учета токенов распределенного реестра за счет использования синхронных протоколов с сохранением безопасности и децентрализации блокчейна. Хорошим примером является предложенный в 2014 году Д. Ларимером протокол Delegated Proof-of-Stake (DPoS, делегированное доказательство доли), его внедрение позволило значительно повысить производительность системы учета токенов распределенного реестра (число транзакций в секунду). Но данный протокол имеет ряд технологических особенностей, существенно ограничивающих возможности его применения.

Учитывая актуальные вопросы в развитии технологии распределенного реестра (TPP), а также те особенности и недостатки, которые выявлены в современных

консенсусных протоколах, компанией Simcord было разработано собственное решение — алгоритм достижения консенсуса Community PoS (совместное доказательство доли).

Алгоритм достижения консенсуса Community PoS представляет собой усовершенствованный протокол DPoS за счет следующих решений:

• введение системы автоматического распределения долей Регистраторов (выраженных в виде мощностей Assetbox) между узлами сети, что приводит к устранению проблемы централизации голосования участников в DPoS, обусловленной принципом Парето 80/20;

• введение системы рейтинга узлов для предотвращения некорректного поведения узлов сети;

• развитие системы peer-to-peer (P2P) протоколов и коммуникаций;

• децентрализованная верификация блока всеми узлами сети и формирование рейтингов узлов в зависимости от результатов верификации;

• введение состояния сети «отказ в обслуживании» для фиксации значения периода неопределенности, когда бизнес-приложение не может квалифицировать состояние операции, выполняемой узлами сети;

• организация системы распределения вознаграждения за участие в обеспечении консенсуса Community PoS с учетом мотивации развития Социальной сети «Система **Bit**bon».

1. Основная идея Community PoS

Основная идея Community PoS заключается в организации Провайдеров в роли Регистраторов, которые объединяют свои Assetbox в пулы Регистраторов посредством предоставления единиц цифрового актива **Bit**bon на таких Assetbox для автоматического распределения мощности таких Assetbox между узлами системы учета токенов распределенного реестра для выполнения процедуры голосования с целью формирования последовательности блок-продюсеров, которые будут подписывать и публиковать блоки.

Каждый Провайдер в роли Регистратора стремится привлечь новых Регистраторов с целью увеличения мощности своего пула. Пулы Регистраторов, обладающие большей мощностью, имеют больше шансов поучаствовать в формировании блоков, так как их поддерживает большее количество участников большим количеством **Bit**bon. Каждый новый Регистратор автоматически получает возможность принять участие как в валидации блоков, генерируемых другими Регистраторами, так и в формировании новых блоков, что существенно усложняет организацию атак злоумышленниками на сеть. Каждый новый Регистратор уменьшает вероятность попадания в группу блок-продюсеров узлов-злоумышленников и в то же время увеличивает требования к аппаратным ресурсам и количеству **Bit**bon, которые злоумышленник должен иметь для проведения атаки на сеть. Как показывают проведенные расчеты, вероятность предсказания злоумышленником момента, когда он сможет выполнить атаку, то есть у него будет контроль над Assetbox и узлами, обеспечивающий попадание таких узлов в состав блок-продюсеров, составляет 10^{-40}. Следовательно, увеличение количества Регистраторов приводит к дальнейшему повышению устойчивости к атакам на систему учета токенов распределенного реестра Системы **Bit**bon, делая любую из видов атак невозможной.

Рисунок 1. Схема работы алгоритма Community PoS

Компанией Simcord запланированы два этапа реализации алгоритма обеспечения консенсуса. В данной статье описан алгоритм, реализуемый на первом этапе (Рисунок 1), который является подготовительным к переходу на полностью децентрализованную модель Системы **Bit**bon в соответствии с Дорожной картой децентрализации. При переходе на второй этап будет расширен алгоритм и обеспечена привязка Assetbox к нодам Пользователей (Рисунок 1*). Это позволит Провайдерам запускать собственные ноды системы учета токенов распределенного реестра и гарантировать их надежную работу, что в свою очередь увеличит диверсификацию, надежность и безопасность Системы **Bit**bon в целом.

