УДК 004.75
DOI: 10.26102/2310-6018/2025.49.2.015

Приложение задачи поиска минимального покрытия в графе для повышения надежности системы цифровой личности

А.С. Акутин✉, В.В. Печенкин

Саратовский государственный технический университет имени Гагарина Ю.А., Саратов, Российская Федерация

Резюме. В работе рассматриваются особенности построения систем цифровой личности для управления информационно-технологическими процессами предприятия, архитектура которых зависит от децентрализованных реестров данных – блокчейнов. В работе рассмотрены блокчейны как взвешенные графы и сформулирован ряд тезисов, говорящих об особенности функционирования таких распределенных сетей на реальных информационно-технологических предприятиях. Рассмотрены особенности различных топологий сетей и возможных архитектурных уязвимостей и недочетов, которые могут повлиять на работу всей сети – централизация майнинга, централизация стейкинга, различные атаки на функционирующую сеть (топологическая и атака 51 % процента). Рассмотрены блокчейны, использующие различные алгоритмы достижения консенсуса с учетом их особенностей. В работе рассматривается задача поиска минимального покрытия в графе и подчеркивается важность приложения этой задачи к описываемой системе цифровой личности для увеличения надежности компьютерной сети блокчейна за счет анализа ее топологии. Рассмотрены различные методы нахождения минимального покрытия в графе – точные и эвристические алгоритмы. В работе проанализировано приложение, реализующее алгоритм муравьиной колонии для решения поставленной задачи, приводятся численные характеристики работы алгоритма и его формальное описание.

Ключевые слова: система цифровой личности, блокчейн, распределенные системы, графы, поиск минимального покрытия.

Для цитирования: Акутин А.С., Печенкин В.В. Приложение задачи поиска минимального покрытия в графе для повышения надежности системы цифровой личности. *Моделирование, оптимизация и информационные технологии.* 2025;13(2). URL: https://moitvivt.ru/ru/journal/pdf?id=1883 DOI: 10.26102/2310-6018/2025.49.2.015

Application of the task of finding the minimum vertex coverage in a graph to improve the robustness of digital identity system

A.S. Akutin✉, V.V. Pechenkin

Yuri Gagarin State Technical University of Saratov, Saratov, the Russian Federation

Abstract. This paper examines the features of building digital identity systems for managing information technology processes in an enterprise, the architecture of which depends on decentralized data registers - blockchains. The paper considers blockchains as weighted graphs and formulates a number of theses that speak about the specifics of the functioning of such distributed networks in real information technology enterprises. The features of various network topologies and possible architectural vulnerabilities and flaws that can affect the operation of the entire network are considered – centralization of mining, centralization of staking, various attacks on a functioning network (topological and 51% percent attack). Blockchains using various consensus-building algorithms, taking into account their features, are considered. The paper considers the task of finding the minimum coverage in a graph and emphasizes the importance of applying this task to the described digital personality system in order to increase the reliability of the blockchain computer network by analyzing its topology. Various

methods of finding the minimum coverage in a graph are considered – exact and heuristic algorithms. The paper analyzes an application that implements the ant colony algorithm to solve the problem, provides numerical characteristics of the algorithm and its formal description.

Keywords: digital identity system, blockchain, distributed systems, graphs, minimum coverage search.

For citation: Akutin A.S., Pechenkin V.V. Application of the task of finding the minimum vertex coverage in a graph to improve the robustness of digital identity system. *Modeling, Optimization and Information Technology*. 2025;13(2). (In Russ.). URL: https://moitvivt.ru/ru/journal/pdf?id=1883 DOI: 10.26102/2310-6018/2025.49.2.015

Введение

В современном мире распределенные системы встречаются все чаще – наука и корпоративные приложения тяготеют к распределенной архитектуре при решении поставленных задач. Одним из передовых подходов построения системы управления информационно-технологическими процессами предприятия является система цифровой личности – механизм, позволяющий перенести процесс верификации личности в публичное пространство, сократив количество потенциальных атак со стороны злоумышленников [1, 2]. Система цифровой личности основывается на хорошо зарекомендовавшей себя модели данных – публичном распределенном децентрализованном реестре данных (блокчейне). Блокчейн в качестве альтернативы традиционным централизованным подходам стал рассматриваться недавно, однако уже существует множество проектов, говорящих об успешности данного подхода [3].

Блокчейн представляет из себя peer-to-peer сеть без ведущего узла – каждая часть системы имеет такое же значение, как и все другие. Все компоненты блокчейна распределены и публикуются на разных физических устройствах, а общение между ними происходит по сети – по синхронной или асинхронной модели, в зависимости от способа решения той или иной задачи. Плюс децентрализации состоит в том, что количество узлов не регламентировано и может быть переменной величиной – во время эксплуатации такой системы участники-узлы могут как добавляться в сеть, так и удаляться из нее [4]. В конечном варианте мы имеем сложную сетевую модель, где узлы не соединены по принципу «каждый с каждым» и нет единого узла, который мог бы заниматься оркестрацией – информация о состоянии сети передаются при помощи gossip-протокола [5].

В работе [6] приводятся методы анализа сетей блокчейна, позволяющие находить уязвимые места и «бутылочные горлышки», что дает пути дальнейшей оптимизации.

