

УДК 004.942, 004.896

Модель распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированными системами интеллектуальных зданий

Р. В. Душкин^{1,а}, А. И. Мохов²

¹ООО «Ростелематика»,
Россия, 603159, Нижегородская обл., г. Нижний Новгород, ул. Карла Маркса, д. 60в

²Национальный исследовательский московский государственный строительный университет (НИУ МГСУ),
Россия, 129337, г. Москва, Ярославское ш., д. 26

E-mail: ^аroman.dushkin@gmail.com

Получено 28.04.2020, после доработки — 08.11.2020.

Принято к публикации 20.04.2021.

Настоящая статья описывает разработанную авторами модель построения распределенной вычислительной сети и осуществления в ней распределенных вычислений, которые выполняются в рамках программно-информационной среды, обеспечивающей управление информационными, автоматизированными и инженерными системами интеллектуальных зданий. Представленная модель основана на функциональном подходе с инкапсуляцией недетерминированных вычислений и различных побочных эффектов в монаδικские вычисления, что позволяет применять все достоинства функционального программирования для выбора и исполнения сценариев управления различными аспектами жизнедеятельности зданий и сооружений. Кроме того, описываемая модель может использоваться совместно с процессом интеллектуализации технических и социотехнических систем для повышения уровня автономности принятия решений по управлению значениями параметров внутренней среды здания, а также для реализации методов адаптивного управления, в частности применения различных техник и подходов искусственного интеллекта. Важной частью модели является направленный ациклический граф, который представляет собой расширение блокчейна с возможностью существенным образом снизить стоимость транзакций с учетом выполнения смарт-контрактов. По мнению авторов, это позволит реализовать новые технологии и методы (распределенный реестр на базе направленного ациклического графа, вычисления на краю и гибридную схему построения искусственных интеллектуальных систем) и все это вместе использовать для повышения эффективности управления интеллектуальными зданиями. Актуальность представленной модели основана на необходимости и важности перевода процессов управления жизненным циклом зданий и сооружений в парадигму Индустрии 4.0 и применения для управления методов искусственного интеллекта с повсеместным внедрением автономных искусственных когнитивных агентов. Новизна модели вытекает из совокупного рассмотрения распределенных вычислений в рамках функционального подхода и гибридной парадигмы построения искусственных интеллектуальных агентов для управления интеллектуальными зданиями. Работа носит теоретический характер. Статья будет интересна ученым и инженерам, работающим в области автоматизации технологических и производственных процессов как в рамках интеллектуальных зданий, так и в части управления сложными техническими и социотехническими системами в целом.

Ключевые слова: автоматизация, интеллектуализация, управление, система управления, автономность, адаптивность, функциональный подход, распределенные вычисления, вычисления на краю, интеллектуальное здание

UDC: 004.942, 004.896

Distributed computing model for the organization of a software environment that provides management of intelligent building automation systems

R. V. Dushkin^{1,a}, M. I. Mohov²

¹JSC “Rostelematika”,
60v Karl Marx st., Nizhny Novgorod, 603159, Russia

²National research Moscow state University of civil engineering (NRU MGSU)
26 Yaroslavskoe shosse, Moscow, 129337, Russia

E-mail: ^aroman.dushkin@gmail.com

*Received 28.04.2020, after completion — 08.11.2020.
Accepted for publication 20.04.2021.*

The present article describes the authors' model of construction of the distributed computer network and realization in it of the distributed calculations which are carried out within the limits of the software-information environment providing management of the information, automated and engineering systems of intellectual buildings. The presented model is based on the functional approach with encapsulation of the non-determined calculations and various side effects in monadic calculations that allows to apply all advantages of functional programming to a choice and execution of scenarios of management of various aspects of life activity of buildings and constructions. Besides, the described model can be used together with process of intellectualization of technical and sociotechnical systems for increase of level of independence of decision-making on management of values of parameters of the internal environment of a building, and also for realization of methods of adaptive management, in particular application of various techniques and approaches of an artificial intellect. An important part of the model is a directed acyclic graph, which is an extension of the blockchain with the ability to categorically reduce the cost of transactions taking into account the execution of smart contracts. According to the authors it will allow one to realize new technologies and methods — the distributed register on the basis of the directed acyclic graph, calculation on edge and the hybrid scheme of construction of artificial intellectual systems — and all this together can be used for increase of efficiency of management of intellectual buildings. Actuality of the presented model is based on necessity and importance of translation of processes of management of life cycle of buildings and constructions in paradigm of Industry 4.0 and application for management of methods of an artificial intellect with universal introduction of independent artificial cognitive agents. Model novelty follows from cumulative consideration of the distributed calculations within the limits of the functional approach and hybrid paradigm of construction of artificial intellectual agents for management of intellectual buildings. The work is theoretical. The article will be interesting to scientists and engineers working in the field of automation of technological and industrial processes both within the limits of intellectual buildings, and concerning management of complex technical and social and technical systems as a whole.

Keywords: automation, intellectualization, management, control system, autonomy, adaptability, functional approach, distributed computing, edge computing, intelligent building

Citation: *Computer Research and Modeling*, 2021, vol. 13, no. 3, pp. 557–570 (Russian).

Введение

Авторы предлагают использовать функциональный подход при автоматизации и интеллектуализации технических систем [Душкин, 2018]. Этот подход заключается в рассмотрении исполняемых в технических системах процессов в качестве последовательности вызовов функций с передачей им параметров и возвращением результатов, как это принято в функциональной парадигме программирования [O'Sullivan et al., 2008]. Проблемным моментом является наличие в технологических процессах разнообразных побочных эффектов и недетерминированности вычислений [Lipovača, 2011], что конфликтует с функциональным подходом, однако при использовании специальных техник эти ограничения можно обойти [Душкин, 2019].

