

ПРЕДСТАВЛЕНИЕ ОБЪЁМНО ПЛАНИРОВОЧНЫХ РЕШЕНИЙ ЗДАНИЙ КОНЕЧНЫМИ АВТОМАТАМИ В ПАРАДИГМЕ¹ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

УДК 721:004
ББК 85.11:32.81

В.Г. Куликов

Московский государственный строительный университет (МГСУ), Москва, Россия

Аннотация

В статье рассмотрены актуальные проблемы проектирования объёмно-планировочных решений зданий с использованием теории множеств, теории графов и теории автоматов. Разработана методика проектирования объёмно планировочных решений путём построения конечного автомата в виде сетевой модели для различных типов зданий. Проектируемые по приведенной методике модели конечных автоматов могут выполнять различные функции консультанта архитектора в виде систем искусственного интеллекта, или экспертных систем при разработке многовариантных проектных решений.

Ключевые слова: объёмно планировочные решения, компоновка, множество, граф, математическая логика, сетевая модель, искусственный интеллект

REPRESENTATION OF SPACE-PLANNING DECISIONS OF BUILDINGS OF THE STATE MACHINES IN THE PARADIGM OF ARTIFICIAL INTELLIGENCE

V.G. Kulikov

Moscow State Construction University (MGSU), Moscow, Russia

Abstract

In the article actual problems of designing space-planning decisions of buildings using set theory, graph theory, and theory of finite automata. The technique of designing space planning solutions by building a finite state machine in the form of network models for different types of buildings. Designed according to the above method models finite state machines can perform a variety of functions of a consultant architect in the form of artificial intelligence or expert systems in the development of multiple design solutions.

Keywords: space-planning decisions, layout, set, graph, mathematical logic, network model, artificial intelligence

Целью настоящей работы является разработка методики представления процесса архитектурного проектирования помещений и их коммуникационных связей конечными автоматами для разработки алгоритма решения такого рода задач с использованием нейронных сетей как базовых элементов текущего положения дел при разработке систем искусственного интеллекта.

Расположение помещений заданных размеров и формы в едином комплексе, подчиненное функциональным, техническим, архитектурно-художественным и

¹ Парадигма (от греч. *παράδειγμα* – пример, образец) – теория или модель постановки проблем.

экономическим требованиям, **объемно-планировочным решением** здания (ОПР)² будем называть компоновкой.

Компоновка с горизонтальными коммуникационными помещениями предусматривает связь между основными помещениями здания через коммуникационные помещения (коридоры, открытые галереи, рекреации). Это позволяет основные помещения проектировать непроходными. При этом проектируемые помещения могут располагаться по одну или по обе стороны галереи (рис. 1).

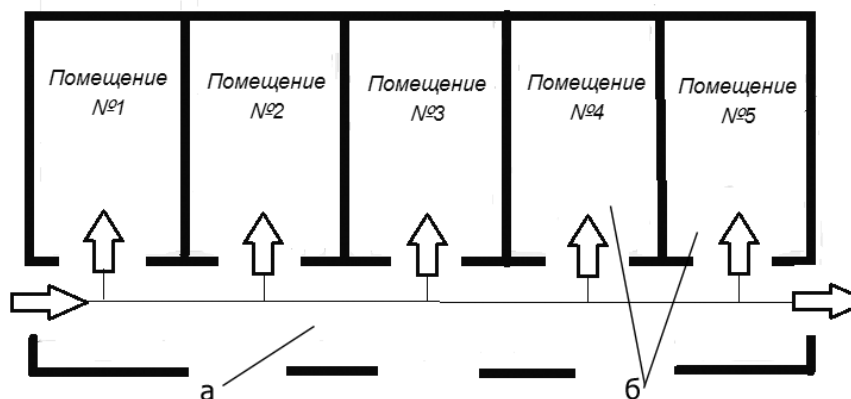


Рис. 1. Галерейная и коридорная система (ОПР): а) галерея, б) проектируемые помещения

На рис. 1 представлена галерейная система, состоящая из элементов здания 1 и 2. Проанализируем показанную на рис. 1 коммуникационную схему в парадигме сетевой модели. При этом будем использовать систему представления текущего анализа соответствующим терминологическим языком.

На рис. 1 система помещений связана системой входов, причём вход в любое из указанных (условно) пяти помещений возможен только из галерейного пространства. Выход из любого помещения также возможен только в галерейное пространство. Такую коммуникацию помещений можно представить ориентированным графом приведенном на рис. 2.