В текущей работе рассматриваются особенности анализа сети блокчейна с точки зрения теории графов для решения задачи поиска минимального покрытия, что позволит увеличить надежность сети за счет анализа изменения ее топологии.

Материалы и методы

Следует обозначить ряд определений и методов теории графов, которые можно использовать в алгоритмах анализа и управления распределенными децентрализованными реестрами. В дальнейшем эти понятия необходимы для того, чтобы рассматривать более сложные концепции приложения теории графов к распределенным системам.

– Центральность (степень центральности, показатель центральности) – определяет «высокосвязанные» узлы, которые взаимодействуют с большим количеством других узлов. Степень центральности – важнейший параметр для определения ключевых узлов в блокчейне [7].

– Степень центральности по расстоянию определяет, как быстро информация из одного узла достигает других узлов за минимальное количество переходов. Для блокчейнов этот параметр демонстрирует, как быстро данные из одного узла смогут быть реплицированы через gossip-протокол [8].

– Кратчайшие пути, диаметр сети и средняя длина пути показывают, насколько легко узлы могут достичь друг друга.

– Мосты и точки сочленения являются важнейшими соединителями, удаление которых может привести к фрагментации графов.

Центральность посредничества: Определим центральность посредничества $C_B(v)$ узла $v \in V$ как

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}, \quad (1)$$

где σ_{st} – общее количество кратчайших путей из s в t, а $\sigma_{st}(v)$ – количество этих путей, проходящих через узел v.

Узлы с более высокими значениями $C_B(v)$ используют больше кратчайших путей в сети. Удаление таких узлов приводит к отключению компонентов, ранее подключенных через них. Анализ распределения показателей центральности сети с течением времени позволяет получить представление о появлении критических узлов и количественно оценить устойчивость сети к топологическим атакам.

В системе цифровой личности очень важным фактором является сложность производства новых блоков – вся значимая информация хранится в блокчейне [9]. Сложность майнинга – процесса создания нового блока и добавления его в публичный реестр – напрямую зависит от выбора алгоритма подтверждения производства нового блока (proof-of-work, proof-of-stake или другие). В работе [10] рассматривается проблема нахождения минимального покрытия графа – понимая, что сложность майнинга является ключевым фактором для динамической модели графа, можно смоделировать поведение графа блокчейна для формализации этого процесса.

Модель блокчейн-сети определяется как динамический граф $G = (V, E)$, где V представляет набор пулов майнинга, а E представляет связи между ними. Степень узла $d(v)$ в G представляет количество соединений, которые пул имеет с другими пулами (в данном случае, речь идет о пересечении узлов через gossip-протокол). Новые узлы (майнинг–пулы) предпочитают подключаться к существующим узлам с более высокой степенью защиты (модель предпочтительного присоединения, аналогичная модели Барабаси-Альберта) [11, 12]. Следствие описанных утверждений можно сформулировать следующим образом: в блокчейн-сети, смоделированной в виде графа в соответствии с приведенными выше предположениями, сеть естественным образом эволюционирует в сторону большей централизации майнинговых мощностей. На основе вышеизложенного можно определить ряд утверждений, характерных для майнинга в блокчейне, смоделированном как граф:

– Пусть $G_0 = (V_0, E_0)$ – начальное состояние блокчейн-сети, где $|V_0|$ – относительно большое, а распределение степеней $D_0 = \{d(v): v \in V_0\}$ является константным.

– Определим вероятность подключения к сети для нового узла, которая подключается к сети и соединяется с существующим узлом v из множества V_0 как

$$P_{attach} = \frac{d(v)}{\sum_{u \in V} d(u)}. \quad (2)$$

– Новый узел v_{new} подключается к существующему узлу v с более высокой вероятностью $P_{attach}(v)$, если степень последнего относительно высока. Каждый новый

узел подключается к k другим узлам, каждое из этих соединений выбирается в соответствии с вероятностями подключения.

– Со временем распределение степеней D_t для $G_t = (V_t, E_t)$ становится искаженным, в пользу узлов с изначально более высокими степенями.

– Асимметрия может быть представлена увеличением дисперсии с течением времени: $Var(D_t) > Var(D_0)$ при $t > 0$ (var – здесь и дальше функция для вычисления дисперсии). Дисперсия может увеличиваться, но все еще стремится к пределу. Похоже, что обычным аргументом в пользу асимметричности такой сети является то, что распределение степеней подчиняется степенному закону. Это действительно так в модели Барабаси-Альберта [12].

– Пусть мощность майнинга $M(v)$ узла v пропорциональна его степени: $M(v) \propto d(v)$. Таким образом, $M(v)$ также все больше искажается по мере того, как D_t становится искаженным.

– Централизацию можно количественно оценить с помощью показателя $C(G_t)$, где $C(G_t)$ увеличивается по мере увеличения асимметрии в $M(v)$. Формально, $C(G_t) > C(G_0)$ при $t > 0$, что указывает на усиление централизации с течением времени.

Таким образом, исходя из вышеописанного, можно сделать вывод, что наличие алгоритма майнинга из разряда труднорешаемых задач, такого как proof-of-work (PoW), ведет к централизации сети, что является контрпродуктивным в разрезе рассматриваемой децентрализованной системы.

