

Алгебра конечных автоматов как математическая модель цифрового двойника умного производства

Д. А. Гапанович*, В. А. Сухомлин

ФГБОУ ВО «Московский государственный университет имени М. В. Ломоносова», г. Москва, Российская Федерация

Адрес: 119991, Российская Федерация, г. Москва, ГСП-1, Ленинские горы, д. 1

* dim.gapanovich@gmail.com

Аннотация

Статья посвящена разработке алгебры конечных автоматов специального вида DTA (Digital Twin Algebra), предназначенной для математического моделирования поведения цифровых двойников производства. Основное внимание уделяется изучению операций композиции детерминированных конечных автоматов, позволяющих строить модели составных активов из более простых. В алгебру DTA включены как общеизвестные операции, например, операция прямого произведения автоматов и операция суперпозиции автоматов, так и вновь введенные операции, включая: операции группирования конечных автоматов в параллельный блок, группирования иерархической структуры, системной композиции, ассоциации между автоматами, рекурсивного вызова автомата, асинхронного вызова автомата. Предложенный математический аппарат может использоваться для разработки формальных моделей физических активов производства, их цифровых двойников, а также моделей поведения киберфизических систем, включающих как физические, так и виртуальные активы.

Ключевые слова: цифровой двойник, кибер-физическая система, ЦПС, алгебра, автомат, алгебра автоматов, математическое моделирование, Индустрия 4.0

Авторы заявляют об отсутствии конфликта интересов.

Для цитирования: Гапанович Д. А., Сухомлин В. А. Алгебра конечных автоматов как математическая модель цифрового двойника умного производства // Современные информационные технологии и ИТ-образование. 2022. Т. 18, № 2. С. 353-366. doi: <https://doi.org/10.25559/SITITO.18.202202.353-366>

© Гапанович Д. А., Сухомлин В. А., 2022



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Algebra of Finite Automata as a Mathematical Model of the Digital Twin of Smart Production

D. A. Gapanovich*, V. A. Sukhomlin

Lomonosov Moscow State University, Moscow, Russian Federation

Address: 1 Leninskie gory, Moscow 119991, GSP-1, Russian Federation

* dim.gapanovich@gmail.com

Abstract

The article is devoted to the development of the finite automata algebra of a special type DTA (Digital Twin Algebra), designed for mathematical modeling of production digital twins. The main attention is paid to the study of the composition operations of deterministic finite automata, which allow building models of composite assets from simpler ones. The DTA includes both well-known operations, for example, the operation of direct product of automata and the operation of superposition of automata, and newly introduced operations, including: operations of grouping finite automata into a parallel block, grouping of hierarchical structure, system composition, association between automata, recursive invocation of the automaton, asynchronous call of the automaton. The proposed mathematical apparatus can be used to develop formal models of behavior of production physical assets, their digital twins, as well as models of cyber-physical systems behavior, including both physical and virtual assets.

Keywords: digital twin, DT, cyber-physical system, CPS, algebra, automata, algebra of automata, mathematical modeling, Industry 4.0

The authors declare no conflict of interest.

For citation: Gapanovich D.A., Sukhomlin V.A. Algebra of Finite Automata as a Mathematical Model of the Digital Twin of Smart Production. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2022; 18(2):353-366. doi: <https://doi.org/10.25559/SITITO.18.202202.353-366>



Введение. Концепция цифровых двойников

В эпоху Индустрии 4.0, имеющей непосредственное отношение к текущему времени, характерной тенденцией в экономике становится масштабный переход промышленных предприятий к технологиям интеллектуального производства или к так называемому умному производству. Эффективным инструментом осуществления таких преобразований является концепция цифровых двойников (Digital Twins – DT)¹, направленная на достижение высокой степени сходства между целевыми активами (физическими, природными, социальными) и их цифровыми или виртуальными образами/двойниками, а также на обеспечение зеркальной информационной связи в реальном времени между активами и двойниками [1].

Создание цифрового двойника (DT) для актива превращает физический актив (устройство, систему, предприятие) в целостную киберфизическую систему (Cyber-Physical System – CPS), предоставляя возможность управления такой системой, как на физическом, так и виртуальном уровнях. Применение DT подразумевает высокоточное моделирование свойств (статистики) и процессов функционирования (динамики, операционной деятельности) активов с использованием интеллектуальных методов принятия решений для оптимизации их функционирования и, в частности, расширения производства².

В серии стандартов ПНСТ «Умное производство. Двойники цифровые производства»³ DT производства рассматривается как детальное моделирование конфигураций физических сущностей и динамическое моделирование изменений продукции, производственной деятельности и ресурсов в процессе производства. DT производства представляется цифровой моделью, которая постоянно обновляется и изменяется по мере изменения физического аналога с целью синхронного представления состояния, условий работы, конфигурации продукта и состояния ресурсов. Представление DT производства предусматривает постоянное взаимодействие с реальными производственными элементами посредством обмена эксплуатационными

данными и данными об условиях эксплуатации [2]–[7].

Под представлением цифрового двойника (digital twin representation) понимается отображение DT в реализационную программно-аппаратную среду в виде целостной системы, отражающей в полной мере для поставленных целей свойства физических элементов производственного процесса, таких, как, персонал, продукты производства, активы и процессы⁴.

Разработка представления DT нетривиального актива, тем более такого каким является современное производство, представляет собой сложную научно-техническую задачу, а сами DT – сложные многофункциональные программно-аппаратные системы, с помощью которых моделируется жизненный цикл актива/производства и осуществляется взаимодействие с приложениями, поддерживающими принятие решений по управлению им. Ввиду сложности двойников важную роль играют методы их архитектурного проектирования. В работе авторов⁵ рассматривался анализ различных архитектурных решений, лежащих в основе создания таких систем, в частности, отмечалось, что при создании DT производства в качестве базового архитектурного решения может быть взята архитектура цифрового двойника, рассмотренная в ПНСТ 432-2020⁶ и показанная на рисунке 1.

Рассмотренная выше структура DT производства показывает конструктивную сложность таких DT, представляющих собой множества двойников элементов/сущностей производства и функциональных сущностей, объединенных в целостную систему виртуального представления физического производства. Концепция DT имеет всеобъемлющий характер и применяется к различным видам активов на всех уровнях их конструктивной иерархии. В результате чего, DT составных активов, состоящих из более простых компонент, конструируются в виде композиций из DT составляющих их компонент.

¹ Прохоров А., Лысачев М. Цифровой двойник. Анализ, тренды, мировой опыт / Под ред. А. И. Боровкова. М.: ООО «АльянсПринт», 2020. 401 с. URL: https://data.atomcloud.ru/books/digital_twin_book.pdf (дата обращения: 26.04.2022).

² Онтологии кибер-физических систем национального цифрового двойника Великобритании и BIM на примерах умных городов, железных дорог и других проектов / В. П. Куприяновский, Д. Е. Намиот, А. А. Климов [и др.] // International Journal of Open Information Technologies. 2021. Т. 9, № 3. С. 91-130. URL: <https://www.elibrary.ru/item.asp?id=44853588> (дата обращения: 26.04.2022).

³ ПНСТ 429-2020 Умное производство. Двойники цифровые производства. Часть 1. Общие положения: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 38-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020; ПНСТ 430-2020 Умное производство. Двойники цифровые производства. Часть 2. Типовая архитектура: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 39-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020; ПНСТ 431-2020 Умное производство. Двойники цифровые производства. Часть 3. Цифровое представление физических производственных элементов: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 40-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020; ПНСТ 432-2020 Умное производство. Двойники цифровые производства. Часть 4. Обмен информацией: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 41-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020.

⁴ Царев М. В., Андреев Ю. С. Цифровые двойники в промышленности: история развития, классификация, технологии, сценарии использования // Известия высших учебных заведений. Приборостроение. 2021. Т. 64, № 7. С. 517-531. doi: <https://doi.org/10.17586/0021-3454-2021-64-7-517-531>

⁵ Анализ подходов архитектурного проектирования цифровых двойников / Д. А. Гапанович, В. А. Тарасова, В. А. Сухомлин, В. П. Куприяновский // International Journal of Open Information Technologies. 2022. Т. 10, № 4. С. 71-83. URL: <https://www.elibrary.ru/item.asp?id=48247621> (дата обращения: 26.04.2022).

⁶ ПНСТ 432-2020 Умное производство. Двойники цифровые производства. Часть 4. Обмен информацией: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 41-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020.



