

ЦЕНТРАЛИЗОВАННО КООРДИНИРУЕМЫЙ ДРЕВОВИДНЫЙ БЛОКЧЕЙН НА БАЗЕ АЦИКЛИЧЕСКОГО НАПРАВЛЕННОГО ГРАФА С ПЛАВАЮЩИМ ГЕНЕЗИС-БЛОКОМ ДЛЯ КРУПНЫХ МАСШТАБИРУЕМЫХ СИСТЕМ¹

Калинин М.О., Крундышев В.М., Бушмелев А.С.

Санкт-Петербургский политехнический университет Петра Великого,

Санкт-Петербург, Россия

max@ibks.spbstu.ru, vmk@ibks.spbstu.ru, sci@ibks.spbstu.ru

Аннотация. Представлена гибридная архитектура блокчейна, объединяющая централизованно координируемую древовидную структуру ациклического направленного графа с плавающим механизмом генезис-блока. Решение обеспечивает работу блокчейна в крупных системах, устраняя проблемы масштабируемости, роста объема хранения и вычислительных издержек.

Ключевые слова: блокчейн, ациклический направленный граф, плавающий генезис-блок, древовидный блокчейн.

Введение

Использование технологии блокчейн в различных прикладных областях становится все более популярным из-за ее ключевых свойств безопасности и прозрачности [1, 2]. Как отмечено в [3], блокчейн обеспечивает контроль целостности, повышение доверия в цепочках поставок и других процессах управления информацией. Однако высокие требования к емкости хранения цепочек блоков и требуемой вычислительной мощности на узлах для проверки транзакций создают значительные барьеры для широкого внедрения блокчейна в крупномасштабных средах с множеством маломощных устройств, особенно в системах на базе Интернета вещей (IoT), таких как умный город, умная фабрика, беспроводная сенсорная сеть, интеллектуальная транспортная система. В таких легковесных и ограниченных в ресурсах системах масштабируемость и оптимизация вычислительных процессов блокчейна имеют решающее значение [4, 5].

Хотя некоторые существующие модификации блокчейна снижают требования к вычислительной мощности устройств, проблема неконтролируемого роста объема хранения блокчейна остается нерешенной до сих пор [6, 7]. Устройства с ограниченными ресурсами не способны вмещать постоянно растущие цепочки блоков и заголовков, и это может привести к отказу в работе блокчейна [8]. Традиционные архитектуры блокчейна, широко используемые в финансовых и других приложениях, оказались неэффективны для масштабируемых, самоорганизующихся и динамических систем. Это особенно актуально для систем с большим количеством устройств IoT, которым требуется высокая производительность и надежная обработка больших объемов данных.

Исследователи предлагают различные модификации классического блокчейна, включая добавление новых типов блоков и варианты расширения связей между ними [9-11]. Однако эти решения не обеспечивают должной масштабируемости и производительности. В этой связи особое внимание уделяется архитектурам на основе ациклического направленного графа (АНГ), которые могут эффективно выполнять параллельную генерацию блоков и обеспечивать высокую пропускную способность [12-15]. В частности, известен протокол обмена сообщениями для блокчейна на базе АНГ с поддержкой древовидной структуры (tree-based gossip protocol, TBGP), который строит дерево взаимодействия узлов (tree-based gossip network, TBGN) и тем самым существенно повышает устойчивость и балансировку нагрузки в блокчейн-системе [12]. В то же время известен метод плавающего генезис-блока для решения проблемы хранения данных и сокращения времени подключения новых узлов [16]. Этот механизм периодически фиксирует текущее состояние блокчейна, что позволяет удалять устаревшие блоки и снижать требования к локальному хранилищу, что критично для устройств с ограниченными ресурсами.

В настоящем исследовании представлена комбинация древовидной архитектуры блокчейна на базе АНГ и механизма плавающего генезис-блока. Основная цель данного исследования — обеспечение масштабируемости, стабильности и безопасности блокчейн-сети в условиях ограниченных вычислительных ресурсов и высокой сетевой динамики.

¹ Исследование выполнено за счет гранта Российского научного фонда №24-11-20005, <https://rscf.ru/project/24-11-20005/>, грант Санкт-Петербургского научного фонда (договор №24-11-20005 о предоставлении регионального гранта)

1. Анализ известных решений

В последнее время исследователи пытаются разработать новые эффективные блокчейн-системы, предлагая новые типы блоков и их соединений, но масштабируемость сети по-прежнему остается основной проблемой при внедрении технологии блокчейна в современных гибких системах. Сегодня решения на основе АНГ, такие как TBGP, предлагают наиболее высокую производительность, обеспечивая параллельную обработку блоков и улучшая балансировку нагрузки в сети [12-15].