В работе [9] был исследован оптимальный алгоритм консенсуса применительно к технологии цифровой личности в управлении информационно-технологическими процессами предприятия. В работе был сделан вывод, что корректным выбором нужного алгоритма консенсуса будет являться именно proof-of-stake (PoS). Данный алгоритм решает ряд проблем, возникающих в классических алгоритмах достижения консенсуса (например, PoW). В обозначенной работе подробно описана механика, преимущества и недостатки обоих алгоритмов.

Необходимо также определить два важных термина из области распределенных децентрализованных реестров, которые впоследствии будут использоваться:

Пул ставок. Пул ставок в блокчейн-системах представляет собой механизм коллективного участия участников в процессе подтверждения транзакций и создания новых блоков в сети с использованием принципа консенсуса, основанного на ставках (staking). Этот механизм широко используется в блокчейнах, работающих на алгоритме консенсуса (PoS) и его вариациях, таких как delegated proof-of-stake (DPoS). Основной функцией пула ставок является объединение ресурсов участников с целью повышения вероятности получения вознаграждения за участие в процессе создания блоков или валидации транзакций. В рамках пула ставок каждый его участник делает определенную ставку, представляющую собой залог в криптовалюте, который служит подтверждением его участия в сети и его доверия к механизму консенсуса. Ставки могут быть как индивидуальными, так и объединенными в пул, что позволяет минимизировать риски, связанные с низким уровнем ставок, а также увеличивает вероятность того, что пул станет одним из валидаторов или участников сети, получающих вознаграждения за свою работу. Основной принцип функционирования пула ставок заключается в том, что ресурсы участников объединяются, и пул в совокупности получает возможность участвовать в процессе валидации блоков. При этом вознаграждение, получаемое в результате участия в процессе консенсуса, распределяется пропорционально объемам ставок, сделанных каждым из участников пула. Такая модель позволяет пользователям с ограниченными ресурсами принимать участие в блокчейн-сетях, которые требуют значительных инвестиций для успешного участия в процессе консенсуса.

Одним из ключевых аспектов пулов ставок является их роль в обеспечении безопасности и децентрализации сети. За счет коллективного участия множества участников увеличивается распределение власти и принятия решений, что способствует снижению риска манипуляций и атак на сеть. Кроме того, пулы ставок стимулируют участников к долгосрочному удержанию активов в сети, что улучшает ее устойчивость и снижает волатильность токенов. Таким образом, пул ставок представляет собой важный элемент экосистемы блокчейн, который способствует повышению доступности и эффективности процессов консенсуса в сетях с использованием алгоритмов PoS и его производных [9].

Хешрейт. Хешрейт (от англ. hash rate) в контексте блокчейн-технологий представляет собой показатель вычислительной мощности сети, который измеряет количество хеш-функций, выполняемых в единицу времени для решения задачи, связанной с поиском корректного значения хеша блока в процессе майнинга или валидации транзакций. Этот параметр является важным индикатором эффективности и безопасности блокчейн-сети, особенно в системах, использующих алгоритм консенсуса PoW.

В блокчейн-сети, смоделированной как граф, каждый узел $v \in V$ представляет пул ставок, а каждая дуга $(u, v) \in E$ представляет делегирование ставок из пула u в пул v. Вес каждого узла $w(v)$ соответствует общей доле, делегированной пулу v. Влияние или сила пула ставок прямо пропорциональна его общей сумме. Новые заинтересованные стороны, как правило, делегируют свои доли пулам с более высокими существующими ставками, моделируя преимущественную привязанность, аналогичную модели Барабаси-Альберта.

В блокчейн-сети, смоделированной в виде ориентированного графа в соответствии с вышеуказанными предположениями, сеть эволюционирует к состоянию, когда небольшое количество пулов ставок накапливает непропорционально большую сумму общих ставок, что указывает на тенденцию к централизации управления.

Формализуем указанные выше утверждения:

– Пусть $G_0 = (V_0, E_0)$ представляет начальное состояние графика, а $w(v_0)$ обозначает ставку для каждого пула $v_0 \in V_0$, где $w(v_0)$ относительно равномерно распределено по V_0.

– Определим вероятность делегирования ставки пулу от нового связанного через gossip-протокол узла как подключение:

$$P_{delegate}(v) = \frac{w(v)}{\sum_{u \in V} w(u)}. \quad (3)$$

Новые заинтересованные стороны с большей вероятностью делегируют свои полномочия пулам с более высоким $w(v)$ в соответствии с преимущественным подключением.

– С течением времени эта динамика приводит к появлению эволюционировавшего графа $G_t = (V_t, E_t)$ в момент времени t, где распределение $w(v)$ для $v \in V_t$ становится все более искаженным. Эта асимметрия представлена возрастающей дисперсией: $Var(w(V_t)) > Var(w(V_0))$ для $t > 0$.

– Количественно определим централизацию в момент времени t с помощью $C_{stake}(G_t)$, которая увеличивается с асимметрией $w(v)$. Формально, $C_{stake}(G_t) > C_{stake}(G_0)$ для $t > 0$, что указывает на тенденцию к увеличению централизации майнинг-пулов [11].

Таким образом, можно сделать вывод, что рассматриваемая в этой работе система цифровой личности может являться более централизованной не только из-за особенностей блокчейна, но и из-за топологических особенностей сети.