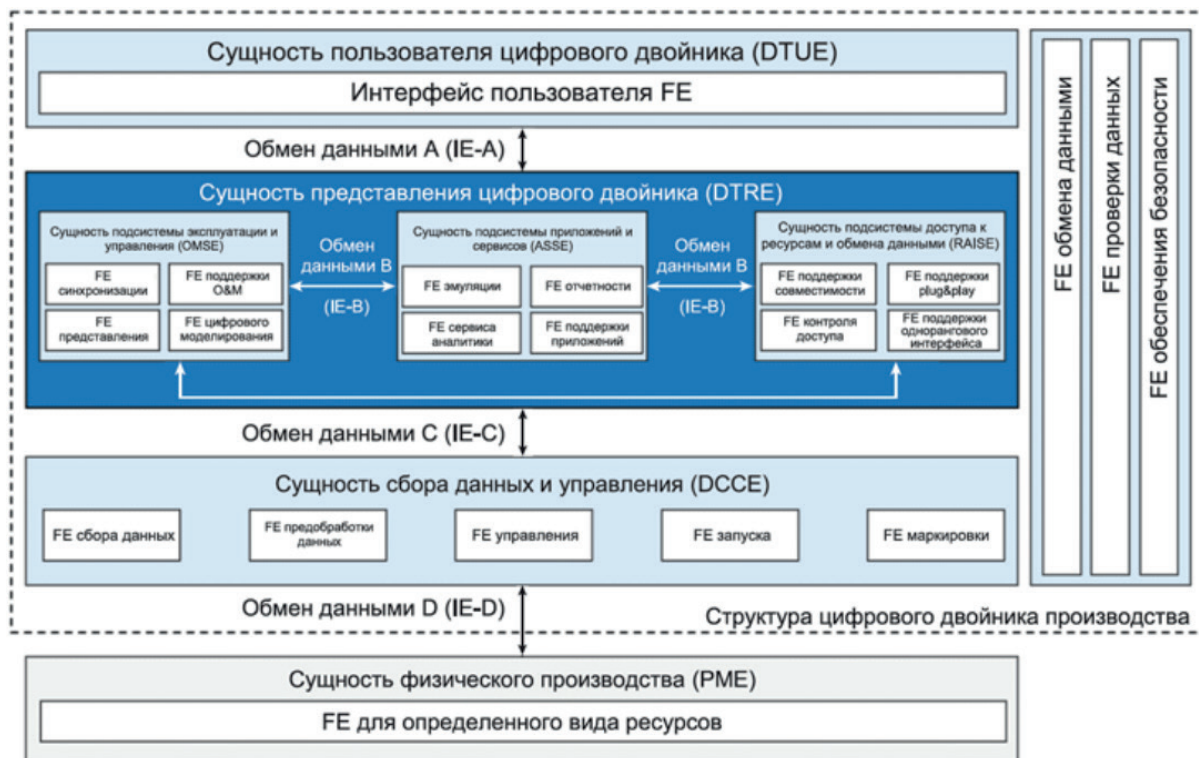
Р и с. 1. Структура цифрового двойника производства⁷

Fig. 1. The structure of the digital twin of production

В широко используемых на практике спецификациях Asset Administration Shell (AAS)⁸ [8], ориентированных на задачи Индустрии 4.0, предложена типовая архитектура для DT производственного актива. Эта архитектура включает статическую часть модели актива, динамическую часть модели, а также, возможно, приложения, расширяющие функциональность двойника. При этом рассматриваются следующие варианты конструирования двойников:

- DT отдельного актива, когда тот представляет собой единый объект, который поддерживает свое приложение в данном контексте использования без необходимости дальнейшей разработки на компоненты;
- DT составного актива как комбинации двойников, соответствующих отдельным компонентам или частям актива, включая следующие основные способы их композиции⁹:

1) Иерархическая структура: так же, как реальные аналоги компонентных DT, например, микросхем, могут быть собраны в двойник устройства, а набор DT устройств может быть со-

бран в цифровой двойник компьютерной системы, а из набора двойников систем, может быть собран двойник производственной линии и так далее, вплоть до двойника предприятия или конгломерата предприятий.

2) Ассоциативные связи: между двойниками могут быть различные ассоциации, как и их в реальных аналогах. Ассоциативные отношения используются не только для описания связей между двойниками вещей и способов взаимодействия между ними, но и для аналитических вычислений.

3) Одноранговая связь: одноранговые отношения наблюдаются в группе оборудования одного и того же или аналогичного типа, которое выполняет одинаковые или аналогичные функции. В этих случаях общий эффект всего оборудования представляет собой простую сумму эффекта, производимого каждой единицей оборудования.

Рассмотренные композиции и ассоциации двойников¹⁰ иллюстрируются на рисунке 2.

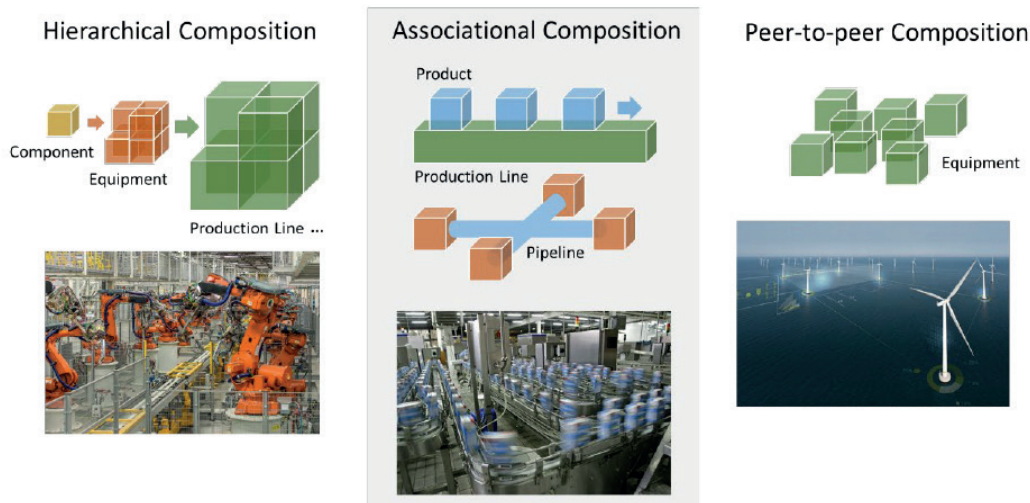
⁷ Там же.

⁸ Boss B., Malakuti S., Lin S.-W., Usländer T., Clauer E., Hoffmeister M., Stojanovic L. Digital Twin and Asset Administration Shell Concepts and Application in the Industrial Internet and Industrie 4.0. An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper. Industrial Internet Consortium: Milford, MA, USA, 2020. URL: <https://www.iiconsortium.org/pdf/Digital-Twin-and-Asset-Administration-Shell-Concepts-and-Application-Joint-Whitepaper.pdf> (дата обращения: 26.04.2022).

⁹ Malakuti S., van Schalkwyk P., Boss B., Sastry C. R., Runkana V., Lin S.-W., Rix S., Green G., Baechle K., Nath S. V. Digital Twins for Industrial Applications: Definition, Business Values, Design Aspects, Standards and Use Cases. Industrial Internet Consortium White Paper. Version 1.0. 2020-02-18. Industrial Internet Consortium: Milford, MA, USA, 2020. P. 4. URL: <https://www.iiconsortium.org/digital-twins-for-industrial-applications> (дата обращения: 26.04.2022).

¹⁰ ПНСТ 432-2020 Умное производство. Двойники цифровые производства. Часть 4. Обмен информацией: предварительный национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 7 августа 2020 г. № 41-пнст: введен впервые: дата введения 2021-01-01 / подготовлен АО «ВНИИС», АО «РВК». М.: Стандартинформ, 2020.





Р и с. 2. Отношение между цифровыми двойниками в возможных композициях¹¹
F i g. 2. The relationship between digital twins in possible compositions

Возвращаясь к рассмотренной на рисунке 1 структуре DT производства, заметим, что вместе с уровнем физических сущностей, т.е. физических активов или вещей, представление двойника производства образуют целостную киберфизическую систему производства или CPS [9]-[10].

Выделим в этой киберфизической системе следующие ее основные компоненты/подсистемы:

- P – множество сущностей физического производства (физические вещи),
- C – множество цифровых близнецов-представителей для каждой физической вещи (кибервещь),
- ST – структурную композицию подсистем для формирования сложных вещей,
- E – множество событий, возникающих на протяжении жизненного цикла производства, определяющих динамику функционирования DT производства,
- R – множество отношений между вещами,
- FE – множество функциональных элементов (FE), в том числе реализующих реакцию на события, происходящие в процессе работы производства,
- AV – множество приложений и веб-сервисов DT производства,
- D – база данных DT производства,
- L – потоки данных в двойнике,
- MIC – механизм зеркальной информационной связи между активом и двойником (mirror information communication).

Таким образом, конфигурацией CPS производства назовем сле-

дующую конструкцию:

$$CPS = \langle P, C, ST, E, R, FE, D, AV, L, MIC \rangle \quad (1)$$

При этом следует иметь в виду, что множества P и C состоят из физических и виртуальных сущностей, являющихся как простыми вещами, так и составными, т.е. композициями из более простых.

Для решения задач исследования и разработки DT в качестве базового математического аппарата будем использовать алгебру конечных автоматов с набором операций, обеспечивающих возможность представлять модели как простых, так и составных физических и виртуальных вещей.

Алгебра конечных автоматов для описания цифровых двойников

Системы управления производством, как правило, представляют собой дискретные системы, функционирование которых основано на событийном подходе. В частности, характерным примером таких систем может служить система дистанционного контроля (СДК) промышленной безопасности в горнодобывающей отрасли¹².

Для моделирования динамических дискретных систем, функционирующих на основе событийного подхода, наиболее адекватным математическим аппаратом является теория конечных автоматов¹³ [11].

Под конечным автоматом M (Мили) понимается следующая

¹¹ Malakuti S., van Schalkwyk P., Boss B., Sastry C. R., Runkana V., Lin S.-W., Rix S., Green G., Baechle K., Nath S. V. Digital Twins for Industrial Applications: Definition, Business Values, Design Aspects, Standards and Use Cases. Industrial Internet Consortium White Paper. Version 1.0. 2020-02-18. Industrial Internet Consortium: Milford, MA, USA, 2020. P. 4. URL: <https://www.iiconsortium.org/digital-twins-for-industrial-applications> (дата обращения: 26.04.2022).