Также довольно важным аспектом является вопрос устойчивости алгоритма достижения консенсуса Community PoS к атакам злоумышленников посредством использования уязвимостей технологии блокчейн. Анализ концепции Community PoS позволяет нам с уверенностью сказать, что использование данного метода обеспечения консенсуса в системе учета токенов распределенного реестра Системы **Bit**bon позволит ей успешно пройти тесты на перечисленные ниже виды атак:

- **Атака 51%.**
- **Sibill.**
- **Timejacking.**
- **Каннибализм пулов.**
- **Удержание блока.**

- **Дезорганизующая атака.**
- **Eclipse атака.**
- **Атака P + epsilon.**
- **Черный список.**
- **Гибкость транзакций.**
- **Эгоистичный майнинг.**
- **Отмена всех транзакций.**
- **Двойное расходование.**
- **Случайные хардфорки.**

Организация инфраструктуры системы учета токенов распределенного реестра Системы Bitbon на базе консенсуса Community PoS дает возможность построить такую децентрализованную среду исполнения Системы Bitbon, социально-правовые, архитектурные и технические решения которой позволят оперативно реагировать на запросы современного мира и изменение условий без снижения качества сервиса такой системы для ее Пользователей.

2. Концепция консенсуса Community PoS

2.1. Цель Community PoS и способы ее достижения

Основная цель консенсуса Community PoS — обеспечение реальной децентрализации процесса публикации, верификации и хранения данных распределенного реестра при высоком уровне производительности сети хранения и малом гарантированном интервале ожидания подтверждения завершения транзакции.

Достижение этой цели обеспечивается:

- использованием схемы предварительного согласования последовательности производства блоков блок-продюсерами для предотвращения форков и коллизий блоков;
- централизацией сети в момент формирования блока узлом сети в соответствии с последовательностью формирования блоков;

- введением жесткой синхронной циклограммы работы узлов сети для обеспечения однозначного определения состояния сети;

- введением состояния сети «отказ в обслуживании» для фиксации значения периода неопределенности, когда бизнес-приложение не может квалифицировать состояние операции, выполняемой узлами сети;

- взаимной синхронизацией узлов сети, чтобы обеспечить соблюдение циклограммы голосования и формирования блоков;

- использованием фиксированного максимального времени обработки транзакции (с отменой в случае невыполнения ее в заданный период);

- введением трех типов протоколов функционирования узлов сети:

– протокол контроля кворума и обеспечения синхронизации времени узлов, представляющий собой фоновый процесс, основанный на опросе P2P-сети и обеспечивающий поддержание в актуальном состоянии информации о доступности узлов, которые могут принимать участие в процедурах голосования и формирования блоков;

– протокол жребия-голосования, в рамках которого формируется последовательность узлов сети, согласно которой узлы сети будут выполнять роль блок-продюсера;

– протокол формирования блока, включающий создание блока блок-продюсером, его верификацию остальными узлами сети и систему рейтинга узлов, обеспечивающую корректность выполнения узлами сети функций блок-продюсеров;

- использованием алгоритма случайного распределения долей (мощностей Assetbox) Регистраторов между узлами сети перед жребием-голосованием среди узлов, рейтинг которых позволяет им быть кандидатами в блок-продюсеры. Такое решение позволяет повысить уровень сложности предсказания последовательности блок-продюсеров. Алгоритм основан на значении параметра «nonce» (последнего блока или генезис-блока), которое, в свою очередь, строится на базе случайного числа от аппаратного генератора случайных чисел и временной метки путем генерации хеш-кода от этой информации с

помощью Keccak-256 на базе эллиптических кривых. Полученная числовая последовательность используется как ключ для распределения мощностей Assetbox Регистраторов в пользу узлов сети — кандидатов в блок-продюсеры;

- механизмом децентрализованного жребия-голосования при определении порядка выполнения узлами роли блок-продюсеров, который производится путем сортировки списка узлов по объему мощностей, распределенных в пользу этих узлов, определения границ последовательности и контроля правил участия узлов в сформированной последовательности блок-продюсеров;

- децентрализованной верификацией блока всеми узлами сети и рассылкой сообщения о повышении или понижении рейтинга блок-продюсера, генерировавшего блок, в зависимости от результатов верификации.

2.2. Роли узлов в рамках консенсуса Community PoS

Каждый узел сети может выполнять следующие роли:

• **Узел синхронизации**. В рамках этой роли узел сети при подключении к сети производит синхронизацию с остальными узлами путем получения блоков, транзакций и связанных с ними объектов от других узлов сети, их верификации и сохранения в локальном хранилище. Синхронизация времени узла производится в соответствии с меткой времени последнего валидного блока и метрикой задержки до блок-продюсера, сформировавшего этот блок, а также меток времени от остальных узлов сети. После процедуры синхронизации узел выполняет верификацию и сохранение поступающих к нему транзакций и блоков. Если метрика времени распространения блока для этого узла меньше 1 секунды, то этот узел сети обязан принимать в обработку транзакции от клиентских приложений и после верификации ретранслировать их всем остальным узлам сети. В противном случае или если кворум сети узлов не достигнут, то узел не принимает транзакции в обработку, возвращая ошибку «отказ в обслуживании».

