

УДК 004.043+519.713.1

Анализ структур данных для представления базовых моделей конечных автоматов

Гуренко В. В.^{1,*}, Бычков Б. И.¹

^{*}vgurenko@bmstu.ru

¹МГТУ им. Н.Э. Баумана, Москва, Россия

Применение аппарата теории автоматов в задачах анализа и синтеза структур сложных систем приводит к необходимости программной реализации моделей конечных автоматов. Структуры данных при этом нередко выбираются произвольно, что приводит к необоснованным затратам вычислительных ресурсов, снижает результативность анализа и эффективность компонентов самих систем. Целью статьи является обоснование выбора статических и динамических структур данных для машинной реализации базовых моделей конечных автоматов. Структуры данных подобраны исходя из традиционных способов задания автоматов. Авторами предложена система критериев для оценки структур данных. Критерии сформулированы на основе алгоритмических особенностей задач теории автоматов с учетом вычислительной сложности в емкостной и временной составляющих. Проведен сравнительный анализ структур данных по предложенной системе критериев. Анализ показал преимущество тех структур, которые отвечают способу задания автомата таблицей переходов/выходов. Для таких структур был проведен эксперимент с целью измерения временных параметров. Исследование полученных данных дало возможность разграничить области применения выбранных структур в задачах теории автоматов. Результаты работы позволят более эффективно применять модели конечных автоматов в программных разработках.

Ключевые слова: теория автоматов, конечный автомат, структура данных, вычислительная сложность

Введение

В задачах анализа и структурного синтеза на всех этапах разработки сложных систем возникает необходимость перехода от реальных объектов к их описанию математическими моделями. Хорошо проработанный и развитый аппарат теории автоматов предоставляет широкие возможности для описания компонентов систем и моделей их поведения. В литературе, посвященной теории автоматов, основное внимание уделено двум аспектам. Во-первых, формальному описанию автоматов с последующим решением традиционных задач теории автоматов, включая задачи анализа и синтеза [1-4]. Во-вторых, практическому приложению автоматов в различных областях, в том числе находящихся на стыке традиционных научных направлений и даже за пределами области

технических наук. Так, аппарат теории автоматов успешно применяется в прикладных задачах математической лингвистики, биоинформатики, теории алгоритмов, теории формальных систем, криптологии и других.

Конечные автоматы (КА) как преобразователи дискретной информации с прикладной точки зрения ассоциируются с системами, выполняющими конкретную последовательность действий в соответствии с заданным алгоритмом. Существует ряд программных реализаций автоматов через выполняемые ими функции. Примером подобного подхода может служить система «The State Machine framework» из библиотеки Qt 5 фирмы Digia Plc. Такие реализации направлены на применение, но не на исследование автоматов.

Между тем вопросы, относящиеся к внутренней структуре автоматов в контексте их программной реализации, изучены недостаточно и обычно остаются за рамками публикуемых работ. Вопрос обоснования выбора структур данных для представления КА не исследован. На практике при выполнении синтеза и программном исполнении конечных автоматов структуры данных, как правило, назначают произвольно, что приводит к неоптимальным решениям с точки зрения затрат вычислительных ресурсов. В связи с этим целесообразно найти решение задачи в следующей постановке.

Основываясь на традиционных способах задания базовых моделей КА, определить варианты организации структур данных для машинной реализации автоматов. Рассмотреть, какой из структур данных следует отдать предпочтение при решении той или иной задачи теории автоматов. Для этого выработать систему критериев с целью оценки структур данных с точки зрения вычислительной сложности, т.е. учитывающую как объем занимаемой памяти, так и время, затрачиваемое на выполнение операций над КА. На основании полученной системы критериев провести сравнительный анализ структур данных. Проверить экспериментально полученные результаты и сделать вывод о предпочтении той или иной структуры в зависимости от решаемой задачи теории автоматов.

1. Варианты структур данных

Под базовыми моделями КА традиционно понимают автоматы Мили и Мура [1-4], причем обе модели описывают детерминированные КА [1]. Такой автомат считают заданным, если определены следующие его компоненты:

- множества X и Y входного и выходного алфавита КА;
- множество S состояний автомата;
- начальное состояние s_0 ;
- однозначная функция переходов f и функция выходов g .

В автомате Мили в качестве аргументов функции g выступают текущее состояние и входной сигнал, а в автомате Мура – только текущее состояние.

Для программного описания моделей Мили или Мура необходимо решить вопрос о машинном представлении определяющих компонентов. Для множеств X , Y и S это не

вызывает затруднений: например, для хранения их элементов можно использовать массивы, при этом размерность каждого совпадает с мощностью соответствующего множества. Однако для большинства задач теории автоматов непосредственного хранения элементов множеств не требуется. В случае программной реализации структур достаточно лишь нумерации элементов, например, от нуля и далее по ряду натуральных чисел. Тогда потребность в массивах или иных структурах данных отпадает, требуется знать только мощность соответствующего множества. Задание начального состояния s_0 также тривиально.

Таким образом, задача машинного представления КА сводится к поиску оптимальных структур данных для представления функции переходов f и функции выходов g . Из ряда возможных вариантов [1-4] рассмотрим следующие:

- 1) таблица переходов/выходов;
- 2) матрица соединений;
- 3) взвешенный ориентированный граф переходов КА.

Существуют и другие способы задания КА. Например, конечный автомат-распознаватель может быть задан синтаксической диаграммой. Однако этот вариант выходит за рамки настоящей работы, поскольку базовые модели КА являются преобразователями, а не распознающими автоматами.

Из всего многообразия структур данных, широко рассмотренных в литературе (см. [5-9]), для представления КА указанными выше способами наиболее подходят статические структуры в виде многомерных массивов и матриц, а также динамические в виде списков различной связности. Последние могут быть оптимизированы. В частности, оптимизация может быть проведена с целью устранения фрагментации памяти, что существенно с точки зрения минимизации времени доступа в задачах большой размерности, когда в структуре интегрированы большие объемы данных. Этого достигают, например, за счет создания локальных непрерывных участков данных с указателями (так называемых «пулов»), которые целиком помещают в кэш-память, и дальнейшего соединения их в цепочки. При этом для экономии памяти и сокращения времени адресации указатели внутри и вне пула могут иметь разную длину [10].

Рассмотрим первый из трех обозначенных способов представления КА – таблицу переходов/выходов. Приемлемым вариантом ее представления статической структурой данных является многомерный массив: трехмерный в случае КА Мили и двумерный (прямоугольная матрица) в случае КА Мура. Трехмерность возникает из-за того, что каждому состоянию КА Мили соответствует пара «состояние – выходной сигнал». Наиболее универсальным способом динамического представления является двухуровневый вектор Айлифа – так называемый «список списков», оба измерения которого имеют переменную длину. Использовать многосвязный список для таблицы переходов/выходов нецелесообразно ввиду простоты самой таблицы и довольно громоздкой структуры такого списка. Оправдать его применение можно было бы

наложением условия упорядоченности компонентов таблицы, однако операция сортировки в задачах теории автоматов не является приоритетной.