Ключевой особенностью системы цифровой личности являются существенные улучшения в области безопасности данных и пользователей – в работах [1, 2] были подробно рассмотрены особенности построения архитектуры подобных систем и полученные преимущества, а в работе [9] был проведен анализ механизмов достижения консенсуса, которые могут быть использованы в технологии цифровой личности. В свою очередь, сама архитектура блокчейна подвержена ряду атак со стороны злоумышленников, некоторые из которых могут нанести существенный урон сети. В качестве одной из таких атак можно выделить топологическую атаку на значимые участки блокчейн-сети. Рассмотрим описание этой проблемы.

Критические узлы: узлы, удаление которых приводит к фрагментации топологии сети на множество несвязанных компонентов. Точки сочленения определяются следующим образом: узлы, удаление которых увеличивает количество связанных компонент в графе. Таким образом, в блокчейне может возникнуть состояния partition – когда сеть распалась на N независимых подсетей, не связанных между собой. При состоянии partition система цифровой личности не может выполнять основные функциональные сценарии, таким образом, задача инженеров и архитекторов – сделать все, чтобы не допускать этой ситуации, Мосты определяются как ребра, удаление которых приводит к увеличению числа компонент связности. Узлы с высокой центральностью посредничества являются важными узловыми точками или мостами в блокчейн-сети. Целенаправленное удаление таких узлов представляет собой эффективный топологический вектор атаки для фрагментации сети. Стоит определить ряд важных факторов, которые влияют на понимание топологической атаки.

Таким образом, для исследователей и разработчиков распределенных систем очень важно иметь представления о топологии сети, чтобы как можно быстрее купировать потенциальные атаки недоброжелателей.

Говоря о блокчейн-сетях, нельзя не формализовать одну из самых опасных архитектурных уязвимостей, которой является «Атака 51 %» – если злоумышленник будет контролировать более половины узлов сети (51 % и более), то в таком случае атакующий будет владеть всей сетью блокчейна и размещать в ней любые данные. Ниже приведена формализация данного процесса.

Рассмотрим блокчейн-сеть, представленную в виде взвешенного ориентированного графа $G = (V, E, w)$, где:

– V: Набор узлов, где каждый узел v представляет майнера в сети.

– E: Набор направленных ребер, представляющих вклады в скорость хэширования от одного узла к другому (в случае майнинга пулов).

– $w(e)$: Вес ребра e, представляющий скорость хэширования и доставки новых значений на соседний узел

В сетевом графе G блокчейна объект, управляющий набором узлов $V_A \subseteq V$, таким образом, что сумма весов исходящего хэшрейта от V_A превышает 50 % от общего хэшрейта сети, может выполнить атаку на 51 % [11]. Сформулируем формальные факторы, позволяющие понять особенности описываемого процесса:

– пусть W_T – общий сетевой хэшрейт, вычисляемый как сумма весов всех исходящих ребер в G:

$$W_T = \sum_{e \in E} w(e), \tag{4}$$

– злоумышленник контролирует подмножество узлов V_A; пусть W_A – скорость хэширования, контролируемая злоумышленником, вычисляемая как:

$$W_A = \sum_{v \in V_A} \sum_{e \to v} w(e), \tag{5}$$

где $e \to v$ обозначает ребро e, выходящее из вершины e;

– для того чтобы атакующий выполнил атаку с 51 %-ной эффективностью, W_A должен превышать половину W_T:

$$W_A > \frac{W_T}{2}. \quad (6)$$

Таким образом, контроль узлов и состояния всей блокчейн сети является ключевым фактором в управлении таких систем, как система цифровой личности

Задача минимального покрытия в графах и ее приложение к системе цифровой личности

Описанные выше особенности и архитектурные уязвимости в блокчейн-сетях являются неотъемлемой платой за все те преимущества децентрализации, которые распределенный децентрализованный реестр привносит после внедрения в инфраструктуру проекта. Зная обо всех описанных нюансах, разработчики и архитекторы могут прибегнуть к своевременному анализу системы и предпринять ряд шагов для своевременного купирования потенциальных угроз и проблем. Мощным инструментом в руках разработчиков децентрализованных систем является возможность нахождения минимального покрытия в графе.

Минимальным вершинным покрытием называется вершинное покрытие, состоящее из наименьшего числа вершин инцидентных всем ребрам графа. Задача о поиске минимального вершинного покрытия является *NP*-полной задачей, что обязывает исследователей и разработчиков искать эвристические алгоритмы для ее решения [10].

В предыдущем разделе были сделаны 4 утверждения, моделирующие распределенный цифровой реестр данных (блокчейн) как граф. В работе [11] содержится еще больше формальных моделей, которые позволяют приложить элементы теории графов к блокчейнам и сделать теоретические и практические выводы. Исходя из рассмотренного, можно заметить, что блокчейн не является универсальной моделью данных и, так же как и традиционные централизованные решения, обладает рядом архитектурных особенностей и уязвимостей. Задача разработчиков и исследователей как можно сильнее децентрализовать блокчейн сеть, чтобы рассмотренные топологические атаки и возможности централизации майнинга и стейкинга стремились к нулю. Для анализа блокчейн-сетей подходит метод теории графов – поиск минимального вершинного покрытия. Нахождение минимального вершинного покрытия позволит обнаружить мосты, точки сочленения и критические узлы в сети, а также рассмотреть топологию построенного графа с точки зрения централизации узлов для предотвращения потенциальных проблем, рассмотренных выше.