Рис. 2. Представление коммуникации галерейной системы ОПР ориентированным графом

Представим концепции конечного автомата и регулярного языка для процесса проектирования ОПР зданий. В кибернетике автоматами называют технически или программно-реализованные модули, предназначенные для обработки информации. В связи с этим, для обоснования методики представления процесса проектирования ОПР

² Объемно-планировочным решением здания называется объединение помещений избранных размеров и формы в единую композицию (т.е. - схема расположения помещений).

зданий и их коммуникационных связей конечными автоматами введём следующие определения: понятие **языка** и понятие **операции над языками**.

Назовём **алфавитом** произвольное непустое конечное множество символов. Символы произвольного алфавита будем называть **буквами**. Например, русский алфавит (с включением или не включением в него знаков препинания), латинский алфавит, объединение указанных алфавитов, множество цифр десятичной или двоичной систем счисления. В общем виде алфавит определим через множество $A = \{a_1, a_2, \dots, a_n\}$. Включим в число букв алфавита A специальный символ (символ пробела, пустой буквы), и этот символ обозначим через ϵ ³.

Словом в алфавите A будем называть произвольную конечную последовательность его букв, допуская при этом, что одна и та же буква в этой последовательности может встречаться многократно. Количество букв в слове назовём **длиной слова** α , обозначив её $l(\alpha)$. Символом λ будем обозначать **пустое слово**, имеющее нулевую длину. При этом будем отличать друг от друга пустое слово λ и непустое слово ϵ , состоящее из одной пустой буквы « ϵ ». Будем считать, что длина слова ϵ т.е. пробела, состоящего из одной пустой буквы ϵ , равна единице, т.е. $l(\epsilon)=1$.

В алфавите $A = \{a_1, a_2, \dots, a_n\}$ можно записать « n^l »⁴ различных слов длины l , где $l = 0, 1, 2, \dots, k$. Множество всех слов алфавита A обозначим через A^* . Заметим, что в множество A^* входит и пустое слово λ . Мощность множества всех слов алфавита A – счётное. Будем считать, что если α и β – два произвольных слова в алфавите A , то $\alpha\beta$ – результат приписывания справа слова β к слову α . Для любых слов, α и β будем считать, что $\alpha\lambda = \lambda\alpha = \alpha$, $\alpha\lambda\beta = \alpha\beta$.

Языком в алфавите A назовём произвольное подмножество слов L из A^* . Язык L будем считать **конечным**, если в его составе имеет место быть конечное множество слов; язык L будем считать **бесконечным**, если в его составе есть бесконечное множество слов. Совокупности всех слов русского, всех слов английского языка представляют собой примеры конечных языков. Множество записей всех простых чисел в десятичной или двоичной системе счисления является бесконечным языком. Множество всех языков алфавита A имеет континуальную мощность⁵.

Язык L в алфавите A назовём **универсальным** если $L=A^*$. Язык L называем **пустым**, если множество L пустое ($L = \emptyset$).

Если L_1 и L_2 – языки в алфавите A , то через $L_1 \cup L_2$ и $L_1 \cap L_2$ будем обозначать **объединение** и **пересечение** этих языков. Будем считать, что слово α принадлежит объединению двух языков, если оно принадлежит хотя бы одному из них (дизъюнкция⁶); слово α принадлежит пересечению двух языков, если оно принадлежит как одному, так и другому языку (конъюнкция⁷). Если L – язык в алфавите A , то через L^c обозначим

³ Сокращение англ. слова «empty» – пустой.

⁴ **Правило произведения.** Пусть требуется выполнить последовательно k действий. Если первое действие можно выполнить n_1 способами, второе действие n_2 способами, третье – n_3 способами и так до k -го действия, которое можно выполнить n_k способами, то все k действий вместе могут быть выполнены: $N=n_1 \times n_2 \times \dots \times n_k = n^k$ способами.

⁵ **Континуальный:** 1. спец. непрерывный, постоянный. 2. матем. не имеющий разрыва, определённый на всех точках числовой оси или её отрезка.

⁶ **Дизъюнкция:** сложное логическое выражение, которое истинно, если хотя бы одно из простых логических выражений истинно и ложно тогда и только тогда, когда оба простых логических выражения ложны. Обозначение: $F = A + B$.

⁷ **Конъюнкция:** это сложное логическое выражение, которое считается истинным в том и только том случае, когда оба простых выражения являются истинными, во всех остальных случаях данное сложное выражение ложно. Обозначение: $F = A \& B$.