• **Узел-участник кворума**. В этой роли может выступать любой узел синхронизации, у которого величина сетевой задержки до узлов, входящих в состав кворума, не должна превышать 400 мс. Для реализации протокола работы консенсуса Community PoS в сети должно присутствовать число узлов, большее, чем размер кворума, определенного Операторами Системы **Bit**bon (не меньше 2/3 числа узлов сети). Узел-участник кворума участвует в процедуре формирования рейтинга в соответствии с системой рейтинга узлов. Если в процессе обработки транзакций и блоков участник кворума обнаруживает нарушение правил обработки, то он высылает всем узлам сети соответствующее сообщение о понижении рейтинга источников невалидных данных. Если данные валидны, то отправляется сообщение о повышении рейтинга соответствующих блок-продюсеров.

• **Кандидат в блок-продюсеры**. В данной роли может выступать любой узел-участник кворума с рейтингом выше 10, если у него включен режим провайдинга. В этом случае такой узел будет включен в процедуру распределения мощностей (долей) Assetbox пула Регистраторов.

• **Блок-продюсер**. В данной роли выступает узел сети — кандидат в блок-продюсеры, который был включен в последовательность блок-продюсеров в результате выполнения процедуры жребия-голосования для подписи и публикации только одного блока в заданный момент времени (тайм-слот) (Рисунок 2).

2.3. Обеспечение системы жребия-голосования

Во избежание угроз централизации Системы **Bit**bon и с целью автоматизации процедуры голосования Регистраторов в рамках участия в обеспечении консенсуса в системе учета токенов распределенного реестра Системы **Bit**bon предусмотрена процедура автоматического перераспределения долей (мощностей Assetbox) между кандидатами в блок-продюсеры.

Для участия Assetbox в автоматическом голосовании достаточно единожды осуществить передачу мощности Assetbox Регистратора в пул путем выполнения перевода 0,0001 **Bit**bon с комментарием «/pool» на любой Assetbox пула Регистратора.

Мощности Assetbox, участвующие в автоматическом распределении, ассоциируются случайным образом между всеми узлами сети с соответствующим рейтингом, которые отвечают актуальным на момент голосования требованиям по производительности и качеству канала связи (узел в роли кандидата в блок-продюсеры).

2.4. Процедура жребия-голосования

Целью процедуры жребия-голосования является формирование на основе распределенных мощностей Assetbox между кандидатами в блок-продюсеры последовательности блок-продюсеров, согласно которой данные блок-продюсеры будут подписывать и публиковать блоки в следующем раунде.

Процедура жребия-голосования производится в соответствии с циклограммой голосования и формирования блоков (Рисунок 2) и состоит из раундов, длительность