Другим подходом, который применяется авторами при интеллектуализации технических систем, является гибридная парадигма построения искусственных интеллектуальных агентов [Душкин, Андронов, 2019], характеризуемая одновременным использованием нисходящих и восходящих методов искусственного интеллекта [Душкин, 2018a]. Это позволяет решить важную задачу — обеспечить наличие возможностей машинного обучения у искусственных интеллектуальных агентов с одновременным применением методов объяснения и интерпретации полученных в результате работы результатов, а также обоснования принятых решений. Это важная характеристика гибридных интеллектуальных агентов, которая позволяет им функционировать в так называемых областях доверия, где принимаемые решения влияют на безопасность инфраструктуры или жизнедеятельности в целом.

Кроме того, с учетом того, что современные системы управления зданиями и сооружениями представляют собой комплексы взаимосвязанных и взаимодействующих инженерных систем жизнеобеспечения, а также разнообразных автоматизированных и информационных систем, каждая из которых может обладать набором периферийного оборудования, имеет смысл рассмотреть возможности по построению на основе этого набора периферийного оборудования распределенной вычислительной сети для осуществления вычислительных процессов на оконечных устройствах. Это связано в том числе с развитием вычислительной техники, что приводит к использованию в оконечном оборудовании элементной базы, на порядки превышающей потребности конкретного прибора. Этот подход сегодня рассматривается в качестве одного из приоритетных в рамках Индустрии 4.0 под наименованием «вычисления на краю» или «туманные вычисления» [Bonomi, 2011].

Новизна представленной работы заключается в использовании нескольких высоких цифровых технологий (распределенные реестры, отдельные методы искусственного интеллекта, туманные вычисления) в рамках применяемого авторами подхода к автоматизации и интеллектуализации процессов управления в зданиях и сооружениях. Актуальность работы основана на том, что развитие методов Индустрии 4.0 и высоких цифровых технологий ставит новые задачи перед наукой и техникой в области управления сложными техническими и социотехническими системами и киберфизическими комплексами [Мохов, Максимова, 2020].

Задачами настоящего исследования являются:

- 1) проведение анализа существующих методов децентрализованного хранения данных;
- 2) разработка модели распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированными системами интеллектуальных зданий.

Статья будет интересна ученым, исследователям и инженерам, работающим в области автоматизации сложных технических систем.

1. Методы децентрализованного хранения данных

На сегодняшний день существует несколько моделей организации вычислительных мощностей для автоматизированных и информационных систем [Таненбаум, Стеен, 2003]. Наиболее общая классификация таких моделей в аспекте соотношения конкретной модели на шкале централизованности выглядит следующим образом:

- 1) централизованная вычислительная модель;
- 2) кластерная вычислительная модель;
- 3) распределенная или децентрализованная вычислительная модель.

Централизованная вычислительная модель основана на использовании мощного сервера, на котором исполняются все критически важные приложения и хранятся все наборы данных для них. Доступ к данным и результатам выполнения приложений осуществляется через терминалы [Fields, 1977]. Исторически это была первая вычислительная модель, так как раньше компьютеры и компьютерное время были очень дорогими, для доступа к вычислительным ресурсам организовывались очереди, а в качестве централизованных серверов выступали мощные мэйнфреймы. Сегодня эта модель часто используется для автоматизации технологических процессов на промышленных производствах или для информатизации управленческих процессов в различных областях хозяйственной деятельности человека. Хотя надо отметить, что от мэйнфреймов во многих случаях был осуществлен переход к современным серверным архитектурам [Shyamkumar, 2018].

Кластерная вычислительная модель основана на использовании нескольких центров осуществления вычислений, представляющих собой более или менее мощные серверные станции, к которым подключаются потребители [Lee et al., 2005]. Серверы могут обмениваться друг с другом заданиями, получать друг от друга информацию для использования в вычислениях. Локально каждый сервер обслуживает свою группу терминалов, но при необходимости делится своими вычислительными мощностями с другими кластерами этой сети. Фактически при помощи кластерной вычислительной модели можно достаточно просто решить задачи балансировки нагрузки на вычислительную сеть, а также задачу устойчивости системы в случае локальных неполадок и даже катастроф [Yang et al., 2020].

Наконец, распределенная вычислительная модель представляет собой одноранговую сеть вычислительных устройств, каждое из которых обладает достаточной мощностью для осуществления большей части вычислений [Li et al., 2001]. Однако при возникновении нагрузки сверх имеющихся мощностей сеть автоматически переконфигурируется и балансируется так, чтобы на запрос было предоставлено необходимое количество вычислительных мощностей, распределенных по многим узлам сети. Каждый узел может представлять собой сервер, персональный компьютер, периферийное устройство с контроллером — суть в том, что все они равнозначны и общаются друг с другом на равных, отправляя запросы на предоставление «помощи» в вычислениях. Эта модель является наиболее гибкой.

Схемы описанных вычислительных моделей представлены на рис. 1.