дополнение⁸ этого языка. Язык L^c образуют слова алфавита A , не входящие в состав языка: $\{L: L^c = A^* \setminus L\}$.

Если L_1 и L_2 – языки в алфавите A , то через $L_1 \setminus L_2$ обозначим **разность**⁹ языков L_1 и L_2 . Слово α из A^* будем считать принадлежащим $L_1 \setminus L_2$ тогда и только тогда, когда оно принадлежит языку L_1 , но не принадлежит языку L_2 . Операцию разности представим, как: $L_1 \setminus L_2 = L_1 \cap (L_2)^c$.

Если L_1 и L_2 – языки в алфавите A , то через $L_1 \circ L_2$ обозначим язык, определяемый следующим образом: **слово** α принадлежит языку $L_1 \circ L_2$ тогда и только тогда, когда это слово можно разбить на две последовательные части (т.е. представить в виде: $\alpha = \alpha_1 \alpha_2$) так, что первая часть является словом языка L_1 , а вторая часть – словом языка L_2 . Операцию $\circ$ обозначим **конкатенацией**¹⁰ языков. Будем считать, что знак $\circ$ можно опускать, если конкатенацию языков L_1 и L_2 представить через $L_1 L_2$. Таким образом, язык $L_1 L_2$ формируется путём приписывания к словам языка L_1 слов языка L_2 в качестве окончаний. Заметим, что если к произвольному слову γ приписать в качестве окончания пустое слово λ , то результатом будем считать слово γ . Если языки L_1 и L_2 конечны, причем в составе первого языка m слов, а в составе второго n слов, то язык $L_1 L_2$ может состоять максимум из $m \times n$ слов. Если один из языков L_1, L_2 пуст, то язык $L_1 L_2$ – тоже пустой язык.

Множество, состоящее из следующих операций $L^0 = \{\lambda, L^1 = L, L^2 = L \times L, L^{n+1} = L^n \times L, n = 2, 3, \dots, k\}$, будем считать операцией возведения языка в степень. Обозначим через L^* объединение всех степеней языка L :

$$L^* = \bigcup_{i=0}^{\infty} L^i \quad (1)$$

Приведенную операцию будем называть **итерацией**¹¹. Положим, что пустое слово λ принадлежит итерации любого языка. Будем полагать, что слово α принадлежит итерации языка L тогда и только тогда, когда это слово можно разбить на некоторое количество последовательных частей (подслов) так, что каждая часть принадлежит языку L . Если язык L состоит из всех однобуквенных слов алфавита A , то итерацией этого языка является универсальный язык A^* . Заметим, что для любого языка L имеет место: $(L^*)^* = L^*$.

Ситуацию, изображенную на рис.2, можно представить следующим образом: обработку произвольного слова α во входном алфавите A проектируемый нами конечный автомат начинает из начального номера состояния № (0). В каждый дискретный момент времени на вход будет подаваться очередная буква обрабатываемого слова α . Из-за её «воздействия» автомат будет изменять своё текущее состояние № (N) на следующее № (N+1) состояние. Таким образом, номер состояния, в которое автомат переходит, определяется его предыдущим состоянием и считанной буквой входного слова α . Таким

⁸ **Дополнением** к классу A называется класс не- A (A^c), который при сложении с A образует универсальную область.

⁹ **Разность**. Существуют следующие операции с классами: объединение, пересечение, вычитание (**разность**), дополнение. Операция вычитания классов дает класс, состоящий из элементов, исключающих элементы вычитаемых классов. Вычитая элементы класса «юрист» (A) из класса «депутат Государственной Думы» (B), получаем класс депутатов Государственной Думы, не являющихся юристами.

¹⁰ **Конкатенация** – операция склеивания объектов линейной структуры, обычно строк. Например, конкатенация слов «микро» и «мир» даст слово «микромир». Конкатенация – бинарная операция, определённая на словах данного алфавита.

¹¹ **Итерация**: 1. книжн. очередное повторение какой-либо операции; 2. комп. организация обработки данных, при которой действия повторяются многократно, не приводя при этом к вызовам самих себя (в отличие от рекурсии).

образом, над словом α длины l автомат проработает l -е количество тактов, а результат работы определится номером состояния, в котором автомат окажется после считывания всего слова α .