которых равна числу блок-продюсеров, умноженному на интервал времени, равный

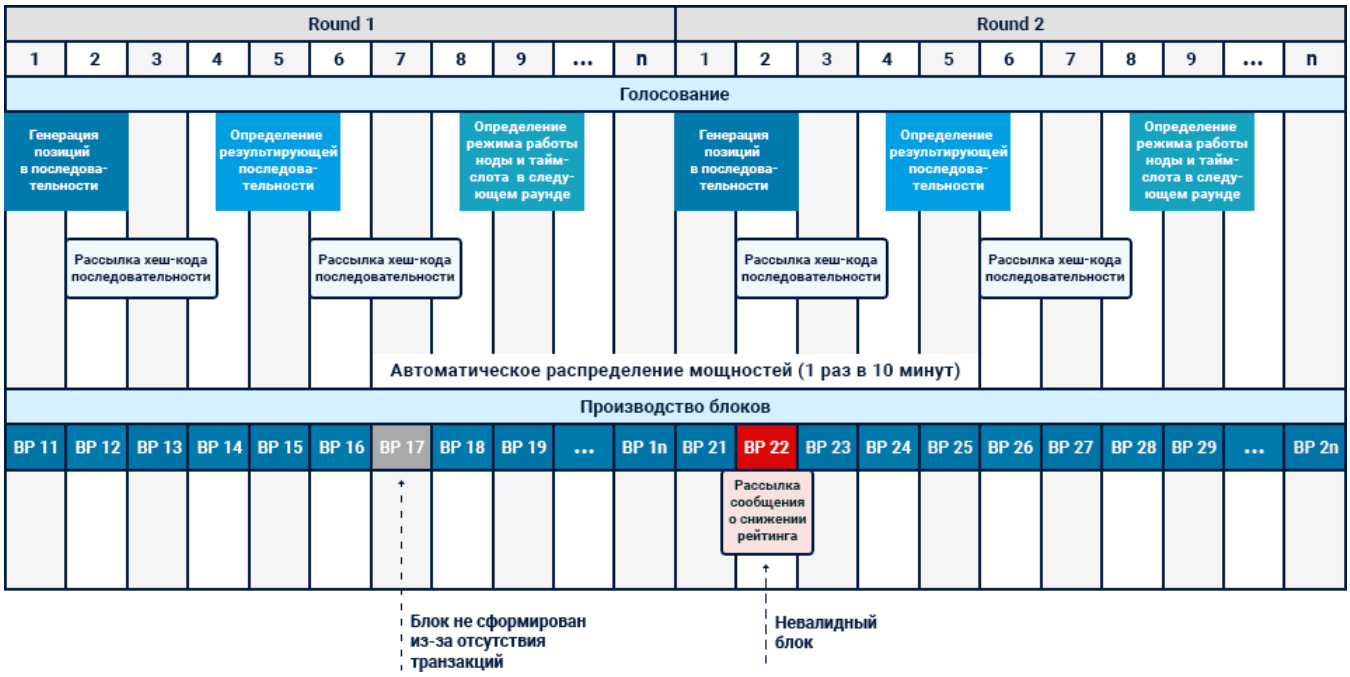


Рисунок 2. Циклограмма консенсуса Community PoS

Процедура жребия-голосования начинается по окончании очередного раунда. Жребий-голосование выполняется в течение всего раунда. Признаком окончания очередного раунда может выступать финальный блок раунда или сообщение отметки времени вместо него, если в этот тайм-слот не было транзакций. Голосование выполняется в следующем порядке:

- в течение первых 2 секунд каждый узел на основе распределенных мощностей Assetbox между кандидатами в блок-продюсеры должен случайным образом сформировать список возможных позиций (последовательность) от 1 до n для каждого из узлов-кандидатов в блок-продюсеры с мощностью выше нижней границы и соответствующим рейтингом, затем отправить хеш этого списка (последовательности) всем узлам-участникам кворума;

- из числа узлов, которые могут быть включены в последовательность, исключается блок-продюсер, который последним формирует блок в текущем раунде;

- элементы последовательности с одинаковыми позициями не допускаются;
- запрещается включать в последовательность 2 элемента с одним идентификатором;
- на 5-й секунде каждый узел-участник кворума должен по количеству голосов за каждый уникальный хеш последовательности определить, набрала ли максимум голосов сформированная им последовательность. Если нет, то узел переходит в режим ожидания получения последовательности блок-продюсеров. В противном случае узел проверяет, что число голосов больше или равно 2/3 кворума. Если число голосов больше или равно 2/3 кворума, то узел публикует полученную последовательность. Иначе узел публикует сообщение об ошибке формирования последовательности и переходит в режим ожидания начала следующего раунда;
- каждый узел до 9-й секунды получает последовательность производства блоков либо ошибку от других узлов.

Каждый узел, независимо от роли, верифицирует приходящие блоки в соответствии с рассчитанной/полученной последовательностью блок-продюсеров. Кандидат в блок-продюсеры, указанный в результирующей последовательности, в свой тайм-слот выполняет роль блок-продюсера.

2.5. Формирование блоков

Блоки формируются блок-продюсерами в раундах длиной *n* блоков (к примеру, длина раунда 21 блок). Блок-продюсер формирует блок из транзакций, находящихся в его транзакционном пуле в период между отметкой времени последней транзакции в последнем валидном сформированном блоке и моментом формирования блока в текущем тайм-слоте, который он обслуживает, согласно правилам:

- если в обрабатываемый тайм-слот не пришлось ни одной транзакции, блок не формируется, но при этом всем узлам рассылается отметка времени завершения тайм-слота;

- узел в роли блок-продюсера формирует блок на базе хеш-кода предыдущего валидного блока из транзакций, находящихся в его транзакционном пуле;

- в рамках раунда блок-продюсер может формировать блок только 1 раз;

- блок-продюсер ни при каких условиях не может формировать 2 блока подряд;

- все узлы (в том числе и блок-продюсеры в текущем раунде) выполняют роль узла синхронизации, получая транзакции, выполняя их и проверяя приходящие блоки:

  - если блок валиден, то он и входящие в него транзакции фиксируются в хранилище;

  - если блок невалиден, узел его игнорирует и ждет прихода валидного блока с таким же номером;

- если блок-продюсер не смог верифицировать предыдущий блок в момент обслуживания своего тайм-слота, то он формирует новый с тем же номером, включая в него все транзакции в транзакционном пуле, в том числе пришедшие в прошлые тайм-слоты, отсортированные по времени создания (исключая те, у которых закончилось время обработки). Как и узлы синхронизации, блок-продюсер рассылает сообщение о понижении рейтинга предыдущего блок-продюсера.