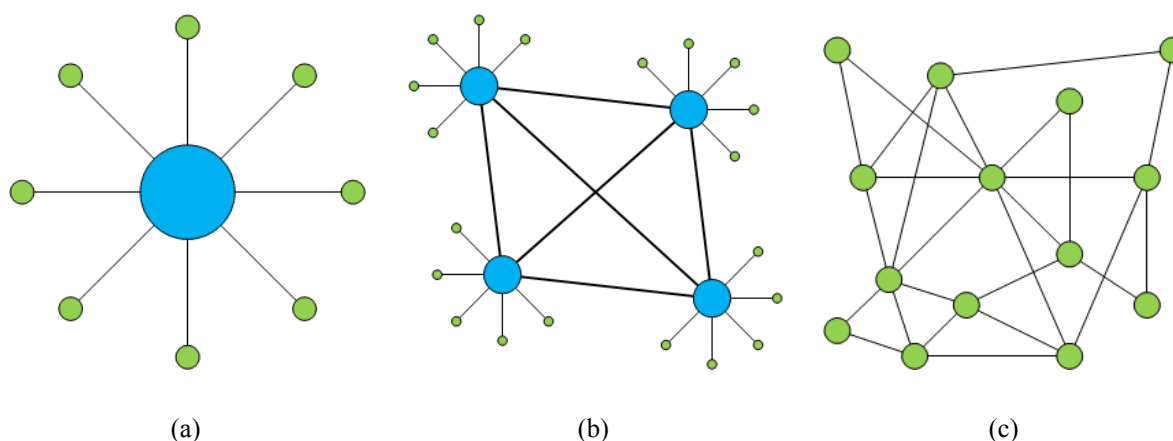


Рис. 1. Модели организации вычислительных мощностей: а) централизованная модель; б) кластерная модель; в) распределенная модель. Голубым цветом на диаграмме обозначено серверное оборудование, зеленым — терминалы и периферийное оборудование

В случае интеллектуального здания резонно рассматривать распределенную вычислительную модель с одним или двумя выделенными узлами, которые играют роль централизованного оборудования для выполнения программного обеспечения центральных систем управления. Тем не менее эти выделенные узлы имеют тот же ранг, что и все прочие вычислительные ресурсы, а выделяются они для обеспечения надежности и отказоустойчивости всей системы управления интеллектуальным зданием.

В работе [Appelquist, 2018] говорится о пользе вычислений на стороне оконечного оборудования. В частности, к положительным сторонам организации вычислений на стороне оконечного оборудования относятся:

- 1) снижение стоимости эксплуатации, поскольку большая часть трафика остается внутри периметра объекта управления — интеллектуального здания;
- 2) сверхнизкое время отклика, так как некоторые технологические процессы можно выполнять без обращения к центральной системе управления, а сразу передать управленческое воздействие от сенсора к исполнительному устройству («рефлекторная» реакция);
- 3) более быстрое время ввода решения в режим эксплуатации по сравнению с облачными решениями промышленного интернета вещей (IIoT);
- 4) повышенный уровень информационной безопасности, так как все данные находятся в периметре объекта управления, который может быть соединен с внешними сетями через демилитаризованную зону и другие средства обеспечения информационной безопасности;
- 5) повышенная устойчивость, поскольку все вычисления осуществляются внутри объекта управления, а потому системе управления не требуется даже быть онлайн;
- 6) сниженные риски процесса разработки, требования к квалификации разработчиков могут быть ниже, так как не требуется умение работать с облачными решениями.

Работа [Браун, Торелл, 2018] также описывает проблемы централизованных центров обработки данных и вводит понятие гибридной среды для организации вычислений, в составе которой находятся центры обработки данных трех различных типов и периферийное оборудование, которое также может выполнять ряд вычислительных задач. В частности, в некоторых случаях возникает потребность поддерживать работу некоторых важных приложений непосредственно на месте. Это позволяет лучше контролировать работу средств автоматизации наряду с соблюдением государственных и отраслевых норм регулирования и требования доступа к информации.

Вычисления на стороне периферийного оборудования (или «вычисления на краю») представляют собой новую вычислительную парадигму и архитектуру построения распределенных вычислительных систем, в рамках которых получение данных о среде и объектах управления, а также осуществление автоматизированных функций, хранение «больших данных» и запуск приложений происходит на фронте соприкосновения системы управления со средой. Вычисления на краю представляют собой логическое завершение методологии интернета вещей в рамках более глобальных архитектур облачных вычислений [Carlini, 2018].

Для организации децентрализованных вычислений и децентрализованного хранения данных на текущий момент используется несколько схем, среди которых можно выделить следующие наиболее интересные:

- 1) блокчейн,
- 2) направленный ациклический граф.

Блокчейн представляет собой распределенный реестр, в котором произвольная информация записывается в виде блоков. В блоки также записываются дополнительные служебные данные — значение сложной хэш-функции для предыдущего блока (хэш, криптографический образ) и в различных вариантах реализации блокчейна какая-либо дополнительная информация, зависящая от назначения и целей конкретной реализации [Lall, 2020]. В частности, для недопущения фальсификации данных блоки криптографически связываются в цепочку посредством включения хеша предыдущего блока в последующий. Присоединение блока к цепочке

происходит только при реализации одного из алгоритмов консенсуса, представляющего собой решение задачи византийских генералов, что гарантирует создание в сети единого источника достоверной информации в среде недоверяющих друг другу участников сети, и таким образом обеспечивается доверие между сторонами в условиях отсутствия централизующей стороны. На текущий момент существует три поколения блокчейнов (третье поколение пока на стадии прототипов), в каждом следующем из которых реализованы дополнительные функции к стандартному базису [Генкин, Михеев, 2017].

Направленный ациклический граф (в отличие от математического термина «ориентированный ациклический граф» в этой работе будет использоваться традиционный технический термин, применяемый при работе с блокчейнами разных типов) является развитием блокчейна и представляет собой сеть блоков информации, в каждом из которых содержится хэш нескольких предыдущих блоков в истории [Lee, 2018]. Эта технология позволяет осуществлять повышенное число транзакций за единицу времени и за меньшую «стоимость» по сравнению с блокчейнами первых двух поколений.