Конечный автомат в парадигме системы искусственного интеллекта определим в виде следующего множества $\{K\}$:

$$K = \{Q, A, q_0, g(x), (F \subseteq Q)\} \quad (2)$$

где: $Q = \{q_0, q_1, q_2, \dots, q_m\}$ – множество состояний автомата; $A = \{a_1, a_2, \dots, a_n\}$ – входной алфавит; q_0 – специально выделенное начальное состояние; $g(x)$ – функция переходов, представляющая собой отображение следующего вида:

$$[Q \times K] \rightarrow [Q], \quad (3)$$

т.е. если $g(q_i, a_j) = q_k$, то автомат из состояния q_i под воздействием буквы a_j должен перейти в состояние q_k ; $F \subseteq Q$ – специально выделенное подмножество удовлетворяющих исходным требованиям состояний автомата.

Так, как $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_p\}$ – входное слово длины $l(\alpha)=p$, то через $q_\alpha(t)$ обозначим состояние, в котором оказывается автомат A через несколько (t) тактов работы над этим словом. Здесь $t = \{0, 1, 2, \dots, p\}$.

С учётом вышеизложенного рассмотрим следующую причинно-следственную серию состояний автомата:

$$q_\alpha(0)=q_0 \rightarrow q_\alpha(1)=g(q_\alpha(0), a_{i1}) \rightarrow q_\alpha(2)=g(q_\alpha(1), a_{i2}) \rightarrow q_\alpha(p)=g(q_\alpha(p-1), a_{ip}) \quad (4)$$

Будем полагать, что слово α принимается автоматом в том случае, когда $q_\alpha(p) \in F$. Тогда для языка $L(K)$: слово α будет принадлежать языку $L(K)$, если данное слово принимается автоматом K . В таком случае язык $L(K)$ будем считать языком, распознаваемым данным конечным автоматом. Язык L назовем **регулярным**, если для него возможно построить распознающий конечный автомат.

Функцию переходов конечного автомата $g(x)$ можно задать **диаграммой** его **переходов**. Диаграмма представляет собой ориентированный граф, вершины которого одноименны текущим состояниям автомата; дуга из вершины q_i в вершину q_k с надписанной над ней буквой a_j проводится тогда и только тогда, когда $g(q_i, a_j) = q_k$. В случае, когда переход из q_i в q_k осуществляется под воздействием любой из букв некоторого подмножества S , $S \subseteq A$, все буквы этого подмножества подписываются над соответствующей дугой (вместо перечня всех букв можно допустить сокращенную запись вида $x \in S$ или просто S). Если произвольное состояние q_i входит в F , то данный факт на диаграмме отмечается жирным кружком, выделяющим вершину q_i .

Очевидно, что конечный автомат полностью определяется своей диаграммой переходов. Принимая во внимание сказанное, под задачей построения конечного автомата, обладающего теми или иными свойствами, будем понимать задачу построения диаграммы его переходов. На рис. 3 показана диаграмма автомата AK_1 , работающего над словами алфавита $A=\{a, b, c\}$. Автомат имеет два состояния, q_0 и q_1 , причем «хорошим» (т.е. $F \subseteq Q$) является только состояние q_1 . Начав работу в состоянии q_0 , автомат под воздействием букв a, b из этого состояния не выходит. Под воздействием буквы c реализуется переход в состояние q_1 . Любая далее поступающая буква оставляет автомат в том же состоянии. Таким образом, автомат K_1 распознает язык L_1 , состоящий из слов, имеющих в своем составе букву c . Таким образом данный язык является регулярным.

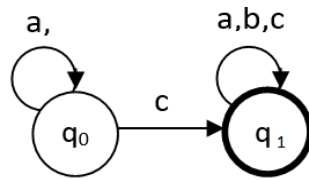


Рис. 3. Диаграмма переходов автомата K_1

Для решения задачи построения конечного автомата галерейной системы ОПР:

1) зададим $Q = \{q_0, q_1, q_2, \dots, q_m\}$ – множество состояний автомата в виде неориентированного графа-дерева, как это показано на рис. 4.

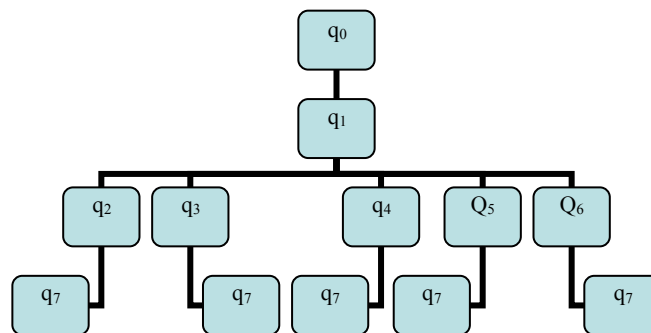


Рис.4 Диаграмма множества состояний автомата ОПР галерейной системы

На рис.4 принята следующая система обозначений:

- а) состоянию q_0 поставлено в соответствие начальное состояние системы, т.е. состоянию входа в галерею на рис.1;
- б) состоянию q_1 поставлено в соответствие состояние системы соответствующее позиции «Галерея» на рис.2;
- с) состоянию q_1 поставлено в соответствие состояние системы соответствующее позиции «Галерея» на рис.2;
- д) состоянию q_2 поставлено в соответствие состояние системы соответствующее позиции «Помещение № 1» на рис.2;
- е) состоянию q_3 поставлено в соответствие состояние системы соответствующее позиции «Помещение № 2» на рис.2;
- ф) состоянию q_4 поставлено в соответствие состояние системы соответствующее позиции «Помещение № 3» на рис.2;
- г) состоянию q_5 поставлено в соответствие состояние системы соответствующее позиции «Помещение № 4» на рис.2;
- х) состоянию q_6 поставлено в соответствие состояние системы соответствующее позиции «Помещение № 5» на рис.2;
- и) состоянию q_7 поставлено в соответствие состояние системы соответствующее конечной позиции, т.е. выходу из галереи на рис. 1;

2) Зададим $A = \{a_1, a_2, \dots, a_n\}$ – входной алфавит в виде: $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$, как это показано на рис.6.

3) Зададим q_0 – специально выделенное начальное состояние системы, т.е. состояние входа в галерею (рис.1а);

4) Зададим $g(x)$ – функцию переходов в виде ориентированного графа, изображённого на рис. 5.

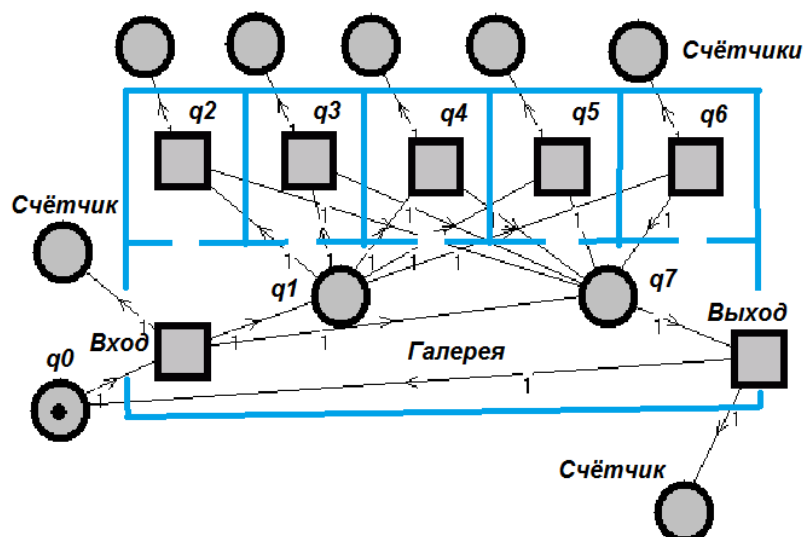


Рис. 5. Функция переходов в виде ориентированного графа

На рис. 6 приведен вариант разработанной сетевой модели конечного автомата ОПР галерейной системы ОПР зданий.

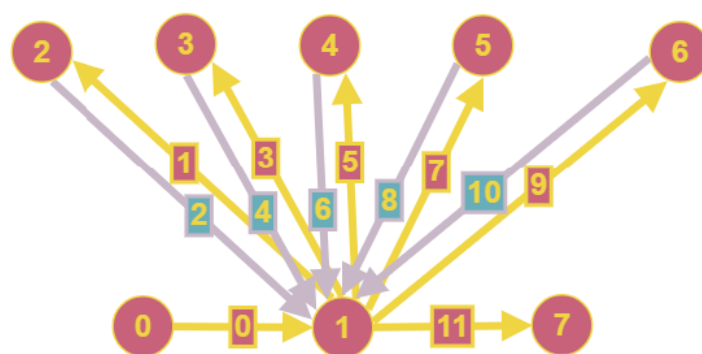


Рис. 6. Сетевая модель конечного автомата ОПР галерейной системы

Выводы:

- 1) В результате выполненного исследования разработана методика представления процесса архитектурного проектирования помещений галерейного типа и их коммуникационных связей формализованным способом типа конечного автомата.
- 2) В качестве постановочной модели использована сетевая модель в виде взвешенного ориентированного графа.
- 3) Приведенная методика позволяет получать конечные автоматы всех известных объёмно-планировочных решений коммуникаций зданий: анфиладную; систему с горизонтальными коммуникационными помещениями; зальную; атриумную; секционную; смешанную (комбинированную).
- 4) Проектируемые по разработанной методике модели конечных автоматов могут выполнять функции помощника архитектора в виде систем искусственного интеллекта, либо экспертных систем при разработке многовариантных проектных решений.