Эти меры позволяют избежать создания форков цепочки блоков, а также атак, связанных с задержкой обработки или игнорированием транзакций, при этом они обеспечивают гарантированное время обработки транзакции.

2.6. Рейтинг узла

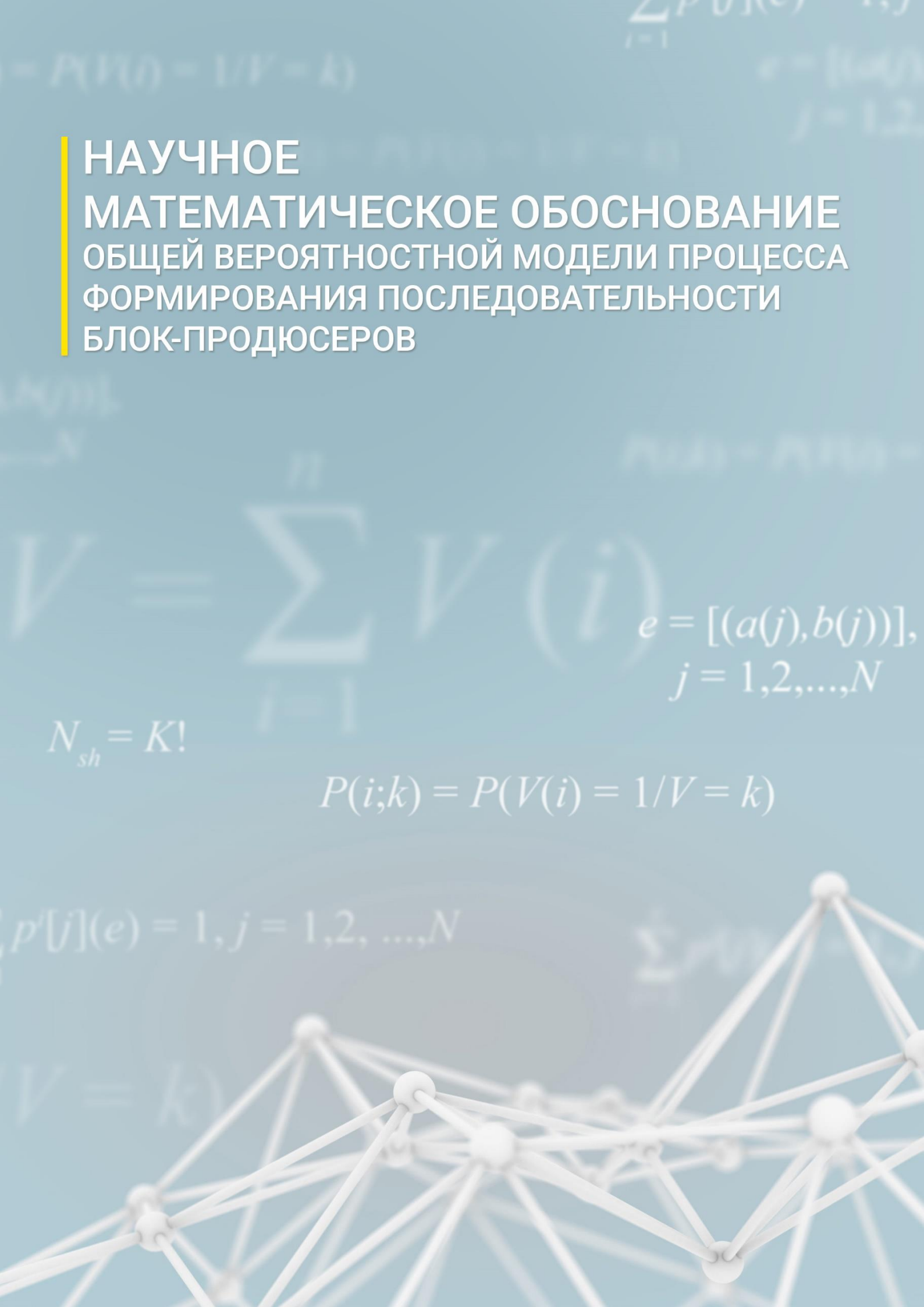
Рейтинг каждого узла формируется посредством сообщений, которые рассылаются по P2P-сети всеми узлами как результат верификации очередного формирования блока. Изменения рейтинга принимаются всеми узлами сети в пользу всех узлов сети, в частности в пользу блок-продюсера, и применяются через период времени, равный длительности 3 тайм-слотов после формирования блока, при условии, что число