¹² ГОСТ Р 58494-2019 Оборудование горно-шахтное. Многофункциональные системы безопасности угольных шахт. Система дистанционного контроля опасных производственных объектов: национальный стандарт РФ: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 22 августа 2019 г. № 522-ст: введен впервые: дата введения 2021-01-01 / подготовлен АО «СУЭК», ЗАО РКСС, ООО НИИПМС, ООО НПФ «Гранч», ООО «НПП «Шахтпожсервис», ООО «Ингортех». М.: Стандартиформ, 2019.

¹³ Кудрявцев В. Б., Алёшин С. В., Подколзин А. С. Введение в теорию автоматов. М.: Наука, 1985. 320 с.; Кудрявцев В. Б., Гасанов Э. Э., Подколзин А. С., Теория интеллектуальных систем: в 4 кн. Книга четвертая. Теория автоматов. М.: Издательские решения, 2018. 302 с.



конфигурация:

$$M = \langle A, B, Q, \delta, \lambda \rangle \quad (2)$$

- A – входной алфавит – непустое конечное множество входных символов (входных событий автомата, например, сигналов от сенсоров),
- B – выходной алфавит – непустое конечное множество выходных символов (например, продуцируемых автоматом событий или сигналов),
- Q – множество состояний автомата – непустое конечное множество символов состояний автомата (его память),
- δ – функции переходов автомата – однозначное отображение вида: $Q \times A \rightarrow Q$
- λ – функцией выходов автомата – отображение вида: $Q \times A \rightarrow B$

Часто в автоматную конфигурацию добавляется начальное состояние автомата q_0 . Тогда автомат называется инициальным, а его конфигурация имеет следующий вид:

$$M = \langle A, B, Q, \delta, \lambda, q_0 \rangle \quad (3)$$

С помощью конечных автоматов будем описывать как физические сущности из множества P , так и их виртуальные образы или кибервещи (ДТ) из множества C .

В случае физических активов компоненты конфигурации автомата будем индексировать буквой « P », а в случае кибервещей – буквой « C ».

Тогда для физической вещи $p \in P$ конфигурация инициального автомата будет иметь следующий вид:

$$p = \langle A_p, B_p, Q_p, \delta_p, \lambda_p, q_{0p} \rangle \quad (4)$$

а для виртуальной вещи $c \in C$:

$$c = \langle A_c, B_c, Q_c, \delta_c, \lambda_c, q_{0c} \rangle \quad (5)$$

Далее будем предполагать, что конечные автоматы, с которыми мы будем иметь дело, являются полными и детерминированными. Детерминированные автоматы – это автоматы, для которых состояние $\delta(q, a)$ однозначно определяется парой $(q, a) \in Q \times A$, т.е. функция переходов $\delta(q, a) = q'$ действительно является функцией.

Говорят, что автомат полный, когда функция переходов предполагается полностью определенной, т.е. значение функции $\delta(q, a)$ должно существовать для каждой пары $(q, a) \in Q \times A$.

Конечные автоматы имеют наглядную форму представления в виде помеченных графов, называемых диаграммами Мура. Такими графами или диаграммами Мура, представляющими конечные автоматы, будем называть абстрактные сущности, состоящие из конечного множества вершин-состояний автомата Q и конечного множества направленных дуг E , соединяющих две вершины (начальную и конечную). В таких графах будут допускаться дуги, имеющие одинаковые начала и концы, называемые также параллельными. Разметка графов осуществляется с помощью отображения $E \rightarrow A \times B$, где каждой дуге ставится в соответствие пара – входной символ $a \in A$ и выходной символ $b \in B$. Таким образом, переход из состояния q в состояние q' , вызванный входной буквой a , изображается дугой с началом q , концом q' и меткой « a », при этом при переходе по дуге осуществляется вывод выходного символа $b = \lambda(q, a)$.

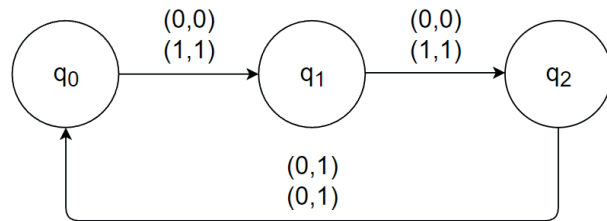
Для обеспечения лучшей наглядности графов помеченные параллельные дуги будут заменять одной дугой, помеченной списком пометок заменяемых дуг.

Рассмотрим автомат $M = \langle A, B, Q, \delta, \lambda, q_0 \rangle$, у которого:

- $A = \{0, 1\}$,
- $B = \{0, 1\}$,
- $Q = \{q_0, q_1, q_2\}$,
- $\{\delta(q_0, 0) = q_1, \delta(q_0, 1) = q_1, \delta(q_1, 0) = q_2, \delta(q_1, 1) = q_2, \delta(q_2, 0) = q_0, \delta(q_2, 1) = q_0\}$,
- $\{\lambda(q_0, 0) = 0, \lambda(q_0, 1) = 1, \lambda(q_1, 0) = 0, \lambda(q_1, 1) = 1, \lambda(q_2, 0) = 1, \lambda(q_2, 1) = 0\}$.

Нетрудно видеть, что работа этого автомата состоит в том, чтобы инвертировать каждый третий символ входного потока.

На рисунке 3 представлен граф этого автомата, в котором параллельные дуги заменены единственными дугами, помеченных списком меток, составленным из меток параллельных дуг.



Р и с. 3. Диаграмма Мура конечного автомата, инвертирующего каждый третий символ входного потока

Fig. 3. Moore diagram of a finite automaton inverting every third character of the input stream

При использовании теории автоматов для моделирования поведения сложных систем важное значение отводится разработке операций для построения композиций автоматов, позволяющих создавать более сложные автоматы из более простых. Ниже рассмотрим такие операции композиции автоматов, начиная с классической теории [11]. Целью изучения операций композиции автоматов является определение набора операций над конечными автоматами, т.е. алгебры конечных автоматов, как основы для математического моделирования поведения ДТ и создаваемых с их помощью киберфизических систем. Такую алгебру в дальнейшем будем называть алгеброй ДТА (Digital Twin Algebra).

Операция прямого произведения автоматов (\times) [11]

Прямым произведением множества автоматов $M_i = \langle A_i, B_i, Q_i, \delta_i, \lambda_i \rangle$, где $i \in \{1, 2, \dots, n\}$, называется автомат $M = \langle A, B, Q, \delta, \lambda \rangle$, множество состояний, а также множества входных и выходных сигналов которого представляют собой прямые произведения соответствующих множеств автоматов:

$$A: (A_1, A_2, \dots, A_n) = (A_1 \times A_2 \times \dots \times A_n),$$

$$B: (B_1, B_2, \dots, B_n) = (B_1 \times B_2 \times \dots \times B_n)$$

$$Q: (Q_1, Q_2, \dots, Q_n) = (Q_1 \times Q_2 \times \dots \times Q_n)$$

Функции переходов $q' = \delta(q, a)$ и выходов $b = \lambda(q, a)$ автомата M задаются соотношениями:

$$q'_i = \delta(q_i, a_i), b_i = \lambda(q_i, a_i), i \in \{1 - n\}, \text{ где}$$

$$q' = (q'_1, q'_2, \dots, q'_n), a = (a_1, a_2, \dots, a_n), b = (b_1, b_2, \dots, b_n),$$

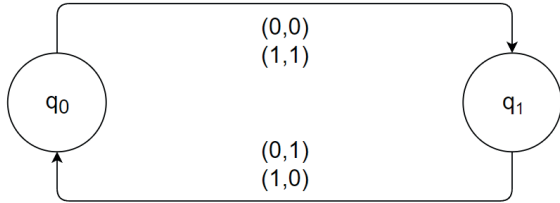
$$a_i \in A_i, b_i \in B_i, q_i \in Q_i, q'_i \in Q_i$$

Рассмотрим пример применения операции прямого произведения двух автоматов M_1 и M_2 .

Первый автомат, M_1 , принимая на вход последовательность из 0 и 1, инвертирует каждую вторую цифру на противополож-



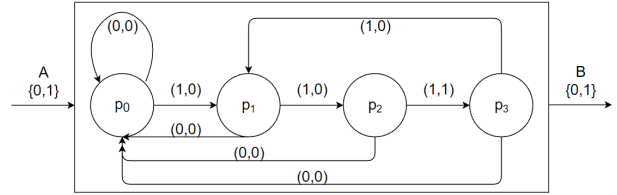
ную. Его конфигурация имеет следующий вид:
 $M_1 = \langle A, B, Q, \delta, \lambda \rangle = \langle \{0, 1\}, \{0, 1\}, \{q_0, q_1\}, \{ \delta(q_0, 0)=q_0, \delta(q_0, 1)=q_1, \delta(q_1, 0)=q_0, \delta(q_1, 1)=q_1 \}, \{ \lambda(q_0, 0)=0, \lambda(q_0, 1)=1, \lambda(q_1, 0)=1, \lambda(q_1, 1)=0 \} \rangle$.
 Диаграмма Мура для этого автомата иллюстрируется на рисунке 4.



Р и с. 4. Диаграмма Мура для автомата M_1
 F i g. 4. Moore diagram for automaton M_1

Второй автомат, M_2 , также принимает на вход последовательность из 0 и 1, распознавая три подряд идущих 1. В случае распознавания слова 111 на выходе он выдает символ 1, иначе 0.
 $M_2 = \langle A, B, Q, \delta, \lambda \rangle = \langle \{0, 1\}, \{0, 1\}, \{p_0, p_1, p_2, p_3\}, \{ \delta(p_0, 0)=p_0, \delta(p_0, 1)=p_1, \delta(p_1, 0)=p_0, \delta(p_1, 1)=p_2, \delta(p_2, 0)=p_0, \delta(p_2, 1)=p_3, \delta(p_3, 0)=p_0, \delta(p_3, 1)=p_3 \}, \{ \lambda(q_0, 0)=0, \lambda(q_0, 1)=0, \lambda(q_1, 0)=0, \lambda(q_1, 1)=0, \lambda(p_0, 0)=0, \lambda(p_0, 1)=1, \lambda(p_1, 0)=0, \lambda(p_1, 1)=0 \} \rangle$.