Рассмотрим архитектуру блокчейна на основе АНГ, адаптированную для распределенных систем с ограниченными ресурсами, таких как Интернет вещей. Данный метод основан на наиболее эффективном блокчейне использующем АНГ, представленном в [12], и использует древовидную структуру блокчейна для координации узлов. Такой подход снижает вычислительную нагрузку на отдельные устройства и обеспечивает более стабильную работу блокчейна. Его структура включает два компонента: механизм формирования АНГ и механизм консенсуса.

Формирование АНГ происходит путем добавления новых транзакций в структуру графа, где каждая транзакция ссылается на одну или несколько предыдущих, создавая направленную сеть. Транзакции распределяются с использованием протокола gossip, который обеспечивает быстрый и децентрализованный обмен данными между узлами.

Для построения АНГ используют три подхода: модель основной цепи, модель параллельной цепи и наивную модель. Блокчейн на базе АНГ представляет собой распределенный реестр в виде АНГ, основанном на модели параллельной цепи, где каждый узел поддерживает свою собственную цепь, и каждая собственная цепь использует определенное правило взаимной ссылки для формирования структуры АНГ. В такой архитектуре блокчейна каждый блок содержит два хеша: родительский хеш и хеш «дяди». Родительский хеш указывает на предыдущий блок, с которым текущий блок напрямую связан, а хеш «дяди» указывает на дополнительный, побочный, блок, связанный с альтернативной ветвью графа. Например, если блок *C* создан узлом, который опирается на блок *B* как на родителя и на блок *A* как на «дядю», то родительский хеш формируется путем хеширования содержимого блока и хеша блока *B*. Хэш «дяди», в свою очередь, вычисляется аналогично на основе содержимого и хеша блока *A*. Такая структура позволяет не только учитывать основную цепочку подтверждений, но и интегрировать параллельные ветви АНГ, что повышает устойчивость и гибкость консенсуса.

Консенсус достигается посредством виртуального голосования и подписания транзакций, что обеспечивает согласованность и безопасность данных в асинхронной и распределенной среде. Для достижения консенсуса в блокчейне на базе АНГ применяется протокол TBGP с виртуальным голосованием, которое включает три этапа: распространение транзакций, голосование и синхронизацию подтверждений. Передача данных осуществляется по протоколу gossip с переключением между методами push, pull и push-pull (в зависимости от этапа). На первом этапе для распространения транзакций используется push. На этапе виртуального голосования используется pull: узлы в конце графа начинают голосование, которое переходит к инициатору. Консенсус достигается, когда транзакция становится «явно видимой», т. е. подтвержденной более чем 2/3 узлов. Такие транзакции считаются надежными и добавляются в АНГ.

Для выбора узлов и построения TBGN авторами [12] используется федеративное обучение для классификации узлов АНГ по их признакам надежности и активности [12]. Однако федеративное обучение требует значительной вычислительной мощности и ресурсов от каждого узла сети, что затрудняет его применение в реальных условиях, особенно в средах с ограниченными ресурсами, таких как IoT. Поэтому в настоящей работе предложено использовать древовидную модель, построенную и размещенную на выделенном сервере, вместо распределенной модели федеративное обучение. Такая модель названы централизованно координируемым блокчейном, поскольку дерево структурируется сервером. Помимо этого, к централизованно координируемому TBGN добавляется механизм плавающего генезис-блока (согласно [16]) для оптимизации хранения данных и сокращения времени подключения новых узлов. Сочетание двух концепций в едином решении обеспечивает высокую производительность, оптимизацию хранения и адаптивность для блокчейна на базе АНГ.

2. Централизованно координируемый древовидный блокчейн на базе АНГ и плавающего генезис-блока

Предлагаемое решение использует централизованно координируемый подход вместо федеративного обучения. Сервер получает информацию об узлах сети, такую как время безотказной работы (Tlive), вычислительная мощность (c), вероятность отказа (Pfault), задержка соединения (Tld) и т.д. На основе этих данных сервер строит древовидную структуру TBGN, назначая узлам

соответствующие позиции (родительские/дочерние), оптимизируя структуру АНГ с учетом надежности и сетевой нагрузки. В результате снижается вычислительная нагрузка на отдельные узлы и упрощается развертывание блокчейн-системы. Построение древовидной структуры блокчейна показано на рисунке 1.