сообщений будет больше или равно кворуму (при этом контролируется количество сообщений от каждого узла). Учитывается только одно сообщение от узла на каждый блок. Ниже приведены основные факторы, влияющие на рейтинг узлов:

- повышение рейтинга:
  - за корректно сформированный блок;
  - за выполнение условий участия в кворуме (выдается Оператором Системы **Bit**bon);
  - если блок-продюсер сформировал за сутки хотя бы один блок и не получил понижение рейтинга;
- понижение рейтинга:
  - если блок-продюсер включил более 10 транзакций, относящихся к предыдущему тайм-слоту;
  - если блок-продюсер сформировал блок не в свой тайм-слот;
  - если блок-продюсер не сформировал блок и не отправил пакет с временной меткой в свой тайм-слот;
  - если узел отправил более 2 сообщений о повышении/понижении рейтинга на один блок (паузу);
  - если узел транслировал невалидную транзакцию или одну и ту же транзакцию повторно (за каждое повторение);
  - обнуление рейтинга, если блок-продюсер сформировал невалидный блок (включил невалидную транзакцию).

Описанная система обратной связи позволяет эффективно пресекать атаки и некорректное поведение потенциальных злоумышленников, а также исключать из кворума нестабильные узлы сети.

НАУЧНОЕ МАТЕМАТИЧЕСКОЕ ОБОСНОВАНИЕ ОБЩЕЙ ВЕРОЯТНОСТНОЙ МОДЕЛИ ПРОЦЕССА ФОРМИРОВАНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ БЛОК-ПРОДЮСЕРОВ



3. Научное математическое обоснование общей вероятностной модели процесса формирования последовательности блок-продюсеров

3.1. Распределение мощностей Assetbox по узлам сети — кандидатам в блок-продюсеры

Исходными данными является группа Assetbox $e=\left[e(1),e(2),...,e(N)\right]$. Каждый Assetbox $e(j)$ характеризуется уникальным идентификатором a из множества A и мощностью (положительной числовой характеристикой) $b(j)$

$$e=\left[\left(a(j),b(j)\right)\right],$$

$$j=1,2,...,N,$$

где N — число Assetbox, участвующих в обеспечении консенсуса в системе учета токенов распределенного реестра Системы **Bit**bon.

В пользу набора узлов сети (кандидатов в блок-продюсеры) случайным (квазислучайным) образом передаются мощности Assetbox. Мощность каждого Assetbox передается только одному из узлов. В зависимости от состояния группы $e=\left[\left(a(j),b(j)\right), j=1,2,...,N\right]$ возникает набор положительных вероятностей

$$\left[p^i[j](e), i=1,...,n;\ j=1,2,...,N\right]$$

передачи мощности j-го Assetbox i-му кандидату в блок-продюсеры. Для каждого Assetbox сумма вероятностей передачи мощности этого Assetbox определенному узлу-кандидату в блок-продюсеры для всех узлов равна единице

$$\sum_{i=1}^{n} p^i[j](e)=1,\ j=1,2,...,N. \qquad (1)$$

Вероятность распределения мощностей группы Assetbox с номерами $J^{[l(i)]} = \left[j(1), j(2),..., j\left(l\left(i\right)\right)\right]$ в пользу i **-го узла** равна

$$P^i(J^{[l(i)]}) = p^i(j^i(1), j^i(2),..., j^i(l(i))) = p^i[j^i(1)](e)\,p^i[j^i(2)](e)...p^i[j^i(l(i))](e). \quad (2)$$

Вероятность же передачи i **-му узлу только мощностей этих Assetbox** равна

$$P^i[J^{[l(i)]}.] = P^i(J^{[l(i)]})\exp[\sum_{j \notin J^{[i]}} \ln[1 - p^i[j](e)]], i = 1,2,...,n\,. \quad (3)$$

Если узлу i в точности передан набор мощностей Assetbox $J^{[l(i)]}\left(i\right)$, а остальным узлам — кандидатам в блок-продюсеры передаются другие непересекающиеся наборы мощностей Assetbox, при этом объединение таких наборов по всем узлам дает все множество мощностей Assetbox, тогда вероятность того, что всем узлам будут переданы мощности своих уникальных наборов Assetbox определяется как произведение по всем узлам сети вероятностей

$$P^i(J^{[L(i)]}) = p^i(j^i(1), j^i(2),..., j^i(l(i))) = p^i[j^i(1)](e)\,p^i[j^i(2)](e)...p^i[j^i(l(i))](e)\,, \quad (4)$$

то есть

$$P[J^{[L(1)]}(1),..., J^{[L(n)]}(n)] = P^1(J^{[L(1)]})P^2(J^{[L(2)]})...P^n(J^{[L(n)]})\,. \quad (5)$$

3.2. Формирование последовательности блок-продюсеров из кандидатов в блок-продюсеры

Каждый узел со своим набором мощностей Assetbox из общего набора в n узлов сети может быть выбран в группу блок-продюсеров из k узлов ($k < n$).