Диаграмма Мура для автомата M_2 иллюстрируется на рисунке 5.



Р и с. 5. Диаграмма Мура для автомата M_2
 F i g. 5. Moore diagram for automaton M_2

Результатом применения операции прямого произведения для рассмотренного выше примера будет автомат M следующего вида:

$M = \langle A, B, Q, \delta, \lambda, q_0 \rangle = M_1 \times M_2 = \langle A_1 \times A_2, B_1 \times B_2, Q_1 \times Q_2, \delta, \lambda, q_0 \times p_0 \rangle$, где

- $A = (A_1 \times A_2) = \{ \{0, 1\}, \{0, 1\} \}$
- $B = (B_1 \times B_2) = \{ \{0, 1\}, \{0, 1\} \}$
- $Q = (Q_1 \times Q_2) = \{ (q_0, p_0), (q_0, p_1), (q_0, p_2), (q_0, p_3), (q_1, p_0), (q_1, p_1), (q_1, p_2), (q_1, p_3) \}$

Функции переходов $q' = \delta(q, a)$ автомата M задаются соотношениями, приведенными в Таблице 1.

Т а б л и ц а 1. Функция переходов для прямого произведения автоматов M_1 и M_2
 T a b l e 1. The transition function for the direct product of the automata M_1 and M_2

| | | | |
|--|--|--|--|
| $\delta((q_0, p_0), (0, 0)) = (q_1, p_0)$, $\delta((q_0, p_0), (1, 0)) = (q_1, p_0)$, $\delta((q_0, p_0), (0, 1)) = (q_1, p_1)$, $\delta((q_0, p_0), (1, 1)) = (q_1, p_1)$, | $\delta((q_0, p_1), (0, 0)) = (q_1, p_0)$, $\delta((q_0, p_1), (1, 0)) = (q_1, p_0)$, $\delta((q_0, p_1), (0, 1)) = (q_1, p_1)$, $\delta((q_0, p_1), (1, 1)) = (q_1, p_1)$, | $\delta((q_0, p_2), (0, 0)) = (q_1, p_0)$, $\delta((q_0, p_2), (1, 0)) = (q_1, p_0)$, $\delta((q_0, p_2), (0, 1)) = (q_1, p_3)$, $\delta((q_0, p_2), (1, 1)) = (q_1, p_3)$, | $\delta((q_0, p_3), (0, 0)) = (q_1, p_0)$, $\delta((q_0, p_3), (1, 0)) = (q_1, p_0)$, $\delta((q_0, p_3), (0, 1)) = (q_1, p_1)$, $\delta((q_0, p_3), (1, 1)) = (q_1, p_1)$ |
| $\delta((q_1, p_0), (0, 0)) = (q_1, p_0)$, $\delta((q_1, p_0), (1, 0)) = (q_1, p_0)$, $\delta((q_1, p_0), (0, 1)) = (q_1, p_1)$, $\delta((q_1, p_0), (1, 1)) = (q_1, p_1)$, | $\delta((q_1, p_1), (0, 0)) = (q_1, p_0)$, $\delta((q_1, p_1), (1, 0)) = (q_1, p_0)$, $\delta((q_1, p_1), (0, 1)) = (q_1, p_2)$, $\delta((q_1, p_1), (1, 1)) = (q_1, p_2)$, | $\delta((q_1, p_2), (0, 0)) = (q_1, p_0)$, $\delta((q_1, p_2), (1, 0)) = (q_1, p_0)$, $\delta((q_1, p_2), (0, 1)) = (q_1, p_3)$, $\delta((q_1, p_2), (1, 1)) = (q_1, p_3)$, | $\delta((q_1, p_3), (0, 0)) = (q_1, p_0)$, $\delta((q_1, p_3), (1, 0)) = (q_1, p_0)$, $\delta((q_1, p_3), (0, 1)) = (q_1, p_3)$, $\delta((q_1, p_3), (1, 1)) = (q_1, p_3)$ |

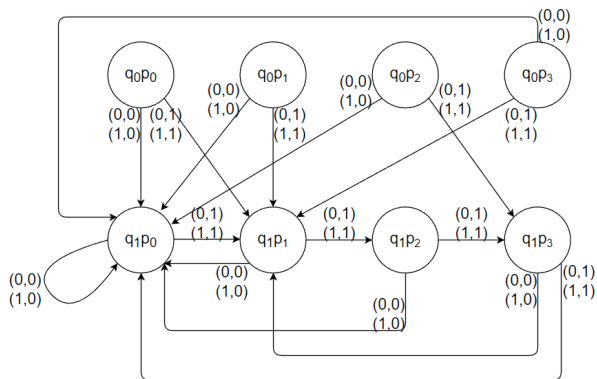
Функция выходов $b = \lambda(q, a)$ для автомата M строится аналогичным образом, соотношениями, приведенными в Таблице 2.

Т а б л и ц а 2. Функция выходов для прямого произведения автоматов M_1 и M_2
 T a b l e 2. The output function for direct product of automata M_1 and M_2

| | | | |
|--|--|--|--|
| $\lambda((q_0, p_0), (0, 0)) = (0, 0)$, $\lambda((q_0, p_0), (1, 0)) = (1, 0)$, $\lambda((q_0, p_0), (0, 1)) = (0, 0)$, $\lambda((q_0, p_0), (1, 1)) = (1, 0)$, | $\lambda((q_0, p_1), (0, 0)) = (0, 0)$, $\lambda((q_0, p_1), (1, 0)) = (1, 0)$, $\lambda((q_0, p_1), (0, 1)) = (0, 0)$, $\lambda((q_0, p_1), (1, 1)) = (1, 0)$, | $\lambda((q_0, p_2), (0, 0)) = (0, 0)$, $\lambda((q_0, p_2), (1, 0)) = (1, 0)$, $\lambda((q_0, p_2), (0, 1)) = (0, 1)$, $\lambda((q_0, p_2), (1, 1)) = (1, 1)$, | $\lambda((q_0, p_3), (0, 0)) = (0, 0)$, $\lambda((q_0, p_3), (1, 0)) = (1, 0)$, $\lambda((q_0, p_3), (0, 1)) = (0, 0)$, $\lambda((q_0, p_3), (1, 1)) = (1, 0)$ |
| $\lambda((q_1, p_0), (0, 0)) = (1, 0)$, $\lambda((q_1, p_0), (1, 0)) = (0, 0)$, $\lambda((q_1, p_0), (0, 1)) = (1, 0)$, $\lambda((q_1, p_0), (1, 1)) = (0, 0)$, | $\lambda((q_1, p_1), (0, 0)) = (1, 0)$, $\lambda((q_1, p_1), (1, 0)) = (0, 0)$, $\lambda((q_1, p_1), (0, 1)) = (1, 0)$, $\lambda((q_1, p_1), (1, 1)) = (0, 0)$, | $\lambda((q_1, p_2), (0, 0)) = (1, 0)$, $\lambda((q_1, p_2), (1, 0)) = (0, 0)$, $\lambda((q_1, p_2), (0, 1)) = (1, 1)$, $\lambda((q_1, p_2), (1, 1)) = (0, 1)$, | $\lambda((q_1, p_3), (0, 0)) = (1, 0)$, $\lambda((q_1, p_3), (1, 0)) = (0, 0)$, $\lambda((q_1, p_3), (0, 1)) = (1, 0)$, $\lambda((q_1, p_3), (1, 1)) = (0, 0)$ |



Результатом прямого произведения автоматов M_1 и M_2 будет автомат с диаграммой Мура, показанный на рисунке 6 (для наглядности без меток, соответствующих символам выхода вида (b_1, b_2) , где $b_1 \in B_1$ и $b_2 \in B_2$).



Р и с. 6. Диаграмма Мура результата прямого произведения автоматов M_1 и M_2
F i g. 6. Moore diagram of the result of the direct product of the automata M_1 and M_2

Прямое произведение автоматов $A_1 \times A_2 \times \dots \times A_n$ можно интерпретировать как множества состояний n параллельно работающих устройств. Поэтому прямое произведение автоматов называют также параллельным соединением (или параллельной композицией) конечных автоматов.

Операция группирования конечных автоматов в параллельный блок «M{»

При использовании теории конечных автоматов на практике может оказаться удобной множественная операция с переменным конечным числом операндов-автоматов, которую называем операцией группирования конечных автоматов в параллельный блок. С помощью этой операции создается именованный автомат-блок, который представляет собой сборку нескольких конечных независимо (параллельно) работающих автоматов в единый именованный блок. Такая сборка реализуется с помощью рассмотренной выше операции прямого произведения.

Синтаксис операции группирования конечных автоматов в параллельный блок имеет следующий вид:

$\langle \text{имя группы} \rangle \{ \langle \text{список автоматов-операндов} \rangle \}$,

где список автоматов состоит из перечисления необязательно однотипных инициальных автоматов.

Как правило, операция группирования конечных автоматов в параллельный блок используется для объединения в одну группу (блок с уникальным идентификатором) автоматов одного и того же или аналогичного типа, выполняющих одинаковые или аналогичные функции. Результатом работы такого автомата-блока является суммарный результат работы, производимой каждым автоматом-операндом, функционирующих независимо друг от друга.