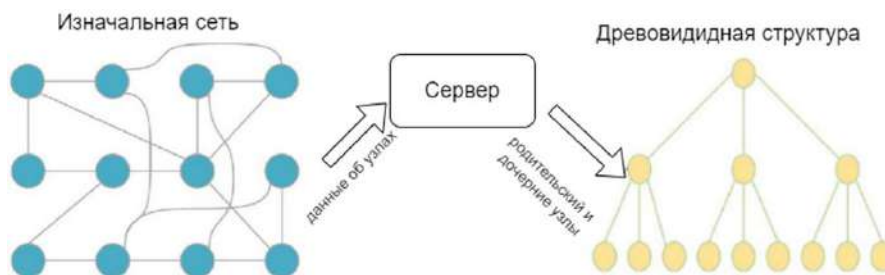


Рис. 1. Построение древовидного блокчейна на базе АНГ

Алгоритм 1 описывает схему построения структуры TBGN. Структура по-прежнему использует протокол gossip для распространения данных, но с той разницей, что после сбора необходимых данных об узлах построение TBGN выполняется централизованно на сервере. Узлы TBGN взаимодействуют иерархически: данные передаются между родительскими и дочерними узлами в дереве, что снижает избыточность и повышает эффективность передачи. При отказе узла механизм устойчивости позволяет дочерним элементам перенастраивать соединения, обходя недоступный узел. Новые узлы, присоединяющиеся к системе, сначала используют классическую схему обмена сообщениями, а затем могут быть включены в обновленную структуру дерева через сервер.

Алгоритм 1 Централизованно координируемое построение TBGN на сервере

Вход: Параметры узла N_i : Tlive, c, Pfault, Tld и др.

Выход: Положение узла в TBGN

1. Узел N_i присоединяется к сети сплетен и передает параметры на центральный сервер.
2. Сервер агрегирует параметры всех активных узлов.
3. На основе полученных данных сервер формирует древовидную структуру TBGN.
4. Узлам назначаются роли (родительский, дочерний) в соответствии с надежностью и топологией.
5. Структура передается всем участникам, и они перестраивают соединения в соответствии с назначением.
6. Добавление новых узлов или обнаружение сбоев инициируют реконфигурацию.

Данный подход обеспечивает более легкую и быструю адаптацию блокчейн-сети к динамическим изменениям масштаба системы, а также упрощает построение дерева TBGN в условиях ограниченных вычислительных ресурсов, обеспечивая отказоустойчивость и структурную эффективность блокчейна.

Для решения проблемы хранения данных в случае больших сетей используется механизм плавающего генезис-блока [16]. Традиционная модель блокчейна (рисунок 2) поддерживает упорядоченный список транзакций, каждая из которых изменяет состояние некоторой переменной, например, s . При этом каждая новая транзакция зависит от предыдущих, и для получения текущего значения s необходимо просмотреть все транзакции, начиная с начального состояния s_0 :

$$s = s_0 + \sum_{i=1}^n \Delta s_i.$$

Это приводит к росту объема блокчейна и размера хранилища на устройства, что становится проблемой для сохранения и обработки новых транзакций в больших масштабируемых системах. Кроме того, по мере увеличения количества транзакций существенно увеличивается и время, необходимое для добавления новых узлов в сеть.

Для решения этой проблемы предлагается модификация блокчейна за счет добавления к нему плавающего генезис-блока. Согласно этому подходу, предлагается сохранять текущее состояние переменной $s_0^{(k)}$ в специальный блок, называемый фиксирующим блоком (рисунок 3). Фиксирующий блок хранит не все изменения переменной, а только ее текущее значение на момент записи блока:

$$s_0^{(k)} = s_0^{(k-1)} + \sum_{i=1}^{n_{k-1}} \Delta s_i^{(k-1)},$$

где n_{k-1} – количество изменений значения переменной между записью блоков $k-1$ и k . Это похоже на то, как начальное значение переменной записывается в генезис-блок в обычном блокчейне.