Обсудим подробнее задачу поиска минимального вершинного покрытия в приложении к блокчейнам и к системе цифровой личности для управления информационно-технологическими процессами предприятия. Распределенный реестр данных (блокчейн), как было описано выше, является децентрализованной системой без единой точки управления. Обсуждения механики построения системы цифровой личности, можно допустить централизованное управление блокчейном, хотя это не является обязательным пунктом. Нахождение минимального вершинного покрытия позволит выявить ключевые узлы, которые являются критическими точками отказа. Рассмотрим пример графа на Рисунке 1.

На Рисунке 1 изображен граф, состоящий из 9 узлов. Пусть этот граф представляет собой блокчейн, где ребро означает наличие сетевой связности между узлами (например, для обеспечения gossip-протокола) – сетевая связанность обеспечивается средствами протокола TCP/IP. Каждый узел в данном случае является

серверным компьютером, который поддерживает функционирование блокчейна (так же называемый как нода, от англ. Node – узел). Минимальное вершинное покрытие *S* содержит 3 вершины, которые выделены на рисунке оранжевым цветом. На этом примере можно заметить наличие следующих проблем:

– Потенциальному злоумышленнику достаточно получить контроль над 3 ключевым узлами, чтобы подделать gossip-протокол

– Отказ 2 из 3 узлов множества *S* приведет сеть практически в неработоспособное состояние, создав множество независимых (сетевым образом) друг от друга кластеров – состояния partition. В этом состоянии сеть просто не может функционировать.

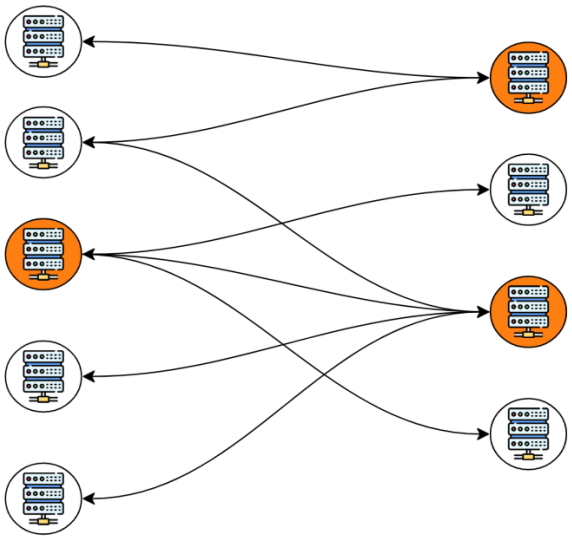

Рисунок 1 – Пример сети блокчейна
Figure 1 – An example of a blockchain network

Предварительный анализ такой сети (обнаружение ключевых узлов путем поиска минимального вершинного покрытия) позволит заранее предсказывать и устранять критичные для системы ситуации проактивно.

Рассмотрев особенности предметной области, можно сформулировать поставленную задачу: в распределенном децентрализованном реестре (блокчейне), смоделированном как граф, необходимо эффективно решать задачу поиска минимального вершинного покрытия.

Способы решения поставленной задачи

Как было рассмотрено раннее, нахождение минимального вершинного покрытия является *NP* – полной задачей, что не позволяет использовать подход полного перебора в реальных системах с большими сетями. Рассмотрим несколько альтернативных сценариев, которые позволят решить задачу более эффективно.

Точные алгоритмы. Точные алгоритмы – это класс алгоритмов, который позволяет гарантированно найти точное решение, хотя время их работы неудовлетворительно для промышленных систем. Среди точных алгоритмов выделяются такие, как полный перебор и алгоритм ветвей и границ.

В работе [13] разбирается и анализируется алгоритм ветвей и границ. Итеративный алгоритм создаст дерево поиска. Чтобы масштабировать размер экземпляра, нам нужно выбрать для него меру. Общей мерой графовой задачи является

количество вершин или ребер в графе. Ограничивая размер дерева поиска функцией меры, мы получим ограничение времени выполнения, связанное с мерой для задачи. Здесь можно определить μ как меру графа, а $C(\mu)$ – как худший случай, то есть наибольший размер поискового дерева при μ. Далее алгоритм предполагает рекуррентный поиск с построением дерева поиска по вершинам графа такой, что

$$C(\mu) \leq \sum_{i=1}^{l} C(\mu - \mu'), \tag{7}$$

где μ' определяется как $\mu_i' = \mu - \mu_i$, а l – количество вершин графа.

Исходя из анализа, проведенного в работе [13], можно сделать однозначный вывод, что применение точных алгоритмов оправдано лишь для сравнительно небольших графов и не подойдет для коммерческого применения в системе цифровой личности на основе распределенного реестра данных.

Эвристические алгоритмы. Эвристические алгоритмы в графах – это методы поиска решения задачи, которые используют эвристические функции (или правила) для приближенного решения, не гарантируя оптимальность, но предлагая достаточно хорошие результаты за разумное время. Основная цель эвристических алгоритмов – найти решение как можно быстрее, не обязательно идеально, что особенно важно в задачах с большими графами или сложными условиями, где полный перебор всех вариантов (например, методом динамического программирования или полного поиска) может быть слишком ресурсоемким. К эвристическим алгоритмам относятся такие алгоритмы, как жадный алгоритм, муравьиный алгоритм, генетический алгоритм и другие

В работе [14] описывается приближенный локальный 2-аппроксимирующий алгоритм поиска минимального покрытия при помощи. Локальный 2-аппроксимирующий алгоритм основывается на построении максимального паросочетания в графе и выборе концов ребер этого паросочетания в качестве покрытия вершин. Алгоритм состоит из следующих шагов:

1. Построение максимального паросочетания: найти максимальное паросочетание в графе *G*. Паросочетание *M* – это, множество ребер, не имеющих общих вершин, при этом никакое расширение *M* парой дополнительных ребер не допускается без нарушения этого условия.