Если даны n конечных автоматов M_i , где $i \in \{1 - n\}$ вида:

$M_i = \langle A_i, B_i, Q_i, \delta_i, \lambda_i, q_{0i} \rangle$,

то результат применения данной операции группирования можно записать следующим образом:

$\langle \text{имя группы} \rangle \{ M_1, M_2, \dots, M_n \} = \langle (A_1, \dots, A_n), (B_1, \dots, B_n), (Q_1, \dots, Q_n), \delta, \lambda, (q_{01}, \dots, q_{0n}) \rangle$,

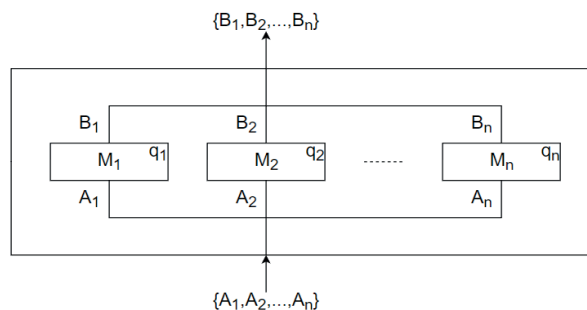
где функции переходов $q' = \delta(q, a)$ и выходов $b = \lambda(q, a)$ автомата M задаются соотношениями:

$q'_i = \delta(q_i, a_i), b_i = \lambda(q_i, a_i) \quad i \in \{1 - n\}$, где

$q' = (q'_1, q'_2, \dots, q'_n), a = (a_1, a_2, \dots, a_n), b = (b_1, b_2, \dots, b_n)$.

Свойством операции группирования является ее независимость от порядка перечисления операндов в наборе аргументов (при сохранении соответствия для каждого автомата с его входным и выходным алфавитами). Таким образом, две сборки с одним и тем же набором аргументов можно считать эквивалентными, если их наборы аргументов эквивалентны с точностью до перестановки аргументов.

На рисунке 7 иллюстрируется применение операции группирования конечных автоматов в параллельный блок.



Р и с. 7. Иллюстрация применения операции группирования конечных автоматов в параллельный блок

F i g. 7. An illustration of the application of the operation of grouping state machines into a parallel block

Производственным примером данной операции может служить объединение в логический кластер одного типа техники (например, грузовиков), работающих на одном уровне шахты и выполняющих аналогичную работу. Результатом работы кластера будет суммарный объем руды, перевезенный ими [12]-[20].

Пример применения операция группирования конечных автоматов в параллельный блок с именем $BLOCK_P$ для рассмотренного выше примера приводит к следующему результату:

$BLOCK_P \{ M_1, M_2 \} = \{ A_1, A_2 \}, \{ B_1, B_2 \}, \{ Q_1, Q_2 \}, \delta, \lambda, (q_{01}, q_{02}) >$

$q'_i = \delta(q_i, a_i), b_i = \lambda(q_i, a_i) \quad i \in \{1 - 2\}$, где

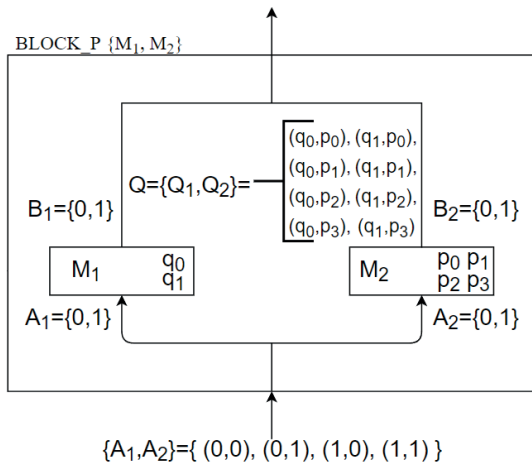
$q' = (q'_1, q'_2), a = (a_1, a_2), b = (b_1, b_2)$.

Определения функций $\delta(q_i, a_i)$ и $\lambda(q_i, a_i)$ для данного примера заданы с помощью Таблицы 1 и Таблицы 2.

Схема применения операция группирования конечных автоматов в параллельный блок с именем $BLOCK_P$ для рассмотренного выше примера показана на рисунке 8.

Применение данной операции может интерпретироваться как группирование некоторого конечного множества автоматов (вещей, например, микросхем) в некоторую более сложную конструкцию, например, блок микросхем, без установления связей между ними. Заметим, что результатом применения рассмотренной операции является конечный автомат, который также может использоваться в качестве аргумента операции группирования. Таким образом с помощью операции группирования возможно конструирование автоматов в виде многоуровневой системы устройств различной конструктивной сложности.





Р и с. 8. Пример применения операции группирования конечных автоматов в параллельный блок для автоматов M_1 и M_2 из примера выше
Fig. 8. An example of applying the operation of grouping finite automata into a parallel block for automata M_1 and M_2 from the example given above

Операция суперпозиции автоматов (*) [11]

Одним из широко применяемых на практике методов композиции автоматов является так называемая суперпозиция автоматов. С помощью двухместной операции суперпозиции создается конечный автомат, работа которого эквивалентна последовательной работе двух автоматов, когда выходные сигналы одного автомата используются как входные сигналы другого.

Формальное определение этой операции предполагает, что даны два автомата:

$$M_1 = \langle A, B, Q_1, \delta_1, \lambda_1 \rangle \text{ и } M_2 = \langle B, C, Q_2, \delta_2, \lambda_2 \rangle$$

таких, что входной алфавит M_2 совпадает с выходным алфавитом M_1 .

Тогда суперпозицией автоматов M_1 и M_2 (будем записывать $M_1 * M_2$) называется автомат $M = \langle A, C, Q, \delta, \lambda \rangle = M_2(M_1)$, множество состояний Q которого совпадает с прямым произведением $Q_1 \times Q_2$ множеств состояний автоматов M_1 и M_2 . Для любого состояния $q = (q_1, q_2)$ автомата M и любого входного сигнала x из A значения функций переходов и выходов автомата M определяются из следующих соотношений: $\delta(q, x) = (\delta_1(q_1, \lambda_1(q_1, x)), \lambda_2(q_2, \lambda_2(q_2, x)))$, при этом $q_0 = (q_{01}, q_{02})$.

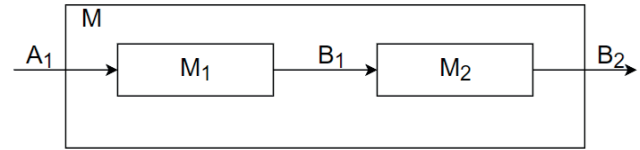
Понятие суперпозиции автоматов допускает естественное обобщение, а именно, обобщается на цепочку автоматов M_1, M_2, \dots, M_k в том случае, когда у каждого последующего автомата множество входных сигналов совпадает с множеством выходных предшествующего автомата в цепочке. Тогда

$$M = M_k(M_{k-1}(\dots M_1)) = M_1 * M_2 * \dots * M_k$$

Пример последовательной суперпозиции двух автоматов M_1 и M_2 из ранее рассмотренного примера иллюстрируется на рисунке 9.

Производственным примером данной операции может служить погрузчик с установленным бортовым компьютером (автомат M) к которому подключено дополнительное оборудование: сенсор подъема ковша (M_1) и весовой комплекс (M_2). Выходной алфавит автомата $M_1 = \{\text{Ковш_Поднят}, \text{Ковш_Опущен}\}$.

Автомат M_2 при получении сигнала Ковш_Поднят производит взвешивание, при получении состояния Ковш_Опущен – сбрасывает значение веса.



Р и с. 9. Суперпозиция последовательно работающих автоматов M_1 и M_2
Fig. 9. Superposition of sequentially operating automata M_1 and M_2

Операция группирования иерархической структуры (Ω)

Операция группирования иерархической структуры предназначена для формирования иерархической ветви, состоящей из автомата-мастера ($M = \langle A_m, B_m, Q_m, \delta_m, \lambda_m \rangle$) и группы подчиненных автоматов (M_1, M_2, \dots, M_k), посредством построения конечного автомата ветви автоматов, удовлетворяющего следующим условиям:

- входным алфавитом ветви иерархической структуры автоматов служит прямое произведение входных алфавитов подчиненных автоматов (т.е. $A_1 \times A_2 \times \dots \times A_k$);
- выходным алфавитом сборки служит выходной алфавит автомата-мастера, т.е. B ;
- множеством состояний является прямое произведение множеств состояний подчиненных автоматов и автомата-мастера (т.е. $Q_1 \times Q_2 \times \dots \times Q_k \times Q_m$).
- функция переходов ветви и функция выходов определяются следующим образом:
 $q'_i = \delta(q_i, a_i)$, $b_i = \lambda(q_i, a_i)$ и $q'_m = \delta_m((q'_1, q'_2, \dots, q'_k, q'_m), (b_1, b_2, \dots, b_k))$,

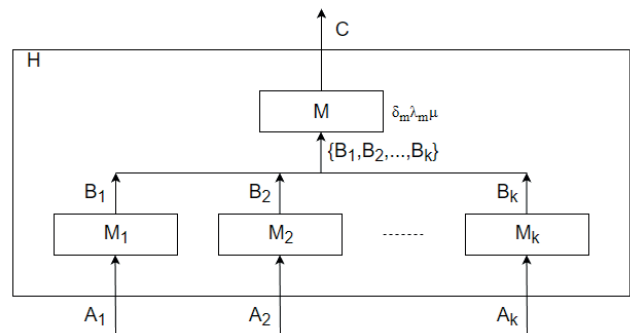
где $i \in \{1 - k\}$; $\lambda_m(q_m, (b_1, b_2, \dots, b_k)) = b_m$

- также с мастером связывается еще одна частично определенная функция – функция обратной связи $\mu(q_m, (b_1, b_2, \dots, b_k)) \rightarrow (Q_1 \times Q_2 \times \dots \times Q_k)$, которая выполняется одновременно с функцией δ_m и применяется для принудительного изменения состояний подчиненных автоматов.