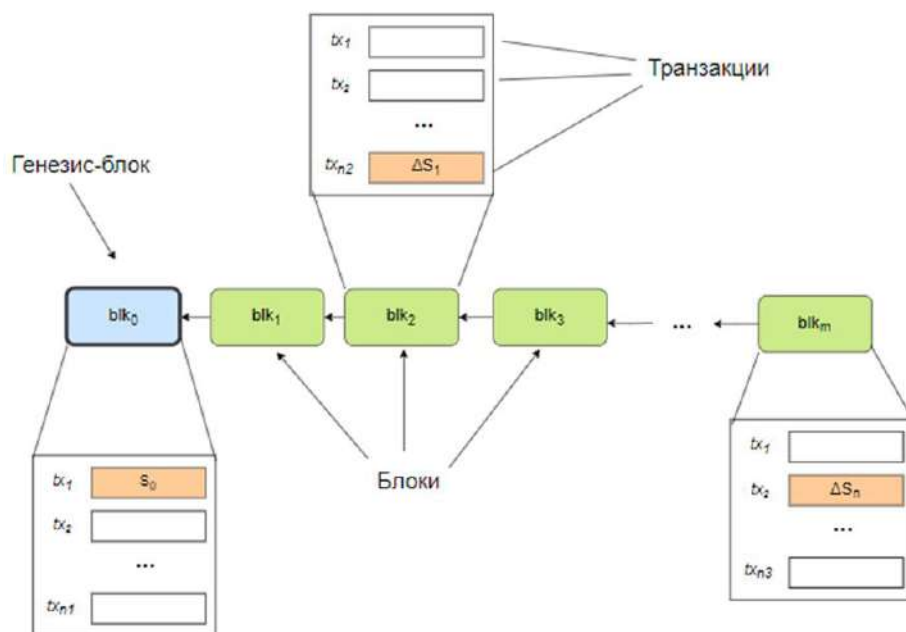


Рис. 2. Блок-схема традиционной модели блокчейна

Получить текущее значение переменной s можно, выполнив все транзакции в списке последовательно, начиная с фиксирующего блока, который является генезис-блоком:

$$s = s_0^{(k)} + \sum_{i=1}^m \sum_{j=1}^{n_i} \Delta S_j^{(i)},$$

где m – количество фиксирующих блоков, следующих за генезис-блоком, в котором хранится значение $s_0^{(k)}$; n_i – количество значений переменной, которые изменились между фиксирующими блоками i и $i + 1$.

Процесс создания и распространения генезис-блока:

1. Создание генезис-блока. При достижении определенного количества транзакций или блоков создается фиксирующий блок, который агрегирует текущее состояние всех активных транзакций, блоков, балансов и других данных, которые важны для дальнейшей работы сети. В отличие от обычного блокчейна, фиксирующий блок не включает в себя всю историю транзакций, а только текущее состояние на момент его создания. Назначенный сетевой узел собирает все необходимые данные, включая ссылку на последний блок, и формирует фиксирующий блок, который будет служить новым генезис-блоком. Этот фиксирующий блок устанавливает точку консенсуса для всей сети в этой точке.

2. Проверка нового генезис-блока. После того, как фиксирующий блок добыт, необходимо проверить этот блок как новый генезис-блок, чтобы он стал стартовой точкой для всех последующих блоков. Для того, чтобы фиксирующий блок стал новым генезис-блоком, необходимо провести голосование среди узлов. Все доверенные узлы голосуют за проверку этого блока. Если блок получает достаточно голосов (например, 2/3 всех узлов), фиксирующий блок подтверждается как новый генезис-блок. Для обеспечения безопасности и предотвращения подделок фиксирующий блок подписывается доверенными узлами. Это гарантирует, что блок не был изменен или подделан и что он действителен. После проверки фиксирующего блока как нового генезис-блока все узлы в сети начинают работать с обновленным блокчейном, при этом этот новый генезис-блок является стартовой точкой.

3. Распространение нового генезис-блока по сети. После одобрения нового генезис-блока он начинает распространяться по всей сети. Узлы, ответственные за распространение данных, передают информацию о новом генезис-блоке в сеть. Это гарантирует, что каждый узел получит актуальную информацию о новом генезис-блоке. Каждый узел в сети, получив новый генезис-блок, обновляет свою локальную копию блокчейна, начиная с нового генезис-блока. Узлы начинают работать с этим новым состоянием, а все старые блоки и ссылки, которые не являются частью активной части сети, исключаются. После распространения нового генезис-блока по сети все блоки, добавленные до него, могут быть удалены или перемещены в архив. Это уменьшает объем данных, которые должны храниться на узлах, минимизируя объем хранилища.

4. Подключение новых узлов и обновление ссылок. Когда новый узел присоединяется к сети, он получает информацию только о текущем генезис-блоке и блоках, добавленных после него. Это позволяет ему быстро синхронизироваться с сетью, не загружая всю цепочку блоков.