Вероятность $p(i)[H]$ выбора i-го узла в избранную группу зависит от состояния узлов $H = (E^1, E^2,..., E^n)$, где состояние узлов определяется мощностями Assetbox, распределенными в их пользу

$$E^i = \left[e\left(j^i(1) \right), e\left(j^i(2) \right), ..., e\left(j^i(l(i)) \right) \right], i = 1,2,...,n. \qquad (6)$$

$V(i)$ — случайная величина, показывающая число попаданий i-го узла в избранную группу, то есть $V(i)$ — индикатор попадания i-го узла в избранную группу, $i = 1,2,...,n$, принимающая значения 0 или 1. $V = \sum_{i=1}^{n} V(i)$ обозначает число узлов в избранной группе. Вероятность попадания i-го узла в избранную группу из k блок-продюсеров равна

$$P(i;k) = P\left(\frac{V(i) = 1}{V = k} \right) \qquad (7)$$

условной вероятности попадания i-го узла в избранную группу при условии, что число узлов в избранной группе равно k.

Данная условная вероятность определяется как отношение вероятности произведения $P(V(i) = 1, V = k)$ этих двух событий к вероятности $P(V = k)$ условия.

Вероятность условия равна

$$P(V = k) = \sum_{\left[\{V(i)\}, V = k \right]} \exp\left[\sum_{i=1}^{n} \ln\left[p(i)^{(V(i))} \left(1 - p(i)\right)^{(1 - V(i))} \right] \right] \qquad (8)$$

сумме вероятностей произведения событий принадлежности определенного узла группе (избранной или неизбранной) событий, когда избранная группа состоит из k узлов.

Далее определяется вероятность произведения $P(V(i) = 1, V = k)$, то есть из этой суммы выделяются те слагаемые, в которых $V(i) = 1$, а сумма остальных индикаторов равна $k - 1$. Такая вероятность определяется

$$P(V(i) = 1, V = k) = p(i) \sum_{[\{V(i)\}, V - V(i) = k - 1]} \exp\left[\sum_{\substack{1 \le r \le n \\ r \ne i}} \ln\left[p(r)^{(V(r))} \left(1 - p(r)\right)^{(1 - V(r))} \right] \right]. \qquad (9)$$

Вероятность попадания i-го узла в избранную группу из k блок-продюсеров равна

$$P(i;k) = P(V(i) = 1 / V = k) \qquad (10)$$

условной вероятности попадания i-го узла в избранную группу при условии, что число узлов в избранной группе равно k и

$$P(i;k) = P(V(i) = 1 / V = k) = \frac{P(V(i) = 1, V = k)}{P(V = k)}$$

или

$$P(i;k) = \frac{p(i) \sum\limits_{[\{V(i)\}, V - V(i) = k-1]} \exp\left[\sum\limits_{\substack{1 \le r \le n \\ r \ne i}} \ln\left[p(r)^{(V(r))} \left(1 - p(r)\right)^{(1 - V(r))} \right] \right]}{\sum\limits_{[\{V(i)\}, V = k]} \exp\left[\sum\limits_{i=1}^{n} \ln\left[p(i)^{(V(i))} \left(1 - p(i)\right)^{(1 - V(i))} \right] \right]}, \qquad (11)$$

где $p(i) = p(i)[H]$.

3.3. События повторного формирования последовательности блок-продюсеров

Условная вероятность попадания i-го узла в избранную группу блок-продюсеров на r-ю позицию при условии, что размер группы блок-продюсеров составляет k узлов, равна $\frac{1}{k}$.