Представление таблицы переходов/выходов статической и динамической структурами данных показано на рис. 1 для КА Мили и рис. 2 для КА Мура. Введены обозначения: $|S|=n$, $|X|=m$, $|Y|=k$. Заметим, что в векторе Айлифа номера столбцов можно не хранить, так как переходы упорядочены.

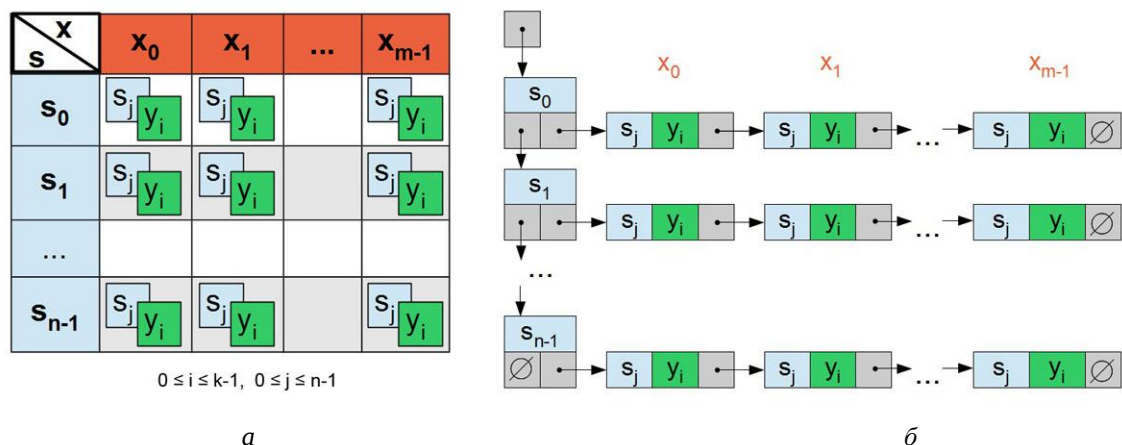


Рис. 1. Таблица переходов/выходов КА Мили в виде:

а – статического трехмерного массива; б – двухуровневого вектора Айлифа

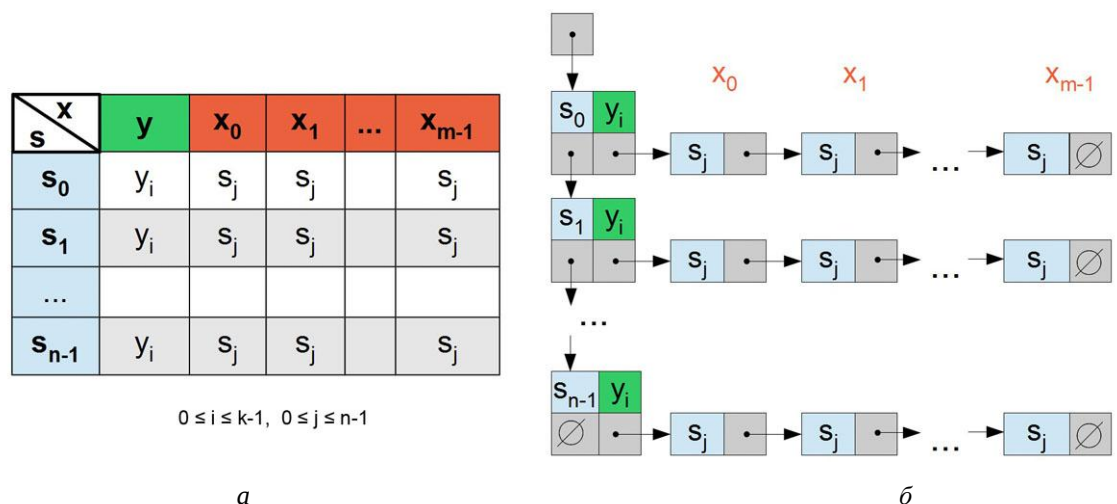


Рис. 2. Таблица переходов/выходов КА Мура в виде:

а – статической матрицы; б – двухуровневого вектора Айлифа

Для представления матрицы соединений КА [2] можно использовать сходные структуры данных, однако каждым элементом такой матрицы является список. Поэтому для организации статической и динамической структур следует применить,

соответственно, матрицу списков и трехуровневый вектор Айлифа. Вид названных структур для КА Мили показан на рис. 3¹.