Литература

1. Прохорский Г.В. Информационные технологии в архитектуре и строительстве : учебное пособие / Г.В. Прохорский. – 2-е изд., стер. – М. : КНОРУС, 2012. – 264 с.
2. Змеул С.Г. Архитектурная типология зданий и сооружений: учебник / С.Г. Змеул, Б.А. Маханько. – М., 2004.
3. Лихненко Е. В. Архитектурные конструкции и основы конструирования [Электронный ресурс]: «Оренбургский гос. ун-т», Каф. архитектуры. – Оренбург : Оренбургский государственный университет, 2011. – 29 с.
4. Серебряков В. А. Теория и реализация языков программирования / В.А. Серебряков, М.П. Галочкин, Д.Р. Гончар, М.Г. Фуругян. – М.: МЗ-Пресс, 2006. – 2-е изд. ISBN 5-94073-094-9.
5. Карпов Ю.Г. Теория автоматов: учебник для вузов. – СПб.: Питер, 2002. – 224 с.
6. Мелихов А.Н. Ориентированные графы и конечные автоматы. – М.: Наука, 1971. – 416 с.
7. Коган Д.И. Учебно-методическая разработка для самостоятельной работы студентов по курсу «Теория алгоритмов и математическая логика» при изучении темы «Концепции конечного автомата и регулярного языка. Операции над регулярными языками» / Д.И. Коган, Т.С. Бабкина. – Нижний Новгород, 2000.

References

1. Pohorski G.V. *Informacionnye tehnologii v arhitekture i stroitel'stve : uchebnoe posobie* [Information technologies in architecture and construction. Textbook. 2nd ed. Erased]. Moscow, 2012, 264 p.
2. Zmeul S.G., Makhanko B.A. *Arhitekturnaja tipologija zdanij i sooruzhenij: uchebnik* [Architectural typology of buildings and structures. Tutorial]. Moscow, 2004.
3. Linenko E.V. *Arhitekturnye konstrukcii i osnovy konstruirovaniya* [Architectural design and design principles [Electronic resource], "Orenburg state University", Russia Architecture]. Orenburg, Orenburg state University, 2011, 29 p.
4. Serebryakov V.A., Galochkin M.P., Potter D.R., Furugen M.G. *Teorija i realizacija jazykov programirovanija* [Theory and implementation of programming languages]. Moscow, 2006, 2nd ed. ISBN 5-94073-094-9.
5. Karpov Yu.G. *Teorija avtomatov: Uchebnik dlja vuzov* [Theory of machines: Textbook for universities]. St. Petersburg, 2002, 224 p.
6. Melikhov A.N. *Orientirovannye grafy i konechnye avtomaty* [Directed graphs and finite state machines]. Moscow, 1971, 416 p.
7. Kogan D.I., Babkina T.S. *Uchebno-metodicheskaja razrabotka dlja samostojatel'noj raboty studentov po kursu «Teorija algoritmov i matematicheskaja logika» pri izuchenii temy «Konceptii konechnogo avtomata i reguljarnogo jazyka. Operacii nad reguljarnymi jazykami»* [The Educational-methodical development for independent work of students in the course "Theory of algorithms and mathematical logic" under the topic "concepts of finite automata and regular language. Operations on regular languages"]. Nizhny Novgorod, 2000.

ДАННЫЕ ОБ АВТОРЕ

Куликов Владимир Георгиевич

К.т.н., доцент, кафедра «Информационных систем, технологий и автоматизации в строительстве», Московский государственный строительный университет (МГСУ), Москва, Россия;

Профессор, кафедра «Высшая математика и строительная механика», Московский архитектурный институт (государственная академия), Москва, Россия

e-mail: 9646379164@mail.ru

DATA ABOUT THE AUTHOR

Kulikov Vladimir

PhD., Associate Professor, Department of Information Systems, Technologies and Automation in Construction, Moscow State Construction University (MGSU), Moscow, Russia;

Professor of «Higher Mathematics and Structural Mechanics», Moscow Institute of Architecture (State Academy), Moscow, Russia

e-mail: 9646379164@mail.ru