Условная вероятность повторения фрагмента последовательности из определенных w узлов в группе блок-продюсеров из k узлов при условии, что узлы, участвовавшие в процедуре выбора, уже включены в последовательность, равна вероятности

$$P(w,k) = \frac{1}{k(k-1)(k-2)...(k-w+1)} \qquad (12)$$

того, что данный фрагмент последовательности размещен в начале последовательности блок-продюсеров, умноженной на число позиций последовательности $(k-w+1)$, в

которых данный фрагмент последовательности может быть размещен, в группе блок-продюсеров из k узлов

$$P^{(r)}(w,k)=\left(k-w+1\right)P(w,k)=\frac{1}{k(k-1)...(k-w+2)}. \qquad (13)$$

Вероятность условия формирования фрагмента последовательности $i(1),i(2),...,i(w)$ равна произведению вероятностей $p(i(1)),p(i(2)),...,p(i(w))$. Вероятность $p^{(r)}=p^{(r)}(i(1),i(2),...,i(w))$ появления фрагмента последовательности ($i(1),i(2),...,i(w)$) на очередном шаге в группе блок-продюсеров равна произведению условной вероятности $P^{(r)}(w,k)$ на вероятность условия $p(i(1))p(i(2))...p(i(w))$ и равна

$$p^{(r)}=P^{(r)}(w,k)\,p(i(1))\,p(i(2))...p(i(w))=\frac{p(i(1))\,p(i(2))...p(i(w))}{k(k-1)...(k-w+2)}. \qquad (14)$$

3.4. Оценка вероятности стационарности результатов процесса формирования последовательности блок-продюсеров относительно этапов распределения мощностей Assetbox и формирования последовательности блок-продюсеров

В качестве оценки вариантов решения можно определить количество возможных способов разбить множество $A=\left\{a^{(1)},a^{(2)},...,a^{(N)}\right\}$ из N элементов на непересекающиеся подмножества $A^{(i)}$, $A=A^{(1)}+A^{(2)}+...+A^{(K)}$, где $i=1,2,...,K$ и $K<N$, равное K^N. Первый элемент $a(1)$ множества A может попасть в любое из K подмножеств, второй элемент $a(2)$ множества A может попасть в любое из K подмножеств... и так N раз. Разбиение исходного множества A на непересекающиеся подмножества $A^{(i)}$, где $i=1,2,...,K$, является результатом этапа распределения мощностей Assetbox в пользу узлов сети — кандидатов в блок-продюсеры.

Далее производится расчет распределения мощности.

Каждое подмножество $A^{(i)}$ таким образом принимает состояние E^i, определяемое попавшими в него элементами $a\left(i(s)\right)$, $s=1,2,...,l=l\left(i\right)$,

$$E^i=\left[a\left(i(1)\right),a\left(i(2)\right),\ldots,a\left(i(l)\right)\right],\ i=1,2,...,K\text{, а }\sum_{i=1}^{K}l(i)=N\text{. (15)}$$

В общем случае состояния подмножеств будут изменяться. Результат реализации этапа **распределения мощности** как число перестановок N_{sh} подмножеств множества A (предполагая, что под перестановкой в данном случае имеется в виду упорядочивание $A^{(i)}$ по величине состояния E^i), которое равно

$$N_{sh}=K!\qquad(16)$$

Каждая перестановка является исходными данными для операции **формирования последовательности блок-продюсеров**, с помощью которой из подмножеств $A^{(i)}$ выбираются привилегированные подмножества $A^{(v)}$, где $v=1,2,...,П$. Общее количество выборов N_{select} привилегированных неупорядоченных подмножеств $П$ из K групп блок-продюсеров равно

$$N_{select}=\frac{K!}{П!(K-П)!}.\qquad(17)$$

В общем случае вероятность попадания элемента $e\left(i\right)$ в конкретное подмножество A^j (событие X), например, первое $j=1$, равна $P(X)=\frac{1}{K}$. Вероятность выбора подмножества A^1 в привилегированную группу (группу блок-продюсеров) при условии, что событие X состоялось (в подмножестве A^1 размещен элемент $e(i^1)$, событие Y), равна $P(Y/X)=\frac{1}{K}$. Вероятность повторного попадания элемента в подмножество, которое впоследствии попадет в привилегированную группу на то же место

$$P(XY) = P(Y / X)P(X) = \frac{1}{K^2}. \quad (18)$$

В подмножество было выбрано u элементов и это подмножество попало в привилегированную группу на определенную позицию в группе блок-продюсеров, событием X будет выбор на следующем шаге такого же количества u элементов в то же подмножество, а событием Y — выбор этого подмножества в привилегированную группу в ту же позицию. Определяем $P(X) = \frac{1}{K^u}$, $P(Y / X) = \frac{1}{K}$ и

$$P(XY) = P(Y / X)P(X) = \frac{1}{K^{(u+1)}}. \quad (19)$$

На основании вышеизложенного определяется значение вероятности распределения мощностей Assetbox в конкретный узел сети и вероятность распределения конкретного узла сети — кандидата в блок-продюсеры в привилегированную группу узлов, которые осуществляют производство блоков.

Вероятность произведения описанных выше событий будет мала при больших значениях аргументов.