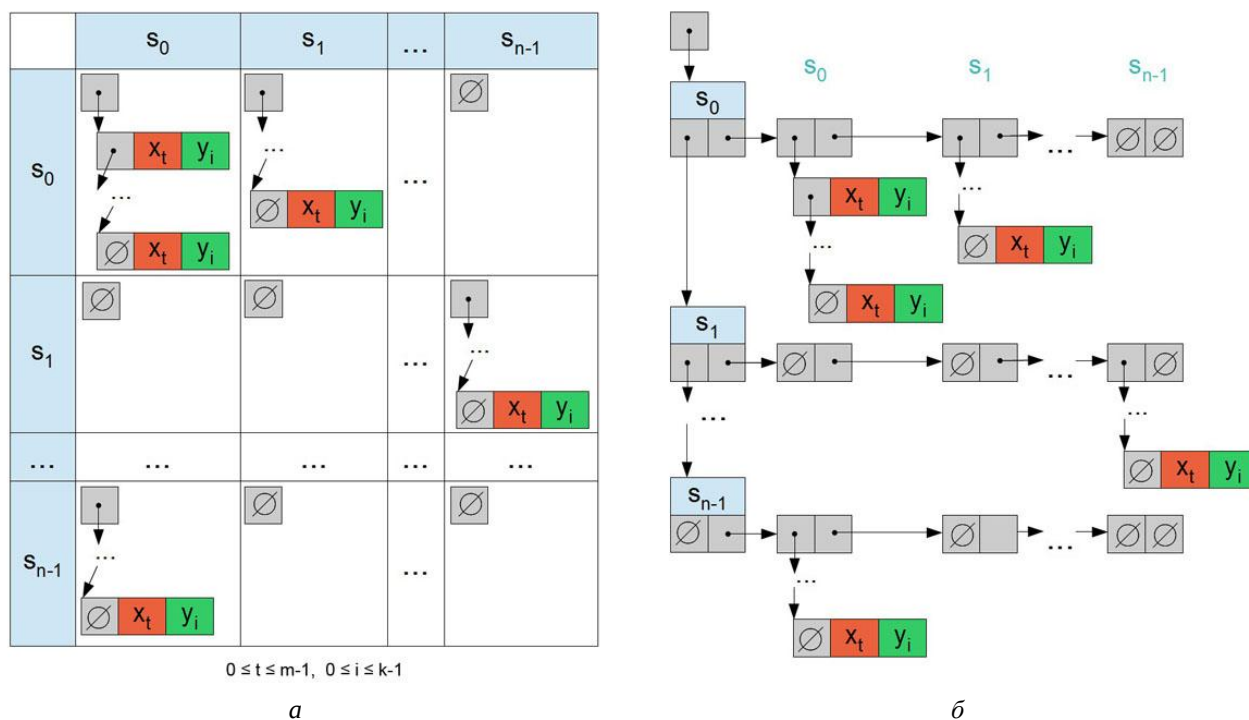


Рис. 3. Матрица соединений КА Мили в виде:

а – матрицы списков; *б* – трехуровневого вектора Айлифа

Взвешенный оргграф переходов КА [1, 2] можно задать любой структурой данных, разработанной или адаптированной для представления графовых моделей [11-14]. Традиционные для графов статические структуры – матрицу смежности и матрицу инцидентий нельзя признать оптимальными с точки зрения машинной реализации, поскольку компенсацией за их простоту и наглядность, как правило, является сильная разреженность, что увеличивает и емкостную, и временную сложность.

Более рациональной следует признать списковую структуру. Граф можно представить не только вектором Айлифа, но и многосвязным списком, в котором каждый элемент соответствует определенной вершине и содержит список меток инцидентных ей ребер. Начальным следует считать элемент списка, отвечающий вершине s_0 . На рис. 4 показан пример графа переходов автомата Мили с $|S|=4$, $|X|=2$, $|Y|=3$ и его представление многосвязным списком.

¹ Переменные индексов i , j и t на рис. 1 – 3 в элементах изображенных структур данных принимают произвольные (не одинаковые) значения из указанных диапазонов.

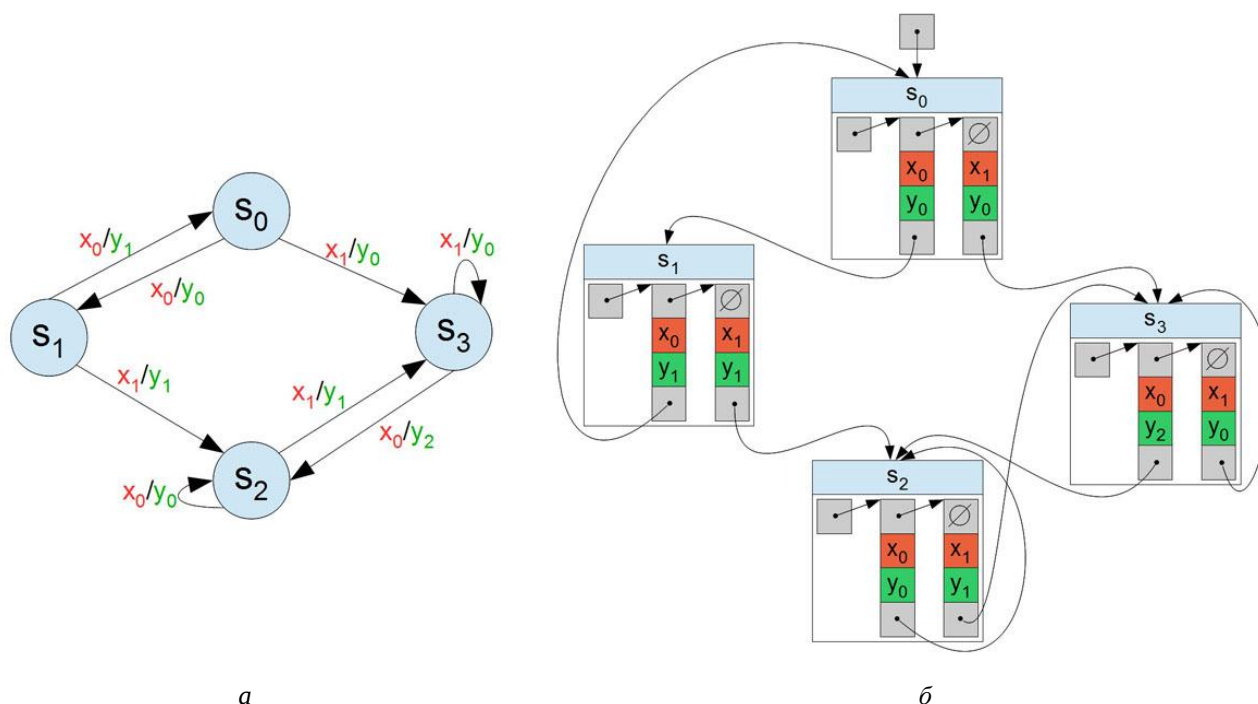


Рис. 4. Граф переходов КА Мили:

a – изображение; b – в виде многосвязного списка

Структуры данных, выделенные для машинной реализации КА, приведены в табл. 1.

Таблица 1. Структуры данных для представления КА

Способ задания КА	Тип структуры данных	
	Статический	Динамический
Таблица переходов/выходов	трехмерный массив (КА Мили), матрица (КА Мура)	двухуровневый вектор Айлифа
Матрица соединений	матрица списков	трехуровневый вектор Айлифа
Взвешенный орграф переходов	–	многосвязный список

2. Критерии оценки структур данных

Эффективность любой структуры данных, в том числе предназначенной для представления базовых моделей КА, следует оценивать с точки зрения удовлетворения параметрам, существенным для решения задач с ее использованием. Среди задач теории автоматов выделим следующие:

- задача эквивалентных преобразований (переход от одной базовой модели КА к другой);
- задачи определения эквивалентности и изоморфизма КА;
- задача минимизации.

Кроме перечисленных, краеугольное положение в теории автоматов занимают задачи анализа и синтеза КА, однако они связаны с алгоритмами функционирования конкретных детерминированных устройств и поэтому в данной работе не рассматриваются.

Для сравнения структур данных уместно применить оценку вычислительной сложности в емкостной и временной составляющих. При этом емкостная сложность даст оценку объема памяти, занимаемого структурой, а временная – времени выполнения алгоритмов конкретных операций над КА с использованием данной структуры [15].

Таким образом, в качестве первого критерия оценки машинных структур представления КА следует принять *память, отводимую под хранение структуры данных*. Заметим, что объем памяти, требуемый для решения самих задач теории автоматов, характеризует алгоритмы, а не структуры данных и потому критерием оценки являться не должен.