5. Обновление АНГ. После одобрения нового генезис-блока все новые блоки в сети начинают ссылаться на новый генезис-блок, создавая обновленный АНГ. Старые блоки, которые больше не являются частью активной сети, исключаются из этих ссылок.

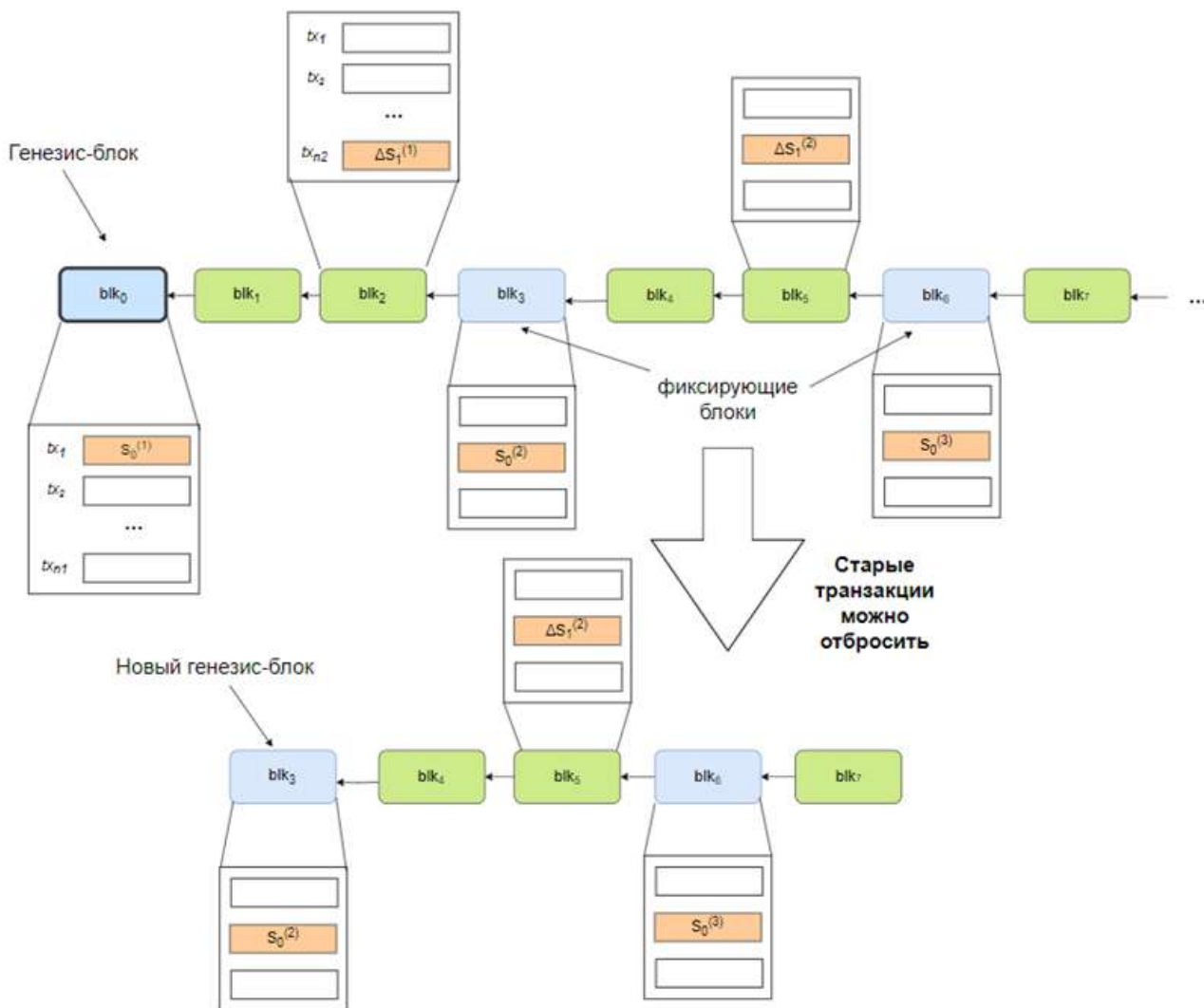


Рис. 3. Схема блокчейна с плавающим генезис-блоком

Таким образом, для решения проблем масштабируемости, хранения данных и производительности блокчейна предложена гибридная архитектура блокчейна. Использование централизованно координируемой древовидной архитектуры АНГ с поддержкой параллельной генерации блоков значительно повышает производительность сети, улучшая ее масштабируемость. Протокол TBGP помогает обеспечить устойчивость сети и балансировку нагрузки, что также повышает ее общую производительность и отказоустойчивость. Объединение древовидной архитектуры АНГ с плавающим генезис-блоком позволяет эффективно справляться с проблемами, связанными с быстрым ростом объемов данных и необходимостью высокой пропускной способности. Механизм плавающего генезис-блока решает проблему хранения, удаляя устаревшие блоки и снижая нагрузку на узлы, что особенно важно для крупных масштабируемых систем, построенных на подключенных устройствах с ограниченными ресурсами.

3. Экспериментальное исследование

Интеграция указанных методов обеспечивает эффективное решение для построения высокопроизводительного и адаптивного блокчейна, способного работать в больших масштабируемых

системах. Для оценки эффективности данного подхода авторами проведена серия экспериментов с использованием платформы Hyperledger Fabric и Python.