Синтаксис операции группирования иерархической структуры:

$$M = \Omega(M_m, M_1, M_2, \dots, M_k)$$

Модель применения операции группирования иерархической структуры иллюстрируется на рисунке 10.



Р и с. 10. Модель операции иерархической сборки
Fig. 10. Hierarchical assembly operation model



Производственным примером данной операции может служить единица техники (бортовой компьютер погрузчика) с дополнительным установленным оборудованием на погрузчике, включающим, например: весовой комплекс, систему позиционирования, систему машинного зрения. В данном примере автоматом-мастером будет единица техники с контроллером (как функция преобразования алфавитов оборудования в выходной алфавит/сигнал единицы техники), который обрабатывает входящие сигналы с дополнительного оборудования (системы позиционирования, весового комплекса, системы машинного зрения – подчиненных автоматов) и устанавливает соответствующее состояние автомату-мастеру, который также может устанавливать начальные состояния подчиненным автоматам [21]-[25].

Операция системной композиции (⊠)

Операция системной композиции является многоместной операцией с переменным конечным числом операндов-автоматов. Она предназначена для интеграции двух и большего числа конечных полных автоматов в систему-автомат – набор совместно функционирующих над общим пространством событий автоматов с согласованными переходами в локальные состояния. Такую систему будем называть системой конечных автоматов, согласованных по событиям или event matched machines – сокращенно – ЕММ-системой.

Синтаксис операции системной композиции:

$$S = \boxtimes (M_1, M_2, \dots, M_k), M_i = \langle E, Q, \delta, q_{0i} \rangle, \text{ где } i = \{1 - k\}$$

Особенностью такой композиции является то, что у автоматов-операндов, как и результирующего автомата множество входных алфавитов одно и тоже – это поле событий E , элементами которого являются события, глобальные для всех автоматов системы, и при наступлении которых автоматы осуществляют переходы из одного состояния в другое локальное состояние. Также для этой операции будем рассматривать конечные полные автоматы в упрощенном виде – без выходного алфавита B и функции выхода λ . Ввиду того, что все автоматы системы работают в одном пространстве событий, отнесем множество событий E к самой системе S . В результате автоматы-операнды операции системной композиции будут описываться следующей конфигурацией:

$$M_i = \langle Q, \delta, q_{0i} \rangle.$$

Такие автоматы будем называть квазиавтоматами. Важными требованиями к квазиавтоматам являются требования их полноты (т.е. полноты функций переходов δ) и инициальности.

На самом деле упрощение автоматов практически не снижает общности решения задачи, так как при разработке автоматных моделей реальных систем с переходами автоматов из одного состояния в другое связываются семантические действия/функции, которые и производят необходимые выходные результаты работы автоматов, например, изменяют состояние информационной базы (базы данных D) системы.

Результатом выполнения операции системной композиции будет система-автомат, имеющая в общем случае следующую конфигурацию:

$$S = \langle E, A, Q, \delta, G, F, D \rangle,$$

где

- E – пространство событий системы,
- $A = E^k$ – входной алфавит для прямого произведения квазиавтоматов системы как множество векторов длины k , элементами которых являются системные события из E ,
- Q – множество состояний – прямое произведение $Q_1 \times Q_2 \times \dots \times Q_k$,
- $\delta: ((Q_1 \times Q_2 \times \dots \times Q_k), E^k) \rightarrow (Q_1 \times Q_2 \times \dots \times Q_k)$,
- при этом $\delta((q_{1i}, q_{2i}, \dots, q_{ki}), (e_1, e_2, \dots, e_k)) = (\delta_1(q_{1i}, e_1), \delta_2(q_{2i}, e_2), \dots, \delta_k(q_{ki}, e_k))$,
- G – блок управления системой S (используется в расширенных операциях над системами автоматов),
- F – семантические функции, связанные с переходами квазиавтоматов из одного состояния в другое или с другими элементами диаграмм автоматов,
- D – база данных – хранит глобальные данные системы и определяет статическое состояние системы в целом, которое может изменяться семантическими функциями,

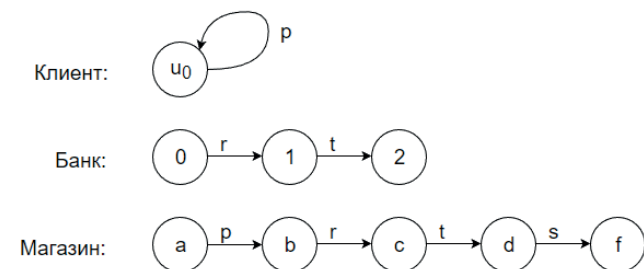
Рассмотрим пример, который в упрощенном виде заимствован из работы¹⁴, состоящий из трех квазиавтоматов, объединив их в систему S с помощью операции \boxtimes :

- покупатель
- магазин
- банк.

Общее пространство событий для системы S образуют следующие события:

- p – оплата покупателем покупки, посылкой файла электронных денег со счета в банке (pay),
- s – доставка покупки магазином покупателю (ship),
- r – запрос магазина банку на выкуп денежного файла покупателя на собственный счет (redeem),
- t – перевод денег банком со счета покупателя на счет магазина.

Диаграммы Мура для квазиавтоматов покупатель, магазин, банк показаны на рисунке 11.



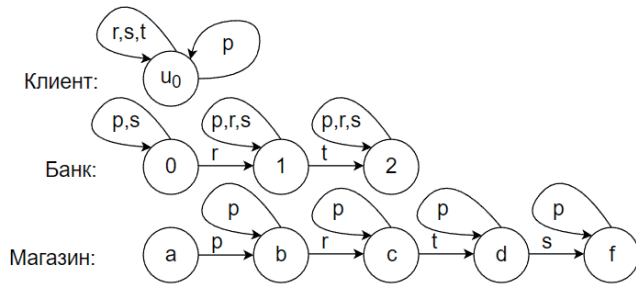
Р и с. 11. Диаграммы Мура для квазиавтоматов покупатель, магазин, банк
F i g. 11. Moore diagrams for quasi-automata buyer, store, bank

Заметим, что для выполнения операции системной интеграции эти квазиавтоматы не годятся, так как они не являются полными. Поэтому модифицируем эти квазиавтоматы, дополняя для каждой вершины циклическую дугу, помеченную списком событий, не используемых функцией переходов для данного состояния и логически допустимых в данном состо-

¹⁴ Hopcroft J. E., Motwani R., Ullman J. D. Introduction to Automata Theory, Languages, and Computation. 3rd ed. Addison-Wesley Longman Publishing Co., Inc., USA, 2006. 560 p.



янии. Таким образом данное состояние становится нейтральным для нефункциональных для него событий и тем самым не прерывает работу системы при возникновении в ней таких событий. В случае, если для вершины q , существуют события, для которых функция перехода не определена, то наступление таких событий в данном состоянии приведет к остановке автомата с сообщением об ошибке. Диаграммы Мура для доработанных до полноты квазиавтоматов покупатель, магазин, банк показаны на рисунке 12.



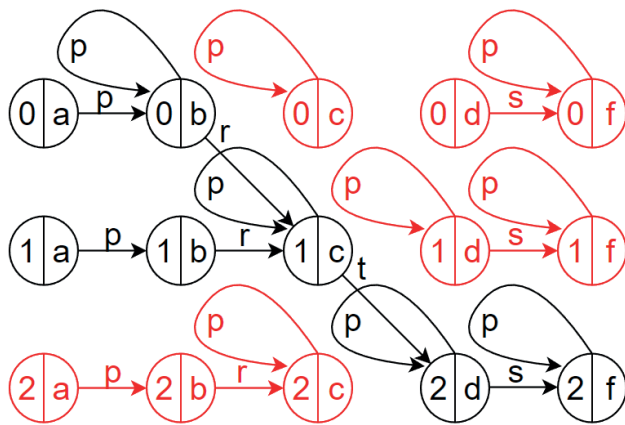
Р и с. 12. Диаграммы Мура для полных квазиавтоматов покупатель, магазин, банк

F i g. 12. Moore diagrams for full quasi-automata buyer, store, bank

В результате применения операции системной композиции к полным квазиавтоматам покупатель, магазин, банк получим систему-автомат:

$S = \boxtimes$ (покупатель, магазин, банк),

диаграмма Мура которой показана рисунке 13 (с учетом проецирования на единственное состояние для диаграммы покупатель).



Р и с. 13. Диаграмма Мура системы (с учетом проецирования на единственное состояние для диаграммы покупатель), где красным цветом выделены элементы диаграммы, которые могут быть удалены без ущерба функционированию системы автоматов. Такие элементы принадлежат несвязным подграфам диаграммы и неработающим ветвям диаграммы. F i g. 13. Moore diagram of the system (taking into account the projection to a single state for the buyer diagram), where the elements of the diagram are highlighted in red, which can be removed without compromising the functioning of the system of automata. Such elements belong to disconnected subgraphs of the diagram and non-working branches of the diagram.