Рассмотрим, время выполнения каких операций существенно при решении каждой из обозначенных задач. Будем считать, что для работы с машинным представлением КА используются известные алгоритмы задач теории автоматов, описанные, например, в [1-4].

Задача эквивалентных преобразований. Алгоритм преобразования КА Мура в КА Мили в случае представления автомата графом переходов предусматривает перенесение выходного сигнала, соответствующего каждой вершине, на все дуги, входящие в эту вершину [1]. При этом значение нужного выходного сигнала извлекают из таблицы соответствия состояний и выходных сигналов, которую следует сформировать до начала обхода графа.

Обратное преобразование КА Мили в КА Мура подразумевает поиск пар «состояние – выходной сигнал», приводящих к переходу автомата в новое состояние. При этом происходит перебор всех возможных переходов КА с извлечением для каждого из них информации об иницирующем его входном сигнале и о вырабатываемом на переходе выходном сигнале. Подробно данный алгоритм изложен в [1, 2].

Основываясь на алгоритмах преобразования базовых моделей КА, можно выделить два временных критерия оценки структур данных: *время формирования таблицы соответствия состояний и выходных сигналов КА Мура* и *время полного перебора переходов КА Мили* с получением для каждого перехода начального и конечного состояний, входного и выходного сигналов.

Задача определения эквивалентности КА. Алгоритм проверки двух КА на эквивалентность основан на теореме Мура, согласно которой из прямого (декартова) произведения компонентных множеств КА необходимо удалить недостижимые состояния. В условиях данной задачи существенны: время перебора всех переходов КА, которое уже включено в число критериев, а также *время создания и заполнения структуры* нового КА, полученного в результате применения теоремы Мура. Кроме того, необходимо учитывать

время удаления недостижимого состояния КА. Отметим, что время поиска всех недостижимых состояний эквивалентно времени полного перебора переходов.

Те же критерии следует выделить и для задачи определения изоморфизма КА.

Задача минимизации КА. Алгоритм минимизации детально разобран в [1]. Он также предусматривает полный обход графа переходов автомата. Для решения этой задачи не требуется выполнения каких-либо действий со структурой данных, неучтенных при рассмотрении предыдущих задач.

Таким образом, алгоритмические особенности задач теории автоматов дают систему критериев оценки структур данных, представленную в табл. 2. Знаком «+» помечены вхождения критериев в соответствующие задачи.

Таблица 2. Критерии оценки структур данных

№ критерия	Наименование критерия	Задачи теории автоматов			
		Эквивалентные преобразования автоматов	Определение эквивалентности автоматов	Определение изоморфизма автоматов	Минимизация автомата
1.	Объем памяти для данных о функциях переходов и выходов	+	+	+	+
2.	Время формирования таблицы соответствия состояний и выходных сигналов КА Мура	+	–	–	–
3.	Время полного перебора переходов КА	+	+	+	+
4.	Время создания и заполнения структуры	–	+	+	–
5.	Время удаления состояния	–	+	+	+

3. Сравнение и анализ

Выработанная система критериев позволяет дать оценку структурам данных с позиции вычислительной сложности алгоритмов решения задач теории автоматов, использующих эти структуры. Как видно из табл. 2, в вычислительной сложности учтены как емкостная, так и временная составляющие.

3.1. Оценка емкостной сложности

С целью проведения оценки приведем аналитические зависимости для объема памяти M , занимаемого каждой из предложенных структур данных. Соответствующие формулы получены из графических представлений этих структур (см. рис. 1-4) и сведены в табл. 3. Приняты следующие обозначения: l_j – размер поля, которое хранит номер элемента множества j (X , Y или S), l_p – размер указателя.

Таблица 3. Формулы для вычисления объема памяти

Способ задания КА	Структуры данных	
	Статические	Динамические
Таблица переходов/выходов КА Мили	трехмерный массив $M = S _{\max} X _{\max} (l_S + l_Y)$	двухуровневый вектор Айлифа $M = l_P + S (2l_P + l_S + X (l_S + l_Y + l_P))$
Таблица переходов/выходов КА Мура	матрица $M = S _{\max} (X _{\max} l_S + l_Y)$	двухуровневый вектор Айлифа $M = l_P + S (2l_P + l_S + l_Y + X (l_S + l_P))$
Матрица соединений КА Мили	Матрица списков $M = S _{\max}^2 l_P + S X (l_X + l_Y + l_P)$	трехуровневый вектор Айлифа $M = l_P + S (2l_P + l_S + 2 S l_P + X (l_X + l_Y + l_P))$
Взвешенный орграф переходов КА Мили	–	многосвязный список $M = l_P + S (l_S + l_P + X (2l_P + l_X + l_Y))$

В формулах для статических структур данных использованы максимальные мощности множеств S и X . Это означает, что в отличие от динамических структур объем памяти в процессе обработки менять нельзя: память выделяют сразу под всю структуру до заполнения ее данными о конкретном автомате. В этом смысле любая из динамических структур дает преимущество, например, при удалении состояния автомата, так как память из-под удаленных элементов списка освобождается сразу. В случае матрицы списков $|S|_{\max}$ задает порядок матрицы.

Формулы для структур данных, соответствующих матрице соединений и графу переходов, составлены для КА Мили. Зависимости для КА Мура аналогичны и опущены для краткости.

В качестве примера поясним получение формулы для вычисления объема памяти, занимаемого трехуровневым вектором Айлифа, который представляет динамическую реализацию матрицы соединений (см. табл. 3). Из рис. 3, б видно, что память требуется под указатель размером l_P на всю структуру данных, а также на $|S|$ двухуровневых векторов Айлифа. В каждый вектор входят: начальный элемент, содержащий информацию о состоянии – его объем складывается из размеров двух указателей l_P и поля с номером состояния l_S , и $|S|$ одинаковых элементов из двух указателей. Таким образом, объем каждой «строки» матрицы равен $2l_P + l_S + 2|S|l_P$ без учета элементов, содержащих данные о сигналах КА. В силу детерминированности рассматриваемых КА каждой строке соответствуют $|X|$ таких элементов, а объем каждого элемента складывается из размеров указателя, поля входного сигнала и поля выходного сигнала: $l_X + l_Y + l_P$. Суммируя описанные составляющие, получаем итоговую формулу, приведенную в соответствующей ячейке таблицы.

Формулы для других структур данных (см. табл. 3) получены в результате аналогичных рассуждений.