В следующей таблице приводится сравнение блокчейнов различных поколений.

Таблица 1. Качественное сравнение поколений блокчейнов

Характеристика	Поколение 1	Поколение 2	Поколение 3
Типичный представитель	Bitcoin	Ethereum	ИОТА (?) Ethereum 2.0
Алгоритм консенсуса	Доказательство работой	Несколько вариантов	Любой
Масштабируемость	Ограничено	Да	Возможность кластеризации
Наличие смарт-контрактов	Очень ограничено	Да	Да
Скорость	Низкая	Средняя	Высокая
Пропускная способность	Низкая	Средняя	Высокая
Стоимость транзакции	Высокая	Средняя	Низкая

Смарт-контракт — это исполняемый в рамках транзакции код, который обычно пишется на тьюринг-полном языке программирования и позволяет реализовывать практически любую функциональность для обработки данных внутри транзакции [Vigliotti, Jones, 2020]. Фактически это механизм осуществления распределенных вычислений, так как код смарт-контракта выполняется всеми узлами сети, присоединяющих блок, относительно которого достигнут консенсус и который содержит транзакцию со смарт-контрактом. Выполнение смарт-контрактов в децентрализованной сети позволяет полностью реализовать функциональный подход к управлению внутренней средой интеллектуального здания.

В соответствии с работой [Маршалко, 2017] можно определить, является ли целесообразным использование технологии «блокчейн» при организации распределенной системы хранения информации и организации вычислительной среды для управления интеллектуальным зданием. В частности, если воспользоваться представленным в работе алгоритмом принятия решений, то получается следующая последовательность выбора и итоговая рекомендация:

1. Есть ли необходимость в хранении истории изменения состояний системы? — Да, есть.
2. Изменения в состояние могут вносить многие пользователи? — Да, любой сенсор или модуль или подсистема системы управления.
3. Можно ли постоянно использовать доступную доверенную сторону? — Нет, так как этим нарушается принцип децентрализации и повышения надежности.
4. Можно ли идентифицировать всех пользователей системы? — Да, так как у любого устройства и внешнего пользователя есть идентификатор и учетная запись.
5. Есть ли доверие ко всем пользователям? — Нет, так как некоторые оконечные устройства могут работать некорректно.

6. Требуется ли публичная проверка? — Нет, так как обычно интеллектуальным зданием управляет частная организация.

7. Результат: подходит *закрытый частный блокчейн*.

Реализация любого блокчейна или его развития в виде направленного ациклического графа может быть основана на любом из имеющихся в открытом доступе вариантов либо может быть реализована с нуля. При этом необходимо принимать во внимание, что сегодня стандартом де-факто для использования в рамках промышленного интернета вещей является именно направленный ациклический граф. Одними из наиболее развитых вариантов реализации этой технологии являются проекты IOTA и Byteball [Ferraro et al., 2018; Попов, 2017].

Таким образом, особенности построения децентрализованных систем хранения и обработки данных для реализации функциональности инженерных систем интеллектуальных зданий заключаются в следующем [Душкин, 2018]:

- 1) использование децентрализованной модели организации вычислительного процесса;
- 2) имплементация вычислений на стороне окончного оборудования;
- 3) наличие семантической определенности данных [Болдачев, Домрачев, 2020];
- 4) применение блокчейна третьего поколения с развитыми смарт-контрактами для ведения децентрализованной базы данных и осуществления децентрализованных вычислений.

2. Математическое описание модели

Внутренняя среда интеллектуальных зданий характеризуется не только сложными процессами, параметры которых необходимо постоянно отслеживать для непрерывного принятия реактивных и проактивных решений, но и наличием большого числа различного периферийного оборудования инженерных систем. Это оборудование и осуществляет как мониторинг, так и воздействие на среду. Важной особенностью является то, что значительная часть такого оборудования представляет собой специализированные или даже универсальные вычислительные устройства [Душкин, 2019].

В свою очередь, это означает, что все периферийное оборудование, которое используется в инженерных системах и может осуществлять вычисления, можно использовать в рамках парадигмы вычислений на краю (или туманных вычислений — англ. fog computing) [Numhauser et al., 2012] для осуществления объемных и высоконагруженных вычислений, хранения и обработки собираемых данных внутри локальной сети интеллектуального здания и всех периферийных устройств. Это позволит как снизить нагрузку на центральное вычислительное оборудование, осуществляющее общий контроль состояния и проактивное моделирование будущих состояний, так и непосредственно использовать гибридную парадигму построения искусственных интеллектуальных систем [Душкин, Андронов, 2019].

Математическая модель распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания, позволит выстроить целостную систему мониторинга, координации и управления вычислениями в сети интеллектуального здания. Это в свою очередь обеспечит эффективное функционирование как всех инженерных систем в реактивном режиме, так и работу проактивной подсистемы управления как в режиме моделирования, так и в режиме направления управленческих воздействий на реактивный контур.

Теоретико-множественный подход к моделированию распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания, дает необходимый инструментарий для построения операционной модели. Такая модель представляет собой следующий кортеж:

$$C = \langle N, G \rangle,$$

где N — множество узлов распределенной вычислительной сети, в которых могут осуществляться вычисления, G — направленный ациклический граф транзакций, каждая из которых представляет собой смарт-контракт на осуществление вычислений или его выполнение.