2. Выбор вершин из паросочетания: для каждого ребра $(u, v) \in M$ добавить обе вершины *u* и *v* в покрытие вершин *C*.

3. Завершение: возвратить множество *C*, содержащее все вершины, выбранные на предыдущем шаге, в качестве покрытия вершин.

Рассмотрим описанный алгоритм более формально. Пусть $G = (V, E)$ – заданный граф. Пусть *M* – максимальное паросочетание в *G*, тогда:

$$[M = (u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)], \tag{8}$$

где

$$u_i, v_i \in V, \tag{9}$$

$$(u_i, v_i) \in E, \tag{10}$$

и для всех $i \in [1, k]$ справедливо ($u_i \neq u_j$) и ($v_i \neq v_j$). Тогда множество покрытия вершин *C* определяется как

$$[C = u_i, v_i \mid (u_i, v_i) \in M], \tag{11}$$

Рассмотрим оптимальное покрытие вершин C^* минимальной мощности для графа *G*. Поскольку *M* является максимальным паросочетанием, такое покрытие вершин

должно включать по крайней мере одну вершину из каждой пары $(u_i, v_i) \in M$ таким образом, оптимальное покрытие должно удовлетворять неравенству:

$$[|C^*| \geq |M|]. \quad (12)$$

Поскольку алгоритм включает обе вершины из каждой пары *M*, размер покрытия *C* удовлетворяет неравенству:

$$[|C| = 2|M|] \leq 2|C^*|. \quad (13)$$

Таким образом, алгоритм является 2-аппроксимирующим, так как размер найденного покрытия вершин не превышает двукратного размера минимального покрытия. Алгоритм предлагает эффективный и простейший подход, основанный на построении максимального паросочетания и выборе его концов в качестве покрытия. Данный алгоритм обладает гарантией аппроксимации вдвое превышающего оптимальное решение, что делает его практически применимым для широкого класса графов.

В работе [15] рассматривается один из способов решения задачи минимального покрытия через муравьиный алгоритм или же алгоритм паукообразной колонии (Ant Colony Algorithm). Муравьиный алгоритм представляет собой стохастический метод оптимизации, основанный на моделировании поведения муравьев при поиске пути к источникам пищи. В контексте задачи покрытия вершинами, алгоритм используется для поиска оптимального или приближенного решения путем координации действий «искусственных муравьев» на графе *G*. Ниже представлено краткое описание шагов алгоритм:

1. Представление Решения: Каждое решение задачи покрытия вершин представляется как подмножество вершин $C \subseteq V$

2. Инициализация Феромонов: на каждой вершине $v \in V$ устанавливается начальное количество феромонов $\tau_0 > 0$

3. Итерационный процесс поиска решений: В каждом итеративном цикле «муравьи» строят покрытия вершин, основываясь на распределении феромонов и эвристической информации.

4. Оценка приспособленности: Качество построенного покрытия оценивается с использованием функции приспособленности, обычно зависящей от мощности покрытия и полноты его покрытия.

5. Обновление феромонов: после построения решений происходит обновление значений феромонов на вершинах с учетом качества найденных покрытий.

6. Условие завершения: Алгоритм завершается при достижении предельного числа итераций или при отсутствии значительного улучшения качества решений.

Важно отметить, что правило перехода состояний представляет собой случайную пропорциональную схему, которая назначает вероятность, с которой муравей *k* на вершине *i* выбирает вершину *j* в качестве следующей вершины для посещения, и эта вероятность может быть рассчитана по следующему уравнению:

$$p_{ij}^k = \begin{cases} 1,\ q < q_0\ and\ j = arg \max\limits_{r \in A_k} \left\{\tau_{ir}\eta_{ir}^{\beta}\right\} \\ 0,\ q < q_0\ and\ j \neq arg \max\limits_{r \in A_k} \left\{\tau_{ir}\eta_{ir}^{\beta}\right\}, \\ \frac{\tau_{ir}\eta_{ir}^{\beta}}{\sum_{r \in A_k}\tau_{ir}\eta_{ir}^{\beta}},\ q \geq q_0 \end{cases} \quad (14)$$

где A_k – множество вершин, которые открыты или доступны для муравья *k*, τ_{ir} – динамическая мера желательности (называемая феромоном в биологическом

смысле) доступа к вершине (i, j), η_{ir} – является статической мерой желательности относительно того же края, основанной на локальной эвристике, специфичной для данной проблемы, а β – параметр, контролирующий относительную значимость между двумя показателями.

Как показано в работе [15], муравьиный алгоритм является алгоритмом вероятностным, что, очевидно, не дает максимально точного результата, но предоставляет приемлемую асимптотическую сложность для работы с большими графами.