На основании формул были построены графики зависимостей объема памяти от мощности множества S при фиксированном числе входных сигналов (рис. 5, а) и от мощности множества X при фиксированном числе состояний автомата (рис. 5, б). Взял наиболее сложный вариант реализации автомата Мили – динамическими структурами данных, т.е. двух- и трехуровневым вектором Айлифа и многосвязным списком (см. формулы в строках 1, 3, 4 правого столбца табл. 3). Для определенности фиксированы: в первом случае $|X|=2$, во втором $|S|=10$. Также принято: $l_p = l_s = l_x = l_y = 4$ байта (стандартный размер указателя и данных целого типа в средах программирования, таких как Microsoft Visual Studio, Borland Delphi и ряде других).

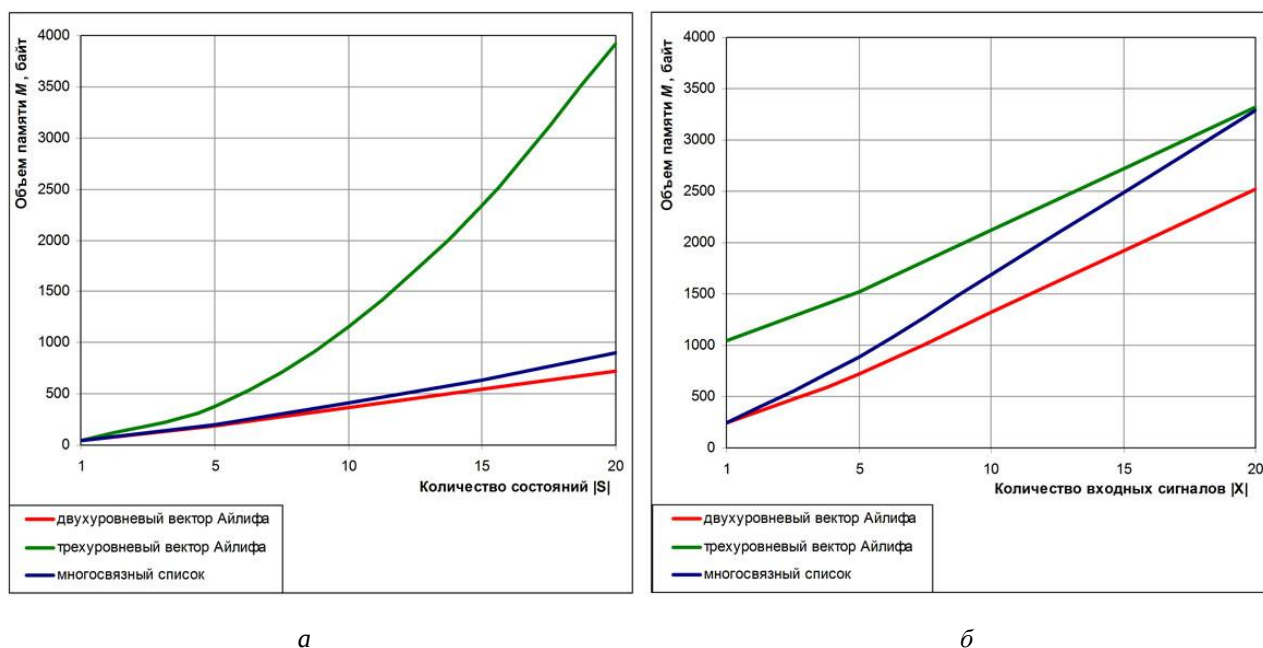


Рис. 5. Зависимости объема памяти для структур данных (КА Мили):

a – от количества состояний при $|X|=2$; $б$ – от числа входных сигналов при $|S|=10$

Анализ полученных зависимостей показывает, во-первых, что объем трехуровневого вектора Айлифа квадратично зависит от $|S|$, в то время как прочие зависимости линейны. Во-вторых, двухуровневый вектор Айлифа, реализующий таблицу переходов/выходов, требует меньше памяти, чем остальные структуры, причем это верно как для зависимости от $|S|$ при фиксированном $|X|$, так и в случае роста $|X|$ при фиксированном $|S|$. Таким образом, среди всех рассматриваемых структур данных динамического типа наименьшее значение емкостной сложности достигается при реализации КА Мили двухуровневым вектором Айлифа.

3.2. Оценка временной сложности

Переходя к оценке по временным критериям, в первую очередь следует отметить существенные особенности матрицы соединений и графа переходов КА, влияющие на

время выполнения операций над КА при представлении соответствующими структурами данных.

Ощутимым недостатком матрицы соединений по сравнению с таблицей переходов/выходов является ее разреженность. Доля заполненных ячеек составляет не более чем $|X|/|S|$. Это говорит о том, что заполнение ячеек (*критерий 4*, см. табл. 2) и полный перебор переходов по матрице соединений (*критерий 3*) потребуют больше времени, чем при работе со статическим массивом либо вектором Айлифа, которыми реализована таблица переходов/выходов (см. рис. 1, 2 и 3). На *критерий 2* данная особенность не влияет, поскольку хранение выходных сигналов КА Мура в обеих структурах – как в матрице, так и в векторе Айлифа организовано одинаково: с помощью столбца матрицы или дополнительных полей элементов вектора Айлифа.

Матрица соединений в случае статической реализации проигрывает также и по *критерию 5*, так как при удалении состояния КА путем сдвига ее строк и столбцов требуется не только переприсваивание указателей, но и освобождение памяти, выделенной под список переходов для каждой пары состояний. В статических массивах, представляющих таблицу переходов/выходов, таких списков не предусмотрено.

Многосвязный список, задающий граф переходов КА (см. рис. 4, б), также обладает рядом недостатков. Они перечислены ниже с указанием критериев, по которым применение многосвязного списка дает неизбежный рост временной сложности сравнительно с двухуровневым вектором Айлифа, представляющим таблицу переходов/выходов:

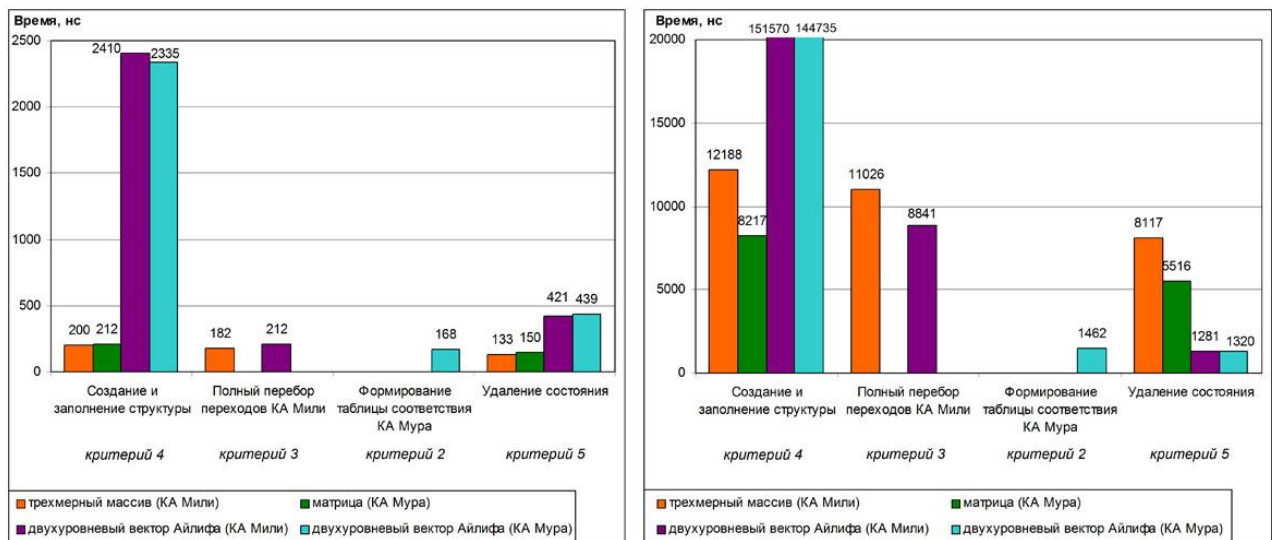
- в процессе перебора переходов КА из-за произвольного порядка элементов в списке состояний необходимо для каждого состояния проверять, было ли оно обработано ранее (*критерии 2 и 3*);
- при построении списка требуется сохранять адреса созданных элементов, чтобы последующие могли на них ссылаться (*критерий 4*);
- удаление состояния предполагает его предварительный поиск по списку полным перебором (*критерий 5*).

Таким образом, рассмотренные структуры данных для матрицы соединений и графа переходов в принятой системе критериев проигрывают структурам, реализующим таблицу переходов/выходов, как по емкостной, так и по временной сложности. Поэтому далее имеет смысл сопоставить статическую и динамическую структуры данных именно для таблицы переходов/выходов.

4. Экспериментальное исследование

Сопоставление статической и динамической структур данных, реализующих таблицу переходов/выходов КА, осуществлено путем анализа результатов эксперимента, целью которого было получение значений по временным критериям для обеих моделей КА. Эксперимент проведен на компьютере с процессором AMD A6-3420M 1.5 ГГц и объемом оперативной памяти 4 ГБ. Программное исполнение структур данных –

трехмерного массива, матрицы и двухуровневого вектора Айлифа – осуществлено средствами среды программирования Qt Creator 3.2.1 (на библиотеке Qt версии 5.3.2). Измерения проводились при помощи встроенных методов класса QElapsedTimer над указанными структурами данных, соответствующими как небольшим автоматам Мили и Мура (2 входных сигнала и 5 состояний), так и автоматам более сложным (10 входных сигналов и 10 состояний). Полученные результаты позволили построить столбчатые диаграммы (рис. 6), которые наглядно иллюстрируют время выполнения операций.



а

б

Рис. 6. Среднее время выполнения операций над КА:

а – при $|X|=2, |S|=5$; б – при $|X|=10, |S|=10$

Заметим, что число выходных сигналов КА не влияет на время выполнения операций и поэтому при программировании структур данных не учитывалось и на диаграммах не отражено. Результаты по операции полного перебора переходов показаны только для автомата Мили, так как в случае КА Мура данные практически неотличимы. Таблица соответствия состояний и выходных сигналов КА Мура для статической реализации не требуется, поэтому соответствующий столбец диаграммы пуст.

Из диаграмм видно, что динамические реализации относительно малого КА (см. рис. 6, а) проигрывают статическим по каждому критерию. Например, создание и заполнение трехмерного массива (*критерий 4*) для КА Мили происходит в среднем за 200 нс, в то время как для вектора Айлифа требуется более чем 2400 нс. Данные по этой операции для КА Мура практически те же: расхождение приблизительно в 12 раз. По времени выполнения операции удаления недостижимого состояния (*критерий 5*) расхождение между статической и динамической структурами трехкратное в пользу статической.

Однако с увеличением мощности компонентных множеств КА (см. рис. 6, б) картина меняется на противоположную: динамические структуры приобретают преимущество перед статическими. По времени удаления состояния (*критерий 5*) различие составляет приблизительно в 6 раз для автомата Мили и в 4 раза для автомата Мура. Заметим, что удвоение числа состояний приводит к увеличению времени выполнения той же операции для КА Мили на трехмерном массиве приблизительно в 60 раз (133 нс и 8117 нс), а на векторе Айлифа – всего в 3 раза (421 нс и 1281 нс). Для КА Мура кратность составляет 36 и 3 соответственно.

Переходя от временных критериев к соответствующим задачам теории автоматов (см. табл. 2), на основании данных эксперимента можно утверждать, что динамические структуры дают выигрыш по временной сложности для алгоритмов решения задач определения эквивалентности и изоморфизма автоматов. Необходимо подчеркнуть, что выигрыш достигается именно в процессе выполнения алгоритмов, поскольку подготовка исходных данных требует создания и заполнения соответствующей структуры, а время этой операции в абсолютном выражении существенно выше в случае динамической реализации. Последнее вполне объяснимо: создание списка сопровождается генерацией адресов, выделением памяти отдельно под каждый элемент, а также простановкой ссылок в адресные поля элементов. Процесс заполнения информационной части элементов списка требует переходов по адресам от текущего элемента к следующему. Создание же и заполнение многомерного массива этих действий не требует и поэтому происходит значительно быстрее.

В то же время удвоение числа состояний в случае автомата Мили дает увеличение времени на создание и заполнение структуры (*критерий 4*) примерно в 60 раз как для статической (200 нс и 12188 нс), так и для динамической (2410 нс и 151570 нс) структур данных. Для автомата Мура эта кратность в случае динамической структуры та же (2335 нс и 144735 нс), а в случае статической структуры – несколько ниже и составляет 38 (212 нс и 8217 нс, см. рис. 6), поскольку рост размерности задачи в меньшей степени влияет на время обработки матрицы, чем трехмерного массива. Таким образом, достичь некоторого снижения временной сложности на операции создания и заполнения структуры в задачах определения эквивалентности и изоморфизма автоматов можно за счет предпочтения статического массива вектору Айлифа, но только в случае автомата Мура.

Таким образом, анализ данных эксперимента приводит к выводу, что для *задач определения эквивалентности и изоморфизма* автоматов предпочтение следует отдать динамической реализации таблицы переходов/выходов двухуровневым вектором Айлифа. Однако при этом необходимо иметь в виду, что подготовка исходных данных потребует в абсолютном выражении заметно больше времени, чем заполнение статических структур данных – трехмерного массива или матрицы.