Каждый элемент (узел) множества $N = \{node_i\}_{i=1}^Q$ представляет собой вычислительное устройство, находящееся во внутренней сети интеллектуального здания. Все эти устройства связаны друг с другом в сеть, фактически представляющую собой полносвязный граф (при этом некоторые связи в конкретные моменты времени могут отсутствовать, однако взаимодействие в этом случае осуществляется через промежуточные узлы). Взаимодействие в этом графе может осуществляться по любому принципу — как «точка–точка», так и через диспетчеризирующий центр (или набор таких центров в случае кластерной организации центрального узла). Это не имеет принципиального значения, так как главная задача каждого узла — осуществлять часть распределенных вычислений, задачи на которые выдаются в транзакциях графа G .

Фактически узел $node \in N$ представляет собой кортеж следующего вида:

$$node_i = \langle ID_i, addr_i, \{n_{ij}, v_{ij}\}_{j=1}^{P_i} \rangle, \quad i = 1, \dots, Q,$$

где:

ID_i — уникальный идентификатор узла $node_i$ в вычислительной сети C , используемый для наименования соответствующего узла.

$addr_i$ — физический адрес узла $node_i$ в вычислительной сети C , по которому осуществляется адресация узла при осуществлении транзакций.

n_{ij} — наименование j -го свойства узла $node_i$.

v_{ij} — значение j -го свойства узла $node_i$.

P_i — количество свойств узла $node_i$. Фактически множество свойств представляет собой набор аналитических показателей, которые хранятся в модели вычислительной сети C для характеристики каждого узла сети. Набор свойств для узлов может быть индивидуальным для каждого узла. И при этом этот набор предназначается исключительно для справочных свойств.

Q — мощность множества N , фактически это мощность вычислительной сети.

Наибольший интерес представляет собой компонент $G = \langle V, E \rangle$ модели распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания. Это направленный ациклический граф, который является расширением концепции блокчейна для высоконагруженных вычислительных сетей [Ferrago et al., 2018]. По своей структуре этот компонент в качестве узлов $t_k \in V$ содержит описания транзакций, а ребра $(t_{k1}, t_{k2}) \in E$ являются ссылками из подтвержденной сетью транзакции t_{k1} на предыдущую транзакцию t_{k2} (таких транзакций может быть несколько, и в этом отличие направленного ациклического графа от простого блокчейна) для обеспечения защиты сети от атаки двойного расходования [Nakamoto, 2008].

Итак, транзакция $t_k \in V$ представляет собой объект следующего вида:

$$t_k = \langle ID_k^*, ID_{ni}, Got, Mes, Anc, Res, SC \rangle,$$

где:

ID_k^* — уникальный идентификатор транзакции, в том числе включающий в себя отметку времени запуска транзакции в вычислительную сеть. Представляет собой строку символов. По этому идентификатору транзакция может быть найдена в направленном ациклическом графе G .

ID_{ni} — идентификатор того узла $node_i$ в вычислительной сети C , который инициировал транзакцию t_k .

Got — количество токенов, которые «заработаны» транзакцией за исполнение смарт-контрактов транзакций-предшественников.

Mes — произвольное сообщение, которое может быть записано в транзакции для каких-либо целей вычислительной системы. Это сообщение — просто набор битов (или тритов, в зависимости от используемого для организации вычислительной системы блокчейна).

Anc — множество идентификаторов транзакций-предшественников $\{ID_{lk}^*\}_{l=1}^{R_k}$, которые подтверждены текущей транзакцией t_k . Подтверждение транзакций-предшественников означает, что для каждой из них выполнены их смарт-контракты, а результаты их выполнения записаны в множестве Res . Это множество идентификаторов фактически определяет связность графа G , так как именно все пары транзакций $\forall k, l (ID_k^*, ID_{lk}^*) \in E$.

Res — множество вычисленных результатов при подготовке транзакции t_k . Каждый вычисленный результат представляет собой кортеж $\langle ID_{mk}^*, var_m, val_m \rangle$, где ID_{mk}^* — идентификатор транзакции, в которой был запрос на вычисление результата, var_m — наименование переменной из запроса, val_m — значение переменной, вычисленное в результате подготовки текущей транзакции.

SC — смарт-контракт, определяющий множество переменных, значения которых необходимо вычислить узлу, который запускает транзакцию. Все дочерние транзакции от этой должны будут вычислить переменные из этого смарт-контракта. Смарт-контракт представляет собой следующий объект:

$$SC = \langle ID_{SC}, ID_n, Time, Pay, \{ \langle var^*, fun^*, \{ \langle argname, argval \rangle_q \}_{q=1}^T \}_{n=1}^S \rangle,$$

где:

ID_{SC} — идентификатор смарт-контракта.

ID_n — идентификатор устройства, для которого предназначен смарт-контракт. Обычно это поле должно быть пустым, однако в некоторых случаях смарт-контракт может быть предназначен для конкретного узла.

$Time$ — отметка времени, до достижения которой необходимо получить результаты.

Pay — количество токенов, которые создатель смарт-контракта готов «заплатить» за его исполнение.

var^* — наименование переменной, значение которой необходимо вычислить.

fun^* — ссылка на функцию, при помощи которой необходимо вычислить значение переменной с идентификатором var^* .

$argname$ — идентификатор аргумента функции fun^* .

$argval$ — значение аргумента функции fun^* .

S — количество переменных в смарт-контракте.

T — количество аргументов у функции fun^* .

Поскольку смарт-контракт содержит в себе множество запросов значений параметров, оно может быть пустым, что означает, что узел, запускающий в сеть транзакцию, не запрашивает никаких вычислений. В этом случае поля ID_{SC} , $Time$ и Pay остаются пустыми (незаполненными).