Результаты и их обсуждение

Как было обсуждено выше, рассматривая систему цифровой личности как систему, функционирующую поверх распределенной децентрализованной сети, возможно представить саму сеть как взвешенный ориентированный граф. Для анализа сети, который позволит найти архитектурные уязвимости (например, найти ключевые узлы для осуществления топологической атаки), будет удобно решить задачу построения пути для минимального покрытия. В качестве примера был использован рассмотренный выше алгоритм муравьиной колонии. После построения всех покрытий вершинами на итерации t происходит обновление феромонов на вершинах согласно следующим правилам:

$$\left[\tau_i^{(t)} = (1 - p) \cdot \tau_i^{(t-1)} + \Delta\tau_i^{(t)}\right], \quad (15)$$

где p – коэффициент испарения феромонов, $p \in [0,1)$. Данный коэффициент отвечает за уменьшение значений феромонов с течением времени. $\Delta\tau_i^{(t)}$ – добавка феромонов, зависящая от качества построенных покрытий, которая вычисляется следующим образом:

$$\left[\Delta\tau_i^{(t)} = \sum_{k=1}^{K} \Delta\tau_i^{(t,k)}\right], \quad (16)$$

где $\Delta\tau_i^{(t,k)}$ зависит от того, принадлежит ли вершина v_i покрытию $C^{(t,k)}$. Применяются следующие правила:

$$\left[\Delta\tau_i^{(t,k)} = Q \,/\, \left|C^{(t,k)}\right|\right], \quad (17)$$

где Q – константа, определяющая общий объем выделяемого феромона, $Q > 0$.

В качестве основы была взята базовая реализация алгоритма муравьиной колонии, которая позволяет выполнять поиск минимального покрытия при помощи описанного выше алгоритма. Программа разработана на языке Python с использованием открытой библиотеки numpy. Используя топологию сетей из прототипов, разработанных в предыдущих работах [1, 9], удалось провести сканирование сетей при помощи описанной методики. Граф представлял из себя квадратную матрицу инцидентности, петель не было предусмотрено. Время, за которое алгоритм находил минимальное покрытие, представлено в Таблице 1.

Таблица 1 – Время поиска минимального покрытия
Table 1 – Minimum coverage search time

Размерность матрицы инцидентности	Количество итераций	Коэффициент распада феромона	Время поиска минимального покрытия (секунды)
100×100	100	0,95	0,356
150×150	100	0,90	1,873

Таблица 1 (продолжение)
Table 1 (continued)

200×200	100	0,90	2,448
300×300	100	0,90	4,468
400×400	100	0,95	5,608
500×500	100	0,90	8,823
1000×1000	100	0,95	15,56
1500×1500	100	0,90	50,60
2000×2000	100	0,95	51,34

Исходя из Таблицы 1, можно сделать вывод, что алгоритм решает поставленную задачу успешно – детерминировано и за конечное время. Возможны дальнейшие уточнения и улучшения применения алгоритма с возможностью влиять на описанные параметры (изменять количество муравьев, коэффициент распада феромона и другие параметры), однако в рамках текущего исследования полученные результаты удовлетворяют поставленной задаче.

Тестирование производилось на процессоре 2 GHz Quad-Core Intel Core i5, оперативная память – 16 Гб DDR4.

Таким образом, возможно сделать вывод, что топологический анализ достаточно больших графов (размерностью 2000 вершин) возможен за вполне обозримое время, достаточное, для быстрого реагирования на потенциальные угрозы и анализа архитектуры на предмет неэффективности.

Важно отметить, что граф, которым моделируется блокчейн-сеть, является динамическим – узлы могут подключаться и отключаться в зависимости от потребностей держателей блокчейна. Однако, рассматривая систему цифровой личности в контексте управления информационно-технологическим предприятием, можно говорить, что все вычислительные узлы или же большая часть из них может быть под управлением администраторов этой системы. Таким образом, подключение и отключение новых узлов – процесс не постоянный, изменение состояние графа происходит достаточно редко, что дает нам возможность приложения задачи поиска минимального покрытия в графе для динамической блокчейн-сети.

После нахождения одного или нескольких минимальных покрытий для графа, стоит рассмотреть вопрос центральности. Как было отмечено выше, центральность вершины блокчейн-сети в разрезе рассматриваемой модели обозначает количество подключений к узлам других узлов (связь через gossip-протокол). Для описанной модели высокая степень центральности говорит о том, что узел уязвим, в частности, к топологической атаке – выведение из строя этого ключевого узла может усложнить коммуникацию других элементов сети или даже привести к состоянию разделения (partition). После нахождения описанным выше алгоритмом муравьиной колонии множества, представляющего минимальное покрытие в графе, следует проанализировать степени центральности вершин, входящих в это множество. Те вершины, которые будут обладать наибольшей степенью, являются наиболее уязвимыми, а потому требуют повышенного внимания системных архитекторов и администраторов системы.

Заключение

Дальнейшие исследования по приложению исследований в области нахождения минимального покрытия в сети системы цифровой личности на основе распределенного реестра данных, которая представлена в виде графа, представляются перспективными.