При решении *задачи эквивалентных преобразований*, как было отмечено выше, не требуется создания таблицы соответствия состояний и выходных сигналов КА Мура для статической структуры данных, что говорит в ее пользу по сравнению с динамической. По

операции полного перебора переходов (*критерий 3*) данные по обоим типам структур сопоставимы (см. рис. 6, а). Однако с ростом числа состояний и входных сигналов время, затрачиваемое на операцию перебора, снижается при реализации КА динамической структурой данных (см. рис. 6, б).

Следовательно, для решения задачи эквивалентных преобразований КА необходимо отдать предпочтение статической структуре данных – матрице или трехмерному массиву в случае малой размерности задачи, т.е. для автоматов с малым числом состояний и входных сигналов. Для автоматов с $|S| \geq 10$ и достаточно нагруженных по числу входных сигналов, что чаще встречается на практике, предпочтительной является динамическая структура – двухуровневый вектор Айлифа.

Время работы алгоритма *минимизации* КА в основном определяется полным перебором переходов автомата (*критерий 3*) и удалением эквивалентных состояний (*критерий 5*). Из диаграмм следует, что рекомендация в данном случае аналогична предыдущей задаче.

В итоге следует отметить, что для рассмотренных задач теории автоматов динамическая таблица переходов/выходов, реализуемая двухуровневым вектора Айлифа, оказалась наиболее приемлемой по предложенной системе критериев.

Заключение

В настоящей работе были проанализированы структуры данных статического и динамического типа, отвечающие трем основным способам представления базовых моделей конечных автоматов. Статические структуры представлены многомерными массивами и матрицей списков, динамические – списками различной связности. На основании анализа алгоритмов решения задач теории автоматов была предложена система критериев для оценки структур данных, учитывающая емкостную и временную сложность. Проведенная оценка позволила сделать вывод о предпочтительном использовании трехмерного массива, матрицы и двухуровневого вектора Айлифа – тех структур, которые соответствуют заданию КА таблицей переходов/выходов. Данный вывод применим к рассмотренному кругу задач теории автоматов.

Экспериментальное исследование, проведенное для трех указанных структур применительно к базовым моделям КА, показало, что для большинства задач теории автоматов более рациональным является применение динамической структуры, т.е. двухуровневого вектора Айлифа. Вычислительная сложность, учитываемая системой критериев, характеризуется меньшим ростом с увеличением размерности задач именно при задании КА вектором Айлифа, чем массивами.

Недостатком динамической структуры по сравнению со статическими являются повышенные затраты времени на заполнение вектора Айлифа данными о КА. Однако преимущество статических структур на всей совокупности рассмотренных задач проявляется по временной сложности только в случае малой размерности задач.

Полученные результаты могут служить рекомендацией разработчикам программных систем, структурный синтез которых предполагает использование аппарата теории автоматов, в частности, моделей конечных автоматов.

Исследования в направлении минимизации вычислительной сложности задач, связанных с применением автоматов, следует продолжить в части оптимизации как алгоритмов задач, так и структур данных, не исключая расширения круга тех и других. Также целесообразно обратить внимание на особенности реализации структурами данных иных моделей автоматов, в частности, недетерминированных и частично определенных, находящихся применение в прикладных системах.

Список литературы

1. Гуренко В.В. Введение в теорию автоматов: электронное учебное пособие по дисциплинам «Теория автоматов», «Прикладная теория цифровых автоматов». М., МГТУ им. Н.Э. Баумана, 2013. Режим доступа: <http://e-learning.bmstu.ru/moodle/mod/data/view.php?rid=206> (дата обращения 29.01.2015).
2. Бобров М., Андриященко Ю., Лаптева Н. Теория автоматов. 2005. Режим доступа: http://ofap.ulstu.ru/vt/Theory_of_automats/content.htm (дата обращения 29.01.2014).
3. Ожиганов А.А. Теория автоматов: учеб. пособие. СПб.: НИУ ИТМО, 2013. 84 с.
4. Лупал А.М. Теория автоматов: учеб. пособие. СПб.: СПбГУАП, 2000. 119 с.
5. Вирт Н. Алгоритмы и структуры данных. Новая версия для Оберона + CD / пер. с англ. Ф.В. Ткачева. М.: ДМК Пресс, 2010. 272 с. [Wirth N. Algorithms and Data Structures. Oberon version: August 2004. Available at: <http://www.inr.ac.ru/~info21/pdf/AD.pdf>].
6. Райли Д. Абстракции и структуры данных: пер. с англ. М.: Мир, 1993. 752 с. [Riley D.D. Data Abstraction and Structures. Boyd & Fraser Pub. Co., 1987. 662 p.].
7. Ахо А.В., Хопкрофт Д., Ульман Д.Д. Структуры данных и алгоритмы: пер. с англ. М.: Издательский дом Вильямс, 2001. 384 с. [Aho A.V., Hopcroft J.E., Ullman J.D. Data Structures and Algorithms. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1983].
8. Иванова Г.С. Математические модели структур данных // Информационные технологии. 2006. № 9. С. 44-52.
9. Guttag J. Abstract data types and the development of data structures // Communications of the ACM. 1977. Vol. 20, issue 6. P. 396-404. DOI: [10.1145/359605.359618](https://doi.org/10.1145/359605.359618)
10. Lattner C. Macroscopic Data Structure Analysis and Optimization. Ph.D. Thesis. Computer Science Dept., University of Illinois at Urbana-Champaign. 2005. 114 p. Available at: <http://llvm.org/pubs/2005-05-04-LattnerPHDThesis-book.pdf> , accessed 17.01.2015.
11. Иванова Г.С. Способы представления структурных моделей // Наука и образование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2007. № 1. Режим доступа: <http://technomag.bmstu.ru/doc/62742.html> (дата обращения 17.01.2015).

12. Овчинников В.А., Иванова Г.С., Ничушкина Т.Н. Выбор структур данных для представления графов при решении комбинаторно-оптимизационных задач // Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение. 2001. № 2 (43). С. 39-51.
13. Пасечников К.А. Анализ проблемы синтеза оптимальных структур данных для алгоритмов решения комбинаторных задач на графах // Информатика и системы управления в XXI веке: сб. трудов молодых ученых, аспирантов и студентов МГТУ им. Н.Э. Баумана. № 5. 2007. С. 118-124.
14. Пасечников К.А. Синтез оптимальных структур данных для алгоритмов решения комбинаторных задач на графах: автореф. дис. ... канд. техн. наук. М., 2009. 16 с.
15. Овчинников В.А. Графы в задачах анализа и синтеза структур сложных систем. М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. 423 с.