Таким образом, одна транзакция в графе G представляет собой объект, структура которого показана на рис. 2.

Поскольку граф G представляет собой направленный ациклический граф, то его общий вид представлен на рис. 3.

На рис. 3 приведен пример того, как выглядит направленный ациклический граф G , который используется для создания распределенной вычислительной среды в интеллектуальном здании. Кружки на схеме обозначают транзакции, а числами в них указан номер поколения соответствующей транзакции. Стрелка от одной транзакции к другой означает, что транзакция t_i вычислила (подтвердила) результаты, которые были запрошены в смарт-контракте транзакции t_{i-1} . Соответственно, вычислительная сеть организована так, что каждая транзакция подтверждена как

минимум одной транзакцией следующего поколения, а также она сама (кроме первоначальной транзакции 0, выделенной на диаграмме цветом) подтверждает одну или несколько транзакций предыдущего поколения. Подтверждение транзакций одного поколения или следующих поколений за текущей не допускается, чем обеспечивается отсутствие в графе G циклов.

Важным нюансом работы представленной модели является то, что узлы распределенной вычислительной сети «платят» за исполнение заказываемых смарт-контрактов. Введение экономической составляющей в модель позволяет использовать экономические методы и инструменты для исследования взаимодействия рациональных агентов, которыми в этом случае представляются узлы распределенной вычислительной сети [Тарасов, 1998]. Этот экономический подход также дает широкое поле для анализа поведения узлов, однако эти вопросы выходят за рамки настоящей работы.

Представленная математическая модель распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания, позволяет организовать децентрализованное вычислительное пространство, в котором периферийные устройства (и центральная система управления тоже) могут «заказывать» друг другу вычисления требуемых параметров для своей работы. Более того, выбор не линейного блокчейна, а направленного ациклического графа в качестве хранилища данных обеспечивает устойчивость и робастность всей распределенной вычислительной системы, что позволит ей работать даже в условиях нештатных ситуаций, в которых может оказаться интеллектуальное здание в процессе своей эксплуатации.

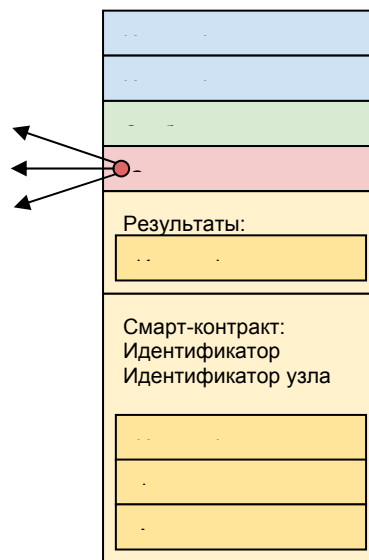


Рис. 2. Структура транзакции t_k в составе графа распределенных вычислений G

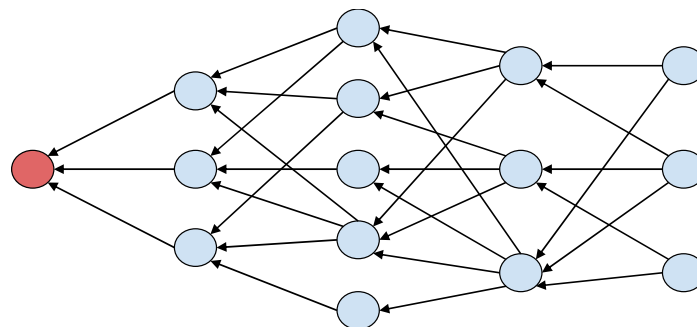


Рис. 3. Пример направленного ациклического графа, используемого для организации распределенной вычислительной среды

Важно отметить, что представленная модель должна реализовываться на нижних уровнях гибридной архитектуры построения искусственного интеллектуального агента в качестве общей системы управления интеллектуальным зданием (в соответствии с положениями работ [Душкин, 2018; Душкин, Андронов, 2019]). Распределенная вычислительная сеть организуется на уровне сенсоров и исполнительных устройств представленной гибридной архитектуры. И, таким образом, незадействованные вычислительные мощности «на краю» включаются в общий контур вычислительной системы, чем достигается решение дополнительной задачи — оптимизации объема используемого оборудования.

Заключение

Представленная в настоящей работе теоретико-множественная модель позволяет целостно охватить процессы, которые могут происходить во внутренней среде интеллектуального здания с точки зрения организации вычислений. На основании этой модели возможна реализация распределенной вычислительной системы, которая может использоваться как для научно-исследовательских работ, так и для непосредственно эксплуатации в зданиях и сооружениях в целях балансировки вычислительной нагрузки на подсистему управления инженерными системами интеллектуального здания.

В работе решены следующие задачи.

1. Проведен анализ существующих методов децентрализованного хранения данных, выделены два класса таких методов — блокчейн и направленный ациклический граф, — которые применимы в описанной в работе модели распределенной вычислительной системы.
2. Разработана теоретико-множественная модель распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированными системами интеллектуальных зданий. Эта модель выявила несколько важных и интересных особенностей, которые необходимо изучить более глубоко.

В работе [Душкин, 2020] представлена база данных для описания режимов функционирования элементов автоматизированных систем интеллектуальных зданий, в которой использованы теоретические результаты настоящей работы. На основе этой базы данных разработана численная имитационная модель, которая позволила проанализировать применение представленной в настоящей работе теоретико-множественной модели в вычислительных сетях различных топологий и архитектур. Результаты моделирования позволяют с определенностью заявить, что описанная модель позволит получить реальную пользу в практических аспектах.