Hyperledger Fabric [17] применяется для моделирования и развертывания прототипа блокчейна, включая реализацию механизма выбора верификатора и алгоритма консенсуса. Данная платформа позволила воссоздать многоуровневую структуру сети и протестировать взаимодействие узлов в условиях, приближенных к реальным. Python используется для разработки скриптов и компонентов, реализующих: сбор и анализ параметров узлов для построения структуры TBGN на сервере; имитацию поведения узлов в сети АНГ, включая генерацию, отправку и обработку транзакций; алгоритмы расчета доверия, виртуального голосования и логики включения блоков в граф; визуализацию результатов и оценку производительности тестируемых архитектур.

В результате экспериментов проведено сравнение двух подходов к распределению транзакций в сетях блокчейн: разработанного решения (древовидного блокчейна на базе АНГ с механизмом плавающего генезис-блока) и протокола RGP (в системе Hashgraph он реализует исходный классический блокчейн на базе АНГ). Данные протоколы похожи по своей структуре за счет использования АНГ в блокчейне, но отличаются по эффективности в различных условиях, таких как размер сети, количество транзакций и избыточность связи.

Рисунки 4 и 5 демонстрируют существенные отличия между протестированными подходами в коммуникационных и временных затратах. На рисунке 4 показан объем связей, требуемый для достижения различных уровней покрытия сети с использованием разработанного решения и RGP. Разработанное решение требует существенно меньше коммуникационных ресурсов, чем RGP, для покрытия того же количества узлов, особенно в крупных системах. Например, для размера сети в 500 узлов разработанное решение использует 16% коммуникаций, требуемых для RGP для распространения сообщения по всей сети. Это показывает, что разработанное решение использует более эффективную топологию для обмена сообщениями, что снижает избыточность и повышает производительность сети. При увеличении размера системы до 2000 узлов разработанное решение продолжает демонстрировать значительное снижение избыточных коммуникаций по сравнению с RGP, показывая свою эффективность в больших сетях.