Операция ассоциации между автоматами (<имя ассоциации>)

Под операцией ассоциации между двумя или большим числом автоматов будем понимать механизм, предоставляющий возможность синхронизации работы автоматов и их взаимодействия, например, посредством обмена сообщениями. Семантика данной операции следующая:

- для каждой ассоциации, например, <обмен сообщениями>, в пространстве событий системы создаются два события:
 - <старт – обмен сообщениями>
 - <финиш – обмен сообщениями>
- для каждого автомата, участвующего в ассоциации, вводится состояние q_r – «реализация ассоциации обмен сообщениями»,
- для состояний, допускающих реакцию на старт ассоциации, вводится дуга, ведущая в состояние q_r , таким образом взаимодействующие автоматы синхронно переходят в состояние <обмен сообщениями>, семантика которого реализует предусмотренный алгоритм обмена данными, например, через базу данных системы автоматов D ,
- в каждом автомате для состояния обработки ассоциации вводится дуга, по которой осуществляется переход по событию завершения ассоциации (<финиш – обмен сообщениями>) к состоянию, продолжающему работу автомата.

Производственным примером данной операции может служить объединение в логический кластер техники, работающей совместно. Например, при наступлении события «начало смены» вся ассоциируемая техника переходит в рабочее состояние, при наступлении события «конец смены» – в нерабочее.

Операция рекурсивного вызова автомата ЕММ-системы

Пусть задана ЕММ-система S , состоящая из k автоматов M_i (квазиавтоматов):

$S = \boxtimes (M_1, M_2, \dots, M_k)$, $M_i = \langle E, Q_i, \delta_i, q_{0i} \rangle$, где $i = \{1 - k\}$.

Ранее система S определялась следующей конфигурацией:

$S = \langle E, A, Q, \delta, G, F, D \rangle$.

Введем в конфигурацию S множество N – множество идентификаторов автоматов системы S , которое будем рассматривать как часть пространства имен S , при этом будем считать, что существует однозначное соответствие $N \rightarrow Q_0$, где Q_0 – множество инициальных состояний автоматов такое, что каждый идентификатор каждого автомата связан со своим инициальным состоянием. Далее алфавит состояний каждого автомата расширим с помощью множества N , т.е. $Q_i' = Q_i \cup N$, т.е. разрешим автомату системы осуществлять переход через глобальный идентификатор на связанное с ним инициальное состояние другого автомата, при этом адрес возврата в вызывающий автомат передается управлению системы G для сохранения в стеке адресов возврата, а вызываемому автомату передается текущее входное событие.

Введем еще одно расширение конфигурации автоматов системы S , добавив в конфигурацию каждого автомата множество конечных состояний H_i . Будем предполагать, что при переходе в состояние из множества H_i запрашивается у системы управления G адрес возврата, находящийся в вершине стека адресов возврата, по которому и происходит возврат управления в точку вызова, т.е. в состояние из которого был осуществлен переход на текущий автомат, а вызывавшему автомату передается



текущее выходное событие, связанное с конечным состоянием текущего автомата. В случае, когда стек адресов возврата пуст, считается, что система S успешно завершила свое выполнение.

Операция асинхронного вызова автомата ЕММ-системы

Определим операцию асинхронного вызова автомата ЕММ-системы - fork, с помощью которой разрешим автомату системы осуществлять переход через глобальный идентификатор другого автомата на связанное с ним инициальное состояние с передачей вызываемому автомату текущего входного события (вместе с его параметрами). При этом вызывающий автомат продолжает автономное функционирование.

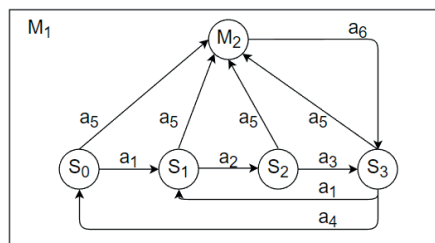
Таким образом новое определение ЕММ-системы следующее:

- E – пространство событий,
- $A = E^k$ – входной алфавит для прямого произведения автоматов (квазиавтоматов) системы как множество векторов длины k , элементами которых являются системные события из E ,
- N – множество идентификаторов автоматов системы S ,
- Q – множество состояний – прямое произведение Q_1, Q_2, \dots, Q_k ,
- Q_i – множество состояний i -го автомата, где $i = \{1 - k\}$,
- $Q_i^* = Q_i \cup N$,
- $H_i \subset Q_i$ – множество конечных состояний i -го автомата,
- $\delta: ((Q_1, Q_2, \dots, Q_k), E_k) \rightarrow (Q_1, Q_2, \dots, Q_k)$,
- при этом $\delta((q_{11}, q_{12}, \dots, q_{1k}), (e_1, e_2, \dots, e_k)) = (\delta_1(q_{11}, e_1), \delta_2(q_{12}, e_2), \dots, \delta_k(q_{1k}, e_k))$,
- G – блок управления системой S содержит стек адресов возвратов для реализации операции рекурсивного вызова автоматов,
- F – множество семантических функций, связанных с переходами квазиавтоматов из одного состояния в другое или с другими элементами диаграмм автоматов,
- D – база данных – хранит глобальные данные системы и определяет статическое состояние системы в целом, которое может изменяться семантическими функциями, связанными с переходами квазиавтоматов из одного состояния в другое,
- $M_i = \langle E, Q_i, H_i, \delta_i, q_{0i} \rangle$ – конфигурация i -го автомата $i = \{1 - k\}$, M_i – идентификатор i -го автомата $i = \{1 - k\}$

Рассмотрим пример применения операции рекурсивного вызова автоматов.

Производственным примером данной операции может служить следующая ситуация: пусть есть погрузчик (автомат M_1), выполняющий заданную работу, представленную на рисунке 14а (получение задания, отправка на место погрузки, погрузка, отправка к месту разгрузки, разгрузка и т.д.), где

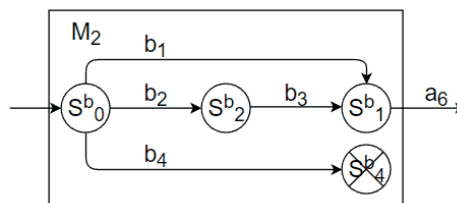
- M_1 – автомат «погрузчик»
- M_2 – автомат системы технического обслуживания
- S_0 – начальное состояние
- S_1 – нахождение на месте погрузки
- S_2 – нахождение на месте разгрузки
- S_3 – технический перерыв
- a_1 – получение задания
- a_2 – погрузка
- a_3 – разгрузка
- a_4 – задание окончено
- a_5 – возникновение неисправности
- a_6 – неисправность ликвидирована



Р и с. 14а. Диаграмма Мура автомата M_1 (погрузчик)
F i g. 14.a. Moore diagram of M_1 automaton (loader)

Во время выполнения работы или перед ее началом может случиться поломка (событие a_5), передающая управление в автомат M_2 , схема работы которого представлена на рисунке 14б (производится соответствующий тип ремонта и выполняются действия согласно типу ремонта), где

- S_0^b – начальное состояние
- S_1^b – ремонт выполнен, техника готова к работе
- S_2^b – ремонт в мастерской
- S_4^b – выход из эксплуатации
- b_1 – ремонт на месте
- b_2 – буксировка в мастерскую
- b_3 – капитальный ремонт
- b_4 – вывод из эксплуатации



Р и с. 14б. Диаграмма Мура автомата M_2 (система технического обслуживания)

F i g. 14.б. Moore diagram of M_2 automaton (maintenance system)

Итак, подведем итоги.

Рассмотренные в работе операции над конечными автоматами позволяют создавать произвольные композиции автоматов из более простых автоматов. Были рассмотрены следующие операции:

- 2.1. Операция прямого произведения автоматов (\times)
- 2.2. Операция группирования конечных автоматов в параллельный блок « $M\{\}$ »
- 2.3. Операция суперпозиции автоматов ($*$)
- 2.4. Операция группирования иерархической структуры (Ω)
- 2.5. Операция системной композиции (\boxtimes)
- 2.6. Операция ассоциации между автоматами ($\langle R \rangle$)
- 2.7. Операция рекурсивного вызова автомата (call).
- 2.8. Операция асинхронного вызова автомата (fork).

С помощью определенных выше операций над конечными автоматами определим алгебру конечных автоматов специального вида DTA (Digital Twin Algebra) над множеством конечных автоматов W , которую будем записывать $DTA = (W, \times, M\{\}, *, \Omega, \boxtimes, \langle R \rangle, call, fork)$.

Данная алгебра применима в качестве математического аппа-



рата для разработки формальных моделей поведения физических активов и их цифровых двойников. Она представляется гибким инструментом для моделирования поведения физических активов, их цифровых двойников и киберфизических систем, объединяющих возможности активов и их DT. Подходы к использованию алгебры DTA для моделирования поведения цифровых двойников и киберфизических систем производства авторы планируют рассмотреть в последующих публикациях.

Заключение

Результатом статьи явилась разработка алгебры конечных автоматов специального вида, именуемой DTA, предназначенной для математического моделирования поведения цифровых двойников производства. Основное внимание уделено изучению операций композиции детерминированных конечных автоматов, позволяющих строить модели составных активов

из более простых. В алгебру DTA включены как общеизвестные операции, такие, как, например, операция прямого произведения автоматов и операция суперпозиции автоматов, так и введены новые операции, включая: операции группирования конечных автоматов в параллельный блок, группирования иерархической структуры, системной композиции, ассоциации между автоматами, рекурсивного вызова автомата, асинхронного вызова автомата. Предложенный математический аппарат может использоваться для разработки формальных моделей поведения физических активов производства, их цифровых двойников, а также комплексных моделей киберфизических систем, включающих как физические, так и виртуальные активы. Построенная алгебра DTA, представляется гибким инструментом для моделирования поведения физических активов, их цифровых двойников и киберфизических систем, объединяющих возможности активов и их DT.