В настоящее время авторами осуществляется практическая реализация описанной модели и связанных с ней методов децентрализованной обработки информации в рамках автоматизированных систем следующих классов:

- 1) системы управления зданиями и сооружениями;
- 2) интеллектуальные транспортные системы;
- 3) системы управления класса «Умный город».

Вместе с тем в целях дальнейшей апробации рассмотренных особенностей построения децентрализованной вычислительной сети для реализации функциональности инженерных систем интеллектуальных зданий будет построена модель децентрализованной вычислительной системы, на которой будут осуществлены оценки результатов распределенных вычислений, обеспечивающих управление инженерными системами интеллектуального здания. Кроме того, необходимо подготовить базу данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания. Эти вопросы являются предметом будущих исследований в рамках настоящей работы.

Также будет проведено исследование экономической составляющей взаимодействия рациональных агентов (узлов, периферийных устройств) в математической модели распределенных вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания, для получения важных теоретических и практических результатов в области теории многоагентных систем.

Список литературы (References)

- Болдачёв А., Домрачев А.* Цифровая поддержка системы стратегического планирования в ЕАЭС: основные подходы и перспективные решения // D-Russia, 03.07.2020. — URL: <https://clck.ru/RoUny> (дата обращения: 07.11.2020).
Boldachev A., Domrachev A. Tsifrovaya podderzhka sistemy strategicheskogo planirovaniya v EAES: osnovnye podkhody i perspektivnye resheniya [Digital support of the strategic planning system in the EAEU: main approaches and promising solutions] // D-Russia, 03.07.2020. — URL: <https://clck.ru/RoUny> (accessed: 07.11.2020) (in Russian).
- Браун К., Торелл В.* Почему облачные вычисления требуют переосмысления отказоустойчивости на периферии // Schneider Electric. — Информационная статья № 256. — 2018.
Brown K., Torell V. Pochemu oblachnye vychisleniya trebuyut pereosmysleniya otkazoustojchivosti na periferii [Why cloud computing requires rethinking fault tolerance on the periphery] // Schneider Electric. — Information article No. 256. — 2018 (in Russian).
- Генкин А., Михеев А.* Блокчейн. Как это работает и что ждет нас завтра. — М.: Альпина Паблшер, 2017. — 592 с.
Genkin A., Mikheev A. Blockchain. Kak eto rabotaet i chto zhdet nas zavtra [Blockchain. How it works and what awaits us tomorrow]. — Moscow: Alpina publisher, 2017. — 592 p. (in Russian).
- Душкин Р. В.* Особенности функционального подхода в управлении внутренней средой интеллектуальных зданий // Прикладная информатика. — 2018. — Т. 13, № 6 (78). — С. 20–31.
Dushkin R. V. Osobennosti funktsional'nogo podkhoda v upravlenii vnutrennej sredoj intellektual'nykh zdaniy [Features of the functional approach to managing the internal environment of intelligent buildings] // Prikladnaya informatika [Applied Informatics]. — 2018. — Vol. 13, No. 6 (78). — P. 20–31 (in Russian).
- Душкин Р. В.* Обзор подходов и методов искусственного интеллекта // Радиоэлектронные технологии. — 2018а. — № 3. — С. 85–89.
Dushkin R. V. Obzor podkhodov i metodov iskusstvennogo intellekta [Overview of artificial intelligence approaches and methods] // Radioelektronnye tekhnologii [Radio-electronic technologies]. — 2018. — No. 3. — P. 85–89 (in Russian).
- Душкин Р. В.* Теоретико-множественная модель функционального подхода к интеллектуализации процессов управления зданиями и сооружениями // Программные продукты и системы. — 2019. — № 2 (32). — С. 306–312. — DOI: 10.15827/0236-235X.126.306-312.
Dushkin R. V. Teoretiko-mnozhestvennaya model' funktsional'nogo podkhoda k intellektualizatsii protsessov upravleniya zdaniyami i sooruzheniyami [Set-theoretic model of a functional approach to the intellectualization of building and structure management processes] // Programmnye produkty i sistemy [Software products and systems]. — 2019. — No. 2. — P. 306–312. — DOI: 10.15827/0236-235X.126.306-312 (in Russian).
- Душкин Р. В.* База данных для описания режимов функционирования элементов автоматизированных систем интеллектуальных зданий // Цифровая экономика. — Май 2020. — URL: https://bit.ly/DB_func
Dushkin R. V. Baza dannykh dlya opisaniya rezhimov funktsionirovaniya elementov avtomatizirovannykh sistem intellektual'nykh zdaniy [Database for describing the modes of functioning of the elements of automated systems of intelligent buildings] // Tsifrovaya ekonomika [Digital Economy]. — May 2020. — URL: https://bit.ly/DB_func (in Russian).
- Душкин Р. В., Андронов М. Г.* Гибридная схема построения искусственных интеллектуальных систем // Кибернетика и программирование. — 2019. — № 4. — С. 51–58. — DOI: 10.25136/2644-5522.2019.4.29809. — URL: http://e-notabene.ru/kp/article_29809.html
Dushkin R. V., Andronov M. G. Gibridnaya skhema postroeniya iskusstvennykh intellektual'nykh sistem [Hybrid scheme for constructing artificial intelligent systems] // Kibernetika i programmirovaniye [Cybernetics and programming]. — 2019. — No. 4. — P. 51–58. — DOI: 10.25136/2644-5522.2019.4.29809. — URL: http://e-notabene.ru/kp/article_29809.html (in Russian).
- Маршалко Г.* Проблемы внедрения технологии блокчейн // Ярославль: Форум «ПРОФ-ИТ», Сентябрь, 2017.
Marshalko G. Problemy vnedreniya tekhnologii blokchejn [Problems of implementing blockchain technology] // Yaroslavl: Forum “PROF-IT”, September, 2017 (in Russian).