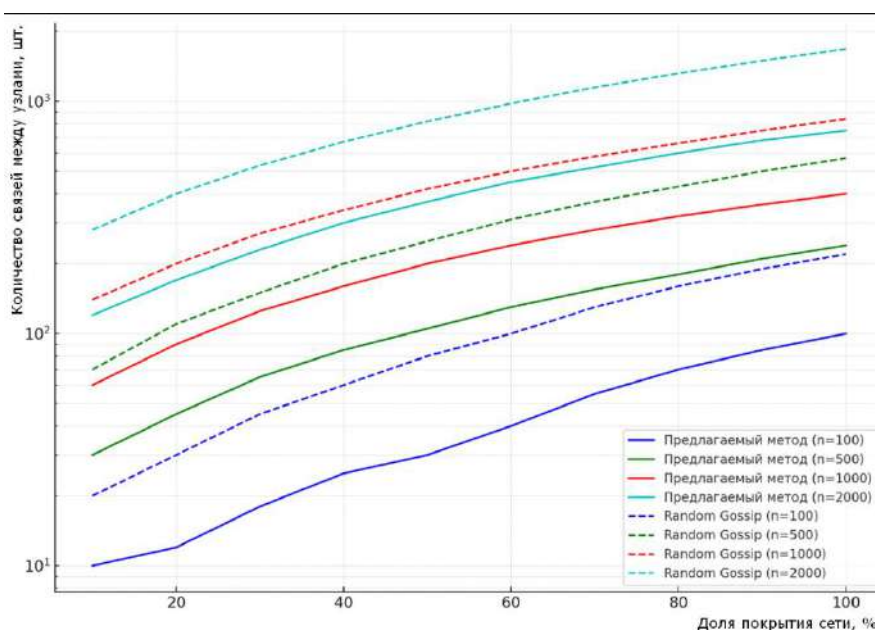


Рис. 4. Объем коммуникаций, необходимый для того, чтобы разработанное решение (древовидный блокчейн на базе АНГ с плавающим генезис-блоком) и RGP (традиционный блокчейн на базе АНГ) охватывали одинаковую долю узлов

На рисунке 5 представлена зависимость времени консенсуса от размера сети. Время достижения консенсуса в сети увеличивается значительно медленнее для разработанного решения, чем для RGP. При увеличении размера сети с 1000 до 2000 узлов время консенсуса для RGP увеличивается на 240%, тогда как для разработанного решения это всего 28% (почти в 10 раз меньше, чем для RGP). Этот результат подтверждает, что разработанное решение гораздо эффективнее справляется с ростом сети, а его влияние на производительность невелико.

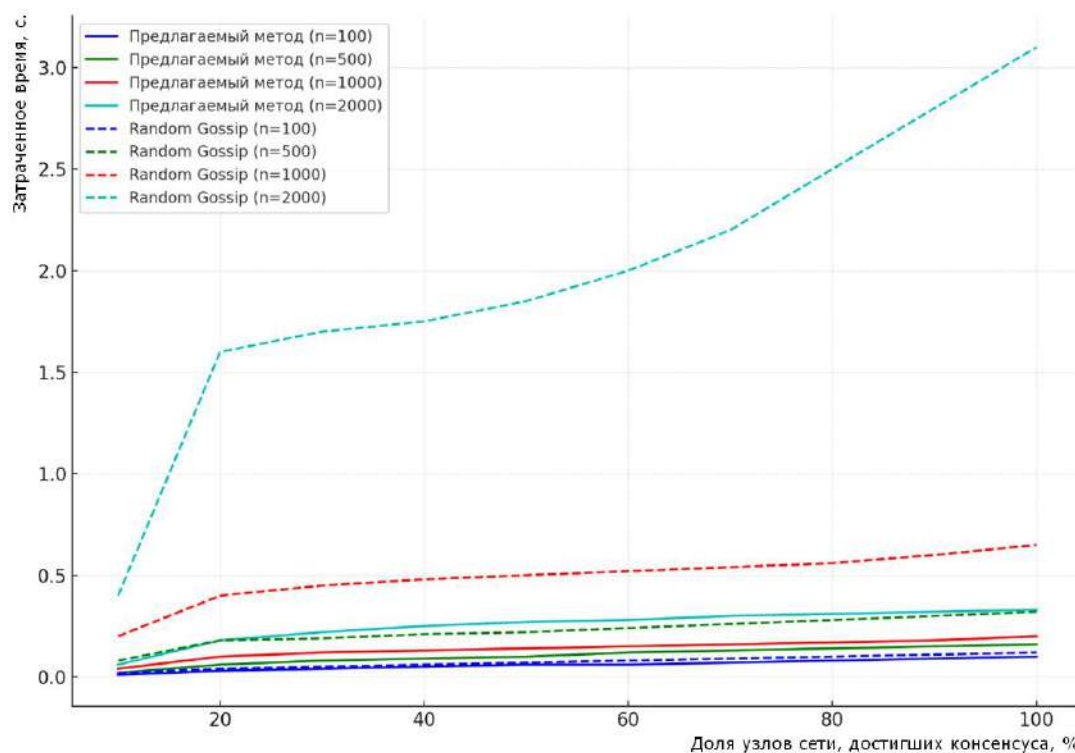


Рис. 5. Доля узлов, достигших консенсуса, и общее затраченное время

Таким образом, экспериментальные результаты сравнения показывают, что разработанный механизм является более эффективным и масштабируемым решением по сравнению с обычным блокчейном АНГ. Он значительно снижает затраты на связь, ускоряет процесс распределения данных и консенсуса и демонстрирует более высокую производительность по мере роста масштаба системы.