Уже сейчас можно делать выводы о топологии сети достаточно больших графов. Также в работе были проанализированы различные методы решения поставленной задачи, а также рассмотрены некоторые приложения теории графов к распределенным сетям.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Акутин А.С. Технология суверенной личности как новый подход в обработке персональных данных. В сборнике: *Проблемы управления в социально-экономических и технических системах: материалы XIX Международной научно-практической конференции, 13–14 апреля 2023 года, Саратов, Россия.* Саратов: ИЦ «Наука»; 2023. С. 128–134.
2. Акутин А.С., Денисова З.П. Технология цифровой личности в управлении технологическим предприятием. В сборнике: *Проблемы управления в социально-экономических и технических системах: материалы XX Международной научно-практической конференции: сборник научных статей, 17–18 апреля 2024 года, Саратов, Россия.* Саратов: ИЦ «Наука»; 2024. С. 154–156.
3. Vayadande K., Baviskar A., Avhad J., Bahadkar S., Bhalerao P., Chimkar A. A Comprehensive Review on Navigating the Web 3.0 Landscape. In: *2024 Second International Conference on Inventive Computing and Informatics (ICICI), 11–12 June 2024, Bangalore, India.* IEEE; 2024. P. 456–463. https://doi.org/10.1109/ICICI62254.2024.00080
4. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. SSRN. URL: https://doi.org/10.2139/ssrn.3440802 [Accessed 4th February 2025].
5. Van Ditmarsch H., Gattinger M., Ramezanian R. Everyone Knows That Everyone Knows: Gossip Protocols for Super Experts. *Studia Logica.* 2023;111(3):453–499. https://doi.org/10.1007/s11225-022-10032-3
6. Song W., Zhang W., Wang J., Zhai L., Jiang P., Huang Sh. Blockchain Data Analysis from the Perspective of Complex Networks: Overview. *Tsinghua Science and Technology.* 2023;28(1):176–206. https://doi.org/10.26599/TST.2021.9010080
7. Gochhayat S.P., Shetty S., Mukkamala R., Foytik P., Kamhoua G.A., Njilla L. Measuring Decentrality in Blockchain Based Systems. *IEEE Access.* 2020;8:178372–178390. https://doi.org/10.1109/ACCESS.2020.3026577
8. Madine M., Salah Kh., Jayaraman R., Al-Hammadi Yo., Arshad J., Yaqoob I. appXchain: Application-Level Interoperability for Blockchain Networks. *IEEE Access.* 2021;9:87777–87791. https://doi.org/10.1109/ACCESS.2021.3089603
9. Акутин А.С., Бровко А.В. Децентрализованный реестр данных в технологии суверенной личности. *Инженерный вестник Дона.* 2023;(6):232–246.
   Akutin A.S., Brovko A.V. Decentralized Data Registry in Sovereign Identity Technology. *Engineering Journal of Don*. 2023;(6):232–246. (In Russ.).
10. Vega F. The Minimum Vertex Cover Problem. [Preprint]. ResearchGate. URL: https://www.researchgate.net/publication/388526292_The_Minimum_Vertex_Cover_Problem [Accessed 13th February 2025].
11. Jayabalasamy G., Pujol C., Latha Bhaskaran K. Application of Graph Theory for Blockchain Technologies. *Mathematics.* 2024;12(8). https://doi.org/10.3390/math12081133
12. Barabási A.-L., Albert R. Emergence of Scaling in Random Networks. *Science.* 1999;286(5439):509–512.
13. Xiao M., Nagamochi H. Exact Algorithms for Maximum Independent Set. *Information and Computation.* 2017;255:126–146. https://doi.org/10.1016/j.ic.2017.06.001

14. Åstrand M., Floréen P., Polishchuk V., Rybicki J., Suomela J., Uitto J. A Local 2-Approximation Algorithm for the Vertex Cover Problem. In: *Distributed Computing: 23rd International Symposium, DISC 2009: Proceedings, 23–25 September 2009, Elche, Spain.* Berlin, Heidelberg: Springer; 2009. P. 191–205. https://doi.org/10.1007/978-3-642-04355-0_21
15. Shyu Sh.J., Yin P.-Ye., Lin B.M.T. An Ant Colony Optimization Algorithm for the Minimum Weight Vertex Cover Problem. *Annals of Operations Research.* 2004;131:283–304. https://doi.org/10.1023/B:ANOR.0000039523.95673.33

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Акутин Артем Сергеевич, аспирант кафедры «Прикладные информационные технологии» Саратовского государственного технического университета имени Гагарина Ю.А., Саратов, Российская Федерация.
e-mail: akutin_artem@mail.ru
ORCID: 0009-0001-1170-5415

Artem S. Akutin, Postgraduate at the Department of Applied Information Technologies, Yuri Gagarin State Technical University of Saratov, Saratov, the Russian Federation.

Печенкин Виталий Владимирович, доктор социологических наук, кандидат физико-математических наук, профессор, профессор кафедры «Прикладные информационные технологии» Саратовского государственного технического университета имени Гагарина Ю.А., Саратов, Российская Федерация.
e-mail: pechenkinvv@mail.ru
ORCID: 0000-0002-5043-1891

Vitaly V. Pechenkin, Doctor of Sociological Sciences, Candidate of Physical and Mathematical Sciences, Professor, Professor at the Department of Applied Information Technologies, Yuri Gagarin State Technical University of Saratov, Saratov, the Russian Federation.

Статья поступила в редакцию 04.04.2025; одобрена после рецензирования 18.04.2025; принята к публикации 24.04.2025.

The article was submitted 04.04.2025; approved after reviewing 18.04.2025; accepted for publication 24.04.2025.