- Мохов А. И., Максимова А. В.* Инновационное развитие технологической инфраструктуры и компетенций персонала промышленных предприятий в условиях их цифровой трансформации // Ускорение процессов цифровизации российской промышленности на основе развития и эффективного использования инновационного человеческого капитала: коллективная монография / под ред. М. Я. Веселовского, М. А. Измайловой. — М.: Научный консультант, 2020. — С. 95–125.
- Mokhov A. I., Maksimova A. V.* Innovatsionnoe razvitie tekhnologicheskoy infrastruktury i kompetentsij personala promyshlennykh predpriyatij v usloviyakh ikh tsifrovoy transformatsii [Innovative development of technological infrastructure and competencies of personnel of industrial enterprises in the context of their digital transformation] // Uskorenie protsessov tsifrovizatsii rossijskoj promyshlennosti na osnove razvitiya i effektivnogo ispol'zovaniya innovatsionnogo chelovecheskogo kapitala: kollektivnaya monografiya / pod red. M. Ya. Veselovskogo, M. A. Izmailovoj [Acceleration of digital industry processes based on the effective use of innovative human capital: collective monograph / ed. M. Ya. Veselovsky, M. A. Izmailova]. — Moscow: Scientific consultant, 2020. — P. 95–125 (in Russian).
- Попов В.* Byteball — криптовалюта без блокчейна и ICO // Forklog. — URL: <https://forklog.com/byteball-kriptovalyuta-bez-blokchejna-i-ico/> (дата обращения: 26.04.2020).
- Popov V.* Byteball — kriptovalyuta bez blokchejna i ICO [Byteball — Cryptocurrency without blockchain and ICO] // ForkLog. — URL: <https://forklog.com/byteball-kriptovalyuta-bez-blokchejna-i-ico/> (accessed: 26.04.2020) (in Russian).
- Таненбаум Э., Стеен М.* Распределенные системы. Принципы и парадигмы. — СПб.: Питер, 2003. — 877 с.
- Tanenbaum E., Steen M.* Raspredeleennye sistemy. Printsipy i paradigm [Distributed Systems. Principles and Paradigms]. — St. Petersburg: Peter, 2003. — 877 p. (in Russian).
- Тарасов В. Б.* Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте // Новости искусственного интеллекта: сборник. — 1998. — № 2. — С. 5–63.
- Tarasov V. B.* Agenty, mnogoagentnye sistemy, virtual'nye soobshchestva: strategicheskoe napravlenie v informatike i iskusstvennom intellekte [Agents, multi-agent systems, virtual communities: a strategic direction in computer science and artificial intelligence] // Novosti iskusstvennogo intellekta: sbornik [News of artificial intelligence]. — 1998. — No. 2. — P. 5–63 (in Russian).
- Appelquist G.* How to Build an Industrial IoT Project Without the Cloud // IIoT World. — 23.10.2018.
- Bonomi F.* Cloud and Fog Computing: Trade-offs and Applications // EON-2011 Workshop, International Symposium on Computer Architecture (ISCA 2011). — San Jose, CA, USA.
- Carlini S.* The Drivers and Benefits of Edge Computing // Schneider Electric — Data Center Science Center. — White Paper 226. — 2018.
- Ferraro P., King C., Shorten R.* IOTA-based Directed Acyclic Graphs without Orphans // Preprint, December 2018. — Lab: Robert Shorten's Lab.
- Fields C.* Terminal access technology of the 1990's // Professional Communication, IEEE Transactions on. PC-20. 2-6. — DOI: 10.1109/TPC.1977.6594165
- Lall M. M.* Blockchain // Wiley Online Library. — P. 1–6. — DOI: 10.1002/9781405165518.wbeos1559
- Lee J., Zhang H., Guha R. K.* Virtual Cluster Computing Architecture // Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 2005, Las Vegas, Nevada, USA, June 27–30, 2005. Vol. 1.
- Lee S.* Explaining Directed Acyclic Graph (DAG), The Real Blockchain 3.0 // Forbes, January 22, 2018.
- Li Ch., Lu Zh., Li L.* A distributed computing model and its application // Proceedings of 2001 International Conference on Computer Networks and Mobile Computing. — 2001. — P. 341–346. — DOI: 10.1109/ICCNMC.2001.962617
- Lipovača M.* Learn You a Haskell for Great Good! — San Francisco: No Starch Press, 2011.
- Nakamoto S.* Bitcoin: A Peer-to-Peer Electronic Cash System. — 2008. — 9 p.
- Numhauser J., Bar-Magen F. J., Casals F., Numhauser P. (eds.)* Fog Computing — Introduction to a new Cloud evolution. — Proceedings from the CIES III Congress, January 2012.

- O'Sullivan B., Goerzen J., Stewart D.* Real World Haskell. — O'Reilly, 2008. — 710 p.
- Shyamkumar I.* Server Architecture for the New Age Datacenter // ACM SIGARCH Computer Architecture News, August 2018.
- Vigliotti M., Jones H.* Smart Contracts. In book: The Executive Guide to Blockchain, Using Smart Contracts and Digital Currencies in your Business. — February, 2020. — DOI: 10.1007/978-3-030-21107-3_8
- Yang Y., Luo X., Chu X., Zhou M.-T.* Fog Computing Architecture and Technologies // Fog-Enabled Intelligent IoT Systems. — DOI: 10.1007/978-3-030-23185-9_2