Data Structure Analysis to Represent Basic Models of Finite State Automation

V.V. Gurenko^{1,*}, B.I. Bychkov¹

^{*}vgurenko@bmstu.ru

¹Bauman Moscow State Technical University, Moscow, Russia

Keywords: automata theory, finite state automaton, data structure, computational complexity

Complex system engineering based on the automaton models requires a reasoned data structure selection to implement them. The problem of automaton representation and data structure selection to be used in it has been understudied. Arbitrary data structure selection for automaton model software implementation leads to unnecessary computational burden and reduces the developed system efficiency. This article proposes an approach to the reasoned selection of data structures to represent finite algorithmic automaton basic models and gives practical considerations based on it.

Static and dynamic data structures are proposed for three main ways to assign Mealy and Moore automata: a transition table, a matrix of coupling and a transition graph. A third-dimensional array, a rectangular matrix and a matrix of lists are the static structures. Dynamic structures are list-oriented structures: two-level and three-level Ayliff vectors and a multi-linked list. These structures allow us to store all required information about finite state automaton model components - characteristic set cardinalities and data of transition and output functions.

A criterion system is proposed for data structure comparative evaluation in virtue of algorithmic features of automata theory problems. The criteria focused on capacitive and time computational complexity of operations performed in tasks such as equivalent automaton conversions, proving of automaton equivalence and isomorphism, and automaton minimization.

A data structure comparative analysis based on the criterion system has been done for both static and dynamic type. The analysis showed advantages of the third-dimensional array, matrix and two-level Ayliff vector. These are structures that assign automaton by transition table. For these structures an experiment was done to measure the execution time of automation operations included in criterion system.

The analysis of experiment results showed that a dynamic structure - two-level Ayliff vector - is more beneficial for an increasing task dimension. Static structures such as the third-dimensional array and matrix should be preferred to represent only automaton with small component set cardinalities.

The article results may find application in software system development using the automata theory when a memory and time consumption minimization of finite state automaton model

processing is required. In prospect, it is necessary to work at an optimization of data structures representing the finite state automaton basic models and algorithms to solve problems with their use. Other models that have a practical application are also of interest, for example, non-deterministic automata. It is advisable to explore machine implementation aspects of such models in terms of computational complexity of applied algorithms.

References

1. Gurenko V.V. *Vvedenie v teoriyu avtomatov* [Introduction to Automata Theory]. Moscow, Bauman MSTU, 2013. Available at: <http://e-learning.bmstu.ru/moodle/mod/data/view.php?rid=206> , accessed 29.01.2015. (in Russian).
2. Bobrov M., Andryushchenko Yu., Lapteva N. *Teoriya avtomatov* [Automata Theory]. 2005. Available at: http://ofap.ulstu.ru/vt/Theory_of_automats/content.htm , accessed 29.01.2014. (in Russian).
3. Ozhiganov A.A. *Teoriya avtomatov* [Automata Theory]. St. Petersburg, NIU ITMO Publ., 2013. 84 s. (in Russian).
4. Lupal A.M. *Teoriya avtomatov* [Automata Theory]. St. Petersburg, SPbGUAP Publ., 2000. 119 p. (in Russian).
5. Wirth N. *Algorithms and Data Structures. Oberon version: August 2004*. Available at: <http://www.inr.ac.ru/~info21/pdf/AD.pdf> (Russ. ed.: Wirth N. *Algoritmy i struktury dannykh. Novaya versiya dlya Oberona + CD*. Moscow, DMK Press, 2010. 272 p.).
6. Riley D.D. *Data Abstraction and Structures*. Boyd & Fraser Pub. Co., 1987. 662 p. (Russ. ed.: Riley D.D. *Abstraktsii i struktury dannykh*. Moscow, Mir Publ., 1993. 752 p.).
7. Aho A.V., Hopcroft J.E., Ullman J.D. *Data Structures and Algorithms*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1983. (Russ. ed.: Aho A.V., Hopcroft J.E., Ullman J.D. *Struktury dannykh i algoritmy*. Moscow, Publishing House "Vil'yams", 2001. 384 p.).
8. Ivanova G.S. Data Structure Mathematical Models. *Informatsionnye tekhnologii = Information Technologies*, 2006, no. 9, pp. 44-52. (in Russian).
9. Guttag J. Abstract data types and the development of data structures. *Communications of the ACM*, 1977, vol. 20, issue 6, pp. 396-404. DOI: [10.1145/359605.359618](https://doi.org/10.1145/359605.359618)
10. Lattner C. *Macroscopic Data Structure Analysis and Optimization. Ph.D. Thesis*. Computer Science Dept., University of Illinois at Urbana-Champaign. 2005. 114 p. Available at: <http://llvm.org/pubs/2005-05-04-LattnerPHDThesis-book.pdf> , accessed 17.01.2015.
11. Ivanova G.S. The methods of structural models representation. *Nauka i obrazovanie MGTU im. N.E. Baumana = Science and Education of the Bauman MSTU*, 2007, no. 1. Available at: <http://technomag.bmstu.ru/doc/62742.html> , accessed 17.01.2015. (in Russian).
12. Ovchinnikov V.A., Ivanova G.S., Nichushkina T.N. Selection of Data Structures for Graph Representation while Solving Combinatorial and Optimizational Problems. *Vestnik MGTU*

im. N.E. Bauman. Ser. Priborostroenie = Herald of the Bauman Moscow State Technical University. Ser. Instrument Engineering, 2001, no. 2 (43), pp. 39-51. (in Russian).

13. Pasechnikov K.A. Analysis of the problem of synthesis of optimal data structures for algorithms for solving combinatorial problems on graphs. *Informatika i sistemy upravleniya v KhKhI veke: Sb. trudov molodykh uchenykh, aspirantov i studentov MGTU im. N.E. Baumana* [Informatics and management systems in the 21 century: collected papers of young scientists, postgraduates and students of the Bauman MSTU]. No. 5. 2007, pp. 118-124. (in Russian).
14. Pasechnikov K.A. *Sintez optimal'nykh struktur dannykh dlya algoritmov resheniya kombinatornykh zadach na grafakh. Avtoref. kand. diss.* [Synthesis of optimal data structures for algorithms for solving combinatorial problems on graphs. Abstract of cand. diss.]. Moscow, 2009. 16 p. (in Russian).
15. Ovchinnikov V.A. *Grafy v zadachakh analiza i sinteza struktur slozhnykh sistem* [Graphs for analysis and synthesis of structures of complex systems]. Moscow, Bauman MSTU Publ., 2014. 423 p. (in Russian).