References

- [1] Borovkov A., Rozhdstvenskiy O., Pavlova E., Glazunov A., Savichev K. Key Barriers of Digital Transformation of the High-Technology Manufacturing: An Evaluation Method. *Sustainability*. 2021, 13(20):11153. (In Eng.) doi: <https://doi.org/10.3390/su132011153>
- [2] Mittal S., Khan M.A., Romero D., Wuest T. Smart manufacturing: Characteristics, technologies and enabling factors. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*. 2019; 233(5):1342-1361. (In Eng.) doi: <https://doi.org/10.1177/0954405417736547>
- [3] Grieves M., Vickers J. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In: Kahlen J., Flumerfelt S., Alves A. (eds.) *Transdisciplinary Perspectives on Complex Systems*. Springer, Cham; 2017. p. 85-113. (In Eng.) doi: https://doi.org/10.1007/978-3-319-38756-7_4
- [4] Lim K.Y.H., Zheng P., Chen C.H. A state-of-the-art survey of Digital Twin: techniques, engineering product lifecycle management and business innovation perspectives. *Journal of Intelligent Manufacturing*. 2020; 31(6):1313-1337. (In Eng.) doi: <https://doi.org/10.1007/s10845-019-01512-w>
- [5] Cheng Y., Yongping Z., Ping J., Wenjun X., Zude Z., Tao F. Cyber-physical integration for moving digital factories forward towards smart manufacturing: a survey. *International Journal of Advanced Manufacturing Technology*. 2018; 97(1-4):1209-1221. (In Eng.) doi: <https://doi.org/10.1007/s00170-018-2001-2>
- [6] Jacoby M., Usländer T. Digital Twin and Internet of Things – Current Standards Landscape. *Applied Sciences*. 2020; 10(18):6519. (In Eng.) doi: <https://doi.org/10.3390/app10186519>
- [7] Souza V., Cruz R., Silva W., Lins S., Lucena V. A Digital Twin Architecture Based on the Industrial Internet of Things Technologies. *2019 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE Computer Society; 2019. p. 1-2. (In Eng.) doi: <https://doi.org/10.1109/ICCE.2019.8662081>
- [8] Jacoby M., Jovicic B., Stojanovic L., Stojanović N. An Approach for Realizing Hybrid Digital Twins Using Asset Administration Shells and Apache StreamPipes. *Information*. 2021; 12(6):217. (In Eng.) doi: <https://doi.org/10.3390/info12060217>
- [9] Haskamp H., Orth F., Wermann J., Colombo A.W. Implementing an OPC UA interface for legacy PLC-based automation systems using the Azure cloud: An ICPS-architecture with a retrofitted RFID system. *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*. IEEE Computer Society; 2018. p. 115-121. (In Eng.) doi: <https://doi.org/10.1109/ICPHYS.2018.8387646>
- [10] Liu Y., Peng Y., Wang B., Yao S., Liu Z. Review on cyber-physical systems. *IEEE/CAA Journal of Automatica Sinica*. 2017; 4(1):27-40. (In Eng.) doi: <https://doi.org/10.1109/JAS.2017.7510349>
- [11] Glushkov V.M. The Abstract Theory of Automata. *Russian Mathematical Surveys*. 1961; 16(5):1-53. (In Eng.) doi: <https://doi.org/10.1070/RM1961v016n05ABEH004112>
- [12] Lu V.Q., et al. Developing a Dynamic Digital Twin at a Building Level: using Cambridge Campus as Case Study. *International Conference on Smart Infrastructure and Construction 2019 (ICSIC): Driving data-informed decision-making*. ICE Publishing; 2019. p. 67-75. (In Eng.) doi: <https://doi.org/10.1680/icsic.64669.067>
- [13] Vrabčič R., Erkoynucu J.A., Butala P., Roy R. Digital twins: Understanding the added value of integrated models for through-life engineering services. *Procedia Manufacturing*. 2018; 16:139-146. (In Eng.) doi: <https://doi.org/10.1016/j.promfg.2018.10.167>
- [14] Shen W., Hao Q., Xue Y. A loosely coupled system integration approach for decision support in facility management and maintenance. *Automation in Construction*. 2012; 25:41-48. (In Eng.) doi: <https://doi.org/10.1016/j.autcon.2012.04.003>
- [15] Luo W., Hu T., Ye Y., Zhang C., Wei Y. A hybrid predictive maintenance approach for CNC machine tool driven by Digital Twin. *Robotics and Computer-Integrated Manufacturing*. 2020; 65:101974. (In Eng.) doi: <https://doi.org/10.1016/j.rcim.2020.101974>



- [16] Abburu S., Berre A.J., Jacoby M., Roman D., Stojanovic L., Stojanovic N. COGNITWIN – Hybrid and Cognitive Digital Twins for the Process Industry. *2020 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. IEEE Computer Society; 2020. p. 1-8. (In Eng.) doi: <https://doi.org/10.1109/ICE/ITMC49519.2020.9198403>
- [17] Hamilton F., Lloyd A.L., Flores K.B. Hybrid modeling and prediction of dynamical systems. *PLOS Computational Biology*. 2017; 13(7):e1005655. (In Eng.) doi: <https://doi.org/10.1371/journal.pcbi.1005655>
- [18] Corallo A., Del Vecchio V., Lezzi M., Morciano P. Shop Floor Digital Twin in Smart Manufacturing: A Systematic Literature Review. *Sustainability*. 2021; 13(23):12987. (In Eng.) doi: <https://doi.org/10.3390/su132312987>
- [19] Aheleroff S., Xu X., Zhong R.Y., Lu Y. Digital Twin as a Service (DTaaS) in Industry 4.0: An Architecture Reference Model. *Advanced Engineering Informatics*. 2021; 47:101225. (In Eng.) doi: <https://doi.org/10.1016/j.aei.2020.101225>
- [20] Sepasgozar S.M.E. Differentiating Digital Twin from Digital Shadow: Elucidating a Paradigm Shift to Expedite a Smart, Sustainable Built Environment. *Buildings*. 2021; 11(4):151. (In Eng.) doi: <https://doi.org/10.3390/buildings11040151>
- [21] Chinesta F., Cueto E.G., Abisset-Chavanne E., Duval J.L., Khaldi F.E. Virtual, Digital and Hybrid Twins: A New Paradigm in Data-Based Engineering and Engineered Data. *Archives of Computational Methods in Engineering*. 2020; 27(1):105-134. (In Eng.) doi: <https://doi.org/10.1007/s11831-018-9301-4>
- [22] Rasheed A., San O., Kvamsdal T. Digital Twin: Values, Challenges and Enablers From a Modeling Perspective. *IEEE Access*. 2020; 8:21980-22012. (In Eng.) doi: <https://doi.org/10.1109/ACCESS.2020.2970143>
- [23] Redelinghuys A.J.H., Basson A.H., Kruger K. A six-layer architecture for the digital twin: a manufacturing case study implementation. *Journal of Intelligent Manufacturing*. 2020; 31(6):1383-1402. (In Eng.) doi: <https://doi.org/10.1007/s10845-019-01516-6>
- [24] Schluse M., Priggemeyer M., Atorf L., Rossmann J. Experimentable Digital Twins – Streamlining Simulation-Based Systems Engineering for Industry 4.0. *IEEE Transactions on Industrial Informatics*. 2018; 14(4):1722-1731. (In Eng.) doi: <https://doi.org/10.1109/TII.2018.2804917>
- [25] Tsinarakis G., Sarantinoudis N., Arampatzis G. A Discrete Process Modelling and Simulation Methodology for Industrial Systems within the Concept of Digital Twins. *Applied Sciences*. 2022; 12(2):870. (In Eng.) doi: <https://doi.org/10.3390/app12020870>

Поступила 26.04.2022; одобрена после рецензирования 02.06.2022; принята к публикации 21.06.2022.

Submitted 26.04.2022; approved after reviewing 02.06.2022; accepted for publication 21.06.2022.

Об авторах:

Гапанович Дмитрий Антонович, аспирант кафедры информационной безопасности факультета вычислительной математики и кибернетики, ФГБОУ ВО «Московский государственный университет имени М. В. Ломоносова» (119991, Российская Федерация, г. Москва, ГСП-1, Ленинские горы, д. 1), ORCID: <https://orcid.org/0000-0002-3222-694X>, dim.gapanovich@gmail.com

Сухомлин Владимир Александрович, заведующий лабораторией открытых информационных технологий факультета вычислительной математики и кибернетики, ФГБОУ ВО «Московский государственный университет имени М. В. Ломоносова» (119991, Российская Федерация, г. Москва, ГСП-1, Ленинские горы, д. 1), доктор технических наук, профессор, ORCID: <https://orcid.org/0000-0001-9468-7138>, sukhomlin@mail.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

About the authors:

Dmitry A. Gapanovich, Postgraduate Student of the Chair of Information Security, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University (1 Leninskie gory, Moscow 119991, GSP-1, Russian Federation), ORCID: <https://orcid.org/0000-0002-3222-694X>, dim.gapanovich@gmail.com

Vladimir A. Sukhomlin, Head of the Open Information Technologies Lab, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University (1 Leninskie gory, Moscow 119991, GSP-1, Russian Federation), Dr. Sci. (Tech.), Professor, ORCID: <https://orcid.org/0000-0001-9468-7138>, sukhomlin@mail.ru

All authors have read and approved the final manuscript.