4. Заключение

Предлагаемая централизованно координируемая блокчейн-система на основе древовидной структуры и плавающего генезис-блока демонстрирует существенные преимущества по сравнению с традиционным блокчейном. Использование древовидного АНГ позволяет структурировать параллельную обработку транзакций, что значительно повышает масштабируемость и отказоустойчивость системы. В сочетании с плавающим генезис-блоком этот механизм решает проблему неконтролируемого роста блокчейна, позволяя новым узлам быстро присоединяться к сети без необходимости загружать всю историю транзакций.

Практическая ценность исследования заключается в возможности использования разработанных методов для создания безопасных и масштабируемых блокчейн-систем, пригодных для использования в реальных крупных масштабируемых системах, включая системы умных городов, умные фабрики, транспортные системы и другие критически важные инфраструктуры.

Представленный метод открывает новые перспективы для развития блокчейн-технологий с повышенной устойчивостью, масштабируемостью и эффективностью. Дальнейшие исследования нацелены на адаптацию предложенного подхода к конкретным сферам применения и системам.

Литература

1. Adhikari N, Ramkumar M. IoT and Blockchain Integration: Applications, Opportunities, and Challenges // Network. – 2023. – Vol. 3, № 1. – P. 115–141.
2. Santhosham K.P., Nallathambi B.G., Kasirajan P.K. Blockchain and TinyML for low powered IoT devices in smart city applications: Current trends and future prospects // Blockchain Enabled Secure Big Data Computing for Smart Cities Using Internet of Things. – 2023. – P. 215–239.
3. Alazab M., Alhyari S., Awajan A., Abdallah A.B. Blockchain technology in supply chain management: an empirical study of the factors affecting user adoption/acceptance // Cluster Computing. – 2021. – Vol. 24, № 1. – P. 83–101.
4. Alkhalifah A., Kayes A.S.M., Chowdhury J., Alazab M., Watters P.A. A Taxonomy of Blockchain Threats and Vulnerabilities // Blockchain for Cybersecurity and Privacy. – 2020. – P. 3–28.
5. Alajlan R., Alhumam N., Frikha M. Cybersecurity for Blockchain-Based IoT Systems: A Review // Applied Sciences (Switzerland). – 2023. – Vol. 13, № 13.

6. *Lu X., Jiang C.* TEEDAG: A High-Throughput Distributed Ledger Based on TEE and Directed Acyclic Graph // *Electronics (Switzerland)*. – 2023. – Vol. 12, № 11.
7. *Alshahrani H.* Sustainability in Blockchain: A Systematic Literature Review on Scalability and Power Consumption Issues // *Energies*. – 2023. – Vol. 16, № 3.
8. *Ismail L., Materwala H.* A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions // *Symmetry*. – 2019. – Vol. 11, № 10.
9. *Park S., Kim H.* DAG-based distributed ledger for low-latency smart grid network // *Energies*. – 2019. – Vol. 12, № 18.
10. *Benčić F.M., Žarko I.P.* Distributed Ledger Technology: Blockchain Compared to Directed Acyclic Graph // *International Conference on Distributed Computing Systems*. – 2018. – Vol. 2018-July. – P. 1569–1570.
11. *Hellani H., Sliman L., Samhat A.E., Exposito E.* Tangle the Blockchain: Towards Connecting Blockchain and GAN // *Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE*. – 2021. – Vol. 2021–October. – P. 63–68.
12. *Li L., Huang D., Zhang C.* An Efficient DAG Blockchain Architecture for IoT // *IEEE Internet of Things Journal*. – 2022. – DOI: 10.1109/JIOT.2022.3206337.
13. *Tokhmetov V., Tanchenko L.* Development of DAG blockchain model // *Scientific Journal of Astana IT University*. – 2024. – DOI: 10.37943/16cgoy7609.
14. *Wang Q., Yu J., Chen S., Xiang Y.* SoK: DAG-based Blockchain Systems // *ACM Computing Surveys*. – 2023. – DOI: 10.1145/3576899.
15. *Bai Y., Lee S., Seo S.-H.* A Survey on Directed Acyclic Graph-Based Blockchain in Smart Mobility // *Sensors*. – 2025. – Vol. 25, № 4. – P. 1108.
16. *Busygin A., Kalinin M., Konoplev A.* Supporting connectivity of VANET/MANET network nodes and elastic software-configurable security services using blockchain with floating genesis block // *SHS Web of Conferences*. – 2018. – Vol. 44. – P. 00020.
17. Hyperledger Fabric. A Blockchain Platform for the Enterprise. <https://hyperledger-fabric.readthedocs.io/en/release-2.5/> (дата обращения: 17.